ADVANCED WIRELESS COMMUNICATIONS
SECOND EDITION
4G
COGNITIVE AND COOPERATIVE BROADBAND TECHNOLOGY
WILEY
www.allitebooks.com
Advanced Wireless Communications

4G Cognitive and Cooperative Broadband Technology

Second Edition

Savo G. Glisic
University of Oulu, Finland

John Wiley & Sons, Ltd
Advanced Wireless Communications
Advanced Wireless Communications

4G Cognitive and Cooperative Broadband Technology

Second Edition

Savo G. Glisic
University of Oulu, Finland

John Wiley & Sons, Ltd
To my family
# Contents

*Preface to the Second Edition*  
xxi

1 Fundamentals 1  
1.1 4G and the book layout 1  
1.2 General Structure of 4G Signals 4  
  1.2.1 Advanced time division multiple access (ATDMA) 5  
  1.2.2 Code division multiple access (CDMA) 5  
  1.2.3 Orthogonal frequency division multiplexing (OFDM) 6  
  1.2.4 Multicarrier CDMA (MC CDMA) 8  
  1.2.5 Ultra wide band (UWB) signals 11  
References 16

2 Adaptive Coding 21  
2.1 Adaptive and reconfigurable block coding 21  
2.2 Adaptive and reconfigurable convolutional codes 26  
  2.2.1 Punctured convolutional codes/code reconfigurability 31  
  2.2.2 Maximum likelihood decoding/Viterbi algorithm 32  
  2.2.3 Systematic recursive convolutional codes 33  
2.3 Concatenated codes with interleavers 36  
  2.3.1 The iterative decoding algorithm 37  
2.4 Adaptive coding, practice and prospects 43  
2.5 Distributed source coding 44  
  2.5.1 Continuous valued source 46  
  2.5.2 Scalar quantization and trellis-based coset construction 48  
  2.5.3 Trellis-based quantization and memoryless 50  
  2.5.4 Performance examples 50  
Appendix 2.1 Maximum *a posteriori* detection 53  
References 56
3 Adaptive and Reconfigurable Modulation 63
  3.1 Coded modulation 63
    3.1.1 Euclidean distance 64
    3.1.2 Examples of TCM schemes 65
    3.1.3 Set partitioning 68
    3.1.4 Representation of TCM 70
    3.1.5 TCM with multidimensional constellation 70
  3.2 Adaptive coded modulation for fading channels 72
    3.2.1 Maintaining a fixed distance 73
    3.2.2 Information rate 74
  References 75

4 Space–Time Coding 79
  4.1 Diversity gain 79
    4.1.1 Two-branch transmit diversity scheme with one receiver 80
    4.1.2 Two transmitters and M receivers 82
  4.2 Space–time coding 84
    4.2.1 The system model 84
    4.2.2 The case of independent fade coefficients 85
    4.2.3 Rayleigh fading 86
    4.2.4 Design criteria for Rayleigh space–time codes 86
    4.2.5 Code construction 87
    4.2.6 Reconfiguration efficiency of space–time coding 91
    4.2.7 Delay diversity 94
  4.3 space–time block codes from orthogonal designs 96
    4.3.1 The channel model and the diversity criterion 96
    4.3.2 Real orthogonal designs 97
    4.3.3 Space–time encoder 97
    4.3.4 The diversity order 97
    4.3.5 The decoding algorithm 98
    4.3.6 inear processing orthogonal designs 98
    4.3.7 Generalized real orthogonal designs 99
    4.3.8 Encoding 99
    4.3.9 The Alamouti scheme 100
    4.3.10 Complex orthogonal designs 100
    4.3.11 Generalized complex orthogonal designs 100
    4.3.12 Special codes 101
    4.3.13 Performance results 102
  4.4 Channel estimation imperfections 102
    4.4.1 Channel estimator 106
  4.5 Quasi-orthogonal space–time block codes 107
    4.5.1 Decoding 108
    4.5.2 Decision metric 108
  4.6 Space–time convolutional codes 109
  4.7 Algebraic space–time codes 111
    4.7.1 Full spatial diversity 116
    4.7.2 QPSK modulation 116
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.8</td>
<td>Differential space–time modulation</td>
<td>116</td>
</tr>
<tr>
<td>4.8.1</td>
<td>The encoding algorithm</td>
<td>122</td>
</tr>
<tr>
<td>4.8.2</td>
<td>Differential decoding</td>
<td>123</td>
</tr>
<tr>
<td>4.9</td>
<td>Multiple transmit antenna differential detection from generalized</td>
<td>125</td>
</tr>
<tr>
<td>4.9.1</td>
<td>Differential encoding</td>
<td>126</td>
</tr>
<tr>
<td>4.9.2</td>
<td>Received signal</td>
<td>126</td>
</tr>
<tr>
<td>4.9.3</td>
<td>Orthogonality</td>
<td>127</td>
</tr>
<tr>
<td>4.9.4</td>
<td>Encoding</td>
<td>127</td>
</tr>
<tr>
<td>4.9.5</td>
<td>Differential decoding</td>
<td>128</td>
</tr>
<tr>
<td>4.9.6</td>
<td>Received signal</td>
<td>129</td>
</tr>
<tr>
<td>4.9.7</td>
<td>Demodulation</td>
<td>130</td>
</tr>
<tr>
<td>4.9.8</td>
<td>Multiple receive antennas</td>
<td>130</td>
</tr>
<tr>
<td>4.9.9</td>
<td>The number of transmit antennas lower than the number of symbols</td>
<td>130</td>
</tr>
<tr>
<td>4.9.10</td>
<td>Final result</td>
<td>131</td>
</tr>
<tr>
<td>4.9.11</td>
<td>Real constellation set</td>
<td>131</td>
</tr>
<tr>
<td>4.10</td>
<td>Layered space–time coding</td>
<td>133</td>
</tr>
<tr>
<td>4.10.1</td>
<td>Receiver complexity</td>
<td>134</td>
</tr>
<tr>
<td>4.10.2</td>
<td>Group interference suppression</td>
<td>134</td>
</tr>
<tr>
<td>4.10.3</td>
<td>Suppression method</td>
<td>134</td>
</tr>
<tr>
<td>4.10.4</td>
<td>The null space</td>
<td>134</td>
</tr>
<tr>
<td>4.10.5</td>
<td>Receiver</td>
<td>135</td>
</tr>
<tr>
<td>4.10.6</td>
<td>Decision metric</td>
<td>135</td>
</tr>
<tr>
<td>4.10.7</td>
<td>Multilayered space–time coded modulation</td>
<td>135</td>
</tr>
<tr>
<td>4.10.8</td>
<td>Diversity gain</td>
<td>136</td>
</tr>
<tr>
<td>4.10.9</td>
<td>Adaptive reconfigurable transmit power allocation</td>
<td>136</td>
</tr>
<tr>
<td>4.11</td>
<td>Concatenated space–time block coding</td>
<td>140</td>
</tr>
<tr>
<td>4.11.1</td>
<td>System model</td>
<td>140</td>
</tr>
<tr>
<td>4.11.2</td>
<td>Product sum distance</td>
<td>141</td>
</tr>
<tr>
<td>4.11.3</td>
<td>Error rate bound</td>
<td>141</td>
</tr>
<tr>
<td>4.11.4</td>
<td>The case of low SNR</td>
<td>142</td>
</tr>
<tr>
<td>4.11.5</td>
<td>Code design</td>
<td>142</td>
</tr>
<tr>
<td>4.12</td>
<td>Estimation of MIMO channel</td>
<td>145</td>
</tr>
<tr>
<td>4.12.1</td>
<td>System model</td>
<td>146</td>
</tr>
<tr>
<td>4.12.2</td>
<td>Training</td>
<td>148</td>
</tr>
<tr>
<td>4.12.3</td>
<td>Performance measure</td>
<td>148</td>
</tr>
<tr>
<td>4.12.4</td>
<td>Definitions</td>
<td>148</td>
</tr>
<tr>
<td>4.12.5</td>
<td>Channel estimation error</td>
<td>148</td>
</tr>
<tr>
<td>4.12.6</td>
<td>Error statistic</td>
<td>149</td>
</tr>
<tr>
<td>4.12.7</td>
<td>Results</td>
<td>149</td>
</tr>
<tr>
<td>4.13</td>
<td>Space–time codes for frequency selective channels</td>
<td>151</td>
</tr>
<tr>
<td>4.13.1</td>
<td>Diversity gain properties</td>
<td>153</td>
</tr>
<tr>
<td>4.13.2</td>
<td>Coding gain properties</td>
<td>154</td>
</tr>
<tr>
<td>4.13.3</td>
<td>Space–time trellis code design</td>
<td>155</td>
</tr>
<tr>
<td>4.14</td>
<td>Optimization of a MIMO system</td>
<td>157</td>
</tr>
<tr>
<td>4.14.1</td>
<td>The channel model</td>
<td>157</td>
</tr>
<tr>
<td>4.14.2</td>
<td>Gain optimization by singular value decomposition (SVD)</td>
<td>158</td>
</tr>
</tbody>
</table>
CONTENTS

4.14.3 The general \((M, N)\) case 159
4.14.4 Gain optimization by iteration for a reciprocal channel 161
4.14.5 Spectral efficiency of parallel channels 162
4.14.6 Capacity of the \((M, N)\) array 163
4.15 MIMO systems with constellation rotation 163
  4.15.1 System model 163
  4.15.2 Performance in a Rayleigh fading channel 165
4.16 Diagonal algebraic space–time block codes 167
  4.16.1 System model 167
  4.16.2 The DAST coding algorithm 169
  4.16.3 The DAST decoding algorithm 170
Appendix 4.1 QR Factorization 173
Appendix 4.2 Lattice code decoder for space–time codes 175
Appendix 4.3 MIMO channel capacity 176
References 180

5 Multiuser Communication 191
  5.1 Pseudorandom sequences 191
    5.1.1 Binary shift register sequences 191
    5.1.2 Properties of binary maximal length sequences 193
    5.1.3 Crosscorrelation spectra 193
    5.1.4 Maximal connected sets of \(m\)-sequences 194
    5.1.5 Gold sequences 194
    5.1.6 Gold-like and dual-BCH sequences 195
    5.1.7 Kasami sequences 196
    5.1.8 JPL sequences 197
    5.1.9 Kronecker sequences 197
    5.1.10 Walsh functions 198
    5.1.11 Optimum PN sequences 199
    5.1.12 Golay code 199
  5.2 Multiuser CDMA receivers 201
    5.2.1 Synchronous CDMA channels 202
    5.2.2 The decorrelating detector 202
    5.2.3 The optimum linear multiuser detector 202
    5.2.4 Multistage detection in asynchronous CDMA [43] 203
    5.2.5 Non-coherent detector 205
    5.2.6 Non-coherent detection in asynchronous multiuser channels [45] 205
    5.2.7 Multiuser detection in frequency non-selective Rayleigh fading channels 207
    5.2.8 Multiuser detection in frequency selective Rayleigh fading channels 210
  5.3 Minimum mean square error (MMSE) linear multiuser detection 216
    5.3.1 System model in multipath fading channels 217
    5.3.2 MMSE detector structures 220
    5.3.3 Spatial processing 221
CONTENTS

5.4 Single user LMMSE receivers for frequency selective fading channels 225
  5.4.1 Adaptive precombining LMMSE receivers 225
  5.4.2 Blind least squares receivers 230
  5.4.3 Least squares (LS) receiver 230
  5.4.4 Method based on the matrix inversion lemma 231

5.5 Signal subspace-based channel estimation for CDMA systems 232
  5.5.1 Estimating the signal subspace 234
  5.5.2 Channel estimation 235

5.6 Iterative receivers for layered space–time coding 236
  5.6.1 LST architectures 237
  5.6.2 LST receivers 241
  5.6.3 QR decomposition/SIC detector 242
  5.6.4 MMSE/SIC detector 244
  5.6.5 Iterative LST receivers 246

Appendix 5.1 Linear and matrix algebra 253
  Definitions 253
  Special matrices 254
  Matrix manipulation and formulas 255
  Theorems 257
  Eigendecomposition of matrices 257
  Calculation of eigenvalues and eigenvectors 258

References 259

6 Channel Estimation and Equalization 269
  6.1 Equalization in the digital data transmission system 269
    6.1.1 Zero-forcing equalizers 269
  6.2 LMS equalizer 275
    6.2.1 Signal model 275
    6.2.2 Adaptive weight adjustment 276
    6.2.3 Automatic systems 276
    6.2.4 Iterative algorithm 277
    6.2.5 The LMS algorithm 277
    6.2.6 Decision feedback equalizer (DFE) 277
    6.2.7 Blind equalizers 278
  6.3 Detection for a statistically known, time varying channel 279
    6.3.1 Signal model 279
    6.3.2 Channel model 279
    6.3.3 Statistical description of the received sequence 281
    6.3.4 The ML sequence (block) estimator for a statistically known channel 281
  6.4 LMS-adaptive MLSE equalization on multipath fading channels 284
    6.4.1 System and channel models 284
    6.4.2 Adaptive channel estimator and LMS estimator model 285
    6.4.3 The channel prediction algorithm 285
CONTENTS

6.5 Adaptive channel identification and data demodulation 288
   6.5.1 System model 288
   6.5.2 Joint channel and data estimation 288
   6.5.3 Data estimation and tracking for a fading channel 292
   6.5.4 The static channel environment 293
   6.5.5 The time varying channel environment 296
6.6 Turbo equalization 301
   6.6.1 Signal format 301
   6.6.2 Equivalent discrete time channel model 302
   6.6.3 Equivalent system state representations 302
   6.6.4 Turbo equalization 302
   6.6.5 Viterbi algorithm 303
   6.6.6 Iterative implementation of turbo equalization 304
   6.6.7 Performance 304
6.7 Kalman Filter based joint channel estimation and data
detection over fading channels 305
   6.7.1 Channel model 308
   6.7.2 The received signal 308
   6.7.3 Channel estimation alternatives 308
   6.7.4 Implementing the estimator 309
   6.7.5 The Kalman filter 310
   6.7.6 Implementation issues 310
6.8 Equalization using higher order signal statistics 311
   6.8.1 Problem statement 311
   6.8.2 Signal model 313
   6.8.3 Derivation of algorithms for DFE 313
   6.8.4 The equalizer coefficients 314
   6.8.5 Stochastic gradient DFE adaptive algorithms 315
   6.8.6 Convergence analysis 316
   6.8.7 Kurtosis-based algorithm 318
   6.8.8 Performance results 321
References 321

7 Orthogonal Frequency Division Multiplexing—OFDM
   and Multicarrier CDMA 329
   7.1 Timing and frequency offset in OFDM 329
      7.1.1 Robust frequency and timing synchronization
            for OFDM 331
   7.2 Fading channel estimation for OFDM systems 334
      7.2.1 Statistics of mobile radio channels 334
      7.2.2 Diversity receiver 335
      7.2.3 MMSE channel estimation 335
      7.2.4 FIR channel estimator 337
      7.2.5 System performance 338
      7.2.6 Reference generation 339
   7.3 64 DAPSK and 64 QAM modulated OFDM signals 339

References 339
CONTENTS

7.4 Space–time coding with OFDM signals 344
  7.4.1 Signal and channel parameters 345
  7.4.2 The wireless asynchronous transfer mode system 349
  7.4.3 Space–time coded adaptive modulation for OFDM 349
  7.4.4 Turbo and space–time coded adaptive OFDM 349
7.5 Layered space–time coding for MIMO OFDM 351
  7.5.1 System model (two times two transmit antennas) 352
  7.5.2 Interference cancellation 353
  7.5.3 Four transmit antennas 353
7.6 Space–time coded TDMA/OFDM reconfiguration efficiency 356
  7.6.1 Frequency selective channel model 356
  7.6.2 Front end prefilter 357
  7.6.3 Time-invariant channel 357
  7.6.4 Optimization problem 358
  7.6.5 Average channel 358
  7.6.6 Prefiltered M-BCJR equalizer 358
  7.6.7 Decision 359
  7.6.8 Prefiltered MLSE/DDFSE equalizer complexity 359
  7.6.9 Delayed decision feedback sequence estimation (DDFSE) 360
  7.6.10 Equalization schemes for STBC 360
  7.6.11 Single-carrier frequency domain equalized space–time block coding SC FDE STBC 361
7.7 Multicarrier CDMA system 369
  7.7.1 Data demodulation 370
  7.7.2 Performance examples 371
7.8 Multicarrier DS-CDMA broadcast systems 371
7.9 Frame by frame adaptive rate coded multicarrier DS-CDMA system 375
  7.9.1 Transmitter 377
  7.9.2 Receiver 378
  7.9.3 Rate-compatible punctured convolutional (RCPC) codes 379
  7.9.4 Rate adaptation 380
7.10 Intermodulation interference suppression in Multicarrier CDMA Systems 382
  7.10.1 Transmitter 382
  7.10.2 Non-linear power amplifier model 383
  7.10.3 MMSE receiver 383
7.11 Successive interference cancellation in Multicarrier DS-CDMA systems 386
  7.11.1 System and channel model 386
7.12 MMSE detection of multicarrier CDMA 387
  7.12.1 Tracking the fading processes 390
7.13 Approximation of optimum multiuser receiver for space–time coded multicarrier CDMA systems 393
  7.13.1 Frequency selective fading channels 397
  7.13.2 Receiver signal model of STBC MC CDMA systems 398
  7.13.3 Blind approach 399
CONTENTS

7.13.4 Bayesian optimal blind receiver 400
7.13.5 Blind Bayesian Monte Carlo multiuser receiver approximation 400
7.13.6 Gibbs sampler 400
7.13.7 Prior distributions 401
7.13.8 Conditional posterior distributions 401
7.13.9 Gibbs multiuser detection 402
7.13.10 Sampling space of data 403
7.13.11 The orthogonality property 403
7.13.12 Blind turbo multiuser receiver 403
7.13.13 Decoder-assisted convergence assessment 404
7.13.14 Performance example 404
7.14 Parallel interference cancellation in OFDM systems in time-varying multipath fading channels 405
7.15 Zero forcing OFDM equalizer in time-varying multipath fading channels 411
7.16 Channel estimation for OFDM systems 415
7.17 Turbo processing for an OFDM-based MIMO system 418
7.18 PAPR reduction of OFDM signals 420
Appendix 424
References 425

8 Ultra Wide Band Radio 433
8.1 UWB multiple access in a gaussian channel 433
8.1.1 The multiple access channel 433
8.1.2 Receiver 434
8.2 The UWB channel 436
8.2.1 Energy capture 436
8.2.2 The received signal model 436
8.2.3 The UWB signal propagation experiment 1 436
8.2.4 UWB propagation experiment 2 437
8.2.5 Clustering models for the indoor multipath propagation channel 438
8.2.6 Path loss modeling 440
8.3 UWB system with $M$-ary modulation 442
8.3.1 Performance in a Gaussian channel 442
8.3.2 Performance in a dense multipath channel 446
8.3.3 Receiver and BER performance 447
8.3.4 Time variations 447
8.3.5 Performance example 448
8.4 $M$-ary PPM UWB multiple access 448
8.4.1 $M$-ary PPM signal sets 451
8.4.2 Performance results 453
8.5 Coded UWB schemes 453
8.5.1 Performance 457
8.5.2 The uncoded system as a coded system with repetition 457
8.6 Multiuser detection in UWB radio 458
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.7</td>
<td>UWB with space–time processing</td>
<td>460</td>
</tr>
<tr>
<td>8.7.1</td>
<td>Signal model</td>
<td>460</td>
</tr>
<tr>
<td>8.7.2</td>
<td>The monopulse tracking system</td>
<td>464</td>
</tr>
<tr>
<td>8.8</td>
<td>Beamforming for UWB radio</td>
<td>467</td>
</tr>
<tr>
<td>8.8.1</td>
<td>Circular array</td>
<td>467</td>
</tr>
<tr>
<td>9</td>
<td>Linear Precoding for MIMO Channels</td>
<td>497</td>
</tr>
<tr>
<td>9.1</td>
<td>Space–time precoders and equalizers for MIMO channels</td>
<td>497</td>
</tr>
<tr>
<td>9.1.1</td>
<td>ISI modelling in MIMO channels</td>
<td>497</td>
</tr>
<tr>
<td>9.1.2</td>
<td>MIMO system precoding and equalization</td>
<td>499</td>
</tr>
<tr>
<td>9.1.3</td>
<td>Precoder and equalizer design for STBC systems</td>
<td>502</td>
</tr>
<tr>
<td>9.2</td>
<td>Linear precoding based on convex optimization theory</td>
<td>504</td>
</tr>
<tr>
<td>9.2.1</td>
<td>Generalized MIMO systems</td>
<td>505</td>
</tr>
<tr>
<td>9.2.2</td>
<td>Convex optimization</td>
<td>506</td>
</tr>
<tr>
<td>9.2.3</td>
<td>Precoding for power optimization</td>
<td>507</td>
</tr>
<tr>
<td>9.2.4</td>
<td>Precoder for SINR optimization</td>
<td>510</td>
</tr>
<tr>
<td>9.2.5</td>
<td>Performance example</td>
<td>512</td>
</tr>
<tr>
<td>9.3</td>
<td>Convex optimization-theory-based beamforming</td>
<td>513</td>
</tr>
<tr>
<td>9.3.1</td>
<td>Multicarrier MIMO signal model</td>
<td>514</td>
</tr>
<tr>
<td>9.3.2</td>
<td>Channel diagonalization</td>
<td>516</td>
</tr>
<tr>
<td>9.3.3</td>
<td>Convex optimization-based beamforming</td>
<td>520</td>
</tr>
<tr>
<td>9.3.4</td>
<td>Constraints in multicarrier systems</td>
<td>526</td>
</tr>
<tr>
<td>9.3.5</td>
<td>Performance examples</td>
<td>527</td>
</tr>
<tr>
<td>References</td>
<td></td>
<td>533</td>
</tr>
<tr>
<td>10</td>
<td>Cognitive Radio</td>
<td>537</td>
</tr>
<tr>
<td>10.1</td>
<td>Energy-efficient cognitive radio</td>
<td>537</td>
</tr>
<tr>
<td>10.1.1</td>
<td>Frame length adaptation</td>
<td>537</td>
</tr>
<tr>
<td>10.1.2</td>
<td>Frame length adaptation in flat fading channels</td>
<td>539</td>
</tr>
<tr>
<td>10.1.3</td>
<td>The adaptation algorithm</td>
<td>542</td>
</tr>
<tr>
<td>10.1.4</td>
<td>Energy-efficient adaptive error control</td>
<td>542</td>
</tr>
<tr>
<td>10.1.5</td>
<td>Processing gain adaptation</td>
<td>545</td>
</tr>
<tr>
<td>10.1.6</td>
<td>Trellis-based processing/adaptive maximum likelihood sequence equalizer</td>
<td>547</td>
</tr>
<tr>
<td>10.1.7</td>
<td>Hidden Markov channel model</td>
<td>548</td>
</tr>
<tr>
<td>10.1.8</td>
<td>Link layer performance with inadequate equalization</td>
<td>549</td>
</tr>
<tr>
<td>10.1.9</td>
<td>Link layer performance with adequate equalization</td>
<td>551</td>
</tr>
<tr>
<td>10.2</td>
<td>A cognitive radio architecture for linear multiuser detection</td>
<td>556</td>
</tr>
<tr>
<td>10.2.1</td>
<td>A unified architecture for linear multiuser detection and dynamic reconfigurability</td>
<td>556</td>
</tr>
<tr>
<td>10.2.2</td>
<td>Experimental results</td>
<td>563</td>
</tr>
<tr>
<td>10.2.3</td>
<td>The effects of quantization</td>
<td>564</td>
</tr>
<tr>
<td>10.2.4</td>
<td>The effect on the ‘near–far’ resistance</td>
<td>565</td>
</tr>
<tr>
<td>10.3</td>
<td>Reconfigurable ASIC architecture</td>
<td>567</td>
</tr>
</tbody>
</table>
CONTENTS

10.3.1 Motivation and present art 569
10.3.2 Alternative implementations 570
10.3.3 Example architecture versus an FPGA 570
10.3.4 DSP against the example architecture 571
10.3.5 Computation of a complex 16-point DFT – the Goertzel FFT mode 571
10.3.6 Fixed coefficient filters 573
10.3.7 Real FIR/correlator 574
10.3.8 Real IIR/correlator 574
10.3.9 Cascading fixed coefficient filters 574
10.3.10 Adaptive filtering 574
10.3.11 Direct digital frequency synthesis 576
10.3.12 CORDIC algorithm [83] 577
10.3.13 Discrete Fourier transform 578
10.3.14 Goertzel algorithm 578
References 580

11 Cooperative Diversity in Cognitive Wireless Networks 587
11.1 System modeling 587
11.1.1 System capacity 588
11.1.2 Probability of outage 591
11.1.3 Cellular coverage 592
11.2 Cooperative diversity protocols 593
11.2.1 System and channel models 593
11.2.2 Cooperative diversity protocols 594
11.2.3 Outage probabilities 595
11.2.4 Performance bounds for cooperative diversity 598
11.3 Distributed space–time coding 600
11.3.1 System description 600
11.3.2 BER analysis in DSTC 603
11.4 Generalization of distributed space–time-coding based on cooperative diversity 605
11.4.1 System and channel model 605
11.4.2 Cooperative diversity based on repetition 608
11.4.3 Cooperative diversity using space–time coding 612
Appendix 11.1 Asymptotic CDF approximations 614
Appendix 11.2 Amplify-and-forward mutual information 619
Appendix 11.3 Input distributions for transmit diversity bound 620
References 621

12 Cognitive UWB Communications 625
12.1 Introduction 625
12.2 Signal and interference models 627
12.3 Receiver structure and performance 628
12.3.1 Interference rejection circuit model 629
12.4 Performance examples 635
References 641
CONTENTS

13 Positioning in Wireless Networks 645
  13.1 Mobile station location in cellular networks 645
    13.1.1 Introduction 645
    13.1.2 MS location estimation using AD and RD measurements 646
    13.1.3 The circular, hyperbolic, and mixed multilateration 646
    13.1.4 WLS solution of the location problem 648
    13.1.5 Accuracy measure 649
    13.1.6 Circular multilateration 650
    13.1.7 Hyperbolic multilateration 651
    13.1.8 Mixed multilateration 652
    13.1.9 Performance results for three stations 652
    13.1.10 Performance results for \( N \) stations 654
  13.2 Relative positioning in wireless sensor networks 655
    13.2.1 Performance bounds 656
    13.2.2 Relative location estimation 659
  13.3 Average performance of circular and hyperbolic geolocation 664
    13.3.1 Signal models and performance limits 664
    13.3.2 Performance of location techniques 666
    13.3.3 Average performance of location techniques 667
  References 671

14 Channel Modeling and Measurements for 4G 675
  14.1 Macrocellular environments (1.8 GHz) 675
    14.1.1 PDF of shadow fading 677
  14.2 Urban spatial radio channels in macro/microcell (2.154 GHz) 681
    14.2.1 Description of environment 682
    14.2.2 Results 682
  14.3 MIMO channels in microcell and picocell environments (1.71/2.05 GHz) 688
    14.3.1 Simulation of channel coefficients 690
    14.3.2 Measurement setups 690
    14.3.3 Validation of the stochastic MIMO channel model assumptions 690
    14.3.4 Input parameters to the validation of the MIMO model 692
    14.3.5 The eigenanalysis method 693
  14.4 Outdoor mobile channel (5.3 GHz) 696
    14.4.1 Path loss models 700
    14.4.2 Window length for averaging fast fading components at 5 GHz 702
    14.4.3 Spatial and frequency correlations 702
    14.4.4 Path number distribution 705
    14.4.5 Rotation measurements in an urban environment 706
  14.5 Microcell channel (8.45 GHz) 708
    14.5.1 Azimuth profile 709
    14.5.2 Delay profile for the forward arrival waves 710
    14.5.3 Short-term azimuth spread (AS) for forward arrival waves 712
CONTENTS

14.6 Wireless MIMO LAN environments (5.2 GHz) 714
  14.6.1 Data evaluation 715
  14.6.2 Capacity computation 716
  14.6.3 Measurement environments 717
14.7 Indoor WLAN channel (17 GHz) 718
14.8 Indoor WLAN channel (60 GHz) 727
  14.8.1 Definition of the statistical parameters 728
14.9 UWB channel model 732
  14.9.1 The large-scale statistics 736
  14.9.2 The small-scale statistics 739
  14.9.3 Correlation of MPCs among different delay bins 741
  14.9.4 The statistical model 741
  14.9.5 Simulation steps 742
14.9 References 745

15 Adaptive 4G Networks 753
  15.1 Adaptive MAC layer 753
    15.1.1 Signal variations and the power control problem 755
    15.1.2 Spectral efficiency and effective load factor of the multirate
      DS-CDMA PRN 755
    15.1.3 CLSP/DS-CDMA packet access and traffic model 756
    15.1.4 Bit rate adaptation 756
    15.1.5 The correlated fading model and optimal packet size 758
    15.1.6 Performance 760
  15.2 Minimum energy peer-to-peer mobile wireless networks 770
    15.2.1 Network layer requirements 770
    15.2.2 The power consumption model 771
    15.2.3 Minimum power networks 772
    15.2.4 Distributed network routing protocol 773
    15.2.5 Distributed mobile networks 775
  15.3 Least resistance routing in wireless networks 778
    15.3.1 Least resistance routing (LRR) 778
    15.3.2 Multimedia least resistance routing (MLRR) 779
    15.3.3 Network performance examples: LRR versus MLRR 780
    15.3.4 Sensitivity to the number of allowable word erasures 783
  15.4 Power optimal routing in wireless networks for
    guaranteed TCP layer QoS 786
    15.4.1 Constant end-to-end error rate 786
    15.4.2 Optimization problem 788
    15.4.3 Error rate models 789
    15.4.4 Properties of power optimal paths 790
  15.4 References 791

16 Cognitive Networks and Game Theory 797
  16.1 Cognitive power control 797
    16.1.1 Noncooperative power control game 797
    16.1.2 Nash equilibrium 799
CONTENTS

16.1.3 Pareto optimality 800
16.1.4 Supermodular games and social optimality 801

16.2 Power control game with QOS guarantee 805

16.3 Power control game and multiuser detection 809

16.4 Power control game in MIMO systems 811

16.5 Game theory based MAC for AD HOC networks 813

16.6 Tit-for-Tat (TFT) game theory based packet forwarding strategies in AD HOC networks 815
  16.6.1 Strategy models 815
  16.6.2 Network nodes dependency graph and system metamodel 817
  16.6.3 The payoff of iterative game 819

16.7 TFT game theory based modeling of node cooperation with energy constraint 823
  16.7.1 Acceptance rate 823
  16.7.2 Pareto optimum 823
  16.7.3 Prisoner’s dilemma and TFT game 825

16.8 Packet forwarding model based on dynamic Bayesian games 828

16.9 Game theoretic models for routing in wireless sensor networks 830
  16.9.1 Cognitive wireless sensor network model 830
  16.9.2 Optimal rout computation 832

16.10 Profit driven routing in cognitive networks 832
  16.10.1 Algorithmic mechanism design 832
  16.10.2 Profit driven pricing mechanism 833
  16.10.3 Truthful behavior in cognitive networks 835
  16.10.4 Collusion of nodes in cognitive networks 836

16.11 Game theoretical model of flexible spectra sharing in cognitive networks with social awareness 838

16.12 A game theoretical modelling of slotted ALOHA protocol 839

16.13 Game-theory-based modeling of admission in competitive wireless networks 842
  16.13.1 System model 842
  16.13.2 Equilibrium solutions 845

16.14 Modelling access point pricing as a dynamic game 846
  16.14.1 The system model 846
  16.14.2 Modelling service reselling 848
  16.14.3 File transfer model 848
  16.14.4 Bayesian model for unknown traffic 849

References 851

Index 855
Preface to the Second Edition

Wireless communications continue to attract the attention of both the research community and industry. In the period since the book was published, significant research and industry activities have brought the fourth generation of wireless communications systems closer to implementation and standardization. Important results for this book, covering the components of common air interface, are in the area of space time coding and multicarrier modulation, especially OFDM. Within OFDM technology, for the application in high mobility environments, the most important problems are channel estimation, synchronization, and problems related to large peak-to-average power ratio. For this reason some new material has been added to Chapters 4 and 7. For the same reason, Chapter 9 is completely new and covers linear precoding in MIMO channels based on complex optimization theory. The third important component of common air interface is media access control. This includes problems of adjacent cell interference, flexible spectra sharing and cooperation between the nodes in ad hoc networks so new material based on game theory modelling is introduced in Chapter 16.

This also includes material on cooperative diversity transmission, introduced in Chapter 11, which is an emerging technology that has attracted a lot of interest lately. In addition to this, the problem of coexistence of different wireless networks is also becoming more and more important and solutions other than frequency planning and standardizations are needed. For this reason Chapter 12 is completely replaced and now presents the latest schemes for interference suppression in ultra wide band (UWB) cognitive systems, like advanced personal area networks (PAN) and discusses its performance. The schemes can be used significantly to improve performance of UWB systems, e.g. high speed Bluetooth, in the presence of interference from mobile communication systems such as GSM and WCDMA. It is also effective in the presence of WLAN systems which are nowadays based on OFDMA technology (e.g. IEEE802.11, 16e, 20) or military communications where the interference is generated by intentional jamming. The chapter also discusses the effectiveness of the scheme to suppress MC CDMA, which is a candidate technology for 4G mobile communications. Chapter 13 is significantly modified to include more details on positioning. This is the result of anticipation that this technique will be gaining more
and more space in advanced wireless communications. This is also supported by activities within the Galileo program in Europe.

In order to keep the book within acceptable limits, a part of the material from the previous edition has been left out. Most of the material omitted is from the sections on software radio. It is believed that this choice will make the content of the book more compact.

Savo G. Glisic
Sent Augustine, Florida
1

Fundamentals

1.1 4G AND THE BOOK LAYOUT

Currently the research community and industry in the field of telecommunications are considering possible choices for solutions in the fourth generation (4G) of wireless communications. This chapter will start with a generic 4G system concept that integrates available advanced technologies and then focuses on system adaptability and reconfigurability as possible options for meeting a variety of service requirements, available resources and channel conditions. The elements of such a concept can be found in Refs [1–51]. The chapter will also try to offer a vision beyond the state of the art, with the emphasis on how advanced technologies can be used for efficient 4G multiple access. Among the relevant issues the focus will be on:

- adaptive and reconfigurable coding and modulation including distributed source coding which is of interest in data aggregation in wireless sensor networks;
- adaptive and reconfigurable space–time coding including a variety of turbo receivers;
- channel estimation and equalization and multiuser detection;
- Orthogonal Frequency Division Multiple Access (OFDMA), Multi Carrier CDMA (MC CDMA) and Ultra Wide Band (UWB) radio;
- linear precoding in MIMO systems;
- cognitive radio including discussion on strategic difference between macro and micro reconfigurability;
- cooperative transmit diversity;
- user location in 4G;
- channel modeling;
- cross-layer optimization including adaptive and power efficient MAC layer design, adaptive and power efficient routing on IP and TCP layer and concept of green wireless network;
- cognitive networks modeling based on game theory.
An important aspect of wireless system design is power consumption. This will be also incorporated in the chapter including several layers in the network.

At this stage of the evolution of wireless communications there is a tendency to agree that 4G will integrate mobile communications as specified by International Mobile Telecommunications (IMT) standards and Wireless Local Area Networks (WLAN) or in general Broadband Radio Access Networks (BRAN). The core network will be based on Public Switched Telecommunications Network (PSTN) and Public Land Mobile Networks (PLMN) based on Internet Protocol (IP) [13, 16, 19, 24, 32, 41, 51]. This concept is summarized in Figure 1.1. Each of the segments of the system will be further enhanced in the future. The inter-technology roaming of the mobile terminal will be based on a reconfigurable cognitive radio concept presented in its generic form in Figure 1.2.

Figure 1.1 IMT2000 and WLAN convergence.

Figure 1.2 Reconfigurable cognitive radio concept intersystem roaming and QoS provisioning.
The material in this book is organized as follows:

Chapter 1 we start with a general structure of 4G signals, mainly advanced time division multiple access (ATDMA), code division multiple access (CDMA), orthogonal frequency division multiplexing (OFDM), multicarrier CDMA (MC CDMA) and ultra wide band (UWB) signal. These signals will be elaborated on later in the book.

Chapter 2 this chapter introduces Adaptive coding. The book has no intention of covering all details of coding but rather of focusing on those components that enable code adaptability and reconfigurability. Within this concept the chapter covers adaptive and reconfigurable block and convolutional codes, punctured convolutional codes/code reconfigurability, maximum likelihood decoding/Viterbi algorithm, systematic recursive convolutional code, concatenated codes with interleaver, the iterative (turbo) decoding algorithm and a discussion on adaptive coding practice and prospective. The chapter also includes presentation of distributed source coding, which is of interest in data aggregation in wireless sensor networks.

Chapter 3 covers adaptive and reconfigurable modulation. This includes coded modulation, trellis coded modulation (TCM) with examples of TCM schemes such as two-, four- and eight-state trellises and QAM with 3 bits per symbol transmission. The chapter further discusses signal-set partitioning, equivalent representation of TCM, TCM with multidimensional constellations, adaptive coded modulation for fading channels and adaptation to maintain fixed distance in the constellation.

Chapter 4 introduces space–time coding. It starts with a discussion on diversity gain, the encoding and transmission sequence, the combining scheme and ML decision rule for two-branch transmit diversity scheme with one and M receivers. In the next step it introduces a general discussion on space–time coding within a concept of space–time trellis modulation. The discussion is then extended to introduce space–time block codes from orthogonal design, mainly linear processing orthogonal designs and generalized real orthogonal designs. The chapter also covers channel estimation imperfections. It continuous with quasi-orthogonal space–time block codes, space–time convolutional codes and algebraic space–time codes. It also includes differential space–time modulation with a number of examples.

Layered space–time coding and concatenated space–time block coding are also discussed. Estimation of MIMO channel and space–time codes for frequency selective channels are discussed in detail. MIMO system optimization, including gain optimization by singular value decomposition (svd) are also discussed. This chapter is extended to include a variety of turbo receivers.

Chapter 5 introduces multiuser detection starting with CDMA receivers and signal subspace-based channel estimation. Then it extends this approach to iterative space time receivers. In Chapter 7 this approach is extended to OFDM receivers.

Chapter 6 deals with equalization, detection in a statistically known time-varying channel, adaptive MLSE equalization, adaptive joint channel identification and data demodulation, turbo-equalization Kalman filter based joint channel estimation and equalization using higher order signal statistics.

Chapter 7 covers orthogonal frequency division multiplexing (OFDM) and MC CDMA. The following topics are discussed: timing and frequency offset in OFDM, fading channel estimation for OFDM systems, 64-DAPSK and 64-QAM modulated OFDM signals, space–time coding with OFDM signals, layered space–time coding for MIMO-OFDM, space–time coded TDMA/OFDM reconfiguration efficiency, multicarrier CDMA system, multicarrier DS-CDMA broadcast systems, frame-by-frame adaptive rate coded multicarrier DS-CDMA systems, intermodulation interference suppression in multicarrier DS-CDMA systems, successive interference cancellation in multicarrier DS-CDMA systems, MMSE detection of multicarrier CDMA, multiuser receiver for space–time coded multicarrier CDMA systems, and peak-to-average power ratio (PAPR) problem mitigation.

Chapter 8 introduces Ultra Wide Band Radio. It covers topics like UWB multiple access in Gaussian channels, the UWB channel, UWB systems with M-ary modulation, M-ary PPM UWB multiple access, coded UWB schemes, multiuser detection in UWB radio, UWB with space–time processing and beamforming for UWB radio.

Chapter 9 covers antenna array signal processing with focus on space–time receivers for CDMA communications, MUSIC and ESPRIT DOA estimation, joint array combining and MLSE receivers, joint combiner and channel response estimation, and complexity reduction in wide-band beamforming.
Chapter 10 discusses adaptive/reconfigurable cognitive radio. The focus is on energy efficient adaptive radio, frame length adaptation, energy-efficient adaptive error control, processing-gain adaptation, trellis based processing/adaptive maximum likelihood sequence equalizer, a software radio architecture for linear multiuser detection, and reconfigurable ASIC architecture.

Chapter 11 introduces cooperative transmit diversity as a power efficient technology to increase the coverage in multihop wireless networks. It is expected that elements of this approach will be used in 4G cellular systems too, especially with relaying as a simple case of this approach.

Chapter 12 covers the problem of the coexistence of different wireless networks as it becomes more and more important, and solutions other than frequency planning and standardization are needed. For this reason Chapter 12 has been completely replaced and now presents an example of the latest schemes for interference suppression in ultra wide band (UWB) cognitive systems, like advanced personal area networks (PAN), and discusses its performance. The schemes can be used significantly to improve performance of UWB systems, e.g. high speed Bluetooth, in the presence of interference from mobile communication systems such as GSM and WCDMA. It is also effective in the presence of WLAN systems, which are nowadays based on OFDMA technology (e.g. IEEE802.11, 16e, 20). The chapter also discusses the effectiveness of the scheme in suppressing MC CDMA, which is a candidate technology for 4G mobile communications. The effectiveness decreases if the number of subcarriers is increased.

Chapter 13 is significantly modified to include more details on positioning. This is the result of the anticipation that this technique will be gaining more and more space in advanced wireless communications. This is also supported by activities within the Galileo program in Europe.

Chapter 14 discusses channel modeling and measurements for 4G. It includes macrocellular environments (1.8 GHz), urban spatial radio channels in macro/micro cell (2.154 GHz), MIMO channels in micro and pico cell environments (1.71/2.05 GHz), outdoor mobile channels (5.3 GHz), microcell channels (8.45 GHz), wireless MIMO LAN environments (5.2 GHz), indoor WLAN channels (17 GHz), indoor WLAN channel (60 GHz) and UWB channel models.

Chapter 15 includes discussion on adaptive 4G networks. It covers adaptive MAC layer, minimum energy routing in peer-to-peer mobile wireless networks, least-resistance routing in wireless networks and power optimal routing in wireless networks for guaranteed TCP layer QoS.

Chapter 16 represents a significant extension of the book to include cognitive networks and models based on game theory. The following topics are covered:

- cognitive power control as a noncooperative game, including power control games with QoS guarantee, multiuser detection and MIMO systems;
- game-theory-based MAC for ad hoc networks;
- tit-for-tat (TFT) game-theory-based packet forwarding strategies in ad hoc networks;
- TFT game-theory-based modelling of node cooperation with energy constraints;
- packet forwarding models based on dynamic Bayesian games;
- game theoretic models for routing in wireless sensor networks;
- profit driven routing in cognitive networks;
- game theoretical model of flexible spectra sharing in cognitive networks with social awareness;
- a game theoretical modelling of slotted ALOHA protocol;
- game-theory-based modeling of admission in competitive wireless networks;
- modelling access point pricing as a dynamic game.

1.2 GENERAL STRUCTURE OF 4G SIGNALS

The evolution of the common air interface in wireless communications can be presented as in Table 1.1. The coding and modulation for the 4G air interface are more or less defined but work on a
new multiple access scheme still remains to be elaborated. In this section we refer to this solution as *intercell interference coordination (IIC) in the MAC layer (IIC MAC)* as a new multiple access scheme for 4G systems.

In this section we will summarize the signal formats used in existing wireless systems and point out possible ways of evolution towards the 4G system. The focus will be on OFDMA, MC CDMA and UWB signals.

### 1.2.1 Advanced time division multiple access (ATDMA)

In a TDMA system, each user is using a dedicated time slot within a TDMA frame as in GSM (Global System of Mobile Communications) or in ADC (American Digital Cellular System). Additional data about the signal format and system capacity are given in [54]. The evolution of the ADC system resulted in TIA (Telecommunications Industry Association) Universal Wireless Communications (UWC) Standard 136 [54]. The evolution of GSM resulted in a system known as Enhanced Data rates for GSM Evolution (EDGE) with parameters summarized also in [54].

### 1.2.2 Code division multiple access (CDMA)

The CDMA technique is based on spreading the spectra of the relatively narrow information signal $S_n$ by a code $c$, generated by much higher clock (chip) rate. Different users are separated by using different uncorrelated codes. As an example, the narrowband signal in this case can be a PSK signal of the form

$$S_n = b(t, T_m) \cos \omega t$$  \hspace{1cm} (1.1)

where $1/T_m$ is the bit rate and $b = \pm 1$ is the information. The baseband equivalent of (1.1.) is

$$S^b_n = b(t, T_m)$$  \hspace{1cm} (1.1a)

A spreading operation, presented symbolically by operator $\varepsilon()$, is obtained if we multiply a narrowband signal by a pseudo noise (PN) sequence (code) $c(t, T_c) = \pm 1$. The bits of the sequence are called chips and the chip rate is $1/T_c \gg 1/T_m$. The wideband signal can be represented as:

$$S_w = \varepsilon(S_n) = cS_n = c(t, T_c)b(t, T_m) \cos \omega t$$  \hspace{1cm} (1.2)

The baseband equivalent of (1.2) is

$$S^b_w = c(t, T_c)b(t, T_m)$$  \hspace{1cm} (1.2a)

Despreading, represented by operator $D()$, is performed if we use $\varepsilon()$ once again and bandpass filtering, with the bandwidth proportional to $2/T_m$, represented by operator $BPF()$ resulting in

$$D(S_w) = BPF(\varepsilon(S_w)) = BPF(c\ c\ b\ \cos\ \omega\ t) = BPF(c\ 2\ b\ \cos\ \omega\ t) = b\ \cos\ \omega\ t$$  \hspace{1cm} (1.3)

The baseband equivalent of (1.3) is

$$D\left(S^b_w\right) = LPF\left(\varepsilon(S^b_w)\right) = LPF(c(t, T_c)c(t, T_c)b(t, T_m)) = LPF(b(t, T_m)) = b(t, T_m)$$  \hspace{1cm} (1.3a)
where LPF(\(t\)) stands for low pass filtering. This approximates the operation of correlating the input signal with the locally generated replica of the code \(\text{Cor}(c, S_w)\). Nonsynchronized despreading would result in

\[
D_x(); \text{Cor}(c, S_w) = \text{BPF}(e^x(S_w)) = \text{BPF}(c_t c b \cos \omega t) = \rho(t)b \cos \omega t \tag{1.4}
\]

In Equation (1.4) BPF would average out the signal envelope \(c_t c\) resulting in \(E(c_t c) = \rho(t)\). The baseband equivalent of Equation (1.4) is

\[
D_x(); \text{Cor}(c_t, S_w) = \int_0^{T_m} c_t \delta_w dt = b(t, T_m) \int_0^{T_m} c_t c dt = bp(t) \tag{1.4a}
\]

This operation would extract the useful signal \(b\) as long as \(\tau \equiv 0\), otherwise the signal will be suppressed because, \(\rho(t) \equiv 0\) for \(\tau \geq T_c\). Separation of multipath components in a RAKE receiver is based on this effect. In other words if the received signal consists of two delayed replicas of the form

\[
r = S_{w_x}(t) + S_{w_y}(t - \tau)
\]

the despreading process defined by Equation (1.4a) would result in

\[
D_x(); \text{Cor}(c, r) = \int_0^{T_m} c r dt = b(t, T_m) \int_0^{T_m} c(c + c_t) dt = bp(0) + bp(t)
\]

Now, if \(\rho(t) \equiv 0\) for \(\tau \geq T_c\), all multipath components reaching the receiver with a delay larger than the chip interval will be suppressed.

If the signal transmitted by user \(y\) is despread in receiver \(x\), the result is

\[
D_{xy}(); \text{BPF}(e_{xy}(S_w)) = \text{BPF}(c_x c_y b_x \cos \omega t) = \rho_{xy}(t)b_y \cos \omega t \tag{1.5}
\]

So in order to suppress the signals belonging to other users (multiple access interference, MAI), the crosscorrelation functions should be low. In other words, if the received signal consists of the useful signal plus the interfering signal from the other user:

\[
r = S_{w_x}(t) + S_{w_y}(t) = b_x c_x + b_y c_y \tag{1.6}
\]

the despreading process at receiver of user \(x\) would produce

\[
D_{xy}(); \text{Cor}(c_x, r) = \int_0^{T_m} c_x r dt = b_x \int_0^{T_m} c_x c_y dt + b_y \int_0^{T_m} c_x c_y dt = b_x \rho_x(0) + b_y \rho_{xy}(0) \tag{1.7}
\]

When the system is properly synchronized \(\rho_x(0) \equiv 1\), and if \(\rho_{xy}(0) \equiv 0\) the second component representing MAI will be suppressed. This simple principle is elaborated in the WCDMA standard resulting in a collection of transport and control channels. The system is based on 3.84 Mcips rate and up to 2 M bits/s data rate. In a special downlink, high data rate, shared channel, the data rate and signal format are adaptive. There shall be mandatory support for QPSK and 16QAM and optional support for 64 QAM based on UE capability which will proportionally increase the data rate. For details see www.3gpp.com.

### 1.2.3 Orthogonal frequency division multiplexing (OFDM)

In wireless communications, the channel imposes a limit on data rates in the system. One way to increase the overall data rate is to split the data stream into a number of parallel channels and use different subcarriers for each channel. The concept is presented in Figures 1.3 and 1.4 and represents the basic idea of OFDM system. The overall signal can be represented as

\[
x(t) = \sum_{n=0}^{N-1} \left\{ D_n e^{j2\pi \frac{n}{N} f_s t} \right\}; \quad \frac{k_1}{f_s} < t < \frac{N + k_2}{f_s} \tag{1.8}
\]
In other words complex data symbols \([D_0, D_1, \ldots, D_{N-1}]\) are mapped in OFDM symbols \([d_0, d_1, \ldots, d_{N-1}]\) such that

\[
d_k = \sum_{n=0}^{N-1} D_n e^{j2\pi \frac{kn}{N}}. \tag{1.9}
\]

The output of the FFT block at the receiver produces data per channel. This can be represented as

\[
\hat{D}_m = \frac{1}{N} \sum_{k=0}^{N-1} r_k e^{-j2\pi \frac{mk}{N}}
\]

\[
r_k = \sum_{n=0}^{N-1} H_n D_n e^{j2\pi \frac{n}{N} k} + n(k) \tag{1.10}
\]

\[
\hat{D}_m = \begin{cases} H_n D_n + N(n), & n = m \\ N(n), & n \neq m \end{cases}
\]

The system block diagram is given in Figure 1.5.

In order to eliminate residual intersymbol interference, a guard interval after each symbol is used as shown in Figure 1.6.
An example of an OFDM signal specified by IEEE 802.11a standard is shown in Figure 1.7. The signal parameters are: 64 points FFT, 48 data subcarriers, 4 pilots, 12 virtual subcarriers, DC component 0, guard interval 800 ns. Discussion on OFDM and an extensive list of references on the topic are included in Chapter 7.

1.2.4 Multicarrier CDMA (MC CDMA)

Good performance and flexibility to accommodate multimedia traffic are incorporated in MC CDMA which is obtained by combining CDMA and OFDM signal formats.

Figure 1.8 shows the DS-CDMA transmitter of the \( j \)-th user for binary phase shift keying/coherent detection (CBPSK) scheme and the power spectrum of the transmitted signal, respectively, where \( G_{DS} = T_m/T_c \) denotes the processing gain and \( \mathbf{C}(t) = [C_1^j C_2^j \cdots C_{G_{DS}}^j] \) the spreading code of the \( j \)-th user.

Figure 1.9 shows the MC-CDMA transmitter of the \( j \)-th user for the CBPSK scheme and the power spectrum of the transmitted signal, respectively, where \( G_{MC} \) is the processing gain, \( N_c \) the number of subcarriers, and \( \mathbf{C}(t) = [C_1^j C_2^j \cdots C_{G_{MC}}^j] \) the spreading code of the \( j \)-th user. The
Figure 1.7 802.11a/HIPERLAN OFDM.

Figure 1.8 DS-CDMA scheme.

Figure 1.9 MC-CDMA scheme.
Figure 1.10 Modification of MC-CDMA scheme: spectrum of its transmitted signals.

MC-CDMA scheme is discussed assuming that the number of subcarriers and the processing gain are all the same.

However, we do not have to choose $N_C = G_{MC}$, and actually, if the original symbol rate is high enough to become subject to frequency selective fading, the signal needs to be first S/P-converted before spreading over the frequency domain. This is because it is crucial for multicarrier transmission to have frequency nonselective lading over each subcarrier.

Figure 1.10 shows the modification to ensure frequency nonselective fading, where $T_S$ denotes the original symbol duration, and the original data sequence is first converted into $P$ parallel sequences, and then each sequence is mapped onto $G_{MC}$ subcarriers ($N_C = P \times G_{MC}$).

The multicarrier DS-CDMA transmitter spreads the S/P-converted data streams using a given spreading code in the time domain so that the resulting spectrum of each subcarrier can satisfy the orthogonality condition with the minimum frequency separation. This scheme was originally proposed for an uplink communications channel, because the introduction of OFDM signaling into a DS-CDMA scheme is effective for the establishment of a quasi-synchronous channel.

Figure 1.11 shows the multicarrier DS-CDMA transmitter of the $j$th user and the power spectrum of the transmitted signal, respectively, where $G_{MD}$ denotes the processing gain, $N_C$ the number of subcarriers, and $C^j(t) = [C_1^j \ C_2^j \ \cdots \ C_{G_{MD}}^j]$ the spreading code of the $j$th user.

The multitone MT-CDMA transmitter spreads the S/P-converted data streams using a given spreading code in the time domain, so that the spectrum of each subcarrier prior to the spreading operation can satisfy the orthogonality condition with the minimum frequency separation. Therefore, the resulting spectrum of each subcarrier no longer satisfies the orthogonality condition. The MT-CDMA scheme uses longer spreading codes in proportion to the number of subcarriers, as compared with a normal (single carrier) DS-CDMA scheme, therefore, the system can accommodate more users than can the DS-CDMA scheme.

Figure 1.12 shows the MT-CDMA transmitter of the $j$th user for CBPSK scheme and the power spectrum of the transmitted signal, respectively, where $G_{MT}$ denotes the processing gain, $N_C$ the number of subcarriers, and $C^j(t) = [C_1^j \ C_2^j \ \cdots \ C_{G_{MT}}^j]$ the spreading code of the $j$th user.
Figure 1.11 Multicarrier DS-CDMA scheme.

Figure 1.12 MT-CDMA scheme.

All these schemes will be discussed in detail in Chapter 7.

1.2.5 Ultra wide band (UWB) signals

For the multipath resolution in indoor environments a chip interval of the order of few nanoseconds is needed. This results in a spread spectrum signal with the bandwidth of the order of few GHz. Such a signal can also be used with no carrier resulting in what is called impulse radio (IR) or ultra wide band (UWB) radio. A typical form of the signal used in this case is shown in Figure 1.13. A collection of pulses received on different locations within the indoor environment is shown in Figure 1.14, and
A typical time-hopping format used in this case can be represented as

\[ s_{tr}^{(k)}(t) = \sum_{j=-\infty}^{\infty} \omega_{tr} \left( t^{(k)} - jT_f - c_j^{(k)}T_c - \delta d_j^{(k)} \right) \]  

(1.11)

where \( t^{(k)} \) is the \( k \)th transmitter’s clock time and \( T_f \) is the pulse repetition time. The transmitted pulse waveform \( \omega_{tr} \) is referred to as a monocycle. To eliminate collisions due to multiple access, each user (indexed by \( k \)) is assigned a distinctive time shift pattern \( \{ c_j^{(k)} \} \) called a time-hopping sequence. This provides an additional time shift of \( c_j^{(k)}T_c \) seconds to the \( j \)th monocycle in the pulse train, where \( T_c \) is the duration of addressable time delay bins. For a fixed \( T_f \) the symbol rate, \( R_s \), determines the number \( N_s \) of monocycles that are modulated by a given binary symbol as \( R_s = \left( 1/N_sT_f \right) \) s\(^{-1}\). The modulation index \( \delta \) is chosen to optimize performance. For performance prediction purposes, most of the time the data sequence \( \{ d_j^{(k)} \}_{j=-\infty}^{\infty} \) is modeled as a wide-sense stationary random process composed of equally likely symbols. For data, a pulse position data modulation is used.
When $K$ users are active in the multiple-access system, the composite received signal at the output of the receiver’s antenna is modeled as:

$$r(t) = \sum_{k=1}^{K} A_k x^{(k)}_{\text{rec}}(t - \tau_k) + n(t)$$  \hspace{1cm} (1.12)$$

The antenna/propagation system modifies the shape of the transmitted monocycle $\omega_{\text{tr}}(t)$ to $\omega_{\text{rec}}(t)$ on its output. An idealized received monocycle shape $\omega_{\text{rec}}(t)$ for a free-space channel model with no fading is shown in Figure 1.13.

The optimum receiver for a single bit of a binary modulated impulse radio signal in additive white Gaussian noise (AWGN) is a correlation receiver

\begin{align*}
\text{‘decide } d_0^{(1)} = 0' \text{ if } & \quad \text{pulsecorrelatoroutput } \Delta_{\alpha(u)} = \sum_{j=0}^{N_t-1} \int_{\tau_1}^{\tau_1+(j+1)T_f} r(u, t) \varphi \left( t - \tau_1 - jT_f - c_j^{(1)}T_c \right) \, dt \\ & \quad \text{teststatistic } \Delta_{\alpha(u)} > 0
\end{align*}  \hspace{1cm} (1.13)$$

where $\varphi(t) = \omega_{\text{rec}}(t) - \omega_{\text{rec}}(t - \delta)$.

The spectra of a signal using TH is shown in Figure 1.16. If instead of TH a DS signal is used the signal spectra is shown in Figure 1.17(a) for pseudorandom code and Figure 1.17(b) for a random code. The FCC (Frequency Control Committee) mask for indoor communications is shown in Figure 1.18. Possible options for UWB signal spectra are given in Figures 1.19 and 1.20 for single band and Figure 1.21 for multiband signal format. For more details see http://www.uwb.org and http://www.uwbnmhiband.org.

The optimal detection in a multiuser environment, with knowledge of all time-hopping sequences, leads to complex parallel receiver designs [2]. However, if the number of users is large and no such multiuser detector is feasible, then it is reasonable to approximate the combined effect of the other users’ dehopped interfering signals as a Gaussian random process. All details regarding the system performance will be discussed in Chapter 8.
Figure 1.16 Spectra of a TH signal.

Figure 1.17 Spectra of pseudorandom DS and random DS signal.
Figure 1.18 FCC frequency mask.

Figure 1.19 FCC mask and possible UWB signal spectra.

Figure 1.20 Single band UWB signal.
Figure 1.21 Multi band UWB signal.

REFERENCES


REFERENCES


REFERENCES


2

Adaptive Coding

Channel coding is a well established technical field that includes both strong theory and a variety of practical applications. Both theory and practice are well documented in open literature. In this chapter we provide a brief review of the basic principles and results in this field, with the emphasis on those parameters which are important for adaptability and reconfigurability of coding and decoding algorithms. This is an important characteristic for applications in wireless systems where a strong request for energy preservation suggests a system operation where quality of service (QoS) is met with minimum effort. In an environment with changing propagation conditions, this requires a possibility to adapt the complexity of the system. This is the focus of the presentation in this chapter and for the conventional details related to channel coding the reader is referred to the classical reference in the field.

2.1 ADAPTIVE AND RECONFIGURABLE BLOCK CODING

The simplest way to improve the probability of correct detection of a bit is to repeat the transmission of the same bit (repetition code) and base the detection of the bit on so-called majority logic. As an example, if each bit is repeated three times, the decoder will base the decision on the observation of the three bits. The error will now occur if two or more bits are received incorrectly. This is a simple solution but rather inefficient from the point of view of bandwidth utilization. The next option is the family of codes based on the parity check principle. An oversimplified example is given in Figure 2.1. For every two input bits \( u = (u_1, u_2) \), a parity check bit, \( x_3 = u_1 + u_2 \), is created so that the transmitted bits are \( x = (x_1, x_2, x_3) = (u_1, u_2, x_3) \) as indicated in the figure. This simple example can be further expanded to include a number of parity check bits. For a number of input bits \( k \), \( n - k \) parity check bits are generated, resulting in a code word of length \( n \). For this we use the notation \((n, k)\) block codes. The art of block code construction consists of finding such parity check rules that would provide the best error correction capabilities with the minimum amount of redundant bits.

The ratio \( R_c = k/n \) is called coding rate. An example of such a code is the Hamming code \((7, 4)\) shown in Figure 2.2.

The Hamming code \((7, 4)\) is defined by the relations:

\[
\begin{align*}
x_i &= u_i, \quad i = 1, 2, 3, 4 \\
x_5 &= u_1 + u_2 + u_3 \\
x_6 &= u_2 + u_3 + u_4 \\
x_7 &= u_1 + u_2 + u_4
\end{align*}
\] (2.1)
ADAPTIVE CODING

22

Figure 2.1 Encoder for the parity check code (3, 2).

<table>
<thead>
<tr>
<th>Data words</th>
<th>Code words</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>000</td>
</tr>
<tr>
<td>01</td>
<td>011</td>
</tr>
<tr>
<td>10</td>
<td>101</td>
</tr>
<tr>
<td>11</td>
<td>110</td>
</tr>
</tbody>
</table>

Figure 2.2 Encoder for the Hamming code (7, 4).

The output code words \( x(x_1, x_2, x_3, x_4, x_5, x_6, x_7) \) for all possible input words \( u(u_1, u_2, u_3, u_4) \) are shown in Table 2.1.

One can see from Table 2.1 that out of \( 2^n \) possible code words, only \( 2^k \) are used in the encoder. A collection of these code words is called a code book. The decoder will decide in the favor of the code word from the code book that is the closest to the given received code word that may contain a certain number of errors. So, in order to minimize the bit error rate, the art of coding consists of choosing those code words for the code book that differ from each other as much as possible. This difference is quantified by Hamming distance \( d \) defined as the number of bit positions where the two words are different. A number of families of block codes are described in the literature. An example is cyclic codes, represented by Bose–Chaudhuri–Hocquengham (BCH) codes and its non-binary subclass known as Reed–Solomon (RS) codes. The RS codes operate with symbols created from \( m \) bits, which are elements in the extension Galois field \( \text{GF}(2^m) \) of \( \text{GF}(2) \). Now, an RS code is defined as a block of \( n \) \( m \)-ary symbols defined over \( \text{GF}(2^m) \), constructed from \( k \) input information symbols by adding an \( n - k = 2t \) number of redundant symbols from the same extension field, giving an \( n \) \( k + 2t \) symbols code word. From now on we will use the notation for such codes: \( \text{RS}(n, k, t) \) over \( \text{GF}(2^m) \) or \( \text{BCH}(n, k, t) \) over \( \text{GF}(2) \). Such codes can correct \( t \) errors. For details of code construction and detection the reader is referred to the classical literature [1–18].

In the case of burst errors, bit interleaving, illustrated in Figure 2.3, is used. The figure represents a scheme for the interpretation of a \( (75, 25) \) interleaved code derived from a \( (15, 5) \) BCH code. A burst of length \( b = 15 \) is spread into \( t = 3 \) error patterns in each of the five code words of the interleaved code.

In general, the decoding algorithms can be based on two different options. If a hard decision is performed for each bit separately and the detected word is compared with the possible candidates from the code book, the process is referred to as hard decision decoding. For such a decision the
Table 2.1 Hamming code (7, 4)

<table>
<thead>
<tr>
<th>Data words u(u₁, u₂, u₃, u₄)</th>
<th>Code words x(x₁, x₂, x₃, x₄, x₅, x₆, x₇)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0000</td>
<td>0000 000</td>
</tr>
<tr>
<td>0001</td>
<td>0001 011</td>
</tr>
<tr>
<td>0010</td>
<td>0010 110</td>
</tr>
<tr>
<td>0011</td>
<td>0011 101</td>
</tr>
<tr>
<td>0100</td>
<td>0100 111</td>
</tr>
<tr>
<td>0101</td>
<td>0101 100</td>
</tr>
<tr>
<td>0110</td>
<td>0110 001</td>
</tr>
<tr>
<td>0111</td>
<td>0111 010</td>
</tr>
<tr>
<td>1000</td>
<td>1000 101</td>
</tr>
<tr>
<td>1001</td>
<td>1001 110</td>
</tr>
<tr>
<td>1010</td>
<td>1010 011</td>
</tr>
<tr>
<td>1011</td>
<td>1011 000</td>
</tr>
<tr>
<td>1100</td>
<td>1100 010</td>
</tr>
<tr>
<td>1101</td>
<td>1101 001</td>
</tr>
<tr>
<td>1110</td>
<td>1110 100</td>
</tr>
<tr>
<td>1111</td>
<td>1111 111</td>
</tr>
</tbody>
</table>

Figure 2.3 Interleaving.

The probability that a wrong code word is selected is given as [1–18]:

\[ P_w(e) \leq (M - 1) \left[ \sqrt{4p(1 - p)} \right]^{d_{\text{min}}} \]  

(2.2)

where \( M = 2^k \) indicates the number of code words in the code book, assumed to be equally likely, and \( d_{\text{min}} \) is the minimum Hamming distance between the code words. Parameter \( p \) represents the bit error rate. The second option is to create a sum of analog signal samples at bit positions and to compare this sum (metrics) with the possible values created from the code book. This is referred to in the literature as \textit{soft decision} decoding. The code word error rate (WER) in this case is given as [1–18]:

\[ P_w(e) \leq \frac{(M - 1)}{2} \text{erfc} \left( \sqrt{\frac{d_{\text{min}}R_cE_b}{N_0}} \right) \]

(2.3)
Figure 2.4  Word error probability for the (7, 4) Hamming code: hard decision and soft decision curves. Binary antipodal transmission.

Figure 2.5  Bit error probability curves for some BCH codes, with rate $R_c$ of about 0.5.

The word error rate for Hamming code (7,4) is given in Figure 2.4. One can see that soft decision decoding offers better performance.

If the word error occurs, then for a high signal to noise ratio, the decoder will choose the word with minimum distance from the correct one. It will make other choices with much lower probability. As a consequence, the decoded word will contain $2t + 1$ errors which can be anywhere in the $n$-bit word. So, the bit error probability can be approximated as:

$$P_b(e) \approx \left( \frac{2t + 1}{n} \right) P_w(e)$$

The bit error rate for some BCH codes is given in Figure 2.5. In general, from the figure, one can see that the longer the code the better the performance. One should be aware that this also means higher complexity. This represents the basis for code adaptability and reconfigurability. Figure 2.6 summarizes these results for a broader class of codes.
For a Hamming code and available signal to noise ratio in the range $E_b/N_0 = 9–10$ dB, we can increase the data rate $R_s/W$ approximately by a factor of two if we reconfigure the code from $H(7, 4, 1)$ to $H(511, 502, 1)$. A similar effect can be achieved by reconfiguration of the BCH codes in the range $(255, 123, 19)$ to $(255, 239, 2)$ if the available SNR changes in the range 6–8 dB. For a fixed data rate $R_s/W \approx 1$, we can reduce the required SNR for $BER = 10^{-6}$, from 9.5 to 5.5 dB (coding gain) by reconfiguring the BCH code from BCH$(15, 7, 2)$ to BCH$(255, 123, 19)$. The dependence of power consumption on the code reconfiguration range will be discussed in Chapter 10.

To relate these tradeoffs analytically we can use Relation (2.5), which represents a bound on word error probability as a function of coding rate, code length and required signal to noise ratio represented through bit error rate $p$.

$$P_w(e) \leq 2^{-n(R_0 - R_c)}, \quad R_c \leq R_0$$ (2.5)

where, for the binary symmetric channel (BSC) with $p(0) = p(1) = 1/2$,

$$R_0 = 1 - \log_2 \left[ 1 + 2\sqrt{p(1-p)} \right]$$ (2.6)

These relations are illustrated in Figure 2.7.

Finally, adaptation/reconfiguration efficiency is defined as:

$$E_{eff} = \frac{-\Delta SNR}{\Delta_r, complexity} = \frac{\text{coding gain}}{\text{relative increase in complexity}} = \frac{10^{E_b/(dB)/10}}{D_r}$$ (2.7)

This parameter will be used throughout the book to compare different schemes.
26 ADAPTIVE CODING

Figure 2.7 Cutoff rate-based bounds on the required signal to noise ratio as a function of the code rate $R_c$ for hard and soft decisions and binary antipodal modulation over the AWGN channel.

2.2 ADAPTIVE AND RECONFIGURABLE CONVOLUTIONAL CODES

In order to further increase the coding gain, a new class of codes, known as convolutional codes, is used [18–45]. A general block diagram of a convolutional encoder in serial form for an $(n_0, k_0)$ code with constraint length $N$ is given in Figure 2.8. $N$ blocks, $k_0$ bits each, are used to produce a code word $n_0$ bits long. For every new input block a new code word is generated.

Figure 2.8 Convolutional encoder.
Figure 2.9 Two equivalent schemes for the convolutional encoder of the (3, 1, 3) code.

One can show that the encoder output can be represented as:

\[ x = uG_\infty \]  

where

\[
G_\infty = \begin{bmatrix}
G_1 & G_2 & \cdots & G_N \\
G_1 & G_2 & \cdots & G_N \\
G_1 & G_2 & \cdots & G_N \\
\vdots & \vdots & \ddots & \vdots \\
G_1 & G_2 & \cdots & G_N \\
\end{bmatrix}
\]  

Submatrices \( G_i \), containing \( k_0 \) rows and \( n_0 \) columns, define connectivity of the \( i \)th block of the input register with \( n_0 \) elements of the output register. In the notation, ‘1’ means a connection and ‘0’ no connection.

We will use the following notation: code rate \( R_c = k_0/n_0 \), memory \( v = (N - 1)k_0 \) and such a code will be denoted as an \((n_0, k_0, N)\) convolutional code. An example of the encoder for \( k_0 = 1 \) is shown in Figure 2.9.

For the example in Fig. 2.9(a) we have:

\[
G_1 = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} \\
G_2 = \begin{bmatrix} 0 & 1 & 1 \end{bmatrix} \\
G_3 = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}
\]  

An equivalent representation is shown in Figure 2.9(b). For this representation we use notation that might be more convenient:

\[
G = \begin{bmatrix}
g_{1,1} & \cdots & g_{1,n_0} \\
\vdots & \ddots & \vdots \\
g_{k_0,1} & \cdots & g_{k_0,n_0}
\end{bmatrix}
\]  

with

\[
g_{1,1} = (100) \\
g_{1,2} = (110) \text{ octal numbers}(110 \rightarrow 6) \\
g_{1,3} = (111)
\]
Figure 2.10 Convolutional encoder for the (3, 2, 2) code.

Figure 2.11 Parallel implementation of the same convolutional encoder shown in Figure 2.10.

An example for $k_0 = 2$ is shown in Figure 2.10. In this case we have:

$$G_1 = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

$$G_2 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix}$$ (2.13)

One can show that the same encoder can be implemented in parallel form, as shown in Figure 2.11. The $N$ general equivalent parallel presentation is shown in Figure 2.12, and two more examples are shown in Figures 2.13 and 2.14.

For the encoder from Figure 2.9 one can verify, and then generalize for any encoder with arbitrary $n_0$ and $k_0 = 1$, that, for each input digit $u_i$, $n_0$ digits will be generated in accordance with:

$$(x_{i1}, x_{i2}, x_{i3}, \ldots, x_{in_0}) = u_iG_1 + u_{i-1}G_2 + u_{i-3}G_3 + \cdots + u_{i-N+1}G_N$$

$$= \sum_{k=1}^{N} u_{i-N+k}G_N$$ (2.14)

This relation has the form of convolution, hence the name convolutional coding.

For an efficient insight into the operation of the encoder, a state diagram, illustrated in Figure 2.15, is used.
Figure 2.12 General block diagram of a convolutional encoder in parallel form for an \((n_0, k_0, N)\) code.

Figure 2.13 Encoder for the \((2, 1, 4)\) convolutional code.

Figure 2.14 Encoder for the \((3, 2, 2)\) convolutional code.

Figure 2.15 State diagram for the \((3, 1, 3)\) convolutional code.
Figure 2.16 Trellis diagram for the (3, 1, 3) convolutional code. The boldface path corresponds to the input sequence 1101.

The (3, 1, 3) encoder, from Figure 2.9, has memory \( v = N - 1 = 2 \). We define the state \( \sigma_l \) of the encoder at discrete time \( l \) as the content of its memory at the same time, \( \sigma_l = (u_{l-1}, u_{l-2}) \). There are \( N_\sigma = 2^v = 4 \) possible states. That is, 00, 01, 10 and 11.

A solid arrow (edge) on Figure 2.15, represents a transition between two states forced by the input digit '0,' whereas a dashed arrow (edge) represents a transition forced by the input digit '1.' The label on each arrow represents the output digits corresponding to that transition. As an example we have:

\[
\begin{align*}
\mathbf{u} &= (1 \ 1 \ 0 \ 1 \ 1 \ldots) \\
\text{path } S_1 \to S_3 \to S_4 \to S_2 \to S_3 \to S_4 \ldots \\
\mathbf{x} &= (111 \ 100 \ 010 \ 110 \ 100 \ldots)
\end{align*}
\]

(2.15)

The concept of a state diagram can be applied to any \((n_0, k_0, N)\) code with memory \( v \). The number of states is \( N_\sigma = 2^v \). There are \( 2^{k_0} \) edges entering each state and \( 2^{k_0} \) edges leaving each state. The labels on each edge are sequences of length \( n_0 \). The equivalent representations using a trellis diagram or tree diagram are shown in Figure 2.16 and Figure 2.17 respectively.

An appropriate measure of the maximum likelihood decoder complexity for a convolutional code is the number of visited edges per decoded bit. Now, a rate \( k_0/n_0 \) code has \( 2^{k_0} \) edges leaving and entering each trellis state and a number of states \( N_\sigma = 2^v \), where \( v \) is the memory of the encoder. Thus, each trellis section, corresponding to \( k_0 \) input bits, has a total number of edges equal to \( 2^{k_0+v} \).

As a consequence, an \((n_0, k_0, N)\) code has a decoding complexity:

\[
D = \frac{2^{k_0+v}}{k_0}
\]

(2.16)

Reconfiguration from code 1 to code 2 can offer the coding gain \( g_{12} \). This process will, in general, increase the complexity from \( D_1 \) to \( D_2 \) resulting in:

\[
D_\epsilon = D_2/D_1 = \frac{k_{01}}{k_{02}} 2^{(k_{2b} - k_{01}) + (v_2 - v_1)}
\]

\[
= \frac{k_{01}}{k_{02}} 2^{\Delta k_0 + \Delta v} = \frac{k_{01}}{k_{02}} 2^{\Delta(k_0+v)} = \frac{k_{01}}{k_{02}} 2^{\Delta(k_0+N)}
\]

(2.17)

where \( \Delta \) should be used as an operator. Equation (2.7) defining the reconfiguration efficiency now becomes:

\[
E_{\text{eff}} = 10^{g_{12}/10} \frac{k_{02}}{k_{01}} 2^{-\Delta(k_0+v)}
\]

(2.18)
The maximum value of $D_r$ will be referred to as the **reconfiguration range** and the maximum value of the gain $g_{12}$ as the **adaptation range**.

### 2.2.1 Punctured convolutional codes/code reconfigurability

The increase of complexity inherent in passing from rate $l/n_0$ to rate $k_0/n_0$ codes can be mitigated using so-called **punctured convolutional codes**. A rate $k_0/n_0$ punctured convolutional code can be obtained by starting from a rate $l/n_0$ and deleting parity check symbols. An example is given in Figure 2.18.

Suppose now, that for every four parity check digits generated by the encoder, one (the last) is punctured, i.e. not transmitted. In this case, for every two input bits, three bits are generated by the encoder, thus producing a rate 2/3 code. The equivalent representation of the encoder is shown in Figure 2.19.

For more details, see [18, 46–48].

From the previous example, we can derive the conclusion that a rate $k_0/n_0$ convolutional code can be obtained considering $k_0$ trellis sections of a rate 1/2 mother code. Measuring the decoding complexity as before for the punctured code we have:

$$D_{\text{punc}} = \frac{k_0 2^{n+1}}{k_0}$$

so that the ratio between the case of the unpunctured to the punctured solution yields:

$$\Delta D_r = \frac{D}{D_{\text{punc}}} = \frac{2^{k_0}}{2k_0}$$
Figure 2.18 Encoder (a) and trellis (b) for a (2, 1, 3) convolutional code. The trellis (c) refers to the rate 2/3 punctured code.

Figure 2.19 Encoder (a) and trellis (b) for the (3, 2, 3) convolutional code equivalent to the rate 2/3 punctured code described.

which shows that, for \( k_0 > 2 \), there is an increasing complexity reduction. Unfortunately, in general, this would also reduce coding gain with respect to the mother code so that the relative increase in the reconfiguration efficiency can be represented as:

\[
\Delta E_{ff} = \Delta D \cdot 10^{\Delta g_{12}/10}
\]

(2.19)

where \( \Delta g_{12} \) is negative.

2.2.2 Maximum likelihood decoding/Viterbi algorithm

The maximum likelihood (ML) decoder will select the code word from the trellis, like the one in Figure 2.16, whose distance from the received sequence is minimal. Theoretically, the decoder must
find the path through the trellis for which:

\[
U(\sigma_{K-1}) \equiv \max_r U^{(r)}(\sigma_{K-1}) \equiv \max_r P\left( y | x^{(r)} \right)
\]

\[
\equiv \max_r \left[ \ln \prod_{l=0}^{K-1} P\left( y_l | x_l^{(r)} \right) \right] = \max_r \left[ \sum_{l=0}^{K-1} \ln P\left( y_l | x_l^{(r)} \right) \right] \tag{2.20}
\]

where \( x_l \) and \( y_l \) are \( n_0 \) binary transmitted and received digits respectively between discrete times \( l \) and \( l + 1 \). The branch metric

\[
V^{(r)}(\sigma_{l-1}, \sigma_l) \equiv \ln P\left( y_l | x_l^{(r)} \right) \tag{2.21}
\]

for antipodal modulation (with transmitted and received energy \( E \)) and assuming that the \( l \)th branch of the \( r \)th path has been transmitted, is:

\[
y_{jl} = \sqrt{E} \left( 2x_{jl}^{(r)} - 1 \right) + v_j
\]

So, we have:

\[
P\left( y_l | x_l^{(r)} \right) = \prod_{j=1}^{n_0} P\left( y_{jl} | x_{jl}^{(r)} \right) = \prod_{j=1}^{n_0} \frac{1}{\sqrt{\pi N_0}} \exp \left\{ - \frac{y_{jl} - \sqrt{E} \left( 2x_{jl}^{(r)} - 1 \right)}{N_0} \right\}^{2} \tag{2.22}
\]

\[
V^{(r)}(\sigma_{l-1}, \sigma_l) = \sum_{j=1}^{n_0} y_{jl} \left( 2x_{jl}^{(r)} - 1 \right) \tag{2.23}
\]

In order to reduce the number of trajectories for which the metric is calculated, the Viterbi algorithm accumulates the result only for those branches with the highest metric (survivors) that have the best chances to be selected at the end as the final choice [1–18]. A maximum a posteriori probability (MAP) detector is defined in Appendix 2.1.

In the remainder of this section we provide several examples of the performance results.

Figure 2.20 demonstrates that increasing the decoding delay (length of the observed trajectories) over 30 bit intervals cannot reduce BER significantly. In other words, for the given code, only trajectories up to 30 bits should be observed. From Figure 2.21 one can see that soft decision decoding provide a roughly 3 dB coding gain compared with hard decision decoding. Figures 2.22 and 2.23 present BER performance for coding rates 1/2 and 1/3 respectively. For these examples, \( k_0 = 1 \) and reconfiguration efficiency, defined by Equation (2.18), becomes:

\[
E_{ff} = 10^{\epsilon_{12}/10} 2^{-\Delta(N-1)} = 10^{\epsilon_{12}/10} 2^{-\Delta N} \tag{2.24}
\]

As an example, one can see from Figure 2.22 that for BER = 10^{-6}, the coding gain \( \epsilon_{12} \approx 6.7 - 4 = 2.7 \) dB can be achieved by reconfiguration of the code from \( N = 3 \) to \( N = 9 \). So the reconfiguration efficiency is \( E_{ff}(R_c = 1/2) = 10^{0.27} 2^{-6} \). On the other hand, from Figure 2.23, one can see that approximately the same coding gain can be achieved by reconfiguration of the rate \( R_c = 1/3 \) code from \( N = 3 \) to \( N = 8 \), giving \( E_{ff}(R_c = 1/3) = 10^{0.27} 2^{-5} \). In other words, we find that the efficiency of the \( R_c = 1/3 \) code is higher by a factor of two.

Achievable coding gains for different codes are shown in Table 2.2. These results can be used for calculation of reconfiguration efficiency for different types of code.

### 2.2.3 Systematic recursive convolutional codes

Systematic convolutional codes generated by feed-forward encoders yield, in general, lower free distances than non-systematic codes. In this section, we will show how to derive a systematic encoder
from every rate $1/n_0$ non-systematic encoder, which generates a systematic code with the same weight enumerating function as the non-systematic one which is relevant for the free distance [18–45]. The systematic codes are used for turbo code construction, to be discussed later.

Consider, for simplicity, a rate 1/2 feed-forward encoder characterized by the two generators (in polynomial form) $g_{1,1}(Z)$ and $g_{1,2}(Z)$. Using the power series $u(Z)$ to denote the input sequence $u$, and $x_1(Z)$ and $x_2(Z)$ to denote the two sequences $x_1$ and $x_2$ forming the code $x$, we have the relationships:

$$x_1(Z) = u(Z)g_{1,1}(Z)$$
$$x_2(Z) = u(Z)g_{1,2}(Z)$$

(2.25)
Figure 2.22 Upper bounds to the soft ML decoding bit error probability of different convolutional codes of rate 1/2.

Figure 2.23 Upper bounds to the soft ML decoding bit error probability of different convolutional codes of rate 1/3.

Table 2.2 Achievable coding gains with some convolutional codes of rate $R_c$ and constraint length $N$ at different values of the bit error probability. Eight-level quantization soft decision Viterbi decoding is used. The last line gives the asymptotic upper bound $10 \log_{10} d f R_c$

| $E_b/N_0$ for uncoded transmission (dB) | $P_b(e)$ | $R_c = 1/3$ | $R_c = 1/2$ | $R_c = 2/3$ | $R_c = 3/4$
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>$1/N$</td>
<td>$1/N$</td>
<td>$2/N$</td>
<td>$3/N$</td>
</tr>
<tr>
<td>6.8</td>
<td>$10^{-3}$</td>
<td>4.2</td>
<td>5.7</td>
<td>6.2</td>
<td>7.0</td>
</tr>
<tr>
<td>9.6</td>
<td>$10^{-5}$</td>
<td>4.3</td>
<td>5.9</td>
<td>6.5</td>
<td>7.3</td>
</tr>
<tr>
<td>11.3</td>
<td>$10^{-7}$</td>
<td>4.9</td>
<td>5.3</td>
<td>5.4</td>
<td>5.7</td>
</tr>
<tr>
<td>$\rightarrow \infty$</td>
<td>$\rightarrow 0$</td>
<td>5.2</td>
<td>6.7</td>
<td>4.8</td>
<td>5.7</td>
</tr>
</tbody>
</table>
ADAPTIVE CODING

To obtain a systematic code we need to have either \( x_1(Z) = u(Z) \) or \( x_2(Z) = u(Z) \). To obtain the first equality, let us divide both equations by \( g_{1,1}(Z) \), so that:

\[
\tilde{x}_1(Z) \doteq \frac{x_1(Z)}{g_{1,1}(Z)} = u(Z) \\
\tilde{x}_2(Z) \doteq \frac{x_2(Z)}{g_{1,1}(Z)} = \frac{u(Z)}{g_{1,1}(Z)}g_{1,2}(Z)
\]  

(2.26)

Defining now a new input sequence \( \tilde{u}(Z) \) as:

\[
\tilde{u}(Z) \doteq \frac{u(Z)}{g_{1,1}(Z)}
\]

(2.27)

the relations become:

\[
\tilde{x}_1(Z) \doteq \tilde{u}(Z)g_{1,1}(Z) \\
\tilde{x}_2(Z) \doteq \tilde{u}(Z)g_{1,2}(Z)
\]

(2.28)

2.2.3.1 Example

Let us assume that the initial encoder is defined by the polynomials \( g_{1}^A(Z) = 1 + Z^2 \), \( g_{2}^A(Z) = 1 + Z + Z^2 \). Its equivalent recursive encoder is obtained as explained previously with generators \( g_{1}^B(Z) = 1 \), \( g_{2}^B(Z) = (1 + Z + Z^2)/(1 + Z^2) \). These steps are illustrated in Figures 2.24 and 2.25.

2.3 CONCATENATED CODES WITH INTERLEAVERS

Further improvements in performance can be achieved if two codes are combined (concatenated) to encode the same message, as illustrated in Figure 2.26 (parallel concatenation) and Figure 2.27 (serial concatenation).
For ML decoding we should consider an equivalent trellis characterized by $2^{v_1+v_2}$ states. The complexity and reconfiguration efficiency are defined by the same equations as before, with $v \Rightarrow v_1 + v_2$. BER curves for the two types of concatenation are shown in Figures 2.28 and 2.29.

### 2.3.1 The iterative decoding algorithm

Due to a significant increase in the number of states in the equivalent trellis of the concatenated codes, the complexity of the ML decoder might become unacceptable. A suboptimal, iterative algorithm, known as a turbo decoder, provides a sufficiently good approximation of the ML decoder and offers performance approaching Shannon’s limit (channel capacity). An heuristic explanation for the iterative decoder is based on the assumption that Equation (2.20) can now be expressed in the form [18]:

$$\hat{u}_k \triangleq \arg \max_i [\text{APP}(k, i)]$$

where the a posteriori probability $\text{APP}(k, i)$ is defined as:

$$\text{APP}(k, i) \triangleq p(u_k = i, y_1, y_2) = \sum_{u_{k-1}} p(y_1 | c_1(u)) p(y_2 | c_2(u)) p_a(u)$$

(2.29)

$$p(y_1 | c_1(u)) = \prod_{j=1}^{N_1} p(y_{1j} | c_{1j}(u))$$

$$p(y_2 | c_2(u)) = \prod_{m=1}^{N_2} p(y_{2m} | c_{2m}(u))$$

$$p_a(u) = \prod_{l=1}^{K} p_a(u_l)$$
Figure 2.28 Average upper bounds to the bit error probability for a rate 1/3 PCCC obtained by concatenating two rate 1/2, four-state convolutional codes (CC), through a uniform interleaver of sizes $N = 100, 1000, 10000$.

Figure 2.29 Average upper bound to the bit error probability for a rate 1/3 SCCC using as outer code a four-state, rate 1/2, non-recursive convolutional encoder, and as inner code a four-state, rate 2/3, recursive convolutional encoder with uniform interleavers of various sizes.
In other words, it is assumed that the interleaver will randomize the data stream so that the output of the two encoders can be considered independent. Based on this assumption we can also write:

\[
p(y_1 | c_1(u)) \equiv \prod_l \tilde{P}_{1l}(u_l)
\]

\[
p(y_2 | c_2(u)) \equiv \prod_l \tilde{P}_{2l}(u_l)
\]

(2.30)

and Equation (2.29) becomes:

\[
\text{APP}(k, i) = \left[ \sum_{u : u_k = i} p(y_2 | c_2(u)) \prod_{l \neq k} \tilde{P}_{2l}(u_l)p_a(u_l) \right] \tilde{P}_{1k}(i)p_a(i)
\]

\[
\text{APP}(k, i) = \left[ \sum_{u : u_k = i} p(y_1 | c_1(u)) \prod_{l \neq k} \tilde{P}_{1l}(u_l)p_a(u_l) \right] \tilde{P}_{2k}(i)p_a(i)
\]

(2.31)

The solution to Equation (2.31) can be represented as:

\[
\tilde{P}_{1k}(i) = \sum_{u : u_k = i} p(y_1 | c_1(u)) \prod_{l \neq k} \tilde{P}_{2l}(u_l)p_a(u_l)
\]

\[
\tilde{P}_{2k}(i) = \sum_{u : u_k = i} p(y_2 | c_2(u)) \prod_{l \neq k} \tilde{P}_{1l}(u_l)p_a(u_l)
\]

(2.32)

Finally, Equation (2.29) becomes:

\[
\text{APP}(k, i) = \tilde{P}_{1k}(i)\tilde{P}_{2k}(i)p_a(u_l)
\]

(2.33)

The maximization of Equation (2.33) can be performed in a number of iterations as:

\[
\tilde{P}^{(0)}_{2k} = 1, \quad k = 1, \ldots, K
\]

\[
\vdots
\]

\[
\tilde{P}^{(m)}_{1k} = \sum_{u : u_k = i} p(y_1 | c_1(u)) \prod_{l \neq k} \tilde{P}^{(m-1)}_{2l}(u_l)p_a(u_l), \quad k = 1, \ldots, K
\]

(2.34)

\[
\tilde{P}^{(0)}_{2k} = \sum_{u : u_k = i} p(y_2 | c_2(u)) \prod_{l \neq k} \tilde{P}^{(m)}_{1l}(u_l)p_a(u_l), \quad k = 1, \ldots, K
\]

For binary signaling, \(u_k = 0, 1\) and the signal probabilities can be replaced by using log likelihood ratios (LLR) defined as:

\[
\lambda_k(\text{APP}) \equiv \log \frac{\sum_{u : u_k = 0} p(y_1 | c_1(u))p(y_2 | c_2(u))p_a(u)}{\sum_{u : u_k = 1} p(y_1 | c_1(u))p(y_2 | c_2(u))p_a(u)}
\]

(2.35)
By introducing the following notation

\[ \lambda_k = \log \frac{p(y_k | 0)}{p(y_k | 1)} \]

\[ \lambda_{1j} = \log \frac{p(y_{1j} | 0)}{p(y_{1j} | 1)} \quad j = 1, \ldots N_1 \]

\[ \lambda_{2m} = \log \frac{p(y_{2m} | 0)}{p(y_{2m} | 1)} \quad m = 1, \ldots N_2 \]

\[ \lambda_a = \log \frac{p_a(0)}{p_a(1)} \]

\[ \pi_{1l} = \log \frac{\tilde{P}_1(l)}{\tilde{P}_1(l)} \]

\[ \pi_{2l} = \log \frac{\tilde{P}_2(l)}{\tilde{P}_2(l)} \]

and

\[ \lambda_k(\text{APP}) = \log \frac{\sum_{u: k_k = 0} p(y_1 | c_1(u))}{\sum_{u: k_k = 1} p(y_1 | c_1(u))} = \log \frac{\sum_{u: k_k = 0} p(y_2 | c_2(u))}{\sum_{u: k_k = 1} p(y_2 | c_2(u))} + \log \frac{p_a(0)}{p_a(1)} \]

(2.36)

Equation (2.35) becomes:

\[ \lambda_k(\text{APP}) = \log \left\{ \sum_{u: k_k = 0} \exp \left[ \sum_{j=1}^{N_1} c_{1j}(u) \lambda_{1j} + \sum_{m=1}^{N_2} c_{2m}(u) \lambda_{2m} + \sum_{l=1}^{K} u_l \lambda_a \right] \right\} - \log \left\{ \sum_{u: k_k = 1} \exp \left[ \sum_{j=1}^{N_1} c_{1j}(u) \lambda_{1j} + \sum_{m=1}^{N_2} c_{2m}(u) \lambda_{2m} + \sum_{l=1}^{K} u_l \lambda_a \right] \right\} \]

(2.38)
CONCATENATED CODES WITH INTERLEAVERS

Figure 2.30 Block diagram of the iterative decoding scheme for binary convolutional codes.

The final value of $\lambda_k(\text{APP})$ is calculated as

$$\lambda_k(\text{APP}) = \pi_{1k} + \pi_{2k} + \lambda_a$$

and the MAP decision is made according to the sign of $\lambda_k$. The details of the numerical evaluation of Equation (2.39) are available in standard literature [49–59], the most relevant being the original work by Berrou [60].

These iterations contain implicitly the trellis constraints imposed by the trellis structure. For these purposes forward and backward recursions are used [49–60].

In the block diagrams shown in Figures 2.30 and 2.31, these calculations are performed in soft input soft output (SISO) blocks [49–59]. Performance curves are given in Figures 2.32, 2.33 and 2.34. In order to avoid repetition, for details of iterative calculations, defined by Equation (2.39), the reader is referred to the classical literature [49–59]. Instead of going into these details we will get back, once again, to the issue of reconfiguration efficiency. As already mentioned, for two concatenated codes with $k_{01} = k_{02}$ and an ML decoder, Equation (2.18) becomes:

$$E_{ff} = 10^{e_{12}/10} \cdot 2^{-\Delta(k_{00})}$$
Figure 2.31 Block diagrams of the encoder and iterative decoder for serially concatenated convolutional codes.

Figure 2.32 Bit error probability obtained by simulating the iterative decoding algorithm. Rate 1/2 PCCC based on 16-state rate 2/3 and 2/1 CCs, and interleaver with size $N = 8920$.

Figure 2.33 Simulated bit error probability performance of a rate 1/4 serially concatenated code obtained with two eight-state constituent codes and an interleaver yielding an input decoding delay equal to 16 384.
Figure 2.34 Comparison of rate 1/3 PCCC and SCCC. The PCCC is obtained concatenating two equal rate 1/2 four-state codes, whereas the SCCC concatenates two four-state rate 1/2 and rate 2/3 codes. The curves refer to six and nine iterations of the decoding algorithm and to an equal input decoding delay of 16 384.

So, for the two codes with \( v_1 = v_2 = v \), the reconfiguration from a single convolutional code (CC) to parallel concatenated CC (PCCC) with \( k_{01} = k_{02} \), gives:

\[
E_{ff} = 10^{g_{12}/10}2^{-v}
\]

If a turbo decoder is used, the relative complexity changes and Equation (2.19) gives:

\[
\Delta E_{ff} = 10^{-|g_{12}/10|}2^{v/2I}
\]

where \( I \) is the number of iterations.

### 2.4 ADAPTIVE CODING, PRACTICE AND PROSPECTS

Efficient error control on time varying channels can be performed by implementing an adaptive control system where the optimum code is selected according to the actual channel conditions.

There is a number of burst error correcting codes that could be used in these adaptive schemes. Three major classes of burst error correcting code are binary fire block codes, binary Iwadare–Massey convolutional codes [61], and non-binary Reed–Solomon block codes. In practical communication systems these are decoded by hard decision decoding methods. Performance evaluation based on experimental data from satellite mobile communication channels [62] shows that the convolutional codes with the soft decision decoding Viterbi algorithm are superior to all of the above burst error correcting codes.

Superior error probability performance and availability of a wide range of code rates without changing the basic coded structure, motivate the use of punctured convolutional codes [27–30] with the soft decision Viterbi decoding algorithm in the proposed adaptive scheme. To obtain the full benefit of the Viterbi algorithm on bursty channels, ideal interleaving is assumed.

An adaptive coding scheme using incremental redundancy in a hybrid automatic repeat request (ARQ) error control system is reported by Wu et al. [63]. The channel model used is BSC with time variable bit error probability. The system state is chosen according to the channel bit error rate. The
error correction is performed by shortened cyclic codes with variable degrees of shortening. When the channel bit error rate increases, the system generates additional parity bits for error correction.

An FEC adaptive scheme for matching code to the prevailing channel conditions was reported by Chase [64]. The method is based on convolutional codes with Viterbi decoding and consists of combining noisy packets to obtain a packet with a code rate low enough (less than \(\frac{1}{2}\)) to achieve the specified error rate. Other schemes that use a form of adaptive decoding are reported in [65–70]. Hybrid ARQ schemes based on convolutional codes with sequential decoding on a memoryless channel were reported by Drukarev and Costello [71, 72] while a Type-II hybrid ARQ scheme formed by concatenation of convolutional codes with block codes was evaluated on a channel represented by two states [73].

In order to implement the adaptive coding scheme it is necessary again to use a return channel. The channel state estimator (CSE) determines the current channel state, based on counting the number of erroneous blocks. Once the channel state has been estimated, a decision is made by the ‘reconfiguration block’ whether to change the code, and the corresponding messages are sent to the encoder and locally to the decoder.

In FEC schemes only error correction is performed, while in hybrid ARQ schemes retransmission of erroneous blocks is requested whenever the decoded data is labeled as unreliable.

The adaptive error protection is obtained by changing the code rates. For practical purposes it is desirable to modify the code rates without changing the basic structure of the encoder and decoder. Punctured convolutional codes are ideally suited for this application. They allow almost continuous change of the code rates, while decoding is done by the same decoder.

The encoded digits at the output of the encoder are periodically deleted according to the deleting map specified for each code. Changing the number of deleted digits varies the code rate. At the receiver end, the Viterbi decoder operates on the trellis of the parent code and uses the same deleting map as in the encoder in computing path metrics [47].

The Viterbi algorithm based on this metric is a maximum likelihood algorithm on channels with Gaussian noise, since on these channels the most probable errors occur between signals that are closest together in terms of squared Euclidean distance. However, this metric is not optimal for non-Gaussian channels. The Viterbi algorithm allows use of channel state information for fading channels [74].

However, a disadvantage of punctured convolutional codes compared to other convolutional codes with the same rate and memory order is that error paths are typically long. This requires quite long decision depths of the Viterbi decoder.

A scheme with ARQ rate compatible convolutional codes was reported by Hagenauer [75]. In this scheme, rate compatible codes are applied. The rate compatibility constraint increases the system throughput, since in transition from a higher to a lower rate code, only incremental redundancy digits are retransmitted. The error detection is performed by a cyclic redundancy check which introduces additional redundancy.

### 2.5 DISTRIBUTED SOURCE CODING

In this section we discuss the problem of compressing correlated distributed sources, i.e. correlated sources that are not co-located or that cannot cooperate to directly exploit their correlation. The most interesting application is data aggregation in wireless sensor networks (WSN). This problem has been studied in the information theory literature under the name of the Slepian–Wolf source coding problem for the lossless coding case, and as ‘rate-distortion with side information’ for the lossy coding case [76–105].

In a WSN, sensors transmit their highly correlated information to a central processing unit (sink) that forms the best picture of the scene based on a fusion of the information collected by all of them.

Data aggregation is reduced to the problem of encoding of sources in the presence of side information available only at the decoder. Consider first the problem where \(X\) and \(Y\) are correlated discrete-alphabet memoryless sources, and we have to compress \(X\) losslessly, with \(Y\) (referred to as side information) being known at the decoder but not at the encoder. If \(Y\) were known at both ends (see Figure 2.35(a)), then the problem of compressing \(X\) is well understood: one can compress \(X\)
Distributed Source Coding

at the theoretical rate of its conditional entropy [76] given \( Y, H(X|Y) \). If \( Y \) were known only at the decoder for \( X \) and not at the encoder (see Figure 2.35(b)) one can still compress \( X \) using only \( H(X|Y) \) bits, as is the case where the encoder \textit{does} know \( Y \). That is, by just knowing the joint distribution of \( X \) and \( Y \), without explicitly knowing \( Y \), the encoder of \( X \) can perform as well as an encoder which explicitly knows \( Y \). This is known as the Slepian–Wolf coding theorem [78].

As an illustration, suppose \( X \) and \( Y \) are equiprobable 3-bit binary words that are correlated in the sense that the Hamming distance between \( X \) and \( Y \) is no more than one. If \( Y \) (side information) is available to both the encoder and the decoder, clearly we can describe \( X \) using 2 bits (there are only four possibilities for the modulo-two binary sum of \( X \) and \( Y \): \{000, 001, 010, 100\}).

If \( Y \) were revealed \textit{only} to the decoder but not the encoder, it is wasteful for \( X \) to spend any bits in differentiating between \{\( X = 000 \) and \( X = 111 \)\}, since the Hamming distance between these two words is 3. Thus, if the decoder knows that either \( X = 000 \) or \( X = 111 \), it can resolve this uncertainty by checking \textit{which of them is closer in Hamming distance to} \( Y \), and declaring that as the value of \( X \). Note that the set \{000, 111\} is a 3-bit repetition code. Likewise, in addition to the set \{000, 111\}, each of the following three sets for \( X \): \{100, 011\}, \{101, 010\}, and \{011, 100\} is composed of pairs of words whose Hamming distance is 3. These are just simple variants or \textit{cosets} of the 3-bit repetition code, and they cover the space of all binary 3-tuples that \( X \) can assume. Thus, instead of describing \( X \) by its 3-bit value, we encode \textit{which coset} \( X \) belongs to, incurring a cost of 2 bits, just as in the case where \( Y \) is known to both encoder and decoder.

In general, by using terminology of algebraic channel codes, a linear block channel code [94] is specified by its 3-tuple \((n, k, d)\), where \( n \) is the code length, \( k \) is the message length, and \( d \) is the minimum distance of the code. In the above example, we considered the cosets of the linear \((3, 1, 3)\) repetition code. Every coset of a linear code is associated with a unique \textit{syndrome} [95]. Recall that the syndrome \( s \) associated with a linear channel code is defined as \( s^T = Hc^T \), where \( H \) is the parity-check matrix of the code, \( c \) is any valid codeword, \( T \) denotes transposition, and \( s, c \) are row vectors.

In general, for \( X, Y \in \{0,1\}^n \) and \( d_H(X, Y) \leq t \), the encoding of \( X \) is done in such a way that the encoder observes \( X \) and sends the index of the coset \( l \) of an appropriately chosen binary linear code with parameters \((n, k, 2t + 1)\), in which it resides. The decoder recovers the value of \( X \) to be that vector in the coset \( l \) which is closest in Hamming distance to \( Y \). The rate of transmission is \((n - k)\) bits per sample.

Figure 2.35 Communication system: (a) both encoder and decoder have access to the side information \( Y \) (which is correlated to \( X \)); (b) only decoder has access to the side information \( Y \).
Similarly to block codes, we can also define a syndrome sequence in convolutional codes. Let \( H(D) = [A(D) | I] \) be the systematic parity-check polynomial matrix of a convolutional code, and let \( x(D) \) be the source sequence in polynomial representation: \( x(D) = \sum_{i=1}^{L} x_i D^{i-1} \), where \( x_i \) denotes the \( i \)th element of \( x \). The encoder sends the syndrome sequence \( s(D) = H(D)x(D) \) to the decoder. Let \( a(D) = [0|s(D)] \) and \( y(D) = y(D) \oplus a(D) \), where \( y(D) \) is the side-information sequence. In the coset with the all-zero syndrome, let \( x'(D) \) be the sequence closest to \( y(D) \), which can be found using the Viterbi algorithm [96] for \( H(D) \). Now the reconstruction \( \hat{x}(D) = x'(D) \oplus a(D) \).

### 2.5.1 Continuous valued source

In this case \( X \) and \( Y \) are correlated memoryless processes characterized by independent and identically distributed (i.i.d.) sequences \( \{X_i\}_{i=1}^{\infty} \) and \( \{Y_i\}_{i=1}^{\infty} \), respectively. For illustration purposes we use a simple case where \( Y \) is a noisy version of \( X \), i.e., \( Y_i = X_i + N_i \), where \( \{N_i\}_{i=1}^{\infty} \) is also continuously valued (defined on the real line \( \mathbb{R} \)), i.i.d, and independent of the \( X \)'s. As before, the decoder alone has access to the \( Y \) process (side information), and the task is to compress optimally the \( X \) process. In the sequel, we consider the case where the \( X \)'s and \( N \)'s are zero-mean Gaussian random variables with known variances, so as to benchmark our performance against the theoretical performance bounds.

The goal is to form the best approximation \( \hat{X} \) to \( X \) given an encoding rate of \( R \) bits per sample. We assume encoding in blocks of length \( L \). Let the distortion measure be \( \rho(\cdot) \) over the \( L \)-sequence, defined as

\[
\rho(x, \hat{x}) = \frac{1}{L} \sum_{i=1}^{L} \rho(x_i, \hat{x}_i), \quad \rho : \mathbb{R} \times \mathbb{R} \to \mathbb{R}^+
\]

This problem can be posed as minimizing the rate of transmission \( R \) such that the reconstruction fidelity \( E[\rho(X, \hat{X})] \) is less than a given value \( D \), where \( E(\cdot) \) is the expectation operator.

The encoder is a mapping from the input space to the index set: \( \mathbb{R}^L \to \{1, 2, \ldots, 2^{LR}\} \), and the decoder is a mapping from the product space of the encoded index set and the correlated \( L \)-sequence \( Y \) to the \( L \)-sequence reconstruction \( \{1, 2, \ldots, 2^{LR}\} \times \mathbb{R}^L \to \mathbb{R}^L \). In the sequel, we focus on the mean-squared error (MSE) distortion: \( \rho(x, \hat{x}) = (x - \hat{x})^2 \).

The encoder and the decoder, as shown in Figure 2.36 consist of five mappings: \( \{M_i\}_{i=1}^{5} \).

**Source coding (\( M_1, M_2 \)):** The source \( X \) is quantized and a source codebook is constructed for a given reconstruction fidelity. The source space \( \mathbb{R}^L \) is partitioned into \( 2^{LR} \) disjoint regions, where \( R_s \) is defined as the source rate. This is referred to as mapping \( M_1 \).

Let \( \Gamma = \{\Gamma_1, \Gamma_2, \ldots, \Gamma_{2^{LR}}\} \) denote the set of \( 2^{LR} \) disjoint regions. Each region in the above partition is associated with a representation codeword. The set of representation codewords is referred to as the source codebook (S). This is a mapping \( M_2 \). We refer to the representation codeword to which \( X \) is quantized as the active codeword. Let the random variable characterizing the active codeword be denoted by \( W \).

**Estimation (\( M_1 \)):** The decoder gets the best estimate of \( X \) (minimizing the distortion) conditioned on the outcome of the side information and the element in \( \Gamma \) containing \( X \). This is given by

\[
\hat{x} = \arg\min_{a \in \mathbb{R}^L} E \left[ \rho(X, a) \mid X \in \Gamma_i \right] \quad \text{for} \quad Y = y
\]
for the received message $i$ and the side-information outcome $y$. It can be interpreted as a mapping $M_4$.

The estimation error is a function of $R_s$, which is chosen to keep this error within the given fidelity criterion.

Channel coding ($M_4, M_3$): The random variable $W$ characterizing the quantized source is correlated to $X$, and this in turn induces a correlation between $W$ and the side information $Y$. This is characterized by a conditional distribution $P(Y|W)$ of the side information given $W$. With this conditional distribution we can associate a fictitious channel with $W$ as input, which is observed at the encoder, and $Y$ as output, which is observed at the decoder, whose information channel capacity is greater than 0 (due to this correlation). Actually to communicate $W$ to the decoder in the absence of side information requires a transmission of $R_y$ bits per sample. With the presence of $Y$ at the decoder, we have this fictitious ‘helper’ channel carrying an amount of information $I(W; Y)$ about $W$. The remaining uncertainty in $W$ after observing the side information $Y$ is $H(W|Y) = H(W) - I(W; Y)$ and this is the desired final rate of transmission. The rebate in the rate of transmission is $I(W; Y)$.

The goal is to get a rebate as close to $I(W; Y)$ as possible by building a practical structured ‘channel code’ (C) for this fictitious channel on the space of $W$. Let $2^{L_{R_s}}$ denote the number of codewords in the designed channel code where $R_s$ is defined as the channel rate. Since, in general, any codeword in the source codebook can be a quantization outcome with a nonzero probability, we partition the source codebook into cosets of this channel code. The channel code is designed in such a way that each of its cosets is also an equally good channel code for the channel $P(Y|W)$. Thus, each quantization outcome belongs to a coset of this channel code, and this alone has to be conveyed to the decoder, which can then proceed to use this coset of the channel code for finding the intended active codeword.

The encoder computes the index of the coset of the channel code containing the active codeword using a mapping $M_4 : \{1, 2, \ldots, 2^{L_{R_s}}\} \rightarrow \{1, 2, \ldots, 2^L\}$ and transmits this information with rate $R = R_s - R_c$ bits per sample to the decoder. The decoder recovers the active codeword in the signaled coset by finding (channel decoding) the most likely codeword given the observed side information. This is characterized by a mapping $M_5 : R^L \times \{1, 2, \ldots, 2^L\} \rightarrow \{1, 2, \ldots, 2^{L_{R_s}}\}$.

In this approach, there is always a nonzero probability of decoding error, where the side information is decoded to a wrong codeword, and this can be made arbitrarily small by designing efficient channel codes. For a given region in $\Gamma$, the choice of the representation codeword determines $I(W; Y)$, and hence $R_c$.

Example: For a fixed-rate ($\log_2 V$) scalar quantizer with $V = 8$ levels, designed for the distribution of $X$ let $\mathcal{V} = \{r_0, r_1, \ldots, r_{V-1}\}$ be the set of reconstruction levels as shown in the Figure 2.37. $\mathcal{V}$ partitions the real line into $V$ intervals each associated with one of the reconstruction levels. Let $\Gamma = \{\Gamma_i\}_{i=0}^{V-1}$ be the partition of $R$, where $\Gamma_i$ is the open interval $\left(\frac{r_i}{2}, \frac{r_{i+1}}{2}\right)$ and we take $r_{-1} = -\infty$ and $r_V = \infty$. We first quantize the source sample by sample using $\mathcal{V}$. Thus, the source codebook $S$ is given by $\mathcal{V}$, and $R_s = 3$ bits/sample. Next we construct $C$ by partitioning the set $\mathcal{V}$ into $M(\leq V)$ cosets. For illustration, let $M = 2$. We group $r_0$, $r_2$, $r_4$, and $r_6$ into one coset. Similarly, $r_1$, $r_3$, $r_5$, and $r_7$
are grouped into another coset. This is done to keep the minimum distance between any two words in every coset as large as possible. Thus, the channel code is defined as $C = \{r_0, r_2, r_4, r_6\}$, making $R_c = 2$.

The information about the coset is transmitted by $R = 1 \text{ bit/sample}$. The representation codeword $r_i$ is the centroid of the disjoint region $\Gamma_i$ for $0 \leq i \leq 2^V - 1$.

The decoder deciphers (with a small probability of error) the active codeword by finding the codeword which is closest to $Y$ in the coset whose index is sent by the encoder and the optimal estimate $\hat{x}$ is computed as:

$$\hat{x} = \arg \min_{a \in R} E \left[ \rho(X, a) \left| X \in \Gamma_i, Y = y \right. \right]$$

(2.42)

where $y$ is the outcome of $Y$, $i$ is the index of the active codeword.

### 2.5.2 Scalar quantization and trellis-based coset construction

In this section we use scalar (memoryless) quantization as before and a coset construction having memory $L$. In other words we still use fixed-length scalar quantizers for quantizing $\{X_i\}_{i=1}^L$, but the cosets are built on the space $\nabla^L$. Thus, for the previous example, the source codebook is given by $S = \nabla^L$ and $R_s = 3 \text{ bits/sample}$.

**Source coding ($M_1, M_2$):** For a general $R_s$, $M_1: R^L \to \{1, 2, \ldots, 2^{LR_s}\}$ is the $L$-product scalar quantizer and $M_2: \{1, 2, \ldots, 2^{LR_s}\} \to \nabla^L$ is the $L$-sequence extension of the mapping $\{1, 2, \ldots, 2^{R_s}\} \to \nabla$.

For the previous example, the space $\nabla^L$ has $2^{3L}$ distinct sequences. The task is to partition this sequence space $S$ into cosets of a set of sequences in such a way that the minimum distance between any two sequences in every coset is made as large as possible, while maintaining symmetry among the cosets. Let $R = 1 \text{ bit/sample}$. In the following, a trellis-based partitioning with an algebraic structure is considered based on convolutional codes and set-partitioning rules of trellis-coded scalar-quantization (TCSQ) [99].

Consider a trellis code built on $\nabla$ with block length $L$, having $2^{2L}$ sequences (code rate is $R_c = 2$ bits/sample), using a rate-$2/3$ convolutional code and a mapping $Q: \{0, 1\}^3 \to \nabla$. For this example,
Let \( x \in \nabla \) of \( a_k \) consists of 22 \( C(\cdot) \).

The encoder sends the index \( t \) of the principal trellis coset (two trellis cosets forms a partition of the trellis code) as a channel code \( C \) for the channel \( P(Y|W) \). Clearly, \( C \subset S \). If \( H(D) \) be the parity-check matrix polynomial of the underlying convolutional code, and \( \theta \) any sequence in \( \nabla^L \) then we have \( Q^{-1}(\theta) \in \{0,1\}^{2L} \).

In the sequel, we associate for any syndrome \( s \in \{0,1\}^L \) a coset of \( C \), given by \( C(s) \), which consists of all the sequences \( z \in \nabla^L \) such that \( H(D)Q^{-1}(z) = s^T \). Each of \( 2^L \) cosets of \( C \) consists of \( 2^L \) sequences. This has resulted in a partition of the space \( \nabla^L \) into \( 2^L \) cosets of \( C \). The encoder sends the index \( s \) of the coset \( C(s) \) containing the quantized source sequence.

**Coset Indexing (M₄):** For the case of general \( R \), and for any \( \theta = \{1,2,\ldots,2^{L(R)}\} \) we use \( M₄(\theta) = D_R[H(D)Q^{-1}[M₄(\theta)]] + 1 \), where \( D_R \) is the R-bit representation to decimal conversion mapping: \( \{0,1\}^{RL} \rightarrow \{0,1,2,\ldots,2^{RL} - 1\} \).

**Decoder (M₅):** The decoder receives \( L \) bits of syndrome \( s \), and \( L \) samples of the process \( Y \) and searches through the list of codeword sequences in a given coset \( C(s) \) for the most likely codeword sequence, given the side-information sequence \( y \). Consider the \( k \)th stage of the four-state trellis as shown in Figure 2.38(a). This is the trellis for the coset of \( C \) with the all-zero syndrome. If the \( k \)th bit of the syndrome sequence is 1 rather than 0, we need to modify the labels on each edge of the principal trellis coset at the \( k \)th stage. Let \( \alpha = [000]^T \), where \( a_k = [000]^T \) for all \( 1 \leq k \leq 7 \), and \( \theta \) is the all-zero sequence. As discussed earlier, for the trellis code under consideration, the sequence \( Q[[000]^T] \) belongs to the coset whose syndrome is \( s \). Thus, at the \( k \)th stage of decoding, if the \( k \)th bit of \( s \) is 1, we need to shift from the principal trellis coset to the other trellis coset (there are only two trellis cosets in the given example: see Figure 2.38(b)).

This is done by defining the trellis coset at the \( k \)th stage as having the edge labels \( \xi = Q[Q^{-1}(r)] \oplus a_k \), where \( r \in \nabla \) is the corresponding label in the principal trellis coset \( a_k = [000]^T \) (this is possible due to the systematic trellis codes). The set of labels on the edges starting from any given state in the two trellis cosets forms a partition of \( \nabla \). For example, in the first state, the edge labels of the principal trellis coset \( \{r_0, r_2, r_4, r_6\} \), and that of the complementary trellis coset \( \{r_1, r_3, r_5, r_7\} \) form a partition of \( \nabla \). Thus, starting from \( k = 1 \) to \( k = L \), at every stage we need to keep relabeling (shifting between the principal trellis coset and the other trellis coset) the edges in the trellis used in the Viterbi decoder. Let \( \mathcal{X} \) be the sequence that is closest to \( y \) obtained using this algorithm (characterizing the mapping \( M₅ \) with relabeled edges. This is illustrated in Figure 2.39 with an example. For a general \( R \), there will be \( 2^R \) trellis cosets. It can be noted that the minimum distance between any two words in every coset of \( C \) has been increased from that in the memoryless coset construction.

After recovering the active codeword, the optimal estimate is given as follows:

\[
\hat{\mathbf{x}} = M₃(y, \theta) = \arg\min_{a \in \mathbb{R}^L} \left\{ E \left\{ \rho(X, a) \left| Y = y, X \in \Gamma \right. \right\} \right. \quad (2.43)
\]
Figure 2.39 An example of computation of the syndrome \((R_s, R_c) = (3, 2)\) bits per sample: let the outcome of quantization be 7, 3, 2, 1, 5. The syndrome is given by 10110 for five samples. Sample numbers 0, 2, and 3 use complementary trellis coset and the rest use principal trellis coset.

where \(\theta\) is the index of the codeword \(x'\) in \(S\), and \(\Gamma\) is the sequence of intervals associated with the elements of \(x'\). Since \(S\) is memoryless, we can simplify the preceding expression as

\[
\hat{x}_k = \arg \min_{a \in \mathbb{R}} E \left\{ \rho(X_k, a) \left| Y_k = y_k, X_k \in \Gamma_i \right. \right\}
\]

(2.44)

where \(\hat{x}_k\) is the \(k\)th sample of \(\hat{x}\), for \(1 \leq k \leq L\), and \(x'_k = r_i \in \nabla\) (the estimation assumes correct decoding).

2.5.3 Trellis-based quantization and memoryless

Coset construction

We now use a more efficient quantizers such as TCSQ as source codebook, and construct cosets on this space. Consider a TCSQ with rate \(R_s\) bits per sample built on a scalar quantizer alphabet \(\nabla\) of size \(2^{R_s+1}\). This has a finite-state machine with a rate-\(R_s/(R_s + 1)\) convolutional code and a mapping \(Q : \{0, 1\}^{R_s+1} \rightarrow \nabla\). The set of \(2^{LR_s}\) ‘valid’ sequences of this TCSQ constitutes the source codebook \(S\). The source is quantized using \(S\) by applying the Viterbi algorithm.

Source coding \((M_1, M_2)\): \(M_1\) corresponds to this trellis-coded quantization, and \(M_2\) corresponds to the rule of assigning the sequences in \(S\) to their corresponding regions. In example with \((R_s, R_c) = (2, 1)\) bits per sample and the trellis as shown in Figure 2.38(a) we consider a TCSQ with parallel transitions, and assign all the label points on a parallel transition to the same coset of a sequence set \(C\), resulting in a reduction in rate by 1 bit/sample. In other words, we partition the scalar quantizer \(\nabla\) into four sets: \(\{r_0, r_4\}, \{r_1, r_5\}, \{r_2, r_6\}, \{r_3, r_7\}\), and the encoder does not expend bits to differentiate between the elements of a set (i.e., labels among parallel transitions between any two connected states in the trellis). Thus, the channel code is given by \(C = \{r_0, r_4\}^L \subset S\). This amounts to discarding the uncoded bit in the 2-bit representation of the TCSQ codewords and sending only the coded bit to the decoder.

The set of codewords in the TCSQ can be viewed as a coset code [101]. \(\nabla\) is partitioned into four cosets of \(\{r_0, r_4\}\), and a bit sequence set is used to index the valid set of \(L\)-dimensional coset sequences.

The extension of the above techniques to the case of trellis based quantization and trellis based coset construction is straightforward.

2.5.4 Performance examples

The following model for \(X\) and \(Y\): \(Y = X + N\) is used in simulation, where \(X\) is i.i.d. Gaussian with zero mean and unit variance and \(N\) is i.i.d. Gaussian with zero mean and independent of \(X\). For memoryless source and channel codes 4-, 8-, and 16-level scalar quantizers are used, each partitioned into two
cosets, with each coset containing two, four, and eight codewords, respectively. Distortion during correct decoding only is plotted versus correlation-SNR (which is the ratio of the variance of $X$ and $N$) for these three schemes in Figure 2.40(a). Figure 2.40(b) shows the probability of decoding error ($P_e$) for the same system. These are the results of Monte Carlo simulations. As can be noted, there is a tradeoff between the distortion and probability of decoding error. For a given correlation-SNR, as the number of levels in the quantizer is increased, the distortion decreases and the probability of decoding error increases.

For the trellis-based coset construction we use 4- and 8-state trellises built on a 4-level scalar quantizer as shown in Figure 2.41(a). These trellises are designed based on the set partitioning rules of [100] and [101]. Figure 2.42(a) gives the probability of decoding error versus correlation-SNR. Note that by using trellis-based cosets, we get gains of around 3–4 dB in correlation-SNR over memoryless coset construction. Thus, without increasing the rate, at $P_e \leq 10^{-4}$, we can operate at correlation-SNRs no less than 12 dB (see Figure 2.42(a)) compared with 15.5 dB (see Figure 2.40(b)). Similar coset constructions are done on an 8-level quantizer using 4-, 8-state trellises for $R = 1$ bit/sample as shown Figure 2.41(b). Figure 2.42(b) gives the performance in terms of $P_e$. Here again we get 3-dB gain over memoryless coset construction with four- and eight-state trellis at $P_e \leq 10^{-4}$.

We now construct trellis-based quantizers and coset construction. We construct such TCSQs and coset partitions on 8- and 16-level quantizers. The trellises for the source and the channel codes for the 8-level quantizer are shown in Figure 2.43. Recall that the encoder uses the trellis of Figure 2.43(a) in quantization, while the receiver uses the trellis of Figure 2.43 (b) in decoding the side information. The probability of decoding error as a function of correlation-SNR is plotted in Figure 2.44 with the number states ranging from 2 to 32. The performance of these systems is better than that of the corresponding constructions with memoryless cosets. Using these methods we can now operate at correlation-SNRs as low as 8.5 dB with 32 states $P_e \leq 10^{-4}$.

In order to analyze the system behavior for higher rates of transmission let us consider the constructions of memoryless source codes and trellis based coset construction for $R = 2$ bits/sample. For a trellis code (characterizing $C$ with $R_c = 1$) built on an 8-level quantizer (characterizing $S$ with $R_s = 3$), we need a rate-1/3 convolutional code, and a mapping rule $Q$ to maximize the coding gain. In
Figure 2.41 Trellises for the channel code C for \( R = 1 \) bit/sample using memoryless source codes (a) for 4-level quantizer; (b) for 8-level quantizer. Both (a) and (b) have the same structure for a given number of states.

principle, for this trellis code, the set partition rule works by increasing the alphabet size four times, as compared to twice in most of the trellis designs. An alternative is to choose a subset (containing four elements) of the available alphabet (eight levels), and use only them in one coset. We partition the 8-level quantizer into two cosets (as in memoryless coset construction for \( R = 1 \)), and construct trellis codes on each of these memoryless cosets. Let Coset-1 and Coset-2 specify the codeword sets \{0, 2, 4, 6\} and \{1, 3, 5, 7\}, respectively. A rate-1/2 convolutional code with set partitioning on Coset-1 can be used to get trellis codes with a 4-state trellis as shown in Figure 2.45(a). The trellis uses an alphabet of size \( 2^{R_c+2} \). There are four trellis cosets, with two of them built on Coset-1 and two built on Coset-2. The index of one of the four trellis cosets containing the quantized source sample is sent to the decoder. Figure 2.46(a) shows the decoding error performance of this construction. Similarly to the case of \( R = 1 \), we get 3–4-dB gains over memoryless coset construction when \( P_e \leq 10^{-4} \). As \( R \) is increased from 1 to 2 bits/sample, for the same trellis complexity and scalar quantizer, the probability of error is decreased, and the correlation-SNR that can be supported is decreased from 18 to 8.5 dB.

Similar codes are constructed on the 16-level quantizer using a rate-2/3 (instead of rate-2/4) convolutional code with memoryless partitioning of the alphabet into two cosets as shown in Figure 2.45(b). The probability of decoding error is shown in Figure 2.46(b). The operating range of the 16-level quantizer has been increased (from a correlation-SNR of \( \geq 24 \) dB when \( R = 1 \) bit/sample) to \( \geq 15 \) dB when \( R = 2 \) bits/sample corresponding to \( P_e \leq 10^{-4} \).
Appendix 2.1 Maximum a posteriori detection

The BCJR algorithm

The maximum likelihood decoder defined by Equation (2.20) minimizes the probability that the whole detected sequence is in error. In an alternative concept presented in this appendix we will be interested in minimizing the symbol error probability. The starting point is the average symbol a posteriori probability [18, 4–17]:

$$\text{APP} = \mathbb{E} \{ p(x_k/y) \}$$  \hspace{1cm} (A2.1)

that should be maximized. In other words, the detector will decide in favor of the symbol for which the probability of correct detection is maximized. For simplicity $y$ will now be replaced by $y$. For binary transmission, this means finding out which of the two probabilities $P(x_k = 0 \mid y)$ and $P(x_k = 1 \mid y)$ is larger. For this we have to compare:

$$\Lambda_k = \frac{P(x_k = 1 \mid y)}{P(x_k = 0 \mid y)}$$
Figure 2.44 Probability of error for $R = 1$ bit/sample, trellis-based quantization and coset construction. (a) $R_s = 2$ bits/sample, $S$ is TCSQ based on the 8-level quantizer. (b) $R_s = 3$ bits/sample, $S$ is TCSQ based on the 16-level quantizer.

Figure 2.45 Codes obtained using the same convolutional code for memoryless quantization and trellis-based coset construction for $R = 2$ bits/sample. (a) 4-state trellis (of $C$) on the 8-level quantizer. (b) 4-state trellis (of $C$) on the 16-level quantizer.

with a unit threshold. The transmitted symbol $x_k$ is associated with one or more branches of the trellis stage at time $k$, and each one of these branches can be characterized by the pair of states, say $(\delta_k, \delta_{k+1})$, that it joins. Thus, we can write:

$$\Lambda_k = \frac{\sum_{(\delta_k, \delta_{k+1}) \mid \mathbf{y} = 1} P(y, \delta_k, \delta_{k+1})}{\sum_{(\delta_k, \delta_{k+1})} P(y, \delta_k, \delta_{k+1})}$$
we may write

\[ y = (y_k^-, y_k, y_k^+) \]

which results in:

\[
p(y, \delta_k, \delta_{k+1}) = p(y_k^-, y_k, y_k^+, \delta_k, \delta_{k+1})
\]

\[
= p(y_k^-, y_k, \delta_k, \delta_{k+1}) p(y_k^+ | y_k^-, y_k, \delta_k, \delta_{k+1})
\]

\[
= p(y_k^+, \delta_k) p(y_k, \delta_{k+1} | y_k^-, \delta_k) p(y_k^+ | y_k^-, y_k, \delta_k, \delta_{k+1})
\]

Due to the dependences among observed variables and trellis states, reflected by the trellis structure or, equivalently, by the Markov chain property of the trellis states, \( y_k^- \) depends on \( \delta_k, \delta_{k+1}, y_k^- \), and \( y_k \) only through \( \delta_k \), and similarly, the pair \( y_k, \delta_{k+1} \) depends on \( \delta_k, y_k^- \), only through \( \delta_k \). Thus, by defining the functions:

\[
\alpha_k(\delta_k) \equiv p(y_k^-, \delta_k)
\]

\[
\beta_{k+1}(\delta_{k+1}) \equiv p(y_k^+ | \delta_{k+1})
\]

\[
\gamma_{k,k+1}(\delta_k, \delta_{k+1}) \equiv p(y_k, \delta_{k+1} | \delta_k) = p(y_k | \delta_k, \delta_{k+1}) p(\delta_{k+1} | \delta_k)
\]

we may write

\[
p(y, \delta_k, \delta_{k+1}) = \alpha_k(\delta_k) \gamma_{k,k+1}(\delta_k, \delta_{k+1}) \beta_{k+1}(\delta_{k+1})
\]

So, the \textit{a posteriori} probability ratio can be rewritten in the form:

\[
\Lambda_k = \frac{\sum_{\delta_k, \delta_{k+1}, y_k} \alpha_k(\delta_k) \gamma_{k,k+1}(\delta_k, \delta_{k+1}) \beta_{k+1}(\delta_{k+1})}{\sum_{\delta_k, \delta_{k+1}, y_k=0} \alpha_k(\delta_k) \gamma_{k,k+1}(\delta_k, \delta_{k+1}) \beta_{k+1}(\delta_{k+1})} \quad (A2.2)
\]
Finally, we now describe how the functions $\alpha_k(\delta_k)$ and $\beta_k+1(\delta_{k+1})$ can be evaluated recursively. We represent the forward recursion as:

$$\alpha_{k+1}(\delta_{k+1}) = p(y^+_{k+1}, \delta_{k+1})$$

$$= p(y^+_{k}, y_k, \delta_k, \delta_{k+1})$$

$$= \sum_{\delta_k} p(y^+_{k}, \delta_k) p(y_k, \delta_{k+1}|\delta_k)$$

$$= \sum_{\delta_k} \alpha_k(\delta_k) \gamma_{k,k+1}(\delta_k, \delta_{k+1})$$

with the initial condition $\alpha_0(s_1) = 1$ ($s_1$ denotes the initial state of the trellis) and the backward recursion as:

$$\beta_k(\delta_k) = p(y^-_{k-1} | \delta_k)$$

$$= \sum_{\delta_{k+1}} p(y_k, y^-_{k+1}, \delta_{k+1} | \delta_k)$$

$$= \sum_{\delta_{k+1}} p(y_k, \delta_{k+1} | \delta_k) p(y^-_{k+1} | \delta_{k+1})$$

$$= \sum_{\delta_{k+1}} \gamma_{k,k+1}(\delta_k, \delta_{k+1}) \beta_{k+1}(\delta_{k+1})$$

with the final value $\beta_K(s_K) = 1$. The combination of the latter two recursions with Equation (A2.2) forms the BCJR algorithm, named after the authors who first derived it, Bahl, Cocke, Jelinek and Raviv [27]. Roughly speaking, we can state that the complexity of the BCJR algorithm is about three times that of the Viterbi algorithm.

REFERENCES

REFERENCES


ADAPTIVE CODING


3

Adaptive and Reconfigurable Modulation

3.1 CODED MODULATION

In general we can use an \(M = 2^b\) point constellation to transmit \(b\) bits of information. An example for \(b = 2\) is shown in Figure 3.1(a). For this example the output symbol rate is \(R_s = R_b/2\). If we use coding, for example a rate 2/3 convolutional encoder, and the same constellation as shown in Figure 3.1(b), the output symbol rate and the bandwidth required will be now higher, \(R_s = (3/4)R_b\).

The third option is shown in Figure 3.1(c). Instead of 4PSK, 8PSK (8 points constellation) is used to transmit the encoded bits and the output symbol rate now remains the same. Because there are only \(2^2 = 4\) possible code words and \(2^3 = 8\) available constellation points, a proper choice of constellation points used in adjacent symbol intervals provides a way to encode the signal. This subset of signal trajectories, generated in \(K\) symbol intervals will again be referred to as a trellis in Euclidean space and the modulation is referred to as Trellis Coded Modulation (TCM) [1–38].

The above example illustrates a need to further elaborate the efficiency of reconfiguration in such a way as to explicitly incorporate constraints imposed by the limited available bandwidth. For these purposes, let us represent the coding gain \(g_{12}\) as the gain in energy per bit per noise density:

\[
g_{12} = \frac{\Delta E}{N_0} = \frac{\Delta (PT) / N_0}{10^g(dB)}
\]

If there is no bandwidth limitation, the coding gain may be used in a number of ways.

1. Operate with reduced power and save the battery life.
2. Keep the same transmit power and data rate and increase the coverage of the network.
3. Increase bit rate (reduce the bit interval).

If the bandwidth is fixed and the coding gain is not available, we may have to reduce the data rate in order to maintain the required \(E_b/N_0\) for a specified QoS. This suggests that the reconfiguration efficiency, defined by Equation (2.18), be further modified as follows:

\[
E_{\text{ff}} = \frac{10^{g_{12}(dB)/10} b_e}{D_t} = \left( \frac{k_{02}}{k_{01}} \right) \frac{10^{g_{12}(dB)/10}}{D_t}
\]

(3.1)
ADAPTIVE AND RECONFIGURABLE MODULATION

Figure 3.1 Three digital communication schemes transmitting two bits every $T$ seconds.
(a) Uncoded transmission with 4PSK; (b) 4PSK with a rate $2/3$ encoder and bandwidth expansion; (c) 8PSK with a rate $2/3$ encoder and no bandwidth expansion.

where $b_r = k_{02}/k_{01}$ is the relative change in the number of bits per symbol for the same symbol period $T$. One should notice, that in Chapter 2, coding gain was defined by taking into account this effect through the coding rate $R_c$. In the next chapter, Equation (3.1) will be further modified to replace $b_r$ by the relative change in the system capacity. This way we remove the portion of the gain due to bit rate reduction and take into account only those contributions due to increased efficiency of the decoding algorithm.

3.1.1 Euclidean distance

The demodulator will decide in favor of the trajectory in the trellis which is closest to the received signal (minimum distance from the received trajectory). This is characterized by the Euclidean distance, defined as:

$$\delta^2 = \sum_{i=0}^{K-1} |r_i - x_i|^2$$  \hspace{1cm} (3.2)

where $r_i$ is the received signal and $x_i$ the possible transmitted signal. In other words, the Euclidean distance is minimized by taking $x_i = \hat{x}_i$, $i = 0, \ldots, K - 1$, if the received sequence is closer to $\hat{x}_i, \ldots, \hat{x}_{K-1}$ than to any other allowable signal sequence.

By increasing the constellation size $M'$ to $M > M'$, and selecting $M'^K$ sequences as a subset of $S^K$, we can have sequences which are less tightly packed and hence increase the minimum distance among them. We obtain a minimum distance, $\delta_{\text{free}}$, between any two sequences, which turns out to be
greater than the minimum distance, $\delta_{\text{min}}$, between signals in $S'$. Hence, use of maximum likelihood sequence detection will yield a ‘distance gain’ of a factor of $\delta_{\text{free}}^2/\delta_{\text{min}}^2$.

The free distance of a TCM scheme is the minimum Euclidean distance between two paths forming an error event.

### 3.1.2 Examples of TCM schemes

In order to analyze Equation (3.1) in more detail, let us assume transmission with two bits per symbol. For such transmission a 4PSK modulation ($M' = 4$) would be enough. We can expand the constellation to $M = 8$, as shown in Figure 3.2, and use the trellis with $S = 2$ states, as shown in Figure 3.3.

The asymptotic coding gain of a TCM scheme is defined as:

$$\gamma = \frac{\delta_{\text{free}}^2 (M)/\varepsilon}{\delta_{\text{min}}^2 (M')/\varepsilon'}$$

For PSK signals, $M' = 4$ and $\delta_{\text{min}}^2/\varepsilon' = 2$. For a TCM scheme based on the 8PSK constellation whose signals we label $\{0, 1, 2, \ldots, 7\}$, as shown in Figure 3.2, we have:

$$\varepsilon' = \frac{\sqrt{2}}{4 \sin^2 \pi/8}$$

![Figure 3.2](image-url)  
Figure 3.2 $M = 8$ point constellation.

![Figure 3.3](image-url)  
Figure 3.3 A TCM scheme based on a two-state trellis, $M' = 4$ and $M = 8$. 

3.1.2.1 Two-state trellis

If the encoder is in state $S_1$, the subconstellation $\{0, 2, 4, 6\}$ is used. In state $S_2$, constellation $\{1, 3, 5, 7\}$ is used instead, as shown in Figure 3.3. The free distance of this TCM scheme is the smallest among the distances between signals associated with parallel transitions (error events of length 1) and the distances associated with a pair of paths in the trellis that originate from a common node and merge into a single node at a later time (error events of length greater than 1). The pair of paths yielding the free distance is shown in Figure 3.3. With $\delta(i, j)$ denoting the Euclidean distance between signals $i$ and $j$, we have the following:

$$\frac{\delta_{\text{free}}^2}{\varepsilon} = \frac{1}{\varepsilon} [\delta^2(0, 2) + \delta^2(0, 1)] = 2 + 4 \sin^2 \frac{\pi}{8} = 2.586$$

The asymptotic coding gain over 4PSK is:

$$\gamma = \frac{2.586}{2} = 1.293 \Rightarrow 1.1 \text{ dB} \quad (3.3)$$

In this case, $\Delta k_0 = 0$ and $\Delta v = 1$ so that reconfiguration efficiency, defined by Equation (3.1), gives:

$$E_{ff} = \left(\frac{k_{02}}{k_{01}}\right)^2 \gamma 2^{-(\Delta k_0 + v)} = 1.293/2 = 0.646 \quad (3.4)$$

For a given symbol error probability, the bit error probability (BER) will depend on the mapping of the source bits onto the signals in the modulator’s constellation (see Figure 3.3). To minimize the BER, this mapping should be chosen in such a way that, whenever a symbol error occurs, the signal erroneously chosen by the demodulator differs from the transmitted one by the least number of bits. For high signal to noise ratios, most of the errors occur by mistaking a signal for one of its nearest neighbors. So, a reasonable choice is a mapping where neighboring signal points in the constellation correspond to binary sequences that differ in only one digit. This is called Gray mapping. In this case, the bit and symbol error probabilities are related as $P_s(e)/b = P_b(e)$. For the evaluation of the symbol error probability the reader is referred to the classical references [39–42].

3.1.2.2 Four-state trellis

In this case the trellis is as given in Figure 3.4.

![Figure 3.4 A TCM scheme based on a four-state trellis, $M' = 4$ and $M = 8$.](Image)
Figure 3.5 A TCM scheme based on an eight-state trellis, $M’ = 4$ and $M = 8$.

We associate the constellation $\{0, 2, 4, 6\}$ with states $S_1$ and $S_3$, and $\{1, 3, 5, 7\}$ with $S_2$ and $S_4$. In this case, the error event leading to $\delta_{\text{free}}$ has length 1 (a parallel transition).

$$\delta_{\text{free}} = \delta_{(0, 4)} = 4$$

Thus, $\gamma = \frac{4}{2} = 2 \Rightarrow 3 \text{ dB}$

In this case, $\Delta \nu = 2$ and Equation (3.4) gives $E_{ff} = 2/4 = 0.5$, which is lower than Equation (3.4). This means that effort invested is larger than the gain obtained.

### 3.1.2.3 Eight-state trellis

For the case of eight states, the trellis is shown in Figure 3.5. The four symbols associated with the branches emanating from each node are used as node labels. The first symbol in each node label is associated with the uppermost transition from the node, the second symbol with the transition immediately below it, etc.

The coding gain is calculated as:

$$\delta_{\text{free}} = \frac{1}{\epsilon} [\delta^2(0, 6) + \delta^2(0, 7) + \delta^2(0, 6)] = 2 + 4 \sin^2 \frac{\pi}{8} + 2 = 4.586$$

Thus, $\gamma = \frac{4.586}{2} = 2.293 \Rightarrow 3.6 \text{ dB}$

In this case, $\Delta \nu = 3$ and Equation (3.4) gives $E_{ff} = 2.293/8 = 0.286$, which is lower than in the previous case of the four-state trellis. This means again that effort invested is larger than the gain obtained.

### 3.1.2.4 QAM 3 bits per symbol

In this case, the trellis is as given in Figure 3.6 and the signal constellation as in Figure 3.7. In this case, we have two subsets of points $\{0, 2, 5, 7, 8, 10, 13, 15\}$ and $\{1, 3, 4, 6, 9, 11, 12, 14\}$. For the basic 8QPSK constellation we have $\delta_{\text{min}}^2 / \epsilon = 0.8$, and coding gain can be represented as:

$$\delta_{\text{free}} = \frac{1}{\epsilon} [\delta^2(10, 13) + \delta^2(0, 1) + \delta^2(0, 5)] = \frac{1}{\epsilon} [0.8\epsilon + 0.4\epsilon + 0.8\epsilon] = 2$$

Thus, $\gamma = \frac{2}{0.8} = 2.5 \Rightarrow 3.98 \text{ dB}$
ADAPTIVE AND RECONFIGURABLE MODULATION

Figure 3.6 A TCM scheme based on an eight-state trellis, $M' = 8$ and $M = 16$.

Figure 3.7 The 8 QAM constellation \{0, 2, 5, 7, 8, 10, 13, 15\} and the 16 QAM constellation \{0, 1, \ldots , 15\}.

In this case, $\Delta k_0 = 0$ and $\Delta v = 3$ so that reconfiguration efficiency, defined by Equation (3.1), gives:

$$E_{\text{ff}} = 2.5 \times 2^{-(0+3)} = 0.312$$

which is higher than in the previous example. Additional results for free distances for a number of different schemes are given in Figure 3.8. These results can be used directly to evaluate the reconfiguration efficiency.

3.1.3 Set partitioning

The $M$-ary constellation is successively partitioned into 2, 4, 8, \ldots , subsets with size $M/2$, $M/4$, $M/8$, \ldots , having progressively larger minimum Euclidean distances $\delta_{\text{min}}^{(1)}$, $\delta_{\text{min}}^{(2)}$, $\delta_{\text{min}}^{(3)}$, \ldots as shown in Figure 3.9 and Figure 3.10.

Then, in accordance with Ungerboeck’s rules, the following steps are taken:

1. Members of the same partition with the largest distance are assigned to parallel transitions.

2. Members of the next largest partition are assigned to ‘adjacent’ transitions, i.e. transitions stemming from, or merging into, the same node.
Figure 3.8 Free distance versus bandwidth efficiency of selected TCM schemes based on two-dimensional modulations. (Adapted from [2]) PSK and QAM © 1987, IEEE.

Figure 3.9 Set partition of an 8PSK constellation.
Figure 3.10 Set partition of a 16 QAM constellation.

Figure 3.11 Representation of TCM.

### 3.1.4 Representation of TCM

A TCM encoder can be represented as a convolutional encoder encoding a block of \( m \) input bits \( b_i = (b_i^{(1)}, b_i^{(2)}, b_i^{(3)}, \ldots, b_i^{(m)}) \) into a block of \( m + 1 \) output bits \( c_i = (c_i^{(1)}, c_i^{(2)}, c_i^{(3)}, \ldots, c_i^{(m+1)}) \), followed by a memoryless mapper into points of the constellation of size \( M = 2^{m+1} \) (Figure 3.11). In the case when there are parallel transitions, not all bits are encoded, which is represented explicitly in Figure 3.12.

Uncoded digits cause parallel transitions; a branch in the trellis diagram of the code is now associated with \( 2^{m-\tilde{m}} \) signals. An example for \( m = 2 \) and \( \tilde{m} = 1 \) is shown in Figure 3.13. The trellis nodes are connected by parallel transitions associated with two signals each. The trellis has four states, as does the rate 1/2 convolutional encoder, and its structure is determined by the latter.

### 3.1.5 TCM with multidimensional constellation

In general, we can use \( m \) channels for transmission and generate an \( m \)-dimensional trellis for the overall signal representation.
Figure 3.12 A TCM encoder where the bits that are left uncoded are shown explicitly.

Figure 3.13 A TCM encoder with \( m = 2, \tilde{m} = 1 \) and the corresponding trellis.

Figure 3.14 A two-state TCM scheme based on a \( 2 \times 4 \)PSK constellation. The error event providing the free Euclidean distance is also shown.

As an example of a TCM scheme based on multidimensional signals, consider the four-dimensional constellation obtained by pairing \( (m = 2) \) 4PSK signals. This is denoted as \( 2 \times 4 \)PSK. With the signal labeling of Figure 3.14, the \( 4^2 = 16 \) four-dimensional signals are:

\[
\{00, 01, 02, 03, 10, 11, 12, 13, 20, 21, 22, 23, 30, 31, 32, 33\}
\]

This constellation achieves the same minimum squared distance as two-dimensional 4PSK,

\[
\delta_{\text{min}}^2 = \delta^2(00, 01) = \delta^2(0, 1) = 2
\]
The following subconstellation has eight signals and a minimum squared distance four:

\[ S = \{00, 02, 11, 13, 20, 22, 31, 33\} \]

With \( S \) partitioned into the four subsets:

\[ \{00, 22\}, \{20, 02\}, \{13, 31\}, \{11, 33\} \]

the choice of a two-state trellis provides the TCM scheme shown in Figure 3.14. This has a squared free distance of 8.

If \( m \) channels are used independently, the overall data rate would be \( k_0 = mk_0 \) and the complexity would be \( m \) times the complexity of the demodulation per trellis. So the normalized complexity per bit would be:

\[ D_1 = \frac{m2^{k_0+v}}{mk_0} = \frac{2^{k_0+v}}{k_0} \]

For \( m \) dimensional trellises, only \( k_0 = k_0 \) bits are transmitted:

\[ D_2 = \frac{m2^{k_0+v}}{k_0} \quad \text{and} \quad D_r = \frac{D_2}{D_1} = m \]

Now we have:

\[ E_{ff} = g_{12} \left( \frac{k_0}{mk_0} \right) \left( \frac{1}{m} \right) = \frac{g_{12}}{m^2} \]  

(3.9)

For the previous example, \( g_{12} = \gamma = 8/2 = 4 \) and \( m^2 = 4 \) so that \( E_{ff} = 1 \).

For this reason, in the next chapter we will discuss multidimensional constellations obtained by using multiple antennas.

### 3.2 ADAPTIVE CODED MODULATION FOR FADING CHANNELS

In this section we describe the system which uses reconfiguration to improve performance in time varying fading channels [43–66]. Basically, for a better signal to noise ratio \( \hat{\gamma} \), estimated at the receiver side, the higher constellation \( M \) is used at the receiver, as shown in Figure 3.15.

Let \( \tau_j \) be the average time that the adaptive modulation scheme continuously uses the constellation \( M_j \). Since the constellation size is adapted to an estimate of the channel fade level (instantaneous signal to noise ratio), several symbol times may be required to obtain a good estimate. In addition, hardware and pulse shaping considerations generally dictate that the constellation size must remain constant over tens to hundreds of symbols. This results in the requirement that \( \tau_j \gg T \forall j \), where \( T \) is the symbol time.

![Figure 3.15 Block diagram of a system using adaptive modulation.](image-url)
Figure 3.16 General structure for adaptive coded modulation.

Since each constellation $M_j$ is associated with a range of fading values called the fading region $R_j$, $\tau_j$ is the average time that the fading stays within the region $R_j$. The value of $\tau_j$ is inversely proportional to the channel Doppler and also depends on the number and characteristics of the different fade regions. In Rayleigh fading with an average SNR of 20 dB and a channel Doppler of 100 Hz, $\tau_j$ ranges from 0.7–3.9 ms, and thus, for a symbol rate of 100 ksymbols/s, the signal constellation remains constant over tens to hundreds of symbols. Similar results hold at other SNR values.

The flat fading assumption implies that the signal bandwidth $B$ is much less than the channel coherence bandwidth $B_c = 1/T_M$, where $T_M$ is the root mean square (rms) delay spread of the channel. For Nyquist pulses $B = 1/T$, so flat fading occurs when $T \gg T_M$. Combining $T \gg T_M$ and $\tau_j \gg T$ we get $\tau_j \gg T \gg T_M$.

Wireless channels have rms delay spreads less than 30 $\mu$s in outdoor urban areas and less than around 1 $\mu$s in indoor environments. Taking the minimum $\tau_j = 0.8$ ms, rates of the order of tens of ksymbols/s in outdoor channels and hundreds of ksymbols/s in indoor channels are practical for an adaptive scheme.

Modulation uses ideal Nyquist data pulses with a fixed symbol period $T = 1/B$. We also restrict $M(\gamma)$ to square $M$-QAM constellations of size $M_0 = 0$ and $M_j = 2^{2(j-1)}$, $j = 2, \ldots, J$. Thus, at each symbol time a constellation from the set $\{M_j : j = 0, 2, \ldots, J\}$ is used – the choice of constellation depends on the fade level $\gamma$ over that symbol time. Choosing the $M_0$ constellation corresponds to no data transmission. Since the constellation set is finite, there will be a range of $\gamma$ values over which a particular constellation $M_j$ is used. Within that range the power must also be adapted to maintain the desired distance $d_0$ between the trajectories in the trellis. Thus, for each constellation $M_j$, the power adaptation $S_j(\gamma)$ associated with that constellation is a continuous function of $\gamma$. The basic premise for using adaptive modulation is to keep these distances constant by varying the size $M(\gamma)$, transmit power $S(\gamma)$, and/or symbol time $T(\gamma)$ of the transmitted signal constellation relative to $\gamma$, subject to an average transmit power constraint $\bar{S}$ on $S(\gamma)$. By maintaining $d_{\min}(t) = d_{\min}$ constant, the adaptive coded modulation exhibits the same coding gain as coded modulation designed for an AWGN channel with minimum code word distance $d_{\min}$. The detailed system block diagram is given in Figure 3.16.

The details about the coset codes used in the scheme can be found in [67, 68].

### 3.2.1 Maintaining a fixed distance

Define

$$M(\gamma) = \frac{\gamma}{\gamma_k} \quad (3.10)$$
where \( \gamma_k^* \geq 0 \) is a parameter which is optimized relative to the fade distribution to maximize spectral efficiency. For \( \gamma < \gamma_k^* M_2 \) the channel is not used. The constellation size \( M_j \) used for a given \( \gamma \geq \gamma_k^* M_2 \) is the largest \( M_j \) for which \( M_j \leq M(\gamma) \). The range of \( \gamma \) values for which \( M(\gamma) = M_j \) is thus \( M_j \leq \frac{\gamma}{\gamma_k^*} M_{j+1} \), with \( M_{j+1} \approx \infty \). We call this range of fading values the fading region \( R_j \) associated with constellation \( M_j \).

### 3.2.2 Information rate

For each \( \gamma \), one redundant bit per symbol is used for the channel coding, so the number of information bits per symbol is \( \log_2 M(\gamma) - 1 \). Thus, the information rate for a single \( \gamma \) is \( R_\gamma = \left\lfloor \log_2 M(\gamma) - 1 \right\rfloor / T_b/s \), and the corresponding spectral efficiency is \( R_\gamma / B = \log_2 M(\gamma) - 1 \), since we use Nyquist pulses (\( B = 1/T \)).

Spectral efficiency is obtained by averaging the spectral efficiency for each \( \gamma \) weighted by its probability:

\[
\frac{R}{B} = \sum_{j=2}^{J} (\log_2 M_j - 1)p(M_j \leq \frac{\gamma}{\gamma_k} < M_{j+1})
\]

(3.11)

where \( \gamma_k \) is picked to maximize Equation (3.11), subject to the average power constraint:

\[
\sum_{j=2}^{J} \int_{\gamma_k M_j}^{\gamma_k M_{j+1}} S_j(\gamma)p(\gamma) \, d\gamma = \bar{S}
\]

(3.12)

Similarly, the reconfiguration efficiency will now vary in time. Equation (3.1) can now be represented as:

\[
E_{ff}(\gamma) = \left( \frac{k_0(\gamma)}{k_0} \right)^2 2^{-\Delta k(\gamma) + k(\gamma)} g_{12}(\gamma)
\]

(3.13)

and the average efficiency is obtained as:

\[
E_{ff} = \int E_{ff}(\gamma)p(\gamma) \, d\gamma
\]

(3.14)

Table 3.1 presents results of simulations for adaptive and non-adaptive systems with \( M_j \in [0, 4, 16, 64, 256] \). One can see from the table that considerable SNR gains can be achieved with adaptive schemes.

<table>
<thead>
<tr>
<th>Spectral efficiency (bps/Hz)</th>
<th>BER</th>
<th>Trellis states</th>
<th>Average SNR (dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>10^-3</td>
<td>4</td>
<td>10.5</td>
</tr>
<tr>
<td></td>
<td></td>
<td>128</td>
<td>9.0</td>
</tr>
<tr>
<td></td>
<td>10^-6</td>
<td>4</td>
<td>13</td>
</tr>
<tr>
<td></td>
<td></td>
<td>128</td>
<td>11.3</td>
</tr>
<tr>
<td>3</td>
<td>10^-3</td>
<td>8</td>
<td>14.2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>128</td>
<td>13.1</td>
</tr>
<tr>
<td></td>
<td>10^-6</td>
<td>8</td>
<td>16.4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>128</td>
<td>15.3</td>
</tr>
</tbody>
</table>
REFERENCES


4
Space–Time Coding

4.1 DIVERSITY GAIN

In Chapters 2 and 3 the coding gain was used as a performance measure. Before we go into a detailed discussion on space–time coding, diversity gain will be defined and discussed. In Chapter 3 we briefly discussed the multidimensional trellis and pointed out the relatively high efficiency of such a concept. By using an additional dimension we provide a diversity effect which results in a considerable gain. These new dimensions may be additional frequency bands, different time slots or delayed replicas of the signal, or different antennas, resulting in frequency, time or space diversity respectively. In this section we elaborate the concept of diversity gain by using space diversity. In the subsequent sections of the chapter we will discuss space–time coding, where the concept of coding and diversity gain is combined into an integral performance measure.

A classical space diversity set-up with one transmitting and two receiving antennas is shown in Figure 4.1. The antenna diversity is realized in the receiver, hence the name receiver diversity. The following notation is used in the figure: the channel between the transmit antenna and the receiver antenna zero is denoted \( h_0 \); that between the transmit antenna and the receiver antenna one is \( h_1 \), where:

\[
\begin{align*}
    h_0 &= \alpha_0 e^{j\theta_0} \\
    h_1 &= \alpha_1 e^{j\theta_1}
\end{align*}
\]  

(4.1)

The resulting received baseband signals at antennas zero and one are:

\[
\begin{align*}
    r_0 &= h_0 s_0 + n_0 \\
    r_1 &= h_1 s_0 + n_1
\end{align*}
\]  

(4.2)

where \( n_0 \) and \( n_1 \) represent complex noise and interference.

In accordance with the discussion in Chapter 2, the ML decoder will choose signal \( s_i \) if and only if:

\[
d^2(r_0, h_0 s_i) + d^2(r_1, h_1 s_i) \leq d^2(r_0, h_0 s_k) + d^2(r_1, h_1 s_k) \quad \forall i \neq k
\]  

(4.3)

where \( d^2(x, y) \) is the squared Euclidian distance between \( x \) and \( y \):

\[
d^2(x, y) = (x - y)(x^* - y^*)
\]  

(4.4)
A two branch Maximum Ratio Receiver Combiner (MRRC) would first create the signal:

\[
\tilde{s}_0 = h_0^* r_0 + h_1^* r_1 = h_0^* (h_0 s_0 + n_0) + h_1^* (h_1 s_0 + n_1) = \left( \alpha_0^2 + \alpha_1^2 \right) s_0 + h_0^* n_0 + h_1^* n_1
\]

with an equivalent distance defined as:

\[
d_i^2 = (\tilde{s}_0 - \beta s_i)(\tilde{s}_0 - \beta s_i)^*; \quad \beta = \alpha_0^2 + \alpha_1^2
\]

and the ML detector would choose \( s_i \) if:

\[
\left( \alpha_0^2 + \alpha_1^2 \right) |s_i|^2 - \tilde{s}_0 s_i^* - \tilde{s}_0 s_i \leq \left( \alpha_0^2 + \alpha_1^2 \right) |s_k|^2 - \tilde{s}_0 s_k^* - \tilde{s}_0 s_k \quad \forall i \neq k
\]

If Equation (4.6) is used in (4.7), the latter can also be represented in the following form. Choose \( s_i \) if:

\[
\left( \alpha_0^2 + \alpha_1^2 - 1 \right) |s_i|^2 + d^2 (\tilde{s}_0, s_i) \leq \left( \alpha_0^2 + \alpha_1^2 - 1 \right) |s_k|^2 + d^2 (\tilde{s}_0, s_k) \quad \forall i \neq k
\]

For PSK signals (equal energy constellations):

\[
|s_i|^2 = |s_k|^2 = E_s \quad \forall i, k
\]

where \( E_s \) is the energy of the signal. So, for PSK signals, the decision rule (4.8) may be simplified to:

Choose \( s_i \) if

\[
d^2 (\tilde{s}_0, s_i) \leq d^2 (\tilde{s}_0, s_k) \quad \forall i \neq k
\]

### 4.1.1 Two-branch transmit diversity scheme with one receiver

Using two antennas in a mobile receiver might prove difficult in practice. For this reason, in this subsection we demonstrate how the same effect and diversity gain may be obtained by using two antennas at the transmitter and only one antenna at the receiver [1]. Implementing two antennas at the base station in mobile communication networks is a much simpler task. The system block diagram is shown in Figure 4.2.
Figure 4.2 Two branch transmit diversity scheme with one receiver [1] © 1998, IEEE.

<table>
<thead>
<tr>
<th>Table 4.1 Space–time coding rules</th>
</tr>
</thead>
<tbody>
<tr>
<td>Antenna 0</td>
</tr>
<tr>
<td>time $t$</td>
</tr>
<tr>
<td>time $t + T$</td>
</tr>
</tbody>
</table>

**4.1.1.1 The encoding and transmission sequence**

The encoding is done in space and time (space–time (ST) coding) as defined in Table 4.1. The encoding may also be done in space and frequency. Instead of two adjacent symbol periods, two adjacent frequency subbands may be used (space–frequency coding).

**4.1.1.2 The received signal**

Assuming that the fading is constant across two consecutive symbols we have:

\[
\begin{align*}
    h_0(t) &= h_0(t + T) = h_0 = \alpha_0 e^{j\theta_0} \\
    h_1(t) &= h_1(t + T) = h_1 = \alpha_1 e^{j\theta_1}
\end{align*}
\]

(4.11)

where $T$ is the symbol interval. The received signals in two adjacent symbol intervals can be represented as:

\[
\begin{align*}
    r_0 &= r(t) = h_0 s_0 + h_1 s_1 + n_0 \\
    r_1 &= r(t + T) = -h_0 s_1^* + h_1 s_0^* + n_1
\end{align*}
\]

(4.12)
4.1.1.3 The combining scheme

The signals received in two adjacent symbol intervals are combined as follows:

\[
\tilde{s}_0 = h^*_0 r_0 + h^*_1 r_1^* \\
\tilde{s}_1 = h^*_1 r_0 + h^*_0 r_1^* \\
\tilde{s}_0 = (\alpha^2_0 + \alpha^2_1)s_0 + h^*_0 n_0 + h^*_1 n_1^* \\
\tilde{s}_1 = (\alpha^2_0 + \alpha^2_1)s_1 - h^*_0 n_1^* + h^*_1 n_0
\] (4.13)

\[
\tilde{s}_0 = (\alpha^2_0 + \alpha^2_1)s_0 + h^*_0 n_0 + h^*_1 n_1^* \\
\tilde{s}_1 = (\alpha^2_0 + \alpha^2_1)s_1 - h^*_0 n_1^* + h^*_1 n_0
\] (4.14)

4.1.1.4 ML decision rule

For each of the signals \(s_0\) and \(s_1\) the decision rule is the same, Inequality (4.3) or (4.10). The resulting combined signals are equivalent to that obtained from two branch MRRC given by Equation (4.5). The only difference is phase rotations on the noise components which do not degrade the effective SNR. The resulting diversity order from the new branch transmit diversity scheme with one receiver is equal to that of two branch MRRC. This is known as Alamouti code [1].

4.1.2 Two transmitters and \(M\) receivers

There may be applications where a higher order of diversity is needed and multiple receive antennas are also feasible. In such cases, it is possible to provide a diversity order of \(2M\), with two transmit and \(M\) receive antennas. Figure 4.3 represents an illustration of a special case of two transmit and two receive antennas. The generalization to \(N\) transmit and \(M\) receive antennas will be further elaborated in the subsequent sections. The notation is explained in Tables 4.2 and 4.3.

The space–time coding rule is given again by Table 4.1

<table>
<thead>
<tr>
<th>Table 4.2 Channel notation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
<tr>
<td>(tr) antenna 0</td>
</tr>
<tr>
<td>(tr) antenna 1</td>
</tr>
</tbody>
</table>
Table 4.3  Signal notation

<table>
<thead>
<tr>
<th></th>
<th>rx antenna 0</th>
<th>rx antenna 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>time $t$</td>
<td>$r_0$</td>
<td>$r_2$</td>
</tr>
<tr>
<td>time $t+T$</td>
<td>$r_1$</td>
<td>$r_3$</td>
</tr>
</tbody>
</table>

4.1.2.1 The received signal

The received signals in two adjacent symbol intervals with notation given in Table 4.3 are:

\[
\begin{align*}
    r_0 &= h_0s_0 + h_1s_1 + n_0 \\
    r_1 &= -h_0s_1^* + h_1s_0^* + n_1 \\
    r_2 &= h_2s_0 + h_3s_1 + n_2 \\
    r_3 &= -h_2s_1^* + h_3s_0^* + n_3
\end{align*}
\]

(4.15)

where $n_0$, $n_1$, $n_2$, and $n_3$ are complex random variables representing receiver thermal noise and interference.

4.1.2.2 The combiner

The combiner is defined by the following rule:

\[
\begin{align*}
    \tilde{s}_0 &= h_0^*r_0 + h_1^*r_1^* + h_2^*r_2 + h_3^*r_3^* \\
    \tilde{s}_1 &= h_0^*r_0 - h_0r_1^* + h_1^*r_2 - h_2r_3^*
\end{align*}
\]

(4.16)

\[
\begin{align*}
    \tilde{s}_0 &= (\alpha_0^2 + \alpha_1^2 + \alpha_2^2 + \alpha_3^2) s_0 + h_0^*n_0 + h_1^*n_1^* + h_2^*n_2 + h_3^*n_3^* \\
    \tilde{s}_1 &= (\alpha_0^2 + \alpha_1^2 + \alpha_2^2 + \alpha_3^2) s_1 - h_0n_0^* + h_1n_1^* - h_2n_2^* + h_3n_3^*
\end{align*}
\]

(4.17)

So, the order of $2 \times 2 = 4$ diversity is achieved. For uncorrelated channel and noise, $\alpha_i^2 = \alpha^2$, and perfect channel estimation, the signal to noise ratio is given as:

\[
\frac{(4\alpha^2)^2}{4\alpha^2N_0} = \frac{4\alpha^2}{N_0}
\]

(4.18)

4.1.2.3 ML decoder

The ML decoder will be operating as follows:

For $s_0$, choose $s_i$ if:

\[
\left( \alpha_0^2 + \alpha_1^2 + \alpha_2^2 + \alpha_3^2 - 1 \right) |s_i|^2 + d^2(\tilde{s}_0, s_i) \leq \left( \alpha_0^2 + \alpha_1^2 + \alpha_2^2 + \alpha_3^2 - 1 \right) |s_i|^2 + d^2(\tilde{s}_0, s_i)
\]

(4.19)

For $s_1$, choose $s_i$ if:

\[
\left( \alpha_0^2 + \alpha_1^2 + \alpha_2^2 + \alpha_3^2 - 1 \right) |s_i|^2 + d^2(\tilde{s}_1, s_i) \leq \left( \alpha_0^2 + \alpha_1^2 + \alpha_2^2 + \alpha_3^2 - 1 \right) |s_i|^2 + d^2(\tilde{s}_1, s_i)
\]

(4.20)

4.1.2.4 The BER performance

The BER results are shown in Figure 4.4. Significant diversity gain measured in SNR improvements is evident.
4.2 SPACE–TIME CODING

In the previous section we introduced the concept of space–time coding. We are now going to look more closely at the general model and bring in more detail in the analysis of such a system. In general, we will consider a system where data is encoded by a channel code and the encoded data is split into \( n \) streams that are simultaneously transmitted using \( n \) transmit antennas. The received signal at each receive antenna is a linear superposition of the \( n \) transmitted signals plus noise.

4.2.1 The system model

The system consists of \( n \) antennas in the base station and \( m \) antennas in the mobile. Data is encoded by the channel encoder, S/P converted, and divided into \( n \) streams of data. Each stream of data is used as the input to a pulse shaper. The output of each shaper is then modulated. At each time slot with index \( l \), the output of modulator \( i \) is a signal \( c_i^l \) that is transmitted using transmit antenna (Tx antenna) \( i \) for \( 1 < i < n \).

The \( n \) signals are transmitted simultaneously, each from a different transmit antenna, and all of these signals have the same transmission period \( T \). The signal at each receive antenna is a noisy superposition of the \( n \) transmitted signals corrupted by Rayleigh or Rician fading.

Elements of the signal constellation are contracted by a factor of \( \sqrt{E_s} \) chosen so that the average energy of the constellation is 1.

At the receiver, the demodulator computes a decision statistic based on the received signals arriving at each receive antenna \( 1 < j < m \). The signal \( r_j^l \) received by antenna \( j \) at discrete time \( l \) is given by:

\[
r_j^l = \sum_{i=1}^{n} \alpha_{i,j} c_i^l \sqrt{E_s} + \eta_j^l
\]  

where the noise \( \eta_j^l \) at discrete time \( l \) is modeled as independent samples of a zero mean complex Gaussian random variable with variance \( N_0/2 \) per dimension. The coefficient \( \alpha_{i,j} \) is the path gain from transmit antenna \( i \) to receive antenna \( j \). It is assumed that these path gains are constant during a frame of \( t \) symbol intervals and vary from one frame to another (quasistatic flat fading).
4.2.2 The case of independent fade coefficients

The ML receiver will decide erroneously in favor of a signal:

\[ \mathbf{e} = e_1^i e_2^i \cdots e_1^p e_2^p \cdots e_1^n e_2^n \]

assuming that

\[ \mathbf{c} = c_1^i c_2^i \cdots c_1^p c_2^p \cdots c_1^n c_2^n \]

was transmitted, with a probability that can be approximated by:

\[ P(\mathbf{e} \rightarrow \mathbf{e}|\alpha_{i,j}, \ i = 1, 2, \ldots, n, \ j = 1, 2, \ldots, m) \leq \exp (-d^2(\mathbf{e}, \mathbf{e})E_s/4N_0) \]  

(4.22)

where \( N_0/2 \) is the noise variance per dimension and

\[ d^2(\mathbf{e}, \mathbf{e}) = \sum_{j=1}^{m} \sum_{l=1}^{t} \sum_{i=1}^{n} \alpha_{l,j} \left| c_j^l - e_j^l \right|^2 \]  

(4.23)

is the distance between the two trajectories measured in a time frame of \( t \) symbol intervals. Setting \( \Omega_j = (\alpha_{1,j}, \ldots, \alpha_{n,j}) \), we rewrite Equation (4.23) as:

\[ d^2(\mathbf{e}, \mathbf{e}) = \sum_{j=1}^{m} \Omega_j^* \mathbf{A} \Omega_j \]  

(4.24)

where \( \mathbf{x}^* \) stands for the complex conjugate of \( \mathbf{x} \). After simple manipulations, we observe that:

\[ d^2(\mathbf{c}, \mathbf{e}) = \sum_{j=1}^{m} \Omega_j^* \mathbf{A} \Omega_j \]  

(4.25)

where \( \mathbf{x}^* \) stands for the transpose conjugate, and elements of matrix \( \mathbf{A} \) are defined as \( A_{pq} = \mathbf{x}_p \cdot \mathbf{x}_q \) and \( \mathbf{x}_p = (c_1^p - e_1^p, c_2^p - e_2^p, \ldots, c_n^p - e_n^p) \) for \( 1 \leq p, q \leq n \). Thus:

\[ P(\mathbf{e} \rightarrow \mathbf{e}|\alpha_{i,j}, \ i = 1, 2, \ldots, n, \ j = 1, 2, \ldots, m) \leq \prod_{j=1}^{m} \exp (-\Omega_j^* \mathbf{A} \mathbf{e}) \Omega_j E_s/4N_0) \]  

(4.26)

where \( A_{pq} = \sum_{i=1}^{t} (c_i^p - e_i^p)(c_i^q - e_i^q) \). This can be also represented as:

\[ \mathbf{A} = \mathbf{B}^* \mathbf{B} \]

(4.27)

\[ \mathbf{B} = \{b_{ij}\} = \{e_j^p - c_j^p\} \]

Matrix \( \mathbf{B} \) is given in an explicit form later by Equation (4.48). From now on we will express \( d^2(\mathbf{c}, \mathbf{e}) \) in terms of the eigenvalues of the matrix \( \mathbf{A}(\mathbf{c}, \mathbf{e}) \) defined by \( \mathbf{V} \mathbf{A}(\mathbf{c}, \mathbf{e}) \mathbf{V}^* = \mathbf{D} = \text{diag} \{\lambda_i\} \). For details of eigenvalue decomposition, see Appendix 5.1.

If we define vector \((\beta_1,j, \ldots, \beta_{n,j}) = \Omega_j \mathbf{V}^* \), then we have:

\[ \Omega_j \mathbf{A}(\mathbf{c}, \mathbf{e}) \Omega_j^* = \sum_{i=1}^{n} \lambda_i |b_{ij}|^2 \]  

(4.28)

At this point recall that \( \alpha_{i,j} \) are samples of a complex Gaussian random variable with mean \( \overline{\alpha}_{ij} \).

Let

\[ \mathbf{K}^j = (\overline{\alpha}_{1,j}, \overline{\alpha}_{2,j}, \overline{\alpha}_{3,j}, \ldots, \overline{\alpha}_{n,j}) \]

(4.29)

Since \( \mathbf{V} \) is unitary, \( \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n\} \) is an orthonormal basis of \( \mathbb{C}^n \) and \( \beta_{i,j} \) are independent complex Gaussian random variables with variance 0.5 per dimension and mean \( \mathbf{K}^j \cdot \mathbf{v}_i \). If \( K_{i,j} = |\beta_{i,j}|^2 = |\mathbf{K}^j \cdot \mathbf{v}_i|^2 \), then \( |\beta_{i,j}| \) are independent Rician distributions with pdf:

\[ p(|\beta_{i,j}|) = 2|\beta_{i,j}| \exp (-|\beta_{i,j}|^2 - K_{i,j}) I_0(2|\beta_{i,j}| \sqrt{K_{i,j}}) \]  

(4.30)
for $|\beta_{i,j}| \geq 0$, where $I_0(\cdot)$ is the zero order modified Bessel function of the first kind. To compute an upper bound on the average probability of error, we simply average:

$$\prod_{j=1}^{m} \exp \left( -(E_s/4N_0) \sum_{i=1}^{n} \lambda_i |\beta_{i,j}|^2 \right)$$  \hspace{1cm} (4.30)

with respect to independent Rician distributions of $|\beta_{i,j}|$ to arrive at:

$$P(c \rightarrow e) \leq \prod_{j=1}^{m} \left( \prod_{i=1}^{n} \frac{1}{1 + E_s/4N_0 \lambda_i} \exp \left( - \frac{K_{i,j} E_s}{4N_0 \lambda_i} \left( \frac{1}{1 + E_s/4N_0 \lambda_i} \right) \right) \right)^{m}$$  \hspace{1cm} (4.31)

4.2.3 Rayleigh fading

In this case $\bar{\alpha} = 0$, giving $K_{i,j} = 0$ for all $i$ and $j$. Thus Inequality (4.31) can be written as:

$$P(c \rightarrow e) \leq \left( \prod_{i=1}^{n} \frac{1}{1 + E_s/4N_0 \lambda_i} \right)^{m} \prod_{i=1}^{r} \lambda_i^{-m} (E_s/4N_0)^{-rm}$$  \hspace{1cm} (4.32)

Let $r$ denote the rank of matrix $A$, then the kernel of $A$ has dimension $n - r$ and exactly $n - r$ eigenvalues of $A$ are zero. Say the non-zero eigenvalues of $A$ are $\lambda_1, \lambda_2, \ldots, \lambda_r$, then it follows from Inequality (4.32) that:

Thus, a diversity advantage of $mr$ and a coding advantage of $(\lambda_1\lambda_2\cdots\lambda_r)^{1/r}$ is achieved. Recall that $\lambda_1\lambda_2\cdots\lambda_r$ is the absolute value of the sum of determinants of all the principal $r \times r$ cofactors of $A$. Moreover, it is easy to see that the ranks of $A(c, e)$, and $B(c, e)$, defined as $A(c, e) = B(c, e)^*B(c, e)$, are equal.

4.2.4 Design criteria for Rayleigh space–time codes

The rank criterion. In order to achieve the maximum diversity $mn$, the matrix $B(c, e)$ has to be full rank for any code words $c$ and $e$. If $B(c, e)$ has minimum rank $r$ over the set of two tuples of distinct code words, then a diversity of $rm$ is achieved.

The determinant criterion. Suppose that a diversity benefit of $rm$ is our target. The minimum of $r$ roots of the sum of determinants of all $r \times r$ principal cofactors of $A(c, e) = B(c, e)^*B(c, e)$ taken over all pairs of distinct code words $e$ and $c$ corresponds to the coding advantage, where $r$ is the rank of $A(c, e)$. Special attention in the design must be paid to this quantity for any code words $e$ and $c$. The design target is making this sum as large as possible. If a diversity of $mn$ is the design target, then the minimum of the determinant of $A(c, e)$ taken over all pairs of distinct code words $e$ and $c$ must be maximized.

For large signal to noise ratios,

$$P(c \rightarrow e) \leq \left( \frac{E_s}{4N_0} \right)^{-rm} \left( \prod_{i=1}^{r} \lambda_i^{-m} \prod_{j=1}^{m} \prod_{i=1}^{r} \exp(-K_{i,j}) \right)$$  \hspace{1cm} (4.34)
Thus, a diversity of $rm$ and a coding advantage of:

$$ \left( \lambda_1 \lambda_2 \cdots \lambda_r \right)^{-1/r} \left[ \prod_{j=1}^{m} \prod_{i=1}^{r} \exp(-K_{i,j}) \right]^{1/rm} $$

is achievable. The derivation in this section is based on the original work presented in [2].

## 4.2.5 Code construction

In the presence of one receive antenna, little can be gained in terms of capacity increase by using more than four transmit antennas. Similarly, if there are two receive antennas, almost all the capacity increase can be obtained using $n = 6$ transmit antennas.

As has been indicated earlier, we can use multidimensional trellis codes for a wireless communication system that employs $n$ transmit antennas and (optional) receive antenna diversity where the channel is a quasistatic flat fading channel. The encoding for these trellis codes is obvious, with the exception that at the beginning and the end of each frame, the encoder is required to be in the zero state. At each time $t$, depending on the state of the encoder and the input bits, a transition branch is chosen. If the label of this branch is $q_1^t q_2^t \cdots q_n^t$, then transmit antenna $i$ is used to send constellation symbols $q_i^t$, $i = 1, 2, \cdots, n$ and all of these transmissions are simultaneous. Let us consider the 4PSK and 8PSK constellations as shown in Figure 4.5.

### 4.2.5.1 Examples

A number of results have been published on space–time code design [1–46]. We will present several illustrations.

In Figure 4.5 the first signal constellation is 4PSK, where the signal points are labeled by the elements of $\mathbb{Z}_4$, the ring of integers modulo 4. We consider the four-state trellis code shown in Figure 4.6 [2]. The edge label $x_1 x_2$ indicates that signal $x_1$ is transmitted over the first antenna and that signal $x_2$ is transmitted over the second antenna. This code has a very simple description in terms of

![Figure 4.5 4PSK and 8PSK constellations.](image)

![Figure 4.6 2-space–time code, 4PSK, four states, 2 b/s/Hz.](image)
Figure 4.7 2-space–time codes, 4PSK(a), 8 and (b)16 states, 2 b/s/Hz.

(a) \( (x_1^k, x_2^k) = a_{k-2}(2, 2) + b_{k-1}(2, 0) + a_{k-1}(1, 0) + b_k(0, 2) + a_k(0, 1) \)

(b) \( (x_1^k, x_2^k) = b_{k-2}(0, 2) + a_{k-2}(2, 0) + b_{k-1}(2, 0) + a_{k-1}(1, 2) + b_k(0, 2) + a_k(0, 1) \)

Figure 4.7 represents 2-space–time codes for the 4PSK constellation and 8 and 16 state encoders for 2 b/s/Hz. The two output bits \((x_1^k, x_2^k)\) as a function of the input bits are also shown in the figure.
Several additional examples are given in Figures 4.8–4.11 [2].

\[
\begin{align*}
(x_1^k, x_2^k) &= a_{k-3}(2, 2) + b_{k-2}(3, 3) + a_{k-2}(2, 0) \\
&+ b_{k-1}(2, 2) + a_{k-1}(1, 1) \\
&+ b_k(0, 2) + a_k(0, 1)
\end{align*}
\] (4.37)

Assuming that the input to the encoder at time \( k \) is the three input bits \((d_k, b_k, a_k)\), the output of the encoder at time \( k \) is:

\[
\begin{align*}
(x_1^k, x_2^k) &= a_{k-1}(4, 0) + b_{k-1}(2, 0) + a_{k-1}(5,0) \\
&+ d_k(0, 4) + b_k(0, 2) + a_k(0, 1)
\end{align*}
\] (4.38)

where the computation is performed in \( \mathbb{Z}_8 \), the ring of integers modulo 8, and the elements of the 8PSK constellation have the labeling given in Figure 4.5.

If \( r_j^t \) is the received signal at receive antenna \( j \) at time \( t \), the branch metric for a transition labeled \( q_1^t q_2^t \cdots q_n^t \) is given by:

\[
\sum_{j=1}^m \left| r_j^t - \sum_{i=1}^n \alpha_{i,j} q_i^t \right|^2
\] (4.39)
The Viterbi algorithm is then used to compute the path with the lowest accumulated metric. The frame error probability for four different examples of the coding is shown in Figures 4.12–4.14. The gain shown in these figures should be used in expressions for $E_{ff}$ discussed in Chapters 2 and 3 to evaluate the overall reconfiguration efficiency for different schemes. In general, the expression for efficiency should be further modified as follows.
4.2.6 Reconfiguration efficiency of space–time coding

Let $rm$ be the diversity advantage of the system with $n$ transmit and $m$ receive antennas. For a block of $l$ symbols, with constellation $Q$ of $2^b$ elements, the equivalent rate of transmission $R$ in a system with ST coding satisfies:

$$R \leq \frac{\log[A_{2^b}(n, r)]}{l}$$

in bits per second per Hertz, where $A_{2^b}(n, r)$ is the maximum size of a code length $n$ and minimum Hamming distance $r$ defined over an alphabet of size $2^b$ [2].

On the other hand, if $b$ is the transmission rate of a multiple antenna system employed in conjunction with an $r$-space–time trellis code, the trellis complexity of the space–time code is at least $2^{b(r-1)}$ [2].

The reconfiguration gain is defined as the solution to:

$$P(c \rightarrow e | r, m, g(ST)E_s/4T) = P(c \rightarrow e | r = 1, m = 1, E_s/4T)$$

Using Inequality (4.33) for Rayleigh fading, Equation (4.41) gives:

$$\left( \prod_{i=1}^{r} \lambda_i \right)^{-m} (g(ST)E_s/4N_0)^{-rm} = \lambda_i^{-1}(E_s/4N_0)$$
Figure 4.12 Codes for 4PSK with rate 2 b/s/Hz that achieve diversity 2 with one receive and two transmit antennas.

resulting in

\[ g(ST) = \left[ \lambda_1^{-1} (E_s/4N_0) \right]^{r_m} \left\{ E_s/4N_0 \left( \prod_{i=1}^{r} \lambda_i \right)^{-1/r} \right\} \]

The relative complexity over the signal constellation Q with \(2^b\) elements is:

\[ D_r(ST) > 2^{b(r-1)/2} = 2^{b(r-2)} \]

Now the reconfiguration efficiency defined by Equation (3.1) becomes:

\[ E_{eff} = \left( \frac{R}{b} \right) \frac{g(ST)}{D_r} = \left[ \lambda_1^{-1} (E_s/4N_0) \right]^{r_m} \left\{ E_s/4N_0 \left( \prod_{i=1}^{r} \lambda_i \right)^{-1/c} \right\} \left( \frac{R}{b} \right) 2^{-b(r-2)} \]

Now we continue with different examples of ST codes. Consider the 8PSK signal constellation, where the encoder maps a sequence of three bits \(a_k b_k c_k\) at time \(k\) to \(ii\) with \(i = 4a_k + 2b_k + c_k\).

It is easy to show that the equivalent space–time code for this delay diversity code has the trellis representation given in Figure 4.15. The minimum determinant of this code is \((2 - \sqrt{2})^2\). As in the 4PSK case, one can improve the coding advantage of the above codes by constructing encoders with more states. An example is given in Figure 4.16 [2] with the constellation in Figure 4.17.

### 4.2.6.1 An \(r\) space–time trellis code for \(r > 2\)

As an example, a 4-space–time code for four transmit antennas is considered. The limit on the transmission rate is 2 b/s/Hz. Thus, the trellis complexity of the code is bounded below by 64. The input to the encoder is a block of length two of bits \(a_1, b_1\), corresponding to an integer \(i = 2a_1 + b_1 \in \mathbb{Z}_4\). The 64 states of the trellis correspond to the set of all three tuples \((s_1, s_2, s_3)\) with \(s_i \in \mathbb{Z}_4\) for
1 ≤ i ≤ 3. At state \((s_1, s_2, s_3)\) upon input data \(i\), the encoder outputs \((i, s_1, s_2, s_3)\) elements of the 4PSK constellation (see Figure 4.5) and moves to state \((i, s_1, s_2)\). Given two code words \(c\) and \(e\), the associated paths in the trellis diverge at time \(t_1\) from a state and remerge in another state at a later time \(t_2 ≤ l\). It is easy to see that the \(t_1\)th, \((t_1 + l)\)th, \((t_2 - 1)\)th, and \(t_2\)th columns of the matrix \(B(c, e)\) are independent. Thus, the above design gives a 4-space–time code.
Figure 4.14 Codes for 8PSK with rate 3 b/s/Hz that achieve diversity 2 with one receive and two transmit antennas.

Figure 4.15 Space–time realization of a delay diversity 8PSK code constructed from a repetition code.

4.2.7 Delay diversity

The encoder block diagram of a delay diversity transmitter is given in Figure 4.18, with

\[
\begin{align*}
    c^1_t &= \tilde{c}^1_{t-1} \\
    c^2_t &= \tilde{c}^2_t
\end{align*}
\]

(4.45)

where \(c^1_t\) and \(c^2_t\) are the symbols of the equivalent space–time code at time \(t\) and \(c^1_t c^2_t\) is the output of the encoder at time \(t\).
Figure 4.16 2-space–time 16 QAM code, 16 states, 4 b/s/Hz.

Figure 4.17 The 16 QAM constellation.

Figure 4.18 The block diagram of a delay diversity transmitter.
Next, we consider the block code \[ C = \{00, 15, 22, 37, 44, 51, 66, 73\} \] of length two defined over the alphabet 8PSK instead of the repetition code. This block code is the best in the sense of product distance \[ |c_1 - e_1||c_2 - e_2| \] among all the codes of cardinality eight and of length two defined over the alphabet 8PSK. This means that the minimum of the product distance \[ |c_1 - e_1||c_2 - e_2| \] between pairs of distinct code words \[ c = c_1c_2 \in C \] and \[ e = e_1e_2 \in C \] is maximal among all such codes. The delay diversity code constructed from this block code is identical to the space–time code given by the trellis diagram of Figure 4.9. The minimum determinant of this delay diversity code is thus 2.

The 16-state code for the 16 QAM constellation given in Figure 4.16, is obtained from the block code

\[ \{00, 11, 22, 39, 44, 55, 66, 71, 88, 93, 1010, 111, 1212, 137, 1414, 155\} \]

using the same delay diversity construction. Again, this block code is optimal in the sense of product distance.

The delay diversity code construction can also be generalized to systems having more than two transmit antennas. For instance, the 4PSK 4-space–time code given before is a delay diversity code. The corresponding block code is the repetition code. By applying the delay diversity construction to the 4PSK block code

\[ \{0000, 1231, 2123, 3312\} \]

one can obtain a more powerful 4PSK 4-space–time code having the same trellis complexity.

### 4.3 SPACE–TIME BLOCK CODES FROM ORTHOGONAL DESIGNS

ML decoding of a multidimensional trellis requires a large complexity. In this section we present a special class of space–time codes for which maximum likelihood decoding is achieved in a simple way through decoupling of the signals transmitted from different antennas rather than joint detection. This uses the orthogonal structure of the space–time block code and gives a maximum likelihood decoding algorithm which is based only on linear processing at the receiver. The presentation is based on [5, 39–46].

Space–time block codes are designed to achieve the maximum diversity order for a given number of transmit and receive antennas subject to the constraint of having a simple decoding algorithm. Unfortunately, space–time block codes constructed in this way only exist for few sporadic values of \( n \).

#### 4.3.1 The channel model and the diversity criterion

At time \( t \) the signal \( r_j^t \), received at antenna \( j \), is given by Equation (4.21) which for \( \sqrt{E_s} = 1 \) becomes:

\[ r_j^t = \sum_{i=1}^{n} \alpha_{i,j} c_i^t + \eta_j^t \]  

(4.46)

Assuming perfect channel state information is available, the receiver computes the decision metric for \( l \) symbols and Expression (4.39) gives:

\[ \sum_{i=1}^{l} \sum_{j=1}^{m} \left| r_j^t - \sum_{i=1}^{n} \alpha_{i,j} c_i^t \right|^2 \]  

(4.47)
In order to achieve the maximum diversity \( mn \), the matrix \( \mathbf{A}(\mathbf{c}, \mathbf{e}) = \mathbf{B}(\mathbf{c}, \mathbf{e})^* \mathbf{B}(\mathbf{c}, \mathbf{e}) \) with

\[
\mathbf{B}(\mathbf{c}, \mathbf{e}) = \begin{bmatrix}
e_1^t - e_1^t & e_2^t - e_2^t & \cdots & e_1^t - e_1^t \\
e_1^t - e_1^t & e_2^t - e_2^t & \cdots & e_2^t - e_2^t \\
e_1^t - e_1^t & e_2^t - e_2^t & \ddots & \vdots \\
e_1^t - e_1^t & e_2^t - e_2^t & \cdots & e_n^t - e_n^t
\end{bmatrix}
\quad (4.48)
\]

has to be full rank for any pair of distinct code words \( \mathbf{c} \) and \( \mathbf{e} \). If \( \mathbf{B}(\mathbf{c}, \mathbf{e}) \) has minimum rank \( r \) over the set of pairs of distinct code words, then a diversity of \( rm \) is achieved.

### 4.3.2 Real orthogonal designs

For \( n = 2, 4 \) or 8, mathematics literature offers orthogonal sets of signals defined as:

\[
\mathbf{O}_2(\mathbf{x}_1, \mathbf{x}_2) = \begin{bmatrix} x_1 & x_2 \\ -x_2 & x_1 \end{bmatrix}
\quad (4.49)
\]

The \( 4 \times 4 \) design

\[
\mathbf{O}_4(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4) = \begin{bmatrix} x_1 & x_2 & x_3 & x_4 \\ -x_2 & x_1 & -x_4 & x_3 \\ -x_3 & x_4 & x_1 & -x_2 \\ -x_4 & -x_3 & x_2 & x_1 \end{bmatrix}
\quad (4.50)
\]

and \( 8 \times 8 \) design

\[
\mathbf{O}_8(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_8) = \begin{bmatrix} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 & x_8 \\ -x_2 & x_1 & -x_3 & x_6 & -x_5 & -x_8 & -x_7 & x_7 \\ -x_3 & -x_4 & x_1 & x_2 & x_7 & x_8 & -x_5 & -x_6 \\ -x_4 & -x_3 & x_1 & x_4 & x_2 & -x_7 & x_6 & -x_5 \\ -x_5 & -x_6 & -x_7 & -x_8 & x_1 & x_2 & x_3 & x_4 \\ -x_6 & x_5 & -x_8 & x_7 & -x_2 & x_1 & -x_4 & x_3 \\ -x_7 & x_8 & -x_5 & -x_6 & -x_3 & x_4 & x_1 & -x_2 \\ -x_8 & -x_7 & x_6 & x_5 & -x_4 & -x_3 & x_2 & x_1 \end{bmatrix}
\quad (4.51)
\]

### 4.3.3 Space–time encoder

At time slot 1, \( nb \) bits arrive at the encoder and select constellation signals \( s_1, \ldots, s_n \). Setting \( x_i = s_i \) for \( i = 1, 2, \ldots, n \), we arrive at a matrix \( \mathbf{C} = \mathbf{O}(s_1, \ldots, s_n) \) with entries \( \pm s_1, \pm s_2, \ldots, \pm s_n \). At each time slot \( t = 1, 2, \ldots, n \), the entries \( C_{nt}, i = 1, 2, \ldots, n \) are transmitted simultaneously from transmit antennas 1, 2, \ldots, \( n \). The rate of transmission is \( b \) bits/s/Hz.

### 4.3.4 The diversity order

The rank criterion requires that the matrix \( \mathbf{O}(\tilde{s}_1, \ldots, \tilde{s}_n) - \mathbf{O}(s_1, \ldots, s_n) \) be non-singular for any two distinct code sequences \( (s_1, \ldots, s_n) \neq (\tilde{s}_1, \ldots, s_n) \). Clearly, \( \mathbf{O}(\tilde{s}_1 - s_1, \ldots, \tilde{s}_n - s_n) = \mathbf{O}(\tilde{s}_1, \ldots, \tilde{s}_n) - \mathbf{O}(s_1, \ldots, s_n) \) where \( \mathbf{O}(\tilde{s}_1 - s_1, \ldots, \tilde{s}_n - s_n) \) is the matrix constructed from \( \mathbf{O} \) by replacing \( x_i \) with \( \tilde{s}_i - s_i \) for all \( i = 1, 2, \ldots, n \). The determinant of the orthogonal matrix \( \mathbf{O} \) is easily seen to be:

\[
\det(\mathbf{O}\mathbf{O}^T)^{1/2} = \left[ \sum_i x_i^2 \right]^{n/2}
\quad (4.52)
\]
where $O^T$ is the transpose of $O$. Hence:

$$\det(O(\tilde{s}_1 - s_1, \ldots, \tilde{s}_n - s_n)) = \left[ \sum_i |\tilde{s}_i - s_i|^2 \right]^{n/2}$$  (4.53)

which is non-zero. It follows that $O(\tilde{s}_1, \ldots, \tilde{s}_n) - O(s_1, \ldots, s_n)$ is non-singular and the maximum diversity order $nm$ is achieved.

### 4.3.5 The decoding algorithm

Rows of $O$ are all permutations of the first row of $O$ with possibly different signs. Let $e_1, \ldots, e_n$ denote the permutations corresponding to these rows and let $\delta_k(i)$ denote the sign of $x_i$ in the $k$th row of $O$. Then $e_k(p) = q$ means that $x_p$ is up to a sign change the $(k, q)$th element of $O$. Since the columns of $O$ are pairwise-orthogonal, it turns out that minimizing the metric of Expression (4.47) amounts to minimizing:

$$\sum_{i=1}^n S_i$$  (4.54)

where

$$S_i = \left( \left[ \sum_{t=1}^n \sum_{j=1}^m r^*_t \alpha_{e_t(i), j} \delta_t(i) \right] - s_i \right)^2 + \left( -1 + \sum_{k,l} |\alpha_{k,l}|^2 \right) |s_i|^2$$  (4.55)

The value of $S_i$ only depends on the code symbol $s_i$, the received symbols $\{r_t^j\}$, the path coefficients $\{\alpha_{i,j}\}$, and the structure of the orthogonal design $O$. It follows that minimizing the sum given in Expression (4.54) amounts to minimizing Equation (4.55) for all $1 \leq i \leq n$. Thus, the maximum likelihood detection rule is to form the decision variables:

$$R_i = \sum_{t=1}^n \sum_{j=1}^m r^*_t \alpha_{e_t(i), j} \delta_t(i)$$  (4.56)

for all $i = 1, 2, \ldots, n$ and decide in favor of $s_i$ among all the constellation symbols $s$ if:

$$s_i = \arg \min_{s \in A} |R_i - s|^2 + \left( -1 + \sum_{k,l} |\alpha_{k,l}|^2 \right) |s_i|^2$$  (4.57)

This is a very simple decoding strategy that provides diversity.

### 4.3.6 Linear processing orthogonal designs

The above properties are preserved even if we allow linear processing at the transmitter. Therefore, we relax the definition of orthogonal designs to allow linear processing at the transmitter. Signals transmitted from different antennas will now be linear combinations of constellation symbols.

A linear processing orthogonal design in variables $x_1, x_2, \ldots, x_n$ is an $n \times n$ matrix $E$ such that:

- the entries of $E$ are real linear combinations of variables $x_1, x_2, \ldots, x_n$;
- $E^*E = D$, where $D$ is a diagonal matrix with $(i, i)$th diagonal element of the form $(l_1^i x_1^2 + l_2^i x_2^2 + \cdots l_n^i x_n^2)$, with the coefficients $(l_1^i, l_2^i, \ldots, l_n^i)$ strictly positive numbers.

It is easy to show that transmission using a linear processing orthogonal design provides full diversity and a simplified decoding algorithm as above.
4.3.7 Generalized real orthogonal designs

Since the simple maximum likelihood decoding algorithm described above is achieved because of orthogonality of columns of the design matrix, we may generalize the definition of linear processing orthogonal designs. This creates new and simple transmission schemes for any number of transmit antennas.

A generalized orthogonal design \( G \) of size \( n \) is a \( p \times n \) matrix with entries \( 0, \pm x_1, \pm x_2, \ldots, \pm x_p \), such that \( G^T G = D \) where \( D \) is a diagonal matrix with diagonal \( D_{ii}, i = 1, 2, \ldots, n \) of the form \( (l_1 x_1^2 + l_2 x_2^2 + \cdots + l_n x_n^2) \) and coefficients \( (l_1, l_2, \ldots, l_n) \) strictly positive numbers. The rate of \( G \) is \( R = k / p \).

A full-rate generalized orthogonal design has entries of the form \( \pm x_1, \pm x_2, \ldots, \pm x_p \).

The generalized orthogonal signal sets are:

\[
G_3 = \begin{bmatrix}
  x_1 & x_2 & x_3 \\
  -x_2 & x_1 & -x_4 \\
  -x_3 & x_4 & x_1 \\
  -x_4 & -x_3 & x_2
\end{bmatrix}
\]

(4.58)

\[
G_5 = \begin{bmatrix}
  x_1 & x_2 & x_3 & x_4 & x_5 \\
  -x_2 & x_1 & x_4 & -x_3 & x_6 \\
  -x_3 & -x_4 & x_1 & x_2 & x_7 \\
  -x_4 & x_3 & -x_2 & x_1 & x_8 \\
  -x_5 & -x_6 & -x_7 & -x_8 & x_1 \\
  -x_6 & x_5 & -x_8 & x_7 & -x_2 \\
  -x_7 & x_8 & x_5 & -x_6 & -x_3 \\
  -x_8 & -x_7 & x_6 & x_5 & -x_4
\end{bmatrix}
\]

(4.59)

\[
G_6 = \begin{bmatrix}
  x_1 & x_2 & x_3 & x_4 & x_5 & x_6 \\
  -x_2 & x_1 & x_4 & -x_3 & x_6 & -x_5 \\
  -x_3 & -x_4 & x_1 & x_2 & x_7 & x_8 \\
  -x_4 & x_3 & -x_2 & x_1 & x_8 & -x_7 \\
  -x_5 & -x_6 & -x_7 & -x_8 & x_1 & x_2 \\
  -x_6 & x_5 & -x_8 & x_7 & -x_2 & x_1 \\
  -x_7 & x_8 & x_5 & -x_6 & -x_3 & x_4 \\
  -x_8 & -x_7 & x_6 & x_5 & -x_4 & -x_3
\end{bmatrix}
\]

(4.60)

\[
G_7 = \begin{bmatrix}
  x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 \\
  -x_2 & x_1 & x_4 & -x_3 & x_6 & -x_5 & -x_8 \\
  -x_3 & -x_4 & x_1 & x_2 & x_7 & x_8 & -x_5 \\
  -x_4 & x_3 & -x_2 & x_1 & x_8 & -x_7 & x_6 \\
  -x_5 & -x_6 & -x_7 & -x_8 & x_1 & x_2 & x_3 \\
  -x_6 & x_5 & -x_8 & x_7 & -x_2 & x_1 & -x_4 \\
  -x_7 & x_8 & x_5 & -x_6 & -x_3 & x_4 & x_1 \\
  -x_8 & -x_7 & x_6 & x_5 & -x_4 & -x_3 & x_2
\end{bmatrix}
\]

(4.61)

4.3.8 Encoding

At time slot 1, \( kb \) bits arrive at the encoder and select constellation symbols \( s_1, s_2, \ldots, s_n \). The encoder populates the matrix by setting \( x_i = s_i \) and at time \( t = 1, 2, \ldots, p \) the signals \( G_{1t}, \ldots, G_{nt} \) are transmitted simultaneously from antennas \( 1, 2, \ldots, n \). Thus, \( kb \) bits are sent during each \( p \) transmissions. It can be proved, as in Equation (4.53), that the diversity order is \( nm \). It should be mentioned that the rate of a generalized orthogonal design is different from the throughput of the associated code. To motivate the definition of the rate, we note that the theory of space–time coding proves that for a diversity order of \( nm \), it is possible to transmit \( b \) bits per time slot and this is the best possible. Therefore, the rate \( R \) of this coding scheme is defined to be \( kb / pb \) which is equal to \( k / p \).
4.3.9 The Alamouti scheme

The space–time block code, already used in Section 4.1 was proposed by Alamouti [1]. The code uses the complex orthogonal signal set:

\[
\begin{bmatrix}
  x_1 & x_2 \\
  -x_2^* & x_1^*
\end{bmatrix}
\]  

(4.62)

Suppose that there are \(2^b\) signals in the constellation. At the first time slot, \(2^b\) bits arrive at the encoder and select two complex symbols \(s_1\) and \(s_2\). These symbols are transmitted simultaneously from antennas one and two, respectively. At the second time slot, signals \(-s_2^*\) and \(s_1^*\) are transmitted simultaneously from antennas one and two, respectively.

The ML detector will minimize the decision statistic:

\[
\sum_{j=1}^{m} \left( \left| r_j^1 - \alpha_{1,j} s_1 - \alpha_{2,j} s_2 \right|^2 + \left| r_j^2 - \alpha_{1,j} s_1^* - \alpha_{2,j} s_2^* \right|^2 \right)
\]

(4.63)

over all possible values of \(s_1\) and \(s_2\). The minimizing values are the receiver estimates of \(s_1\) and \(s_2\), respectively. This is equivalent to minimizing the decision statistics:

\[
\left( \sum_{j=1}^{m} \left( r_j^1 \alpha_{1,j} + r_j^2 \alpha_{2,j} \right) - s_1 \right)^2 + \left( -1 + \sum_{j=1}^{m} \sum_{i=1}^{2} |\alpha_{i,j}|^2 \right) |s_1|^2
\]

(4.64)

for detecting \(s_1\) and the decision statistics:

\[
\left( \sum_{j=1}^{m} \left( r_j^1 \alpha_{2,j} + r_j^2 \alpha_{1,j} \right) - s_2 \right)^2 + \left( -1 + \sum_{j=1}^{m} \sum_{i=1}^{2} |\alpha_{i,j}|^2 \right) |s_2|^2
\]

(4.65)

for decoding \(s_2\). The scheme provides full diversity of \(2m\) using \(m\) receive antennas.

4.3.10 Complex orthogonal designs

Given a complex orthogonal signal set \(O_c\) of size \(n\), we replace each complex variable \(x_i = x_1^i + x_2^i\), \(1 \leq i \leq n\) by the \(2 \times 2\) real matrix:

\[
\begin{bmatrix}
  x_1^i & x_2^i \\
  -x_2^i & x_1^i
\end{bmatrix}
\]

(4.66)

In this way, \(x_i^*\) is represented by:

\[
\begin{bmatrix}
  x_1^i & -x_2^i \\
  x_2^i & x_1^i
\end{bmatrix}
\]

(4.67)

\(ix_i\) is represented by:

\[
\begin{bmatrix}
  -x_2^i & x_1^i \\
  x_1^i & -x_2^i
\end{bmatrix}
\]

(4.68)

and so forth. It is easy to see that the \(2n \times 2n\) matrix formed in this way is a real orthogonal design of size \(2n\).

A complex orthogonal signal set of size \(n\) exists if and only if \(n = 2\).

4.3.11 Generalized complex orthogonal designs

Let \(G_c\) be a \(p \times n\) matrix whose entries are \(0, \pm x_1, \pm x_1^*, \pm x_2, \pm x_2^*, \ldots, \pm x_k, \pm x_k^*\) or their product with \(i\). If \(G_c^* G_c = D_c\) where \(D_c\) is a diagonal matrix with \((i, i)\)th diagonal element of the
SPACE–TIME BLOCK CODES FROM ORTHOGONAL DESIGNS

form:

\[
\left( l_1 |x_1|^2 + l_2 |x_2|^2 + \cdots + l_k |x_k|^2 \right)
\]

and the coefficients \((l_1, l_2, \ldots, l_k)\) all strictly positive numbers, then \(G_c\) is referred to as a generalized orthogonal design of size \(n\) and rate \(R = k/p\). For instance, rate 1/2 codes for transmission using three and four transmit antennas are given by:

\[
G_3^c = \begin{bmatrix}
    x_1 & x_2 & x_3 \\
    -x_2 & x_1 & -x_4 \\
    -x_3 & x_4 & x_1 \\
    -x_4 & -x_3 & x_2 \\
    x_1^* & x_2^* & x_3^* \\
    -x_2^* & x_1^* & -x_4^* \\
    -x_3^* & x_4^* & x_1^* \\
    -x_4^* & -x_3^* & x_2^*
\end{bmatrix}
\]

(4.69)

and

\[
G_4^c = \begin{bmatrix}
    x_1 & x_2 & x_3 & x_4 \\
    -x_2 & x_1 & -x_4 & x_3 \\
    -x_3 & x_4 & x_1 & -x_2 \\
    -x_4 & -x_3 & x_2 & x_1 \\
    x_1^* & x_2^* & x_3^* & x_4^* \\
    -x_2^* & x_1^* & -x_4^* & x_3^* \\
    -x_3^* & x_4^* & x_1^* & -x_2^* \\
    -x_4^* & -x_3^* & x_2^* & x_1^* \\
\end{bmatrix}
\]

(4.70)

These transmission schemes and their analogs for higher \(n\) give full diversity but lose half of the theoretical bandwidth efficiency.

**4.3.12 Special codes**

It is natural to ask for higher rates than 1/2 when designing generalized complex linear processing orthogonal designs for transmission with \(n\) multiple antennas. For \(n = 2\), Alamouti’s scheme gives a rate one design. For \(n = 3\) and 4, rate 3/4 generalized complex linear processing orthogonal designs are given by:

\[
H_3 = \begin{bmatrix}
    x_1 & x_2 & \frac{x_3}{\sqrt{2}} \\
    -x_2^* & x_1^* & \frac{x_3}{\sqrt{2}} \\
    \frac{x_1}{\sqrt{2}} & \frac{x_2}{\sqrt{2}} & \frac{(-x_1-x_2^*+x_3^*)}{2} \\
    \frac{x_1}{\sqrt{2}} & -\frac{x_2}{\sqrt{2}} & \frac{(x_1-x_2^*-x_3^*)}{2}
\end{bmatrix}
\]

(4.71)
4.3.13 Performance results

A collection of results is shown in Figures 4.19–4.26. The transmission using two transmit antennas employs the 8PSK constellation and the code $G_2$. For three and four transmit antennas, the 16 QAM constellation and the codes $H_3$ and $H_4$, respectively, are used. Since $H_3$ and $H_4$ are rate $3/4$ codes, the total transmission rate in each case is 3 bits/s/Hz. It is seen that at the bit error rate of $10^{-5}$ the rate $3/4$ 16 QAM code $H_4$ gives about 7 dB gain over the use of an 8PSK $G_2$ code.

Transmission using two transmit antennas employs the 4PSK constellation and the code $G_2$. For three and four transmit antennas, the 16-QAM constellation and the codes $G_3$ and $G_4$, respectively, are used. Since $G_3$ and $G_4$ are rate 1/2 codes, the total transmission rate in each case is 2 bits/s/Hz. It is seen that at the bit error rate of $10^{-5}$ the rate 1/2 16-QAM code $G_4$ gives about 5 dB gain over the use of a 4PSK $G_2$ code.

The transmission using two transmit antennas employs the binary PSK (BPSK) constellation and the code $G_2$. For three and four transmit antennas, the 4PSK constellation and the codes $G_3$ and $G_4$, respectively, are used. Since $G_3$ and $G_4$ are rate 1/2 codes, the total transmission rate in each case is 1 bit/s/Hz. It is seen that at the bit error rate of $10^{-5}$ the rate 1/2 4PSK code $G_4$ gives about 7.5 dB gain over the use of a BPSK $G_2$ code.

If number of receive antennas is increased, this gain reduces to 3.5 dB. The reason is that much of the diversity gain is already achieved using two transmit and two receive antennas.

4.4 CHANNEL ESTIMATION IMPERFECTIONS

So far we have been assuming that a perfect channel estimation is available for the operation of the ML decoder collecting statistics defined by Expression (4.39) or Expression (4.47). Let us look at this assumption more carefully. First of all, the errors in the channel estimation will depend on the estimator structure.
Figure 4.20 Bit error probability versus SNR for space–time block codes at 3 bits/s/Hz; one receive antenna.

Figure 4.21 Bit error probability versus SNR for space–time block codes at 2 bits/s/Hz; one receive antenna.
Figure 4.22 Symbol error probability versus SNR for space–time block codes at 2 bits/s/Hz; one receive antenna.

Figure 4.23 Bit error probability versus SNR for space–time block codes at 1 bit/s/Hz; one receive antenna.
Figure 4.24 Symbol error probability versus SNR for space–time block codes at 1 bit/s/Hz; one receive antenna.

Figure 4.25 BER versus SNR for space–time block codes at 1 bit/s/Hz; two receive antennas.
4.4.1 Channel estimator

At the beginning of each frame of symbols to be transmitted from transmit antenna \( i \), a sequence \( W_i \) of length \( k \) pilot symbols:

\[ W_i = (W_{i,1}, W_{i,2}, \ldots, W_{i,k}) \]

is appended. The sequences \( W_1, W_2, \ldots, W_n \) are designed to be orthogonal to each other:

\[ W_p \bar{W}_q = \sum_{j=1}^{k} W_{p,j} \bar{W}_{q,j} = 0 \]

whenever \( p \neq q \). In the previous expression \( \bar{W} \) stands for the conjugate of \( W \) and \( \bar{W} \) is the conjugate transpose of \( W \). Let \( r^j = (r^j_1, r^j_2, \ldots, r^j_k) \) be the observed sequence of received signals at antenna \( j \) during the training period. Then:

\[ r^j_i = \sum_{i=1}^{n} \alpha_{i,j} W_{i,t} + \eta_{i,j}, \quad 1 \leq j \leq m, \quad 1 \leq t \leq k \]

where the channel coefficients \( \alpha_{i,j} \) are independent samples of a complex Gaussian random variable with mean zero and variance 0.5 per dimension, and \( \eta_{i,j} \) are independent samples of a zero mean complex Gaussian random variable with variance \( N_0/2 \) per dimension. Let \( \eta^j = (\eta_{1,j}, \eta_{2,j}, \ldots, \eta_{k,j}) \). Our goal is to estimate \( \alpha_{i,j}, i = 1, 2, \ldots, n, j = 1, 2, \ldots, m \) using the statistic \( r^j \).

The unbiased estimator \( \beta_{i,j} \) having the least variance is given by the ratio of inner products \( (r^j_i, \bar{W}^i) / (W^i, \bar{W}^i) \). Indeed, since \( W^i \bar{W}^i = 0 \) it is easy to see that:

\[ r^j_i \cdot \bar{W}^i = \alpha_{i,j} (W^i \cdot \bar{W}^i) + \eta^j \cdot \bar{W}^i \]

thus

\[ \alpha_{i,j} = \frac{r^j_i \cdot \bar{W}^i}{W^i \cdot \bar{W}^i} - \frac{\eta^j \cdot \bar{W}^i}{W^i \cdot \bar{W}^i} \]
Figure 4.27 Performance of four and 16 state 4PSK codes in the presence of channel estimation error, 2 bits/s/Hz, two receive and two transmit antennas.

In other words,

$$\beta_{i,j} = \alpha_{i,j} + \frac{\eta^j \cdot \bar{W}^i}{W^i \cdot \bar{W}^i}$$

The random variable $\beta_{i,j}$ has zero mean. The variance of the estimation error is $\frac{N_0}{2kE_s}$ per dimension which is the minimum given by the Cramer–Rao bound. Simulation results for imperfect channel estimation (mismatch) with $n = 2$, $k = 8$, and the frame length 130 symbols are shown in Figure 4.27.

More details on system imperfections can be found in [7, 47, 48].

4.5 QUASI-ORTHOGONAL SPACE–TIME BLOCK CODES

It was shown in Section 4.3 that a complex orthogonal design that provides full diversity and full transmission rate for a space–time block code is not possible for more than two antennas. Previous attempts have been concentrated in generalizing orthogonal designs which provide space–time block codes with full diversity and a high transmission rate. In this section we discuss rate one codes which are quasi-orthogonal and provide partial diversity. The decoder for these codes works with pairs of transmitted symbols instead of single symbols.

An example of a full rate full diversity complex space–time block code is the Alamouti scheme already discussed in Section 4.3. In this section we will use the notation:

$$A_{12} = \begin{bmatrix} x_1 & x_2 \\ -x_2^* & x_1^* \end{bmatrix}$$

Here we use the subscript 12 to represent the indeterminates $x_1$ and $x_2$ in the transmission matrix. Now, let us consider the following space–time block code with block length $T$ symbol intervals where $K$ bits are transmitted over $N$ transmit antennas and received with $M$ receive antennas, for $N = T = K = 4$:

$$A = \begin{bmatrix} A_{12} & A_{34} \\ -A_{34}^* & A_{12}^* \end{bmatrix} = \begin{bmatrix} x_1 & x_2 & x_3 & x_4 \\ -x_2^* & x_1^* & -x_4^* & x_3^* \\ -x_3^* & -x_4^* & x_1^* & x_2^* \\ x_4 & -x_3 & -x_2 & x_1 \end{bmatrix}$$
It is easy to see that the minimum rank of matrix \( \mathcal{A}(s_1 - \bar{s}_1, s_2 - \bar{s}_2, s_3 - \bar{s}_3, s_4 - \bar{s}_4) \), the matrix constructed from \( \mathcal{A} \) by replacing \( x_i \) with \( s_i - \bar{s}_i \), is 2. Therefore, a diversity of \( 2M \) is achieved while the rate of the code is one. Note that the maximum diversity of \( 4M \) for a rate one code is impossible in this case.

### 4.5.1 Decoding

Assuming perfect channel state information is available, the ML receiver computes the decision metric:

\[
\sum_{m=1}^{M} \sum_{t=1}^{T} \left| r_{1,m} - \sum_{n=1}^{N} a_{n,m} A_{tn} \right|^2
\]

over all possible \( x_k = s_k \in \mathcal{A} \) and decides in favor of the constellation symbols \( s_1, \ldots, s_K \) that minimize this sum.

### 4.5.2 Decision metric

Now, if we define \( \mathcal{V}_i, i = 1, 2, 3, 4, \) as the \( i \)th column of \( \mathcal{A} \), it is easy to see that

\[
\langle \mathcal{V}_1, \mathcal{V}_2 \rangle = \langle \mathcal{V}_1, \mathcal{V}_3 \rangle = \langle \mathcal{V}_2, \mathcal{V}_4 \rangle = \langle \mathcal{V}_3, \mathcal{V}_4 \rangle = 0
\]

where \( \langle \mathcal{V}_i, \mathcal{V}_j \rangle = \sum_{n=1}^{4} (\mathcal{V}_i)_n (\mathcal{V}_j)_n^* \) is the inner product of vectors \( \mathcal{V}_i \) and \( \mathcal{V}_j \). Therefore, the subspace created by \( \mathcal{V}_1 \) and \( \mathcal{V}_2 \) is orthogonal to the subspace created by \( \mathcal{V}_2 \) and \( \mathcal{V}_3 \). Using this orthogonality, the maximum likelihood decision metric, Expression (4.74), can be calculated as the sum of two terms \( f_{14}(x_1, x_4) + f_{23}(x_2, x_3) \), where \( f_{14} \) is independent of \( x_2 \) and \( x_3 \) and \( f_{23} \) is independent of \( x_1 \) and \( x_4 \). Thus, the minimization of Expression (4.74) is equivalent to minimizing these two terms independently. In other words, first the decoder finds the pair \((s_1, s_4)\) that minimizes \( f_{14}(x_1, x_4) \) among all possible \((x_1, x_4)\) pairs. Then, or in parallel, the decoder selects the pair \((s_2, s_3)\) which minimizes \( f_{23}(x_2, x_3) \). This reduces the complexity of decoding without sacrificing the performance.

Simple manipulation of Expression (4.74) provides the following formulas for \( f_{14}(\cdot) \) and \( f_{23}(\cdot) \):

\[
f_{14}(x_1, x_4) = M \sum_{m=1}^{M} \left( \sum_{n=1}^{4} |a_{n,m}|^2 \right) \left( |x_1|^2 + |x_4|^2 \right)
+ 2 \text{Re}\{-(\alpha_{1,m} r_{1,m}^* - \alpha_{2,m} r_{2,m} - \alpha_{3,m} r_{3,m} - \alpha_{4,m} r_{4,m}) x_1 \}
+ (\alpha_{2,m} \alpha_{3,m} - \alpha_{3,m} \alpha_{4,m} + \alpha_{4,m} \alpha_{1,m}) x_1 x_4^*
\]

\[
f_{23}(x_2, x_3) = M \sum_{m=1}^{M} \left( \sum_{n=1}^{4} |a_{n,m}|^2 \right) \left( |x_2|^2 + |x_3|^2 \right)
+ 2 \text{Re}\{-(\alpha_{2,m} r_{1,m}^* - \alpha_{1,m} r_{2,m} - \alpha_{4,m} r_{3,m} - \alpha_{3,m} r_{4,m}) x_2 \}
+ (\alpha_{1,m} \alpha_{4,m} - \alpha_{2,m} \alpha_{3,m} + \alpha_{4,m} \alpha_{1,m}) x_2 x_3^*
\]
There are other structures which provide behaviors similar to those of Equation (4.73). A few examples are given below:

\[
\begin{bmatrix}
A_{12} & A_{34} \\
-A_{34} & A_{12}
\end{bmatrix}
\begin{bmatrix}
A_{12} & A_{34} \\
A_{34} & -A_{12}
\end{bmatrix}
\begin{bmatrix}
A_{12} & A_{34} \\
A_{34}^* & -A_{12}^*
\end{bmatrix}
\]  

(4.78)

A similar idea can be used to combine two rate 3/4 transmission matrices (4 \times 4) to build a rate 3/4 transmission matrix (8 \times 8) and so on. An example of an 8 \times 8 matrix which provides a rate 3/4 code is given below:

\[
\begin{bmatrix}
x_1 & x_2 & x_3 & 0 & x_4 & x_5 & x_6 & 0 \\
-x_2^* & x_1^* & 0 & -x_3 & x_5^* & -x_4^* & 0 & x_6 \\
x_3^* & 0 & -x_1^* & -x_2 & -x_6^* & 0 & x_4^* & x_5 \\
0 & -x_3^* & x_2^* & -x_1 & 0 & x_6^* & -x_5^* & x_4 \\
-x_4 & -x_5 & -x_6 & 0 & x_1 & x_2 & x_3 & 0 \\
-x_5^* & x_4^* & 0 & x_6 & -x_2^* & x_1^* & 0 & x_3 \\
x_6^* & 0 & -x_4^* & x_5 & x_3^* & 0 & -x_1^* & x_2 \\
0 & x_6^* & -x_5^* & -x_4 & 0 & x_3^* & -x_2^* & -x_1
\end{bmatrix}
\]  

(4.79)

Here, n = 8 antennas, k = 6 symbols, p = 8 transmissions. In this code, if we define \( V_i \), \( i = 1, 2, \ldots, 8 \), as the \( i \)th column, we have:

\[
\langle V_1, V_i \rangle = 0, \ i \neq 5 \quad \langle V_2, V_i \rangle = 0, \ i \neq 6 \\
\langle V_3, V_i \rangle = 0, \ i \neq 7 \quad \langle V_4, V_i \rangle = 0, \ i \neq 8 \\
\langle V_5, V_i \rangle = 0, \ i \neq 1 \quad \langle V_6, V_i \rangle = 0, \ i \neq 2 \\
\langle V_7, V_i \rangle = 0, \ i \neq 3 \quad \langle V_8, V_i \rangle = 0, \ i \neq 4
\]  

(4.80)

The maximum likelihood decision metric, Expression (4.74), can be calculated as the sum of three terms \( f_{14}(x_1, x_4) + f_{25}(x_2, x_5) + f_{36}(x_3, x_6) \) and similarly the decoding can be done using pairs of constellation symbols.

Figure 4.28 illustrates the performance of transmission of 2 bits/s/Hz using four transmit antennas and the rate one quasi-orthogonal code, the rate 1/2 full diversity orthogonal code and the uncoded 4PSK. The appropriate modulation schemes to provide the desired transmission rate for the space–time block codes, are 4PSK for the rate one code and 16 QAM for the rate 1/2 code.

Figure 4.29 presents the performance for four transmit antennas for space–time block codes at 3 bits/s/Hz. The rate one code and the uncoded system use 8PSK and the rate 3/4 code uses 16 QAM.

More details on the topic can be found in [42].

### 4.6 SPACE–TIME CONVOLUTIONAL CODES

In Section 4.2 the coding gain was defined as:

\[
\eta = \min_{c,e} \left( \prod_{i=1}^{r} \lambda_i \right)^{1/r}
\]
Figure 4.28 Bit error probability versus SNR for space–time block codes at 2 bits/s/Hz; 1 receive antenna.

Figure 4.29 Bit error probability versus SNR for space–time block codes at 3 bits/s/Hz; 1 receive antenna.
over all code word pairs. Due to the similarity of Inequality (4.33) to an error bound for trellis coded modulation, \( \eta \) is called the diversity gain (the slope of the pairwise error probability on a log–log plot) and \( \eta \) is called the coding gain (\( \eta^{-\theta} \) is an offset on a log–log plot).

Consider only codes that achieve maximum diversity gain. Of these, we search for codes that give the largest possible coding gain.

Similarly to Equations (2.8)–(2.13) in Chapter 2, consider a set of convolutional codes whose output at time \( k \) is:

\[
[x_1(k), x_2(k), \ldots, x_n(k)] = \mathbf{b}(kR)\mathbf{G}
\]

\[
= \mathbf{b}(kR)[G_1G_2\cdots G_n]
\]

where \( G_i, i = 1, \ldots, n \) is the \( i \)th column of the matrix \( \mathbf{G} \) which is given by:

\[
\mathbf{G} = \begin{bmatrix}
g_{11} & g_{12} & \cdots & g_{1n} 
g_{21} & g_{22} & \cdots & g_{2n} 
g_{31} & g_{32} & \cdots & g_{3n} 
g_{41} & g_{42} & \cdots & g_{4n} 
\vdots & \vdots & \ddots & \vdots 
g_{QR,1} & g_{QR,2} & \cdots & g_{QR,n}
\end{bmatrix}
\]

and \( \mathbf{b}(kR) = [b_1, \ldots, b_{QR}] \) is a length QR binary row vector, \( b_i \in \{0, 1\} \) for \( i = 1, \ldots, n \) are taken to be in an alphabet of size \( s \) so that \( g_{ij} \in \{0, 1, \ldots, s-1\} \). The arithmetic in Equation (4.81) is mod-s so \( x_i(j) \in \{0, 1, \ldots, s-1\} \) also. Thus, the output code word is \( (c_1(k), \ldots, c_n(k)) = (z(x_1(k)), \ldots, z(x_n(k))) \) where \( z[x] = \exp(j(2\pi x/s + \phi)) \) and \( 0 \leq \phi \leq 2\pi \) allows arbitrary rotation.

At each time slot, \( R \) bits are input into the convolutional encoder and the state is determined by \((Q-1)R\) bits. For space–time convolutional coding using Equation (4.81), the outputs \( x_1(k), \ldots, x_n(k) \) are each mapped into a constellation of size \( s \) and transmitted simultaneously from \( n \) antennas. The input to the encoder at time slot \( k \) is \( b_1, \ldots, b_R \). The state is given by \( b_{R+1}, \ldots, b_{QR} \). For the next time slot, new input bits are received and the old input bits are shifted to the right into the state which explains the subscript on \( \mathbf{b}(kR) \).

The results of the search for the good codes are presented in Tables 4.4–4.9 with the following notation:

- \( \eta_\iota \) (4.2) – coding gain for the trellis space–time codes defined in Section 4.2.
- \( \eta_c \) (4.6) – coding gain for the convolutional space–time codes defined in Section 4.6.
- \( \epsilon_{p\min} \) – minimum effective product distance.
- \( L \) – total number of time slots considered in the bound calculation.
- \( \eta_{AP}(L) \) – considers all trellis paths describing error events of length \( L \) or longer.
- \( \eta_{CP}(L) \) – considers sets of code word pairs (continuous error paths) and searches for smallest upper bound.
- \( \tilde{Q} \) – the minimum length error event of \( \tilde{Q} \) slots.

Performance results are shown in Figures 4.30–4.34.

More details on the topic can be found in [35, 49–55].

### 4.7 ALGEBRAIC SPACE–TIME CODES

Let \( C \) be a binary convolutional code of rate \( 1/L_\iota \) with the transfer function encoder \( \mathbf{Y}(D) = X(D)\mathbf{G}(D) \), where \( \mathbf{G}(D) = [G_1(D)G_2(D)\cdots G_{L_\iota}(D)] \). In the natural space–time formatting of \( C \) for BPSK transmission, the output sequence corresponding to \( Y_i(D) = X(D)G_i(D) \) is assigned to the \( i \)th transmit antenna. The number of transmit antennas is \( L_\iota \). The resulting space–time code \( C \) satisfies the binary rank criterion under relatively mild conditions on the connection polynomials \( G_i(D) \).
### Table 4.4  Optimum $q$-state 2 b/s/Hz 4PSK space–time codes [35] © 2002, IEEE

<table>
<thead>
<tr>
<th>$q$</th>
<th>$\eta_t(4.2)$</th>
<th>$\eta_c(4.6)$</th>
<th>$\bar{\eta}_{AP}(\hat{Q})$</th>
<th>$\bar{\eta}_{CP}(\hat{Q})$</th>
<th>$G^T$</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>2</td>
<td>$\sqrt{8}$</td>
<td>3.54</td>
<td>3.80</td>
<td>$\begin{bmatrix} 2 &amp; 0 &amp; 1 &amp; 2 \ 2 &amp; 2 &amp; 2 &amp; 1 \end{bmatrix}$</td>
</tr>
<tr>
<td>8</td>
<td>$\sqrt{12}$</td>
<td>4</td>
<td>3.42</td>
<td>4.00</td>
<td>$\begin{bmatrix} 0 &amp; 2 &amp; 1 &amp; 0 &amp; 2 \ 2 &amp; 1 &amp; 0 &amp; 2 &amp; 2 \end{bmatrix}$</td>
</tr>
<tr>
<td>16</td>
<td>$\sqrt{12}$</td>
<td>$\sqrt{32}$</td>
<td>5.76</td>
<td>6.24</td>
<td>$\begin{bmatrix} 0 &amp; 2 &amp; 1 &amp; 1 &amp; 2 &amp; 0 \ 2 &amp; 2 &amp; 1 &amp; 2 &amp; 0 &amp; 2 \end{bmatrix}$</td>
</tr>
<tr>
<td>32</td>
<td>$\sqrt{12}$</td>
<td>6</td>
<td>5.63</td>
<td>6.33</td>
<td>$\begin{bmatrix} 2 &amp; 0 &amp; 1 &amp; 2 &amp; 1 &amp; 2 &amp; 2 \ 2 &amp; 2 &amp; 0 &amp; 1 &amp; 2 &amp; 0 &amp; 2 \end{bmatrix}$</td>
</tr>
</tbody>
</table>

### Table 4.5  Optimum $q$-state 2 b/s/Hz 4PSK space–time codes [35] © 2002, IEEE

<table>
<thead>
<tr>
<th>$q$</th>
<th>$\eta_t$</th>
<th>$\eta_c$</th>
<th>$e_{\rho \text{ min}}$</th>
<th>$\bar{\eta}_{AP}(L)$</th>
<th>$\bar{\eta}_{CP}(L)$</th>
<th>$G^T$</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>2</td>
<td>$\sqrt{8}$</td>
<td>8</td>
<td>3.54</td>
<td>3.80</td>
<td>$\begin{bmatrix} 2 &amp; 0 &amp; 1 &amp; 2 \ 2 &amp; 2 &amp; 2 &amp; 1 \end{bmatrix}$</td>
</tr>
<tr>
<td>8</td>
<td>$\sqrt{12}$</td>
<td>4</td>
<td>16</td>
<td>3.42</td>
<td>4.00</td>
<td>$\begin{bmatrix} 0 &amp; 2 &amp; 1 &amp; 0 &amp; 2 \ 2 &amp; 1 &amp; 0 &amp; 2 &amp; 2 \end{bmatrix}$</td>
</tr>
<tr>
<td>16</td>
<td>$\sqrt{12}$</td>
<td>$\sqrt{32}$</td>
<td>32</td>
<td>5.76</td>
<td>6.24</td>
<td>$\begin{bmatrix} 0 &amp; 2 &amp; 1 &amp; 1 &amp; 2 &amp; 0 \ 2 &amp; 2 &amp; 1 &amp; 2 &amp; 0 &amp; 2 \end{bmatrix}$</td>
</tr>
<tr>
<td>32</td>
<td>$\sqrt{12}$</td>
<td>6</td>
<td>36</td>
<td>5.63</td>
<td>6.33</td>
<td>$\begin{bmatrix} 2 &amp; 0 &amp; 1 &amp; 2 &amp; 1 &amp; 2 &amp; 2 \ 2 &amp; 2 &amp; 0 &amp; 1 &amp; 2 &amp; 0 &amp; 2 \end{bmatrix}$</td>
</tr>
</tbody>
</table>

### Table 4.6  Optimum $q$-state 1 b/s/Hz BSK space–time codes [35] © 2002, IEEE

<table>
<thead>
<tr>
<th>$q$</th>
<th>$\eta_c$</th>
<th>$e_{\rho \text{ min}}$</th>
<th>$\bar{\eta}_{AP}(L)$</th>
<th>$\bar{\eta}_{CP}(L)$</th>
<th>$G^T$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>4</td>
<td>4</td>
<td>4.00</td>
<td>4.00</td>
<td>$\begin{bmatrix} 0 &amp; 1 \ 1 &amp; 1 \end{bmatrix}$</td>
</tr>
<tr>
<td>4</td>
<td>$\sqrt{48}$</td>
<td>12</td>
<td>7.33</td>
<td>6.93</td>
<td>$\begin{bmatrix} 0 &amp; 1 &amp; 1 \ 1 &amp; 0 &amp; 1 \end{bmatrix}$</td>
</tr>
<tr>
<td>4</td>
<td>$\sqrt{32}$</td>
<td>8</td>
<td>7.33</td>
<td>7.73</td>
<td>$\begin{bmatrix} 0 &amp; 1 &amp; 1 \ 1 &amp; 1 &amp; 1 \end{bmatrix}$</td>
</tr>
<tr>
<td>8</td>
<td>$\sqrt{80}$</td>
<td>20</td>
<td>10.06</td>
<td>10.47</td>
<td>$\begin{bmatrix} 1 &amp; 0 &amp; 1 &amp; 1 \ 1 &amp; 0 &amp; 1 \ 0 &amp; 1 &amp; 1 &amp; 1 \end{bmatrix}$</td>
</tr>
<tr>
<td>16</td>
<td>$\sqrt{112}$</td>
<td>28</td>
<td>11.38</td>
<td>14.27</td>
<td>$\begin{bmatrix} 1 &amp; 1 &amp; 0 &amp; 1 &amp; 1 \ 0 &amp; 1 &amp; 1 &amp; 1 &amp; 1 \end{bmatrix}$</td>
</tr>
</tbody>
</table>

† denotes that the code is catastrophic
### Table 4.7 Optimum $q$-state 1 b/s/Hz BSK 3-space–time codes [35] © 2002, IEEE

<table>
<thead>
<tr>
<th>$q$</th>
<th>$\eta_c$</th>
<th>$e_p$</th>
<th>$\bar{\eta}_{AP}(L)$</th>
<th>$\bar{\eta}_{CP}(L)$</th>
<th>$G_T$</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>4</td>
<td>$\frac{64}{27}$</td>
<td>5.08</td>
<td>5.08</td>
<td>$\begin{bmatrix} 0 &amp; 1 &amp; 1 \ 1 &amp; 0 &amp; 1 \ 1 &amp; 1 &amp; 1 \end{bmatrix}$</td>
</tr>
<tr>
<td>8</td>
<td>$256^{1/3}$</td>
<td>$\frac{256}{27}$</td>
<td>7.65</td>
<td>7.05</td>
<td>$\begin{bmatrix} 1 &amp; 0 &amp; 0 &amp; 1 \ 1 &amp; 0 &amp; 1 &amp; 0 \ 1 &amp; 1 &amp; 1 &amp; 1 \end{bmatrix}$</td>
</tr>
<tr>
<td>8</td>
<td>$192^{1/3}$</td>
<td>$\frac{64}{9}$</td>
<td>7.65</td>
<td>8.32</td>
<td>$\begin{bmatrix} 1 &amp; 0 &amp; 1 &amp; 1 \ 1 &amp; 1 &amp; 0 &amp; 1 \ 1 &amp; 0 &amp; 1 &amp; 1 \end{bmatrix}$</td>
</tr>
<tr>
<td>16</td>
<td>8</td>
<td>$\frac{512}{27}$</td>
<td>10.00</td>
<td>10.18</td>
<td>$\begin{bmatrix} 1 &amp; 0 &amp; 0 &amp; 1 &amp; 1 \ 1 &amp; 1 &amp; 0 &amp; 1 &amp; 0 \ 1 &amp; 1 &amp; 1 &amp; 0 &amp; 1 \end{bmatrix}$</td>
</tr>
</tbody>
</table>

† denotes that the code is catastrophic

### Table 4.8 Optimum $q$-state 1 b/s/Hz BSK 4-space–time codes [35] © 2002, IEEE

<table>
<thead>
<tr>
<th>$q$</th>
<th>$\eta_c$</th>
<th>$e_p$</th>
<th>$\bar{\eta}_{AP}(L)$</th>
<th>$\bar{\eta}_{CP}(L)$</th>
<th>$G_T$</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>4</td>
<td>1</td>
<td>5.97</td>
<td>5.97</td>
<td>$\begin{bmatrix} 0 &amp; 1 &amp; 0 &amp; 1 \ 0 &amp; 1 &amp; 1 &amp; 1 \ 1 &amp; 0 &amp; 1 &amp; 0 \ 1 &amp; 1 &amp; 1 &amp; 0 \end{bmatrix}$</td>
</tr>
<tr>
<td>16</td>
<td>$1280^{1/4}$</td>
<td>5</td>
<td>8.11</td>
<td>8.37</td>
<td>$\begin{bmatrix} 1 &amp; 0 &amp; 0 &amp; 0 &amp; 1 \ 1 &amp; 0 &amp; 1 &amp; 1 &amp; 1 \ 1 &amp; 1 &amp; 0 &amp; 1 &amp; 1 \ 1 &amp; 1 &amp; 1 &amp; 1 &amp; 0 \end{bmatrix}$</td>
</tr>
<tr>
<td>16</td>
<td>$1024^{1/4}$</td>
<td>4</td>
<td>7.99</td>
<td>9.32</td>
<td>$\begin{bmatrix} 0 &amp; 1 &amp; 1 &amp; 0 &amp; 1 \ 1 &amp; 1 &amp; 0 &amp; 0 &amp; 1 \ 1 &amp; 1 &amp; 1 &amp; 1 &amp; 0 \ 1 &amp; 1 &amp; 1 &amp; 1 &amp; 1 \end{bmatrix}$</td>
</tr>
<tr>
<td>32</td>
<td>$4352^{1/4}$</td>
<td>17</td>
<td>9.80</td>
<td>10.38</td>
<td>$\begin{bmatrix} 1 &amp; 0 &amp; 0 &amp; 0 &amp; 0 &amp; 1 \ 1 &amp; 0 &amp; 1 &amp; 1 &amp; 1 &amp; 1 \ 1 &amp; 1 &amp; 1 &amp; 0 &amp; 1 &amp; 0 \ 1 &amp; 1 &amp; 1 &amp; 1 &amp; 0 &amp; 0 \end{bmatrix}$</td>
</tr>
</tbody>
</table>

† denotes that the code is catastrophic
Table 4.9 Optimum $q$-state 2 b/s/Hz 4 PSK space–time code using three transmit antennas [35] © 2002, IEEE

<table>
<thead>
<tr>
<th>$q$</th>
<th>$\eta_c$</th>
<th>$e_{\min}$</th>
<th>$\bar{N}_{AP}(L)$</th>
<th>$\bar{N}_{CP}(L)$</th>
<th>$G^T$</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>$32^{1/3}$</td>
<td>$\frac{256}{27}$</td>
<td>3.90</td>
<td>4.72</td>
<td>$\begin{bmatrix} 0 &amp; 2 &amp; 1 &amp; 2 &amp; 2 &amp; 0^T \ 1 &amp; 2 &amp; 2 &amp; 0 &amp; 0 &amp; 2 \ 2 &amp; 2 &amp; 0 &amp; 2 &amp; 1 &amp; 2 \end{bmatrix}$</td>
</tr>
</tbody>
</table>

Figures 4.30 and 4.31 demonstrate the performance comparisons of some best 2 b/s/Hz, QPSK, $q$-state STCs with two transmit and two receive antennas. Figure 4.30 illustrates the performance comparison with the SNR per receive antenna set to $nE_s/N_0$. Figure 4.31 presents the performance comparisons of some 2 b/s/Hz, QPSK, four-state STCs with two transmit and two receive antennas.
Figure 4.32 Performance comparisons of some 2 b/s/Hz, QPSK, 16- and 32-state STCs with two transmit and two receive antennas.

Figure 4.33 Performance comparisons of some 2 b/s/Hz, QPSK, eight-state STCs with two transmit and two receive antennas.
Figure 4.34 Performance of some 1 b/s/Hz, BPSK, \( q \)-state STCs with two transmit and two receive or three transmit and three receive antennas (performance of the best noncatastrophic codes similar).

4.7.1 Full spatial diversity

Code \( C \), associated with the \( 1/L_t \) convolutional code \( C \), satisfies the binary rank criterion, and thus achieves full spatial diversity for BPSK transmission if and only if the transfer function matrix \( G(D) \) of \( C \) has full rank \( L_t \) as a matrix of coefficients over the binary field \( \mathbb{F} \). This result follows directly from the stacking construction and can be easily generalized to recursive convolutional codes.

It is straightforward to see that the zeros symmetry codes satisfy the stacking construction conditions. However, the set of binary rate \( 1/L_t \) convolutional codes with the optimal free distance \( d_{\text{free}} \) offers a richer class of space–time codes as their associated natural space–time codes usually achieve full spatial diversity. Furthermore, these codes outperform the zeros symmetry codes uniformly. A collection of these codes is given in Table 4.10.

4.7.2 QPSK modulation

Experimentally, codes obtained by replacing each zero in the binary generator matrix by a two yield the best simulated frame error rate performance in most cases. These codes are reported in Table 4.11 for different numbers of transmit antennas and constraint lengths \( v \) and are used in generating the simulation results in the next section.

A space–time code has zeros symmetry if every baseband code word difference \( f(c) - f(e) \) is upper and lower triangular and has appropriate non-zero entries to ensure full rank. The zeros symmetry property is sufficient for full rank but not necessary. Some simulation results for the system performance are shown in Figures 4.35–4.40.

4.8 DIFFERENTIAL SPACE–TIME MODULATION

When channel estimation is not available, a differential modulation and detection might be a solution. We start with a simple transmission scheme [65] for exploiting diversity given by two transmit
Table 4.10  Natural full diversity space–time convolutional codes with optimal free distance for BPSK modulation [56] © 2002, IEEE

<table>
<thead>
<tr>
<th>$L_t$</th>
<th>$v$</th>
<th>Connection polynomials</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>2</td>
<td>5, 7</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>64, 74</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>46, 72</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>65, 57</td>
</tr>
<tr>
<td>6</td>
<td>5</td>
<td>554, 744</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>54, 64, 74</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>52, 66, 76</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>47, 53, 75</td>
</tr>
<tr>
<td></td>
<td></td>
<td>654, 624, 764</td>
</tr>
</tbody>
</table>

Table 4.11  Linear $\mathbb{Z}_4$ space–time codes for QPSK modulation [56] © 2002, IEEE

<table>
<thead>
<tr>
<th>$L$</th>
<th>$v$</th>
<th>Connection polynomials</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>1</td>
<td>$1 + 2D, 2 + D$.</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>$1 + 2D + D^2, 1 + D + D^2$.</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>$1 + D + 2D^2 + D^3, 1 + D + D^2 + D^3$.</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>$1 + 2D + 2D^2 + D^3 + D^4, 1 + D + D^2 + 2D^3 + D^4$.</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>$1 + D + 2D^2 + D^3 + 2D^4 + D^5, 1 + 2D + D^2 + D^3 + D^4 + D^5$.</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>$1 + 2D + 2D^2, 2 + D + 2D^2, 2 + 2D + D^3$.</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>$1 + 2D + D^2 + D^3, 1 + D + 2D^2 + D^3, 1 + D + D^2 + D^3$.</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>$1 + 2D + D^2 + 2D^3 + D^4, 1 + D + 2D^2 + D^3 + D^4, 1 + 2D + D^2 + D^3 + D^4$.</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>$1 + 2D + 2D^2 + D^3 + D^4 + D^5, 1 + 2D + 2D^2 + 2D^3 + D^4 + D^5$.</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>$1 + 2D + 2D^2 + 2D^3, 2 + D + 2D^2 + 2D^3, 2 + 2D + D^2 + 2D^3, 2 + 2D + 2D^3 + D^3$.</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>$1 + 2D + D^2 + 2D^3 + D^4, 1 + 2D + D^2 + D^3 + D^4, 1 + D + 2D^2 + D^3 + D^4$.</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>$1 + 2D + D^2 + 2D^3 + D^4 + D^5, 1 + D + 2D^2 + D^3 + D^4 + D^5$.</td>
</tr>
<tr>
<td>5</td>
<td>4</td>
<td>$1 + 2D + 2D^2 + 2D^3 + 2D^4, 2 + D + 2D^2 + 2D^3 + 2D^4, 2 + 2D + D^2 + 2D^3 + D^4$.</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>$1 + D + D^2 + D^3 + 2D^4 + D^5, 1 + D + D^2 + D^3 + 2D^4 + D^5$.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$1 + D + D^2 + D^3 + D^4 + D^5, 1 + D + 2D^2 + D^3 + 2D^4 + D^5$.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$1 + 2D + D^2 + D^3 + D^4 + D^5$.</td>
</tr>
</tbody>
</table>
Figure 4.35 Performance of BPSK space–time codes with $L_t = 1$. The number of transmit antennas is represented by $L_t$, receive antennas by $L_r$, and constraint lengths by $K$. The number of bits per frame is 100.

Figure 4.36 FER for BPSK space–time codes with $L_t = 5$ and $K = 6$. 
Figure 4.37 Performance of four-state QPSK space–time codes with $L_t = 2$. Four-state codes due to Tarokh, Seshadri, and Calderbank (TSC) [2], Baro, Bauch, and Hansmann (BBH) [31], Grimm, Fitz, and Krogmeier (GFK) [33], and the new linear $\mathbb{Z}_4$ code in Table 4.11 with two and four receive antennas.

Figure 4.38 Performance of eight-state QPSK space–time codes with $L_t = 2$. 
Figure 4.39 FER for QPSK space–time codes with $L_t = 4$ and $L_r = 1$. The new 64-state space–time code for four transmit antennas and the TSC 64-state code (i.e. delay diversity).

Figure 4.40 Performance of an algebraic short block and an orthogonal code with $L_t = 3$ and $L_r = 1$. The new algebraic code is used with QPSK modulation and the orthogonal code is used with 16 QAM. The increased size of the constellation in the case of the orthogonal code is necessary to support the same throughput [6]. More details on the topic can be found in [56–64].
antennas when neither the transmitter nor the receiver has access to channel state information. At the receiver, decoding is achieved with low decoding complexity. The transmission provides full spatial diversity and requires no channel state side information at the receiver. The scheme can be considered as the extension of conventional differential detection schemes to two transmit antennas.

In traditional differential phase shift keying (DPSK) the data is encoded in the difference of the phase of two consecutive symbols. For $b$ bits per symbol, the symbol would usually have the form:

$$\Delta(l) = e^{i\theta(l)} = e^{2\pi i l/L}, \quad l = 0, 1, \ldots L - 1$$  \hspace{1cm} (4.83)

where $i = \sqrt{-1}$ and $L = 2^b$. If the data sequence is $z_1, z_2, z_3, \ldots$ the transmitted symbol sequences would be $s_0, s_1, s_2, \ldots$ where, at time $t$,

$$s_t = \Delta(z_t)s_{t-1}$$  \hspace{1cm} (4.84)

Bits are mapped into the symbols by using Gray mapping, e.g. for $b = 2$:

$$M(z) \Rightarrow \Delta(z) = \Delta(l)$$

$$\Delta(00) = \Delta(0) = e^{2\pi i \times 0/4} = 1$$
$$\Delta(01) = \Delta(1) = e^{2\pi i \times 1/4} = e^{i\pi/2}$$
$$\Delta(10) = \Delta(3) = e^{2\pi i \times 3/4} = e^{-i\pi/2}$$
$$\Delta(11) = \Delta(2) = e^{2\pi i \times 2/4} = -1$$

This ensures that the most probable symbol errors cause the minimum number of bit errors.

Signal $r_t$ from the received signal sequence $r_0, r_1, r_2, \ldots$ can be represented as:

$$r_t = h_ts_t + n_t$$  \hspace{1cm} (4.86)

where $h_t$ is the channel gain. The receiver would extract the information by processing the received signal samples as follows:

$$\hat{\theta}_t = \arg r_t r_{t-1}^*$$  \hspace{1cm} (4.87)

which, for $h_t \approx h_{t-1}, |h_t| = 1$ and no noise, gives:

$$\hat{\theta}_t = \arg h_t s_t h_{t-1}^* s_{t-1}^* = \arg |h_t|^2 \Delta(z_t) |s_{t-1}|^2 = \arg \Delta(z_t)$$  \hspace{1cm} (4.88)

By using analogy with the previous discussion we will now extend this principle to two-dimensional constellations. Later in the chapter we will generalize these schemes to multidimensional constellations.

Let us restrict the constellation $\mathcal{A}$ to $2^b$-PSK for some $b = 1, 2, 3, \ldots$, but in reality only BPSK, QPSK, and 8PSK are of interest. Thus,

$$\mathcal{A} = \left\{ e^{2\pi i k/2^b} \sqrt{2} |k = 0, 1, \ldots, 2^b - 1 \right\}$$  \hspace{1cm} (4.89)

In order to implement Equation (4.84) in a two-dimensional constellation, let us first consider the following relation between the vectors.

Given a pair of $2^b$-PSK constellation symbols, $x_1$ and $x_2$, we first observe that the complex vectors $(x_1, x_2)$ and $(-x_2^*, x_1^*)$ are orthogonal to each other and have unit lengths defined by $\sum_i x_i x_i^* = 1$. Any two-dimensional vector $X = (x_1, x_2)$ can be uniquely represented in the orthonormal basis given by these vectors. In other words, there exists a unique complex vector $P_X = (A_X B_X)$ such that $A_X$ and $B_X$ satisfy the vector equation:

$$(x_3, x_4) = A_X (x_1, x_2) + B_X (-x_2^*, x_1^*)$$  \hspace{1cm} (4.90)
or
\[
\begin{align*}
x_3 &= A_X x_1 - B_X x_2^* \\
x_4 &= A_X x_2 + B_X x_1^* \\
x_3 x_1^* &= A_X x_1 x_1^* - B_X x_2^* x_1^* \\
x_4 x_2^* &= A_X x_2 x_2^* + B_X x_1^* x_2^*
\end{align*}
\]
giving \(A_X\) and \(B_X\) as:
\[
\begin{align*}
A_X &= x_3 x_1^* + x_4 x_2^* \\
B_X &= -x_3 x_2 + x_4 x_1
\end{align*}
\]

These relations will be used later to implement Equation (4.84). For an equivalent implementation of Equation (4.85) we will need a few additional definitions.

We define the set \(\mathcal{V}_X\) to consist of all the vectors \(P_X, X \notin A \times A\). The set \(\mathcal{V}_X\) has the following properties:

- **Property A**: It has \(2^b\) elements corresponding to the pairs \((x_3,x_4)\) of constellation symbols.

- **Property B**: All elements of \(\mathcal{V}_X\) have unit length.

- **Property C**: For any two distinct elements \(X = (x_1,x_2)\) and \(Y = (y_1,y_2)\) of \(A \times A\)

\[||P_X - P_Y|| = ||(x_1,x_2) - (y_1,y_2)||\]

- **Property D**: The minimum distance between any two distinct elements of \(\mathcal{V}_X\) is equal to the minimum distance of the \(2^b\)-PSK constellation \(A\).

The above properties hold because the mapping \(\mathcal{X} \rightarrow P_X\) is just a change of basis from the standard basis given by vectors \{(1 0), (0 1)\} to the orthonormal basis given by \{(\(x_1 x_2\), \(-x_2^* x_1^*)\)}, which preserves the distances between the points of the two-dimensional complex space.

The first ingredient of construction is the choice of an arbitrary set \(V\) having Properties A and B. It is also handy if \(V\) has Properties C and D as well. As a natural choice for such a set \(V\), we may fix an arbitrary pair \(X \in A \times A\) and let \(V = \mathcal{V}_X\). Because the \(2^b\)-PSK constellation \(A\) always contains the signal point \(1/\sqrt{2}\), we choose to fix \(X = ((1/\sqrt{2})(1/\sqrt{2}))\).

We also need an arbitrary bijective mapping \(M\) of blocks of \(2b\) bits onto \(V\). Among all the possibilities for \(M\), we choose the mapping analogous to Equation (4.85). Given a block \(B\) of \(2b\) bits, the first \(b\) bits are mapped into a constellation symbol \(a_3\) and the second \(b\) bits are mapped into a constellation symbol \(a_4\) using Gray mapping.

Let \(a_1 = a_2 = 1/\sqrt{2}\), then \(M(B) = (A(B) B(B))\) is defined by Equation (4.91):
\[
\begin{align*}
A(B) &= a_3 a_1^* + a_4 a_2^* \\
B(B) &= -a_3 a_2 + a_3 a_1
\end{align*}
\]

Clearly, \(M\) maps any \(2b\) bits onto \(V\). Conversely, given \((A(B) B(B))\), the pair \((a_3, a_4)\) is recovered by Equation (4.90):
\[
(a_3 a_4) = A(B)(a_1 a_2) + B(B)(a_2^* a_1^*)
\]

The block \(B\) is then constructed by inverse Gray mapping of \(a_3\) and \(a_4\).

### 4.8.1 The encoding algorithm

The transmitter begins the transmission by sending arbitrary symbols \(s_1\) and \(s_2\) from transmit antennas one and two respectively at time one, and symbols \(-s_2^*\) and \(s_1^*\) at time two unknown to the receiver. These two transmissions do not convey any information. The transmitter then encodes the rest of the data in an inductive manner. Suppose that \(s_{2t-1}\) and \(s_{2t}\) are sent, respectively, from transmit antennas one and two at time \(2t-1\), and that \(-s_{2t}^*, s_{2t-1}^*\) are sent, respectively, from antennas one and two.
at time \(2t\). At time \(2t + 1\), a block of \(2b\) bits \(B_{2t+1}\) arrives at the encoder. The transmitter uses the mapping \(\mathcal{M}\) given by Equation (4.92) and computes \(\mathcal{M}(B_{2t+1}) = (A(B_{2t+1}) \cdot B(B_{2t+1}))\). Then, in accordance with Equation (4.92) it computes:

\[
(s_{2t+1} s_{2t+2}) = A(B_{2t+1})(s_{2t-1} s_{2t}) + B(B_{2t+1})(s^*_{2t} s^*_{2t-2})
\]  

(4.94)

The transmitter then sends \(s_{2t+1}\) and \(s_{2t+2}\), respectively, from transmit antennas one and two at time \(2t + 1\), and \(-s^*_{2t+2}, s^*_{2t+1}\) from antennas one and two at time \(2t + 2\). This process is inductively repeated until the end of the frame (or end of the transmission). The block diagram of the encoder is given in Figure 4.41.

### 4.8.1.1 Example

We assume that the constellation is BPSK \((b = 1)\) consisting of the points \(-1/\sqrt{2}\) and \(1/\sqrt{2}\). Then the set \(\mathcal{V} = \{(1, 0), (0, 1), (-1, 0), (-0, -1)\}\). Recall that the Gray mapping maps a bit \(i = 0, 1\) to \((-1)^i/\sqrt{2}\). We set \(a_1 = a_2 = 1/\sqrt{2}\). Then the mapping \(\mathcal{M}\) maps two bits onto \(\mathcal{V}\) and is given by:

\[
\mathcal{M}(00) = (1, 0)
\]

\[
\mathcal{M}(10) = (0, 1)
\]

\[
\mathcal{M}(01) = (0, -1)
\]

\[
\mathcal{M}(11) = (-1, 0)
\]  

(4.95)

Now suppose that at time \(2t - 1\), \(s_{2t-1} = 1/\sqrt{2}\) and \(s_{2t} = -1/\sqrt{2}\) are sent, respectively, from antennas one and two, and at time \(2t\), \(-s^*_{2t} = 1/\sqrt{2}\) and \(s^*_{2t-1} = 1/\sqrt{2}\) are sent. Suppose that the input to the encoder at time \(2t + 1\) is the block of bits 10. Since \(\mathcal{M}(10) = (1, 0)\), we have \(A(10) = 0\) and \(B(10) = 1\). Then the values \(s_{2t+1}\) and \(s_{2t+2}\) corresponding to input bits 10 are computed as follows:

\[
(s_{2t+1} s_{2t+2}) = 0 \cdot \left(\frac{1}{\sqrt{2}} \cdot -1\right) + 1 \cdot \left(\frac{1}{\sqrt{2}} \cdot 1\right)
\]

(4.96)

\[
= \left(\frac{1}{\sqrt{2}} \cdot 1\right)
\]

Thus, at time \(2t + 1\), \(s_{2t+1} = 1/\sqrt{2}\) and \(s_{2t+2} = 1/\sqrt{2}\) are sent, respectively, from antennas one and two, and at time \(2t + 2\), \(-s^*_{2t+2} = -1/\sqrt{2}\) and \(s^*_{2t+1} = 1/\sqrt{2}\) are sent, respectively, from antennas one and two. Transmitted symbols at time \(2t + 1\) and \(2t + 2\) correspond to the input bits 00, 10, 01 and 11 for this scenario. The results are summarized in Tables 4.12 and 4.13.

### 4.8.2 Differential decoding

For notational simplicity, we first present the results for one receive antenna. Write \(r_t\) for \(r^1_t\), \(\eta_t\) for \(\eta^1_t\) and \(\alpha_1, \alpha_2\), respectively, for \(\alpha_{1,1}, \alpha_{2,1}\), knowing that this can cause no confusion since there is only one receive antenna.
which gives:

$$r_{2t-1}, r_{2t}, r_{2t+1}, \text{and } r_{2t+2} \text{ are received signal samples. Let}$$

$$\Lambda(\alpha_1, \alpha_2) = \begin{bmatrix} \alpha_1 & \alpha_2^\ast \\ \alpha_2 & -\alpha_1^\ast \end{bmatrix}$$  \hspace{1cm} (4.97)$$

and

$$N_{2t-1} = (\eta_{2t-1} \eta_{2t-1}^\ast)$$  \hspace{1cm} (4.98)$$

Then the received signal can be represented as:

$$(r_{2t-1} r_{2t-1}^\ast) = (s_{2t-1} s_{2t-1}) \Lambda(\alpha_1, \alpha_2) + N_{2t-1}$$  \hspace{1cm} (4.99)$$

and

$$(r_{2t+1} r_{2t+1}^\ast) = (s_{2t+1} s_{2t+1}) \Lambda(\alpha_1, \alpha_2) + N_{2t+1}$$  \hspace{1cm} (4.100)$$

The receiver will create

$$(r_{2t+1} r_{2t+2}^\ast) \cdot (r_{2t-1} r_{2t-1}^\ast) = (s_{2t+1} s_{2t+2}) \Lambda(\alpha_1, \alpha_2) \Lambda^\ast(\alpha_1, \alpha_2) (s_{2t-1} s_{2t-1}^\ast) + \Lambda(\alpha_1, \alpha_2) N_{2t-1}^s + N_{2t+1} N_{2t-1}^s$$  \hspace{1cm} (4.101)$$

which gives:

$$r_{2t+1} r_{2t-1}^\ast + r_{2t+2} s_{2t} = (|\alpha_1|^2 + |\alpha_2|^2) (s_{2t+1} s_{2t-1}^\ast + s_{2t+1} s_{2t-1}^s) + \Lambda(\alpha_1, \alpha_2) N_{2t-1}^s + N_{2t+1} \Lambda^\ast(\alpha_1, \alpha_2) (s_{2t-1} s_{2t})^s + N_{2t+1} N_{2t-1}^s$$

For a more compact representation we use the following notation:

$$R = r_{2t+1} r_{2t-1}^\ast + r_{2t+2} r_{2t}$$

$$N_1 = (s_{2t+1} s_{2t+2}) \Lambda(\alpha_1, \alpha_2) N_{2t-1} + N_{2t+1} \Lambda^\ast(\alpha_1, \alpha_2) (s_{2t-1} s_{2t})^s + N_{2t+1} N_{2t-1}^s$$

<table>
<thead>
<tr>
<th>Input bits at time $2t+1$</th>
<th>Antenna 1</th>
<th>Antenna 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>$\frac{1}{\sqrt{2}}$</td>
<td>$\frac{1}{\sqrt{2}}$</td>
</tr>
<tr>
<td>10</td>
<td>$\frac{1}{\sqrt{2}}$</td>
<td>$\frac{1}{\sqrt{2}}$</td>
</tr>
<tr>
<td>01</td>
<td>$-\frac{1}{\sqrt{2}}$</td>
<td>$-\frac{1}{\sqrt{2}}$</td>
</tr>
<tr>
<td>11</td>
<td>$-\frac{1}{\sqrt{2}}$</td>
<td>$\frac{1}{\sqrt{2}}$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Input bits at time $2t+1$</th>
<th>Antenna 1</th>
<th>Antenna 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>$\frac{1}{\sqrt{2}}$</td>
<td>$\frac{1}{\sqrt{2}}$</td>
</tr>
<tr>
<td>10</td>
<td>$\frac{1}{\sqrt{2}}$</td>
<td>$\frac{1}{\sqrt{2}}$</td>
</tr>
<tr>
<td>01</td>
<td>$-\frac{1}{\sqrt{2}}$</td>
<td>$-\frac{1}{\sqrt{2}}$</td>
</tr>
<tr>
<td>11</td>
<td>$-\frac{1}{\sqrt{2}}$</td>
<td>$-\frac{1}{\sqrt{2}}$</td>
</tr>
</tbody>
</table>
which gives
\[ \mathcal{R}_1 = (|\alpha_1|^2 + |\alpha_2|^2)A(B_{2r-1}) + \mathcal{N}_1 \] (4.102)

For the second vector term in the right side of Equation (4.94) we have:
\[ (r_{2r} - r_{2r-1}^*) = (-s_{2r}^* s_{2r-1}^*)A(\alpha_1, \alpha_2) + N_{2r} \]
where
\[ N_{2r} = (\eta_{2r} - \eta_{2r-1}) \]

The receiver will now create
\[ (r_{2r+1}^* r_{2r+2}) \cdot (r_{2r} - r_{2r-1}^*) = (s_{2r+1} s_{2r+2})A(\alpha_1, \alpha_2)A^*(\alpha_1, \alpha_2)(-s_{2r} s_{2r-1}) \]
\[ + (s_{2r+1} s_{2r+2})A(\alpha_1, \alpha_2)N_{2r}^* \]
\[ + N_{2r+1}A^*(\alpha_1, \alpha_2)(-s_{2r}^* s_{2r-1}^*)^* + N_{2r+1}N_{2r}^* \]

giving:
\[ r_{2r+1}r_{2r}^* - r_{2r+2}^* r_{2r-1} = (|\alpha_1|^2 + |\alpha_2|^2)(-s_{2r+1} s_{2r+2}^* + s_{2r+2} s_{2r-1}^*) \]
\[ + (s_{2r+1} s_{2r+2})A(\alpha_1, \alpha_2)N_{2r}^* \]
\[ + N_{2r+1}A^*(\alpha_1, \alpha_2)(-s_{2r}^* s_{2r-1}^*)^* + N_{2r+1}N_{2r}^* \]

With the notation
\[ \mathcal{R}_2 = r_{2r+1}r_{2r}^* - r_{2r+2}^* r_{2r-1} \]
\[ \mathcal{N}_2 = +(s_{2r+1} s_{2r+2})A(\alpha_1, \alpha_2)N_{2r}^* + N_{2r+1}A^*(\alpha_1, \alpha_2)(-s_{2r}^* s_{2r-1}^*)^* + N_{2r+1}N_{2r}^* \]
we have
\[ \mathcal{R}_2 = (|\alpha_1|^2 + |\alpha_2|^2)B(B_{2r-1}) + \mathcal{N}_2 \] (4.103)

The final result can be represented as:
\[ (\mathcal{R}_1, \mathcal{R}_2) = (|\alpha_1|^2 + |\alpha_2|^2)(A(B_{2r-1}) B(B_{2r-1})) + (\mathcal{N}_1, \mathcal{N}_2) \] (4.104)

Because the elements of \( \mathcal{V} \) have equal length, to compute \((A(B_{2r-1}) B(B_{2r-1}))\), the receiver now computes the closest vector of \( \mathcal{V} \) to \((\mathcal{R}_1, \mathcal{R}_2)\). Once this vector is computed, the inverse mapping of \( \mathcal{M} \) is applied and the transmitted bits are recovered (see Table 4.12 and Table 4.13). The receiver block diagram and BER curves are given in Figures 4.42 and 4.43 respectively. As in the traditional, one dimensional DPSK system, there is loss in the performance (about 3 dB at BER = 0.001).

More results on the topic can be found in [65–76].

4.9 MULTIPLE TRANSMIT ANTENNA DIFFERENTIAL DETECTION FROM GENERALIZED ORTHOGONAL DESIGNS

In this section we explicitly construct multiple transmit antenna differential encoding/decoding schemes based on generalized orthogonal designs. These constructions generalize the two transmit antenna differential detection scheme discussed in the previous section. The presentation is based on [77, 78].

![Figure 4.42 Receiver block diagram.](image)
4.9.1 Differential encoding

We consider the specific example code, $\Gamma_{84}$, and only one receive antenna, $M = 1$. The generalization to other codes is straightforward. The code set is defined as:

$$\Gamma_{84} = \begin{bmatrix}
x_1 & x_2 & x_3 & x_4 \\
-x_2 & x_1 & -x_4 & x_3 \\
-x_3 & x_4 & x_1 & -x_2 \\
-x_4 & -x_3 & x_2 & x_1 \\
x_1^* & x_2^* & x_3^* & x_4^* \\
-x_2^* & x_1^* & -x_4^* & x_3^* \\
-x_3^* & x_4^* & x_1^* & -x_2^* \\
-x_4^* & -x_3^* & x_2^* & x_1^*
\end{bmatrix} \quad (4.105)$$

By using Equation (4.105), one can show that the following relations are valid:

$$\begin{align*}
(r_1, r_2, r_3, r_4, r_5^*, r_6^*, r_7^*, r_8^*) &= (s_1, s_2, s_3, s_4)\Omega + (\eta_1, \eta_2, \eta_3, \eta_4, \eta_5^*, \eta_6^*, \eta_7^*, \eta_8^*) \quad (4.107)
\end{align*}$$

4.9.2 Received signal

At each time, there is only one received signal, $r_{1,1}$, which can be noted by $r_1$, and this is related to the constellation symbols $s_1, s_2, s_3, s_4$ by:

$$\begin{align*}
r_1 &= \alpha_1 s_1 + \alpha_2 s_2 + \alpha_3 s_3 + \alpha_4 s_4 + \eta_1 \\
r_2 &= -\alpha_1 s_2 + \alpha_2 s_1 - \alpha_3 s_4 + \alpha_4 s_3 + \eta_2 \\
r_3 &= -\alpha_1 s_3 + \alpha_2 s_4 + \alpha_3 s_1 - \alpha_4 s_2 + \eta_3 \\
r_4 &= -\alpha_1 s_4 - \alpha_2 s_3 + \alpha_3 s_2 + \alpha_4 s_1 + \eta_4 \\
r_1^* &= \alpha_1 s_1 + \alpha_2 s_2 + \alpha_3 s_3 + \alpha_4 s_4 + \eta_1 \\
r_2^* &= -\alpha_1 s_2 + \alpha_2 s_1 - \alpha_3 s_4 + \alpha_4 s_3 + \eta_2 \\
r_3^* &= -\alpha_1 s_3 + \alpha_2 s_4 + \alpha_3 s_1 - \alpha_4 s_2 + \eta_3 \\
r_4^* &= -\alpha_1 s_4 - \alpha_2 s_3 + \alpha_3 s_2 + \alpha_4 s_1 + \eta_4 \quad (4.106)
\end{align*}$$

By using Equation (4.105), one can show that the following relations are valid:

$$\begin{align*}
(r_1, r_2, r_3, r_4, r_5^*, r_6^*, r_7^*, r_8^*) &= (s_1, s_2, s_3, s_4)\Omega + (\eta_1, \eta_2, \eta_3, \eta_4, \eta_5^*, \eta_6^*, \eta_7^*, \eta_8^*) \quad (4.107)
\end{align*}$$
where

\[
\Omega = \begin{bmatrix}
\alpha_1 & \alpha_2 & \alpha_3 & \alpha_4 & \alpha_1' & \alpha_2' & \alpha_3' & \alpha_4'
\end{bmatrix} \tag{4.108}
\]

If, instead of \( S = (s_1, s_2, s_3, s_4) \), other orthogonal vectors are used, we have:

\[
(-r_2, r_1, r_4, -r_3, -r_6^*, r_5^*, r_8^*, -r_7^*) = (s_2, -s_1, s_4, -s_3)\Omega + (-\eta_2, \eta_1, \eta_4, -\eta_3, -\eta_6^*, \eta_5^*, \eta_8^*, -\eta_7^*)
\]

\[
(-r_3, -r_4, r_1, r_2, -r_7^*, -r_8^*, r_5^*, r_6^*) = (s_3, -s_4, -s_1, s_2)\Omega + (-\eta_3, -\eta_4, \eta_1, \eta_2, -\eta_7^*, -\eta_6^*, \eta_5^*, \eta_8^*)
\]

\[
(-r_4, r_3, -r_2, r_1, -r_8^*, r_7^*, -r_6^*, r_5^*) = (s_4, s_3, -s_2, -s_1)\Omega + (-\eta_4, \eta_3, -\eta_2, -\eta_1, -\eta_8^*, \eta_7^*, -\eta_6^*, \eta_5^*) \tag{4.109}
\]

### 4.9.3 Orthogonality

Note that, in general, for \( S = (s_1, s_2, s_3, s_4)^T \), vectors

\[
V_1(S) = (s_1, s_2, s_3, s_4, s_1^*, s_2^*, s_3^*, s_4^*)^T
\]

\[
V_2(S) = (s_2, -s_1, s_4, -s_3, s_2^*, -s_1^*, s_4^*, -s_3^*)^T
\]

\[
V_3(S) = (s_3, -s_4, -s_1, s_2, s_3^*, -s_4^*, -s_1^*, s_2^*)^T
\]

\[
V_4(S) = (s_4, s_3, -s_2, -s_1, s_4^*, s_3^*, -s_2^*, -s_1^*)^T
\]

are orthogonal to each other. Therefore, for specific constellation symbols \( S \), vectors \( V_1(S), V_2(S), V_3(S), V_4(S) \) can create a basis for the four-dimensional subspace of any arbitrary four-dimensional constellation symbols and their conjugates in an eight-dimensional space. If the constellation symbols are real numbers, vectors

\[
V'_1(S) = (s_1, s_2, s_3, s_4)^T
\]

\[
V'_2(S) = (s_2, -s_1, s_4, -s_3)^T
\]

\[
V'_3(S) = (s_3, -s_4, -s_1, s_2)^T
\]

\[
V'_4(S) = (s_4, s_3, -s_2, -s_1)^T \tag{4.111}
\]

which only contain the first four elements of vectors \( V_1(S), V_2(S), V_3(S), V_4(S) \), create a basis for the space of any arbitrary four-dimensional real constellation symbols.

### 4.9.4 Encoding

**Step 1.** Let us assume that we use a signal constellation with \( 2^b \) elements. For each block of \( Kb \) bits, the encoding is done by first calculating the \( K \)-dimensional vector of symbols \( S = (s_1, s_2, \ldots, s_K)^T \).

**Step 2.** Indeterminates \( s_1, s_2, \ldots, s_K \) in \( \Gamma \) are replaced by \( s_1, s_2, \ldots, s_K \) to establish the matrix \( X \) which is used for transmission in a manner similar to a regular space–time block code. The main issue here is how to calculate \( S = (s_1, s_2, \ldots, s_K)^T \) such that non-coherent detection is possible.

**Step 3.** Let us assume that \( S_n \) is the vector which is used for the \( v \)th block of \( Kb \) bits. Also, \( X(S_n) \) defines what to transmit from each antenna during the transmission of the \( v \)th block. In other words, \( X(S_n), n = 1, 2, \ldots, N \) is the \( n \)th column of \( X(S_n) \) which contains \( T \) symbols which are transmitted from the \( n \)th antenna sequentially.

We fix a set \( V \) which consists of \( 2^{Kb} \) unit length vectors \( P_1, P_2, \ldots, P_{2^{Kb}} \) where each vector \( P_i \) is a \( K \times 1 \) vector of real numbers \( P_i = (P_{i1}, P_{i2}, \ldots, P_{ik})^T \). We define an arbitrary one-to-one mapping \( \beta \) which maps \( Kb \) bits onto \( V \). We start with an arbitrary vector \( S_1 \). Then, let us assume that \( S_v \) is used for the \( v \)th block. For the \( (v + 1) \)th block, we use the \( Kb \) input bits to pick the corresponding
vector $P_i$ in $V$ using the one-to-one mapping $\beta$. Then, we calculate:

$$S_{v+1} = \sum_{k=1}^{K} P_k \mathbf{V}_k(S_v)$$  \hspace{1cm} (4.112)

where $\mathbf{V}_k(S_v)$ is a $K$-dimensional vector which includes the first $K$ elements of $\mathbf{V}_k(S_v)$. We use $\mathbf{X}(S_{v+1})$ for transmission at the following $T$ time slots.

### 4.9.4.1 Example

We pick four BPSK symbols, i.e. symbol $s_k = 1$ if the corresponding bit is zero and $s_k = -1$ otherwise. Then, $P_i$ is defined as the projection of the vector $(s_1, s_2, s_3, s_4)$ onto $\mathbf{V}_i(S)$, $\mathbf{V}_i(S)$, $\mathbf{V}_i(S)$, $\mathbf{V}_i(S)$, where $S = (1, 1, 1)^T$. In fact, the elements of $P_i$ can be calculated using the following equations:

\[
\begin{align*}
P_{i1} &= (s_1 + s_2 + s_3 + s_4)/4 \\
P_{i2} &= (s_1 - s_2 + s_3 - s_4)/4 \\
P_{i3} &= (s_1 - s_2 - s_3 + s_4)/4 \\
P_{i4} &= (s_1 + s_2 - s_3 - s_4)/4
\end{align*}
\hspace{1cm} (4.113)
\]

This completes the one-to-one mapping $\beta$ which is needed for encoding and decoding.

### 4.9.5 Differential decoding

Let us recall that the received signal $r_i$ is related to the transmitted signals by:

$$r_i = \sum_{n=1}^{N} \alpha_n X_{i,n} + \eta_i$$  \hspace{1cm} (4.114)

Assuming $T = 2K$, which results in a rate 1/2 code, and defining

$$\mathbf{R} = (r_1, r_2, \ldots, r_K, r_{K+1}, r_{K+2}, \ldots, r_{2K})^T,$$

we have

$$\mathbf{R} = \mathbf{S}^T \Omega(\alpha_1, \alpha_2, \ldots, \alpha_N) + \mathbf{N}$$  \hspace{1cm} (4.115)

where $\mathbf{N} = (\eta_1, \eta_2, \ldots, \eta_K, \eta_{K+1}, \eta_{K+2}, \ldots, \eta_{2K})$ and

$$\Omega(\alpha_1, \alpha_2, \ldots, \alpha_N) = (\Lambda(\alpha_1, \alpha_2, \ldots, \alpha_N)\Lambda(\alpha_1, \alpha_2, \ldots, \alpha_N))$$  \hspace{1cm} (4.116)

where $\Lambda(\alpha_1, \alpha_2, \ldots, \alpha_N)$ is the $K \times K$ matrix defined by

$$\Lambda_{i,j} = \delta(i, j)\alpha_{\varepsilon(i,j)}, \quad i = 1, 2, \ldots, K, \quad j = 1, 2, \ldots, K$$  \hspace{1cm} (4.117)

where $\varepsilon(i, j) = l \Leftrightarrow B_{i,l} = s_i$ or $B_{j,l} = -s_j$, and

$$\delta(i, j) = \begin{cases} 
1, & \text{if } B_{j,l} = s_i \\
-1, & \text{if } B_{j,l} = -s_i
\end{cases}$$

### 4.9.5.1 Example

For the rate 1/2 code defined by $\Gamma_{84}$ in Equation (4.105), $\Lambda(\alpha_1, \alpha_2, \alpha_3, \alpha_4)$ is a $4 \times 4$ matrix defined by:

$$\Lambda = \begin{bmatrix} 
\alpha_1 & \alpha_2 & \alpha_3 & \alpha_4 \\
\alpha_2 & -\alpha_1 & -\alpha_4 & \alpha_3 \\
\alpha_3 & \alpha_4 & -\alpha_1 & -\alpha_2 \\
\alpha_4 & -\alpha_3 & \alpha_2 & -\alpha_1
\end{bmatrix}$$  \hspace{1cm} (4.118)
Differential decoding is enabled by the fact that $\Omega^* = 2 \sum_{n=1}^{N} |\alpha_n|^2 I_K$. To prove it, let us set $N = 0$ in Equations (4.114) and (4.115). Then, we have $\mathbf{R} = \mathbf{S}^T \Omega^*$ which results in:

$$\mathbf{R}^* = \mathbf{S}^T \Omega^* \mathbf{S}^T$$  \hspace{1cm} (4.119)

On the other hand

$$\mathbf{R}^* = \sum_{t=1}^{T} |r_t|^2 = \sum_{t=1}^{T} \left( \sum_{n=1}^{N} \alpha_n \mathbf{X}_{t,n} \right)^* \left( \sum_{n=1}^{N} \alpha_n \mathbf{X}_{t,n} \right)$$ \hspace{1cm} (4.120)

We rewrite the above formulas using matrix C as follows:

$$\mathbf{R}^* = (\alpha_1, \alpha_2, \ldots, \alpha_N) \mathbf{X}^T \mathbf{X}^*(\alpha_1, \alpha_2, \ldots, \alpha_N)^*$$

$$= (\alpha_1, \alpha_2, \ldots, \alpha_N) (\mathbf{X}^* \mathbf{X})^T (\alpha_1, \alpha_2, \ldots, \alpha_N)^*$$

$$= (\alpha_1, \alpha_2, \ldots, \alpha_N) 2 \sum_{k=1}^{K} |s_k|^2 I_k (\alpha_1, \alpha_2, \ldots, \alpha_N)^*$$ \hspace{1cm} (4.121)

$$= 2 \sum_{k=1}^{K} |s_k|^2 (\alpha_1, \alpha_2, \ldots, \alpha_N) (\alpha_1, \alpha_2, \ldots, \alpha_N)^*$$

Therefore, we have:

$$\mathbf{R}^* = \mathbf{S}^T \Omega^* \mathbf{S}^T = 2 \sum_{k=1}^{K} |s_k|^2 \sum_{n=1}^{N} |\alpha_n|^2$$ \hspace{1cm} (4.122)

Since Equation (4.122) is true for any $\mathbf{S} = (s_1, s_2, \ldots, s_K)^T$, we should have

$$\Omega^* = 2 \sum_{n=1}^{N} |\alpha_n|^2 I_K$$

### 4.9.6 Received signal

Let us recall that $\mathbf{S}_0$ and $\mathbf{S}_{l+1}$ are used for the $v$th and $(v + l)$th blocks of $Kb$ bits, respectively. Using $\Gamma_{84}$, for each block of data we receive eight signals. To simplify the notation, we denote the received signals corresponding to the $v$th block by $r_1^v, r_2^v, \ldots, r_8^v$ and the received signals corresponding to the $(v + l)$th block by $r_{v+1}^{v+1}, r_{v+1}^{v+1}, \ldots, r_{8+1}^{v+1}$. Let us construct the following vectors (see Equation (4.109)).

$$\mathbf{R}_v = \left( r_1^v, r_2^v, r_3^v, r_4^v, r_5^v, r_6^v, r_7^v, r_8^v \right)$$

$$\mathbf{R}_v = \left( -r_2^v, r_1^v, r_4^v, -r_3^v, r_6^v, -r_5^v, r_8^v, -r_7^v \right)$$

$$\mathbf{R}_v = \left( -r_3^v, -r_4^v, r_2^v, -r_5^v, r_7^v, -r_6^v, r_8^v, -r_1^v \right)$$

$$\mathbf{R}_v = \left( -r_4^v, -r_5^v, r_3^v, -r_6^v, r_1^v, -r_7^v, -r_8^v, r_2^v \right)$$

$$\mathbf{R}_{v+l} = \left( r_{v+1}^{v+1}, r_{v+1}^{v+1}, r_{v+1}^{v+1}, r_{v+1}^{v+1}, r_{v+1}^{v+1}, r_{v+1}^{v+1}, r_{v+1}^{v+1}, r_{v+1}^{v+1} \right)$$

By using Equations (4.107) and (4.109), we have:

$$\mathbf{R}_{v+l} \mathbf{R}_v^* = \mathbf{S}_{v+l}^T \Omega^* \mathbf{V}_k^* (\mathbf{S}_v)^* + \mathbf{N}_k$$

$$= 2 \sum_{n=1}^{N} |\alpha_n|^2 \mathbf{S}_{v+l}^T \mathbf{V}_k^* (\mathbf{S}_v)^* + \mathbf{N}_k$$ \hspace{1cm} (4.124)

$$= 2 \sum_{n=1}^{N} |\alpha_n|^2 \mathbf{P}_k + \mathbf{N}_k$$
Therefore, we can write:

\[
R = \left( R_{v+1} R_v^1, R_{v+1} R_v^2, R_{v+1} R_v^3, R_{v+1} R_v^4 \right)
\]

\[
= \left( 2 \sum_{n=1}^{N} |\alpha_n|^2 \right) P + N' \tag{4.125}
\]

where \( N' = (\eta_1, \eta_2, \ldots, \eta_K) \).

### 4.9.7 Demodulation

Because the elements of \( R \) have equal lengths, to compute \( P \), the receiver can compute the closest vector of \( V \) to \( R \). Once this vector is computed, the inverse mapping of \( \beta \) is applied and the transmitted bits are recovered.

### 4.9.8 Multiple receive antennas

If there is more than one receive antenna, the same procedure can be used. In this case, first assuming that only the receive antenna \( m \) exists, we compute \( P^m \) using the same method for \( P \) given above. Then, after calculating \( M \) vectors \( P^m \), \( m = 1, 2, \ldots, M \), the closest vector of \( V \) to \( \sum_{m=1}^{M} P^m \) is computed. The inverse mapping of \( \beta \) is applied to the closest vector to compute the transmitted bits. It is easy to show that \( 4 M \)-level diversity is achieved by using this method.

### 4.9.9 The number of transmit antennas lower than the number of symbols

So far, we have considered an example where \( N = K \), i.e. the real matrix which creates \( \Gamma \) is a square matrix. When the number of transmit antennas is less than the number of symbols, \( N < K \), the same approach works. Let us use the following example:

\[
\Gamma_{83} = \begin{bmatrix}
  x_1 & x_2 & x_3 \\
  -x_2 & x_1 & -x_4 \\
  -x_3 & x_4 & x_1 \\
  -x_4 & -x_3 & x_2 \\
  x_1^* & x_2^* & x_3^* \\
  -x_2^* & x_1^* & -x_4^* \\
  -x_3^* & x_4^* & x_1^* \\
  -x_4^* & -x_3^* & x_2^*
\end{bmatrix} \tag{4.126}
\]

When there is only one receive antenna, \( M = 1 \), the received signals are related to the constellation symbols \( s_1, s_2, s_3, s_4 \) by:

\[
\begin{align*}
  r_1 &= \alpha_1 s_1 + \alpha_2 s_2 + \alpha_3 s_3 + \eta_1 \\
  r_2 &= -\alpha_1 s_1 + \alpha_2 s_1 - \alpha_3 s_4 + \eta_2 \\
  r_3 &= -\alpha_1 s_1 + \alpha_2 s_4 + \alpha_3 s_1 + \eta_3 \\
  r_4 &= -\alpha_1 s_4 - \alpha_2 s_3 + \alpha_3 s_2 + \eta_4 \\
  r_5 &= \alpha_1 s_1^* + \alpha_2 s_2^* + \alpha_3 s_3^* + \eta_5 \\
  r_6 &= -\alpha_1 s_1^* + \alpha_2 s_1^* - \alpha_3 s_4^* + \eta_6 \\
  r_7 &= -\alpha_1 s_4^* + \alpha_2 s_3^* + \alpha_3 s_1^* + \eta_7 \\
  r_8 &= -\alpha_1 s_4^* + \alpha_2 s_4^* + \alpha_3 s_2^* + \eta_8.
\end{align*}
\]  

Equation (4.107) now becomes:

\[
(r_1, r_2, r_3, r_4, r_5^*, r_6^*, r_7^*, r_8^*) = (s_1, s_2, s_3, s_4) \Omega + (\eta_1, \eta_2, \eta_3, \eta_4, \eta_5^*, \eta_6^*, \eta_7^*, \eta_8^*) \tag{4.128}
\]
Therefore, again for specific constellation symbols \( S \), transmit antennas respectively. The only difference in the final result is that \( \sum_{n=1}^{4} |\alpha_n|^2 \) is replaced by \( \sum_{n=1}^{3} |\alpha_n|^2 \).

### 4.9.10 Final result

Instead of Equation (4.125) we now have:

\[
\mathbf{R} = \left( \mathbf{R}_{v+1}, \mathbf{R}_{v}^{1\text{a}}, \mathbf{R}_{v+1}^{2\text{e}}, \mathbf{R}_{v}^{3\text{e}}, \mathbf{R}_{v+1}^{4\text{a}} \right)
\]

\[
= \left( 2 \sum_{n=1}^{3} |\alpha_n|^2 \right) \mathbf{P}_1 + \mathbf{N}
\]

### 4.9.11 Real constellation set

The above rate 1/2 space–time block codes can be applied to any complex constellation set. If the constellation set is real, rate 1 space–time block codes are available and the same approach works. For example, in the case of \( T = K = 4 \), the following space–time block code exists for \( N = 4 \):

\[
\Gamma = \begin{bmatrix}
  x_1 & x_2 & x_3 & x_4 \\
  -x_2 & x_1 & -x_4 & x_3 \\
  -x_3 & x_4 & x_1 & -x_2 \\
  -x_4 & -x_3 & x_2 & x_1
\end{bmatrix}
\]

(4.131)

It is easy to show that:

\[
(r_1, r_2, r_3, r_4, r_5^*, r_6^*, r_7^*, r_8^*) = (s_1, s_2, s_3, s_4) \Omega + (\eta_1, \eta_2, \eta_3, \eta_4, \eta_5^*, \eta_6^*, \eta_7^*, \eta_8^*)
\]

(4.132)

where \( \Omega \) is defined by Equation (4.108). Similar differential encoding and decoding are possible if we use the following vectors for \( \mathbf{R}_i, i = 1, 2, 3, 4 \), and \( \mathbf{R}_{v+1} \):

\[
\mathbf{R}_v^1 = \left( r_1^v, r_2^v, r_3^v, r_4^v, r_1^{v*}, r_2^{v*}, r_3^{v*}, r_4^{v*} \right)
\]

\[
\mathbf{R}_v^2 = \left( -r_2^v, r_1^v, -r_3^v, -r_4^v, -r_1^{v*}, r_2^{v*}, -r_3^{v*}, -r_4^{v*} \right)
\]

\[
\mathbf{R}_v^3 = \left( -r_3^v, -r_4^v, r_1^v, r_2^v, -r_1^{v*}, -r_2^{v*}, r_3^{v*}, r_4^{v*} \right)
\]

\[
\mathbf{R}_v^4 = \left( -r_4^v, -r_3^v, r_2^v, r_1^v, r_1^{v*}, -r_2^{v*}, -r_3^{v*}, r_4^{v*} \right)
\]

(4.133)

\[
\mathbf{R}_{v+1} = \left( r_1^{v+1}, r_2^{v+1}, r_3^{v+1}, r_4^{v+1}, r_1^{(v+1)*}, r_2^{(v+1)*}, r_3^{(v+1)*}, r_4^{(v+1)*} \right)
\]

This results in a full diversity, full rate scheme for differential detection. For example, as defined by Equation (4.133), performance results are shown in Figures 4.44 and 4.45 for four and three transmit antennas respectively.

More details on the topic can be found in [77, 78].
Figure 4.44 Performance results for coherent and non-coherent detection schemes; four transmit antennas, one receive antenna, BPSK.

Figure 4.45 Performance results for coherent and non-coherent detection schemes; three transmit antennas, one receive antenna, BPSK.
4.10 LAYERED SPACE–TIME CODING

In this section we discuss a possibility of partitioning antennas at the transmitter into small groups, and using individual space–time codes, called component codes, to transmit information from each group of antennas. At the receiver, an individual space–time code is decoded by a linear processing technique that suppresses signals transmitted by other groups of antennas by treating them as interference. This receiver structure provides diversity and coding gain over uncoded systems. This combination of array processing at the receiver and coding techniques for multiple transmit antennas can provide reliable and very high data rate communication over narrowband wireless channels. A refinement of this basic structure gives rise to a multilayered space–time architecture that both generalizes and improves upon the layered space–time architecture proposed by Foschini, which is known in the literature as Bell Lab Layered space time (BLAST) coding [10, 77–88].

For $1 \leq j \leq m$ the signal $r_j^t$ received by antenna $j$ at time $t$ is given by:

$$r_j^t = \sum_{i=1}^{n} \alpha_{i,j} c_i^t + \eta_j^t \quad \text{(4.134)}$$

where $c_i^t$ is the encoded signal transmitted from transmit antenna $i$.

For any vector $x$, let $x^T$ denote the transpose of $x$. We can now write Equation (4.134) in the vector form given by:

$$r_t = \Omega c_t + \eta_t \quad \text{(4.135)}$$

where

$$c_t = \left( c_1^t, c_2^t, \ldots, c_n^t \right)^T$$

$$r_t = \left( r_1^t, r_2^t, \ldots, r_m^t \right)^T$$

$$\eta_t = \left( \eta_1^t, \eta_2^t, \ldots, \eta_m^t \right)^T \quad \text{(4.136)}$$

and

$$\Omega = \begin{bmatrix}
\alpha_{1,1} & \alpha_{2,1} & \cdots & \cdots & \alpha_{n,1} \\
\alpha_{1,2} & \alpha_{2,2} & \cdots & \cdots & \alpha_{n,2} \\
\alpha_{1,3} & \alpha_{2,3} & \cdots & \cdots & \alpha_{n,3} \\
\vdots & \vdots & \ddots & \cdots & \vdots \\
\alpha_{1,m} & \alpha_{2,m} & \cdots & \cdots & \alpha_{n,m}
\end{bmatrix} \quad \text{(4.137)}$$

A space–time product encoder accepts a block of $B$ input bits in each time slot $t$ and these bits are divided into $q$ strings of lengths $B_1, B_2, \ldots, B_q$ with $B_1 + B_2 + \cdots + B_q = B$. At the base station, $n$ antennas are partitioned into $q$ groups $G_1, G_2, \ldots, G_q$, respectively, comprising $n_1, n_2, \ldots, n_q$ antennas with $n_1 + n_2 + \cdots + n_q = n$. Each block $B_j$, $1 \leq j \leq q$ is then encoded by a space–time encoder $X_j$. The output of $X_j$ goes through a serial to parallel converter and provides $n_j$ sequences of constellation symbols for $1 \leq j \leq q$ which are simultaneously transmitted from the antennas of the group $G_j$. This gives a total of $n$ sequences of constellation symbols that are transmitted simultaneously from antennas $1, 2, \ldots, n$.

A space–time product encoder can be considered as a set of $q$ space–time encoders, called the component codes, operating in parallel on the same wireless communication channel, with each encoder using $n_j$ transmit and $m$ receive antennas for $1 \leq j \leq q$. It will be denoted an $X_1 \times X_2 \times \cdots X_q$ encoder.
4.10.1 Receiver complexity

One approach to recovering the transmitted data at the receiver is to jointly decode the transmitted code word, but the difficulty here is decoding complexity. Indeed, if we require a diversity of \( r m \), where \( r \leq \min_j (n_j) \), then (see Section 4.2) the complexity of the trellis of \( X_j \) is at least \( 2^{B_j(r-1)} \) states and the complexity of the product code is at least \( 2^{B(r-1)} \) states. This means that if \( B \) is very large, the scheme may be too complex to implement.

4.10.2 Group interference suppression

The idea is to decode each code \( X_j \) separately while suppressing signals from other component codes. This approach has a much lower complexity but achieves a lower diversity order than the full diversity order \( nm \), which is the product of the numbers of transmit and receive antennas.

4.10.3 Suppression method

Without any loss of generality, we take \( j = 1 \) and look to decode \( X_1 \). There are \( n - n_1 \) interfering signals. We assume that there are \( m \geq n - n_1 + 1 \) receive antennas and that the receiver knows the matrix \( \Omega(X_1) \) (channel state information). The matrix

\[
\Lambda(X_1) = \\
\begin{bmatrix}
\alpha_{n_1+1, 1} & \alpha_{n_1+2, 1} & \cdots & \cdots & \alpha_{n_1, 1} \\
\alpha_{n_1+1, 2} & \alpha_{n_1+2, 2} & \cdots & \cdots & \alpha_{n_1, 2} \\
\alpha_{n_1+1, 3} & \alpha_{n_1+2, 3} & \cdots & \cdots & \alpha_{n_1, 3} \\
\vdots & \vdots & \ddots & \ddots & \vdots \\
\alpha_{n_1+1, m} & \alpha_{n_1+2, m} & \cdots & \cdots & \alpha_{n_1, m}
\end{bmatrix}
\]  

has rank less than or equal to the number of its columns. Thus, \( \text{rank} [\Lambda(X_1)] \leq n - n_1 \).

4.10.4 The null space

The null space \( N \) of this matrix is the set of all row vectors \( x \) such that \( x\Lambda(X_1) = (0, 0, \ldots, 0) \). Furthermore,

\[
\dim(N) + \text{rank} [\Lambda(X_1)] = m
\]

Since \( \text{rank}[\Lambda(X_1)] \leq n - n_1 \), it follows that \( \dim(N) \geq m - n + n_1 \). Hence we can compute a (not necessarily unique) set of orthonormal vectors:

\[
\{v_1, v_2, \ldots, v_{m-n+n_1}\}
\]

in \( N \). We let \( \Theta(X_1) \) denote the \((m - n + n_1) \times m \) matrix whose \( j \)th row is \( v_j \). Clearly, \( \Theta(X_1)\Theta(X_1)^* = I_{n_1-n+m} \), where \( \Theta(X_1)^* \) is the Hermitian of \( \Theta(X_1) \) and \( I_{n_1-n+m} \) is the \((m - n + n_1) \times (m - n + n_1) \) identity matrix. We multiply both sides of Equation (4.135) by \( \Theta(X_1) \) to arrive at:

\[
\Theta(X_1)r_t = \Theta(X_1)\Omega c_t + \Theta(X_1)\eta_t
\]  

where

\[
\Omega(X_1) = \\
\begin{bmatrix}
\alpha_{1,1} & \alpha_{2,1} & \cdots & \cdots & \alpha_{n_1,1} \\
\alpha_{1,2} & \alpha_{2,2} & \cdots & \cdots & \alpha_{n_1,2} \\
\alpha_{1,3} & \alpha_{2,3} & \cdots & \cdots & \alpha_{n_1,3} \\
\vdots & \vdots & \ddots & \ddots & \vdots \\
\alpha_{1,m} & \alpha_{2,m} & \cdots & \cdots & \alpha_{n_1,m}
\end{bmatrix}
\]  

Since $\Theta(X_1) \Lambda(X_1) = 0$ is the all zero matrix, Equation (4.139) can be written as

$$\Theta(X_1) r_t = \Theta(X_1) \Omega(X_1) c^l_t + \Theta(X_1) \eta_t,$$

where $c^l_t = (c^l_1, c^l_2, \ldots, c^l_n)^T$. Setting

$$\tilde{r}_t = \Theta(C_1) r_t,$$

$$\Omega = \Theta(C_1) \Omega,$$

$$\tilde{\eta}_t = \Theta(C_1) \eta_t,$$

we arrive at the equation

$$\tilde{r}_t = \Omega c^l_t + \tilde{\eta}_t.$$

This is an equation where all the signal streams out of antennas $n_1 + 1, n_1 + 2, \ldots, n$ are suppressed.

### 4.10.5 Receiver

Let us look at the structure of the decoder for the code $X_1$, given that group interference suppression is performed to suppress all the signal streams out of antennas $n_1 + 1, n_1 + 2, \ldots, n$. To this end, suppose that $\Lambda(X_1)$ is given. The receiver computes a set of orthonormal vectors

$$\{v_1, v_2, \ldots, v_{m-n+n_1}\}$$

and the matrix $\Theta(X_1)$ as described in the previous section. Let $\tilde{\Omega}_{ij}$ and $\tilde{\Omega}_{ik}$ denote the $(i, j)$th and $(l, k)$th elements of $\Omega$. By definition $\tilde{\Omega}_{ij} = v_i \omega_j$ and $\tilde{\Omega}_{ik} = v_l \omega_k$ where $\omega_j$ and $\omega_k$ are, respectively, the $j$th and $k$th columns of $\Omega(X_1)$. The random variables $\tilde{\Omega}_{ij}$ and $\tilde{\Omega}_{ik}$ have zero means given $\Lambda(X_1)$. Moreover,

$$E \left[ \tilde{\Omega}_{ij} \tilde{\Omega}_{ik}^* \right] = E \left[ v_i \omega_j v_l^* v_i^* \right] = v_i E \left[ \omega_j \omega_k^* \right] v_l^* = \delta_{jk} v_i v_l^* = \delta_{jk} \delta_{il}$$

where $\delta$ is the Kronecker delta function given by $\delta_{rs} = 0$ if $r \neq s$ and $\delta_{ss} = 1$ if $r = s$. We conclude that the elements of the $(m - n + n_1) \times n_1$ matrix $\tilde{\Omega}$ are independent complex Gaussian random variables of variance 0.5 per real dimension. Similarly, the components of the noise vector $\tilde{\eta}_t$, $t = 1, 2, \ldots, l$ are independent Gaussian random variables of variance $N_0/2$ per real dimension.

### 4.10.6 Decision metric

Assuming that all the code words of $X_1$ are equiprobable, and given that group interference suppression is performed, the ML receiver for $X_1$ decides in favor of the code word

$$c^1_1 c^2_1 \cdots c^n_1 c^1_2 \cdots c^n_2 \cdots c^1_n \cdots c^n_n$$

if it minimizes the decision metric

$$\sum_{t=1}^f \left| \tilde{r}_t - \Omega c^l_t \right|^2$$

### 4.10.7 Multilayered space–time coded modulation

The previous discussion reduces code design for a multiple antenna communication system with $n$ transmit and $m$ receive antennas to that of designing codes for communication systems with $n_t$ transmit and $n_r = m - n_i$, $i = 1, 2, \ldots, q$, receive antennas where $\sum_{i=1}^q n_i = n$ and $n_i \geq n - m + 1$. Using this insight, we may design a multilayered space–time coded modulation scheme. The idea behind such a system is multistage detection and cancellation.
4.10.8 Diversity gain

Suppose that $X_1$ is decoded correctly using combined array processing and space–time coding. The space–time code $X_1$ affords a diversity gain of $n_1 \times (n_1 + m - n)$. After decoding $X_1$ we may subtract the contribution of these code words to signals received at different antennas. This gives a communication system with $n - n_1$ transmit and $m$ receive antennas.

In the next step the receiver uses combined array processing and space–time coding to decode $X_2$. The space–time code $X_2$ affords a diversity gain of $n_2 \times (n_2 + n_1 + m - n)$. Proceeding in this manner, we observe that by subtracting the contribution of previously decoded code streams $X_j$, $j \leq k - 1$ to the received signals at different antennas, the space–time code $X_k$ affords a diversity gain of $n_k \times (n_1 + \cdots + n_k + m - n)$.

We can choose space–time codes $X_i$, $1 \leq i \leq q$ to provide these diversity gains and such that the sequence

$$n_1 (n_1 + m - n), n_2 (n_2 + n_1 + m - n), \ldots, n_k (n_1 + \cdots + n_k + m - n), \ldots, n_q m$$

is an increasing sequence. Assuming there was no decoding error in steps 1, 2, \ldots, $k - 1$, then at decoding step $k$, the probability of error for the component code $X_k$ is equal to the probability of error for $X_k$ when employed in a communication system using $n_k$ transmit and $(n_1 + \cdots + n_k + m - n)$ receive antennas.

4.10.9 Adaptive reconfigurable transmit power allocation

Since the diversity in each decoding stage $k$ is more than that of the previous decoding stage $k - 1$, the transmit power out of each antenna at level $k$ can be substantially less than that of the previous layer. Thus the transmitter should divide the available transmit power among different antennas in an unequal manner. Power allocation for this scenario is straightforward. In fact, powers at different levels could be allocated based on the diversity gains. In this way, the allocated powers may decrease geometrically in terms of the diversity gains. Other approaches are also possible.

4.10.9.1 Example 1

**Transmitter.** Here, four transmit and four receive antennas are used. The transmission rate is 4 bits/s/Hz. Let $X$ denote the 32-state 4PSK (see Figure 4.46) space–time trellis code given in Figure 4.47. The product code $X_1 \times X_2$ where $X_1 = X_2 = X$ will be used for transmission of 4 bits/s/Hz. At each time slot, upon the arrival of the four bits of the input data, the first two bits are used as the input to the encoder of $X_1$ and the encoded symbols are transmitted by antennas one and two. The second two bits are used as the input to the encoder of $X_2$ and the encoded signals are transmitted by antennas three and four. We assume that the average powers radiated from antennas one and two are equal but each is twice as much as the average power radiated from antennas three and four.

**Receiver.** Interference suppression is used to suppress $X_2$ and decode $X_1$. Upon decoding $X_1$ the contributions of the code words transmitted from antennas one and two are subtracted from the received signals. Finally, $X_2$ is decoded.

4.10.9.2 Simulation results

Figure 4.48 demonstrates the performance of this multilayered space–time coded architecture. Each frame consists of 130 transmissions from each transmit antenna. It is assumed that the channel matrix is perfectly known at the receiver. The horizontal axis shows the receive signal to noise ratio per transmission time. Each transmission time corresponds to the transmission of four bits. Thus, the horizontal axis denotes the receive signal to noise ratio per four bits. For comparison, the graph of the outage capacity versus the signal to noise ratio for four transmit and four receive antennas is presented in Figure 4.49. The outage capacity is defined as the achievable capacity $C_{\text{out}}$ for which the outage
Figure 4.46 The 4PSK constellation.

Figure 4.47 4PSK space–time code, 32 states 2 bits/s/Hz.
Figure 4.48 The performance of the scheme in Example 1.

Figure 4.49 Outage capacity for four transmit and four receive antennas.
probability $P_{\text{out}} = P(C < C_{\text{out}}) < \varepsilon$. More details on MIMO channel capacity will be presented in Section 4.12. One can see that for a frame error probability of $10^{-1}$, the system is about 6 dB away from the capacity.

### 4.10.9.3 Example 2

Here, eight transmit and eight receive antennas are used. The transmission rate is 8 bits/s/Hz. Let $\mathbf{X}$ denote the code given in Figure 4.47. We use the product code $\mathbf{X}_1 \times \mathbf{X}_2 \times \mathbf{X}_3 \times \mathbf{X}_4$ where $\mathbf{X}_1 = \mathbf{X}_2 = \mathbf{X}_3 = \mathbf{X}_4 = \mathbf{X}$ for transmission of 8 bits/s/Hz. At each time instance, upon the arrival of the eight bits of the input data, the first, second, third, and fourth blocks of length two of the input bits are respectively used as the input to encoders of $\mathbf{X}_1$, $\mathbf{X}_2$, $\mathbf{X}_3$ and $\mathbf{X}_4$. The output of encoders of $\mathbf{X}_i$, $1 \leq i \leq 4$ are, respectively, transmitted by antennas $2i-1$ and $2i$. We assume that the average power radiated from antennas one and two is $E_s$, the average power radiated from antennas three and four is $E_s/2$, the average power radiated from antennas five and six is $E_s/4$, and the average power radiated from antennas seven and eight is $E_s/8$. Thus, the total signal to noise ratio at each receive antenna is $15 E_s/4N_0$.

**Decoder.** Group interference suppression is used to decode $\mathbf{X}_1$. Upon decoding $\mathbf{X}_1$, the contributions of the code words transmitted from antennas one and two are subtracted from the received signals. Using this, $\mathbf{X}_2$ is decoded next and so forth. In Figure 4.50, we provide simulation results to demonstrate the performance of this multilayered space–time coded architecture. Each frame consists of 130 transmissions from each transmit antenna. It is assumed that the channel matrix is perfectly known at the receiver. The horizontal axis shows the receive signal to noise ratio per transmission

![4PSK, 4 bits/s/Hz, 8 Recev., 8 Trans. Antennas](image)

Figure 4.50 The performance of the scheme in Example 2.
Figure 4.51 Outage capacity for eight transmit and eight receive antennas.

Figure 4.52 Block diagram of a space–time coded system concatenating STBC and TCM, $N = 2$, $M = 1$.

time. Each transmission time corresponds to the transmission of eight bits. Thus, the horizontal axis denotes the receive signal to noise ratio per eight bits.

For comparison, we provide in Figure 4.51 the graph of the outage capacity versus the signal to noise ratio for eight transmit and eight receive antennas, as computed by Foschini and Gans [89]. We observe that for a frame error probability of $10^{-1}$, we are about 9 dB away from the capacity.

More details on this topic can be found in [10, 77–88].

### 4.11 CONCATENATED SPACE–TIME BLOCK CODING

#### 4.11.1 System model

The system model is given in Figure 4.52.
We begin by performance analysis and design criteria in quasi-static fading, where the fading coefficients are constant during a frame of length $2L$ and vary from one frame to another.

### 4.11.2 Product sum distance

Let $\eta$ be the set of all $i$ for which $c_i \neq e_i$ or $c_{i+1} \neq e_{i+1}$. Denote the number of elements in $\eta$ by $l_\eta$. Then, at high signal to noise ratios (SNRs) [90],

$$P(C \rightarrow E) \leq \left( \frac{E_s}{4N_0} \right)^{l_\eta} d_P(l_\eta)$$

where

$$d_P(l_\eta) = \prod_{i \in \eta} \left( |c_i - e_i|^2 + |c_{i+1} - e_{i+1}|^2 \right)$$

is the product of Euclidean distances associated with two consecutive symbols along the error event path ($C \rightarrow E$). Parameter $d_P(l_\eta)$ is referred to as the product–sum distance over span 2. In addition, $l_\eta$ is referred to as the effective length of this error event over span 2.

Let $P_e$ denote the error event probability, then by using the union bound, an upper bound can be obtained as:

$$P_e \leq \sum_{L=1}^{\infty} \sum_C \sum_{E \neq C} P(C) P(C \rightarrow E)$$

where $P(C)$ is the a priori probability of transmitting the symbol sequence $C$ with length $2L$. By summing over all possible $l_\eta$ and all possible $d_P(l_\eta)$, the error event probability can be further written as:

$$P_e \leq \sum_{l_\eta} \sum_{d_P(l_\eta)} \Xi(l_\eta, d_P(l_\eta)) \left( \frac{E_s}{4N_0} \right)^{l_\eta} d_P(l_\eta)^{-2}$$

where $\Xi(l_\eta, d_P(l_\eta))$ is the average number of error events having the span 2 effective length $l_\eta$ and the product–sum distance $d_P(l_\eta)$.

### 4.11.3 Error rate bound

The smallest $l_\eta$ and the smallest $d_P(l_\eta)$ dominate the error event probability at high SNRs. Denoting $R = \min(l_\eta)$ and $d_{\text{min}}(R) = \min(d_P(R))$, then the error event probability is asymptotically approximated as:

$$P_e \approx \Xi(R, d_{\text{min}}(R)) \frac{1}{\left( \frac{E_s}{4N_0} \right)^{2R} \left[ d_{\text{min}}(R) \right]^2}$$

From Equation (4.149), we observe that the error event probability asymptotically varies with $2R$-power of SNR, so a diversity order of $2R$ is achieved. We further refer to $R$ as the built-in time diversity or effective length of the concatenated space–time code.

The design criteria, in this case, involve the maximization of both the built-in-time diversity and the minimum product–sum distance of the trellis code at high SNRs for Rayleigh fading. This conclusion is, therefore, different from that of conventional TCM where the minimum product distance needs to be maximized. Thus, new optimal codes can be found, based upon these new criteria.
4.11.4 The case of low SNR

For low SNR we have [90]:

\[
P(C \rightarrow E) \leq 1 + \frac{E_s}{4N_0} \sum_{i=0}^{2L-1} |c_i - e_i|^2 + o\left(\frac{E_s}{4N_0}\right)^2
\] (4.150)

where \(o(E_s/4N_0)\) denotes the summation of all the terms which include higher order quantities of \((E_s/4N_0)\). Equation (4.150) indicates that the squared Euclidean distance becomes the main factor. Thus, the dominant factor affecting the performance of trellis coded modulation for use with space–time block coding at low SNRs is the free Euclidean distance rather than the product–sum distance and built-in time diversity.

4.11.5 Code design

Here we explain the code design rules, by using the example with the four-state rate 2/3 8PSK trellis code for use with transmit diversity as it was presented in [90]. It is noted that the built-in time diversity (\(R\)) of a four-state code is equal to one and, therefore, optimized in this simplest case. In order to increase the product–sum distance of the code, parallel transitions in the trellis diagram are avoided. Thus, we can only consider the error events of actual length two to maximize the minimum product–sum distance of the underlying code. The signal transitions between states of consecutive stages can be represented by a \(4 \times 4\) matrix \(G\), where the \(ij\)th element represents the signal transmitted from state \(i\) to state \(j\) between consecutive stages in the trellis diagram. Using set partitioning, the 8PSK signal set shown in Figure 4.53 is partitioned into two subsets \(A_0 = (0, 2, 4, 6)\) and \(A_1 = (1, 3, 5, 7)\). The design rules are given as follows.

**Rule 1.** Elements of each row of the matrix \(G\) are associated with signals from subsets \(A_0\) or \(A_1\). Specifically, signals with distance \(\delta_1\) and \(\delta_3\) are associated with branches diverging from one state to two adjacent states, with a state difference of two and one, respectively, where the state difference is defined as the number of bits in which two states differ (see Figure 4.54).

**Rule 2.** The distance between branches remerging at one state from two adjacent states with a state difference of two is \(\delta_2\). The pair of signals remerging from two states with a state difference of one is associated with distance \(\delta_0\) or \(\delta_3\).

According to Equation (4.149), the minimum product–sum distance should be maximized. Rule 1 associates distance \(\delta_1\) to signals diverging from one state to two adjacent states with a state difference of two (two states with state difference of two are always adjacent) and guarantees that the distance between any two signals diverging from one state is at least \(\delta_1\). Thus, if we assign \(\delta_2\) to signals remerging at one state from two states with a state difference of two (Rule 2), the minimum product–sum distance is maximized.

![Figure 4.53 8 PSK signal set](image_url)

Figure 4.53 8 PSK signal set [90] © 2002, IEEE.
product–sum distance will be:

\[ d_{\text{min}}(R) = \min \left( \delta_1^2 + \delta_2^2, \delta_0^2 + \delta_3^2 \right) = \delta_0^2 + \delta_3^2 = 4.586 E_s \]

which is greater than that of the optimal single antenna four-state 2/3 8PSK code.

Using the above code design rules, the best four-state 2/3 8PSK trellis code for use with transmit diversity when perfect interleavers are assumed is shown in Figure 4.54. An equivalent code constructed using the design tools is also shown in the figure.

The eight-state trellis code can also be constructed. Due to the constraint of the trellis structure, the eight-state code becomes the Ungerboeck code [91]. Its minimum product–sum distance is 6E_s. An eight-state code with a larger product–sum distance exists, but it is catastrophic. Obviously, the total diversity (2R) of both the four and eight-state codes is equal to two, but it will increase to four when the number of states is increased to 16, since R is increased from one to two. After experimentation with various trellis structures and signal assignments, the 16-state code is also found based on the design criteria, and is shown in Figure 4.55. The minimum product–sum distance of this code is equal to:

\[ d_{\text{min}}(R) = (\delta_0^2 \delta_1^1)(\delta_0^2 \delta_1^1) = (2 + 0.586) E_s \times (2 + 0.586) E_s = 6.69E_s^2 \]
Figure 4.56 Quasi-static fading, $M = 1$.

Figure 4.57 Quasi-static fading, $M = 2$. The dashed dotted line denotes the STBC only, and the solid lines denote the concatenated scheme.

The constructed 16-state code has an Ungerboeck representation, which means that it can be generated by a feedback-free convolutional encoder followed by the natural mapping. Performance results for concatenated code obtained using Rules 1 and 2 (R1 and 2 code) and traditional trellis codes are compared in Figures 4.56–4.63 with transmission matrix $\Gamma$ of the STBC by Alamouti.

More details on the topic can be found in [90–98].
Figure 4.58 Quasi-static fading, $M = 3$. The dashed dotted line denotes the STBC only, and the solid lines denote the concatenated scheme.

Figure 4.59 Quasi-static fading, $M = 4$. The dashed dotted line denotes the STBC only, and the solid lines denote the concatenated scheme.

4.12 ESTIMATION OF MIMO CHANNEL

Channel estimation using training sequences is required for coherent detection in BLAST. In this section we present the maximum likelihood channel estimator and the optimal training sequences for block flat fading channels and analyze the estimation error. The optimal training length and training interval that maximize the throughput for a given target bit error rate are presented as functions of the Doppler frequency and the number of antennas.
4.12.1 System model

The system consists of $M$ transmitting antennas and $N$ receiving antennas. The vector of signals at the output of $N$ receive antennas can be represented as:

$$y_i = \sqrt{\frac{\rho}{M}} H_i s_i + w_i$$  \hspace{1cm} (4.151)
Figure 4.62 Perfect interleaving, $M = 3$. Performances without interleaving (dashed lines) are also shown for comparison.

Figure 4.63 R1 and 2 eight-state 8PSK scheme versus eight-state 8PSK STTC at a spectral efficiency of 1.5 bit/s/Hz, $M = 1$. 
where $H_i$ is the $N \times M$ channel matrix, $s_i$ is the $M \times 1$ transmitted signal vector and $w_i$ is the $N \times 1$ vector of complex additive white Gaussian noise with zero mean and unit variance at time instant $i$. The average power of the components in $H_i$ and $s_i$ are normalized to unity, so the average signal to noise ratio (SNR) at each receiving antenna is $\rho$, independent of the number of transmitting antennas.

An alternative presentation without normalization gives:

$$y = Hs + w$$ \hfill (4151.a)

In this case, the signal power is constrained by $E( ss^*) \leq P$ so that $P/M$ is the maximum average power transmitted by each antenna, where $(\cdot)^*$ stands for the Hermitian matrix/vector transpose conjugate.

Starting from the definition of the mutual information exchanged in the MIMO channel, the capacity (maximum mutual information) $C(H)$ can be represented as [see Appendix 4.3, Equation A4.3.21]

$$C(H) = \log \det \left( I_N + \frac{P}{M} HH^* \right)$$ \hfill (4.151.b)

### 4.12.2 Training

During the training phase, training sequences of $L_t$ symbols long are transmitted from all the transmitting antennas. An estimate of the channel, $\hat{H}$, is obtained at the end. During the payload phase, data sequences of $L_d$ symbols long are transmitted and jointly detected. We define $L_t$ as the training length and $L = L_d + L_t$ as the training interval. The duty cycle factor $\eta = 1 - L_t/L$, is the fraction of time spent in data transmission.

### 4.12.3 Performance measure

Since the channel is continuously fading, the actual channel will deviate progressively from the channel estimate obtained at time $i = L_t$. The BER performance will be dominated by the worst channel estimation error. Therefore, we consider $\hat{H} - H_{L_t}$, as a measure of the channel estimation error, where $H_{L_t}$ is the channel at the end of the training period.

### 4.12.4 Definitions

Define the difference between the channel at time $i$ and at time $L$ as:

$$\Delta H_i = H_i - H_L$$ \hfill (4.152)

then, we can rewrite Equation (4.151) as follows:

$$y_i = \sqrt{\frac{\rho}{M}} H_L s_i + \sqrt{\frac{\rho}{M}} \Delta H_i s_i + w_i$$ \hfill (4.153)

Let $S$ be the matrix of training symbols, $S = [s_1 \ s_2 \ \cdots \ s_{L_t}]$, where $s_i$ for $1 \leq i \leq L_t$ is the $M \times 1$ training symbol vector at time $i$. Let the matrix of received signals be $Y = [y_1 \ y_2 \ \cdots \ y_{L_t}]$, and the matrix of noise be $W = [w_1 \ w_2 \ \cdots \ w_{L_t}]$. Then:

$$Y = \sqrt{\frac{\rho}{M}} H_L S + W \times S^* + \sqrt{\frac{\rho}{M}} [\Delta H_1 s_1 \ \Delta H_2 s_2 \ \cdots \ \Delta H_{L_t} s_{L_t}]$$ \hfill (4.154)

### 4.12.5 Channel estimation error

For block fading channels where the channel realization remains constant within a block of certain length and then changes to an independent realization for the next block, the ML channel estimator
is:

\[ \hat{H} = \sqrt{\frac{M}{\rho}} Y \cdot S^* \cdot (SS^*)^{-1} \]  (4.155)

and the optimal training sequences which minimize the mean square estimation error are orthogonal across all transmitting antennas, i.e.

\[ SS^* = L_t I_M \]  (4.156)

where \( I_M \) is the \( M \times M \) identity matrix. A necessary condition for the matrix inversion \((SS^*)^{-1}\) to exist is \( L_t \geq M \).

Equations (4.155) and (4.156) are suboptimal but practically appealing for continuous fading channels too. By applying these to Equation (4.154), we obtain:

\[ \hat{H} = H_L + \Delta H_{\text{noise}} + \Delta H_{\text{Doppler}} \]  (4.157)

where

\[ \Delta H_{\text{noise}} = \frac{1}{L_t} \sqrt{\frac{M}{\rho}} W S^* \]  (4.158)

is the estimation error due to noise and

\[ \Delta H_{\text{Doppler}} = \frac{1}{L_t} \sum_{i=1}^{L_t} \Delta H_i \cdot (s_i s_i^*) \]  (4.159)

is the estimation error due to the temporal variation of the channel.

### 4.12.6 Error statistic

It is easy to show that \( \Delta H_{\text{noise}} \) has i.i.d. complex Gaussian entries of zero mean and variance of \( M / (\rho L_t) \). We assume the components of \( H_i \) are uncorrelated with each other (rich scattering) and Rayleigh fading with respect to \( i \). Let \( \Delta h_n^T \) represent the \( n \)th row of \( \Delta H_{\text{Doppler}} \).

\[
E \{ (\Delta h_{n1}^T)^* \Delta h_{n2}^T \} = \delta_{n1n2} \cdot \frac{1}{L_t^2} \sum_{i_1=1}^{L_t} \sum_{i_2=1}^{L_t} s_{i_1} s_{i_1}^* \cdot [\xi (i_1 - i_2) - \xi (i_1 - L) - \xi (i_2 - L) + 1] s_{i_2} s_{i_2}^*
\]  (4.160)

where \( \delta_{jk} \) is the discrete Dirac delta function. \( \xi (x) = J_0 (2\pi f_{d_{\text{max}}} T \cdot x) \), where \( J_0 (\cdot) \) is the zeroth order Bessel function of the first kind, \( f_{d_{\text{max}}} \) is the maximum Doppler frequency and \( T \) is the symbol period.

For channel estimation tracking, it is reasonable to assume that the phase change during one training period is small, i.e. \( 2\pi f_{d_{\text{max}}} T L_t \ll 1 \). Then Equation (4.160) can be simplified as:

\[
E \{ (\Delta h_{n1}^T)^* \Delta h_{n2}^T \} = \delta_{n1n2} \cdot 2 \left( \frac{\pi f_{d_{\text{max}}} T}{L_t} \right)^2 \cdot \left( \sum_{i=1}^{L_t} (L - i) s_i s_i^* \right)^2
\]  (4.161)

using \( J_0 (x) \approx 1 - x^2/4 \) for small \( x \).

### 4.12.7 Results

The result indicates that the estimation error due to temporal variation increases quadratically with the Doppler frequency. The error also depends on the training length \( L_t \), the training interval \( L \) and the training sequences \( s_i \).
In the simple case where there is only one transmitting antenna and one receiving antenna, i.e. \( M = N = 1 \), the variance of the estimation error in Equation (4.161) can be computed directly:

\[
\sigma^2_{\text{Doppler}} = 2 \left[ \pi f_{\text{d max}} T \left( L - \frac{L_t + 1}{2} \right) \right]^2
\]  

(4.162)

We can see that if \( L \) is fixed and \( L_t \) increases, the error decreases. If \( L_t \) is fixed and \( L \) increases, the error increases. If both \( L_t \) and \( L \) increase at a fixed ratio \( L_t/L \), the error increases. For \( M, N > 1 \), the expression for the mean square estimation error is generally very complicated and it depends on the exact training sequences. A possible choice of orthogonal training sequences is the FFT matrix, i.e.

\[
S_{m,i} = e^{-j2\pi(m-1)(i-1)/L_t}
\]  

(4.163)

where \( S_{m,i} \) is the \((m, i)\)th component of the training matrix \( S \), \( 1 \leq m \leq M, 1 \leq i \leq L_t \). It can be shown that with such training sequences, the leading component of the variance is the same as Equation (4.162). Therefore, the earlier observations also apply to multiple antenna systems.

### 4.12.7.1 Example \( M, N = 4 \)

In this example, \( M = 4 \) transmitting antennas and \( N = 4 \) receiving antennas are used. The training sequences are the fast Fourier transform (FFT) sequences in Equation (4.163). The average receiving SNR is \( \rho = 15 \text{ dB} \). The carrier frequency is \( f_c = 2 \text{ GHz} \) and the maximum Doppler frequency is \( f_{\text{d max}} = 10 \text{ Hz} \), which corresponds to a pedestrian speed. The symbol period is \( T = 41 \mu\text{s} \), corresponding to the IS-136 standard (see Chapter 1). The channel coefficients are generated using the Jakes model and continuously fading. Figure 4.64 shows the mean square error (MSE) of the channel estimation as a function of the training length \( L_t \). Both \( L_t \) and the training interval \( L \) increase at a fixed ratio, \( L_t/L = 20\% \). The MSE due to noise decreases with \( L_t \) but the MSE due to temporal variation increases with \( L_t \) and \( L \). As a result, the overall MSE first decreases and then increases.

Note here that as long as the flat fading model holds, the above results will apply to systems with different symbol period, \( T \), if we scale the maximum Doppler frequency \( f_{\text{d max}} \) appropriately. This is valid because the estimation error due to temporal variation depends only on the product \( f_{\text{d max}} T \). Additional results are given in Figures 4.65 and 4.66.

More details on the topic can be found in [101–112].

![Figure 4.64](image-url)  
Figure 4.64 Channel estimation MSE versus the training length \( L_t \). Four transmitting antennas and four receiving antennas. \( L_t/L = 20\% \), \( (\rho = 15 \text{ dB}, f_{\text{d max}} = 10 \text{ Hz}) \).
Figure 4.65 Optimal training interval versus Doppler frequency ($M, N = 4, \text{BER} = 3\%$).

Figure 4.66 Maximum throughput versus the number of antennas resulting from the optimal training interval. ‘Ideal’ indicates the throughput with ideal channel estimation. ($\rho = 15 \text{ dB}, \text{BER} = 3\%$).

4.13 SPACE–TIME CODES FOR FREQUENCY SELECTIVE CHANNELS

The presentation in this section is based on [113]. If a frequency selective channel is modeled as a symbol spaced, tap delay line of length $L$, the sampled version of one frame ($K + L - 1$ time slots) of the received signal, at antenna $r$, after matched filtering can be represented as

$$y_k^r = \sum_{l=0}^{L-1} \sum_{t=1}^{M_r} h_t^r(l)c_{k-l}^r + n_k^r \quad k = 1, \ldots, K + L - 1$$

(4.164)
In Equation (4.164) $y_r^k$ is the received signal at antenna $r$ and time slot $k$, $n_r^k$ is a complex white Gaussian random noise sample at antenna $r$ and time slot $k$ with variance $N_0$ and $h_r^l(l)$ is a circularly symmetric complex Gaussian random variable with zero mean describing the $l$th tap gain. The variance of $h_r^l(l)$ is denoted as $\sigma^2(l)$ and $h_r^l(l)$ is normalized so that we have:

$$\sum_{l=0}^{L-1} \sigma^2(l) = 1$$

(4.165)

No channel knowledge at the transmitter and a coherent receiver with perfect channel state information are assumed. The vector $\mathbf{v} = (\sigma^2(0), \sigma^2(1), \ldots, \sigma^2(L-1))$ will be referred to as the power delay profile vector and is common for all subchannels. If the antennas are spaced sufficiently far apart, then $h_r^l(l)$ and $h_r^l(l')$ are independent if $t \neq t'$ or $r \neq r'$, which is referred to as spatial independence. The case when $h_r^l(l)$ and $h_r^l(l')$ are independent for $l \neq l'$, is referred to as the uncorrelated tap case. Channels are said to have uniform power delay profiles if all components in the power delay profile vector are equal. Otherwise we have a non-uniform power delay profile [111–114]. Equation (4.164) can be represented in vector form as:

$$y_r = \sum_{l=0}^{L-1} \mathbf{h}^l(l)\mathbf{C}(l) + \mathbf{n}_r$$

(4.166)

where

$$y_r = (y_1^r, \ldots, y_{K+L-1}^r)$$

$$\mathbf{n}_r = (n_1^r, \ldots, n_{K+L-1}^r)$$

$$\mathbf{h}^l(l) = (h_1^l(l), \ldots, y_{MT}^l(l))$$

$$\mathbf{C}(l) = \begin{pmatrix} 
0_{MT \times l} & \mathbf{C} & 0_{MT \times (L-1-l)} 
\end{pmatrix}$$

(4.167)

with

$$\mathbf{C} = \begin{bmatrix} 
\begin{array}{cccc}
\epsilon_1^1 & \epsilon_2^1 & \cdots & \epsilon_K^1 \\
\epsilon_1^2 & \epsilon_2^2 & \cdots & \epsilon_K^2 \\
\vdots & \vdots & \ddots & \vdots \\
\epsilon_1^{MT} & \epsilon_2^{MT} & \cdots & \epsilon_K^{MT} 
\end{array} 
\end{bmatrix}$$

(4.168)

which is the traditional code word matrix from the flat fading channel analysis. In fact, the $l = 0$ term in Equation (4.166) is exactly the flat fading signal term when $MT$ transmit antennas are employed. Due to the similarity of the other terms, the $l$th term in Equation (4.166) for $l > 0$ can be thought of as coming from an $l$th set of $MT$ virtual transmit antennas.

The $r$th row of the matrix $\mathbf{C}(l)$ represents the modulated output symbols transmitted from the $r$th transmit antenna over $K + L - 1$ time periods. Combining these rows together for $l = 0, \ldots, L - 1$ in the order of increasing $l$ gives:

$$\mathbf{C}_r = \begin{bmatrix} 
\epsilon_1^l & \epsilon_2^l & \cdots & \epsilon_K^l & 0 & 0 & \cdots & 0 \\
0 & \epsilon_1^l & \epsilon_2^l & \cdots & \epsilon_K^l & 0 & \cdots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & 0 & \epsilon_1^l & \epsilon_2^l & \cdots & \epsilon_K^l 
\end{bmatrix}$$

(4.169)

Then, Equation (4.166) is also equal to:

$$y_r = \sum_{l=1}^{MT} \mathbf{h}_r^l \mathbf{C}_r + \mathbf{n}_r$$

(4.170)
where $h_r^t = (h_r^t(0), h_r^t(1), \ldots, h_r^t(L - 1))$ is a subchannel impulse response vector between transmit antenna $t$ and receive antenna $r$.

We stack up all the signals received by the $M_R$ receive antennas to get:

$$ Y = HC_s + N $$

(4.171)

where

$$ Y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_{M_R} \end{bmatrix}, \quad N = \begin{bmatrix} n_1 \\ n_2 \\ \vdots \\ n_{M_R} \end{bmatrix}, \quad C_s = \begin{bmatrix} C_1 \\ C_2 \\ \vdots \\ C_{M_R} \end{bmatrix} $$

$$ H = \begin{bmatrix} h_1^1 & h_1^2 & \cdots & h_M^1 \\ h_2^1 & h_2^2 & \cdots & h_M^2 \\ \vdots & \vdots & \ddots & \vdots \\ h_{M_R}^1 & h_{M_R}^2 & \cdots & h_{M_R}^M \end{bmatrix} $$

Assume that the transmitted code word is $C_s$ and the erroneously decoded code word is $E_s$. Define the codeword difference matrix as $B_s = C_s - E_s$. Define the non-negative definite Hermitian matrix $A_s = B_sB_s^H$, where $H$ represents the conjugate transpose. Further, consider the $M_TLM_R \times M_TLM_R$ matrix $D_s = I_{M_R} \otimes A_s$, where $I_{M_R}$ is an $M_R \times M_R$ identity matrix and $\otimes$ is the Kronecker product. Now vectorize the channel matrix $H^T$, where $T$ represents the transpose operation, to define the channel vector $h = \text{vec}(H^T)$. Let $K = E(h^HH)$ denote the correlation matrix of $h$. We only consider the case where $K$ is full rank. Since $K$ is a positive definite matrix, Cholesky factorization yields $K = F^HF$, where $F$ is a lower triangular matrix. Using arguments from Section 4.2, the pairwise error probability is upper bounded by:

$$ P(C_s \rightarrow E_s) \leq \frac{1}{\det(I + \gamma F D_s F^H)} $$

(4.172)

where $\gamma = E_s / 4N_0$, $E_s$ is the average energy per symbol at each transmit antenna. At high SNR, Inequality (4.172) reduces to:

$$ P(C_s \rightarrow E_s) \leq \gamma^{-qM_R} \left( \prod_{i=1}^{qM_R} \lambda_i \right) $$

(4.173)

where $q$ is the rank of the matrix $A_s$ and $\lambda_i$ is the $i$th non-zero eigenvalue of $F D_s F^H$. An equation similar to Inequality (4.172) can be developed by using Equation (4.166) instead of Equation (4.170).

The maximum rank of matrix $A_s$ is $M_T L$. Thus, the maximum diversity gain of an STC employed in an frequency selective channel is $M_T M_R L$, $L$ times greater than that of the same STC for flat fading channels, which is only $M_T M_R$.

### 4.13.1 Diversity gain properties

Similarly to arguments used in Section 4.2, let $C$ and $E$ be two code word matrices from Equation (4.168). Let $C(l)$ and $E(l)$ denote the corresponding matrices from Equation (4.167). Let $B(l) = C(l) - E(l)$, then define $B_{ds}$ as:

$$ B_{ds} = \begin{bmatrix} B(0) \\ B(1) \\ \vdots \\ B(L - 1) \end{bmatrix} $$

(4.174)
which can be easily derived from $B$, and vice versa. Let $B = C - E$ be the flat fading code word difference matrix and assume the rank of the matrix $B$ is $r_b$. The matrix $B$ is similar to an upper triangular matrix $T$ whose first $r_b$ diagonal elements are non-zero and other $M_T - r_b$ diagonal elements are zero. Likewise, the matrix $B(l)$ is similar to $T(l) = 0_{M_T \times l} \ T \ 0_{M_T \times (L - 1 - l)}$. Thus, the difference matrix $B_{ds}$ is similar to the following matrix:

$$T_{ds} = \begin{bmatrix}
T(0) \\
T(1) \\
\vdots \\
T(L - 1)
\end{bmatrix}$$  \hspace{1cm} (4.175)

It is apparent that the collection of vectors consisting of all $r_b$ rows of the matrix $T(0)$ and the $r_b$th row of each of the matrices $T(l)$, $t = 1, \ldots, L - 1$ will be linearly independent, so $T_{ds}$ has rank $r_b + L - 1$ or larger. Therefore, the minimum rank of difference matrix $B_{ds}$ is $r_b + L - 1$.

### 4.13.2 Coding gain properties

Inequality (4.172) shows that the coding gain depends not only on the matrix $A$, but on the channel correlation matrix $K$ as well. With the assumption that the STC provides maximum diversity gain, $K$ is full rank, and SNR is large, Inequality (4.172) is well approximated by:

$$P(C_t \rightarrow E_s) \leq \frac{\gamma^{-M_T M_K L}}{\det(K) \det(D_t)}$$  \hspace{1cm} (4.176)

Denote $K$ as a partitioned matrix with $E(h_i^{r_i}h_j^{r_j})$ as its $(M_T \times (r_1 - 1) + t_1, M_T \times (r_2 - 1) + t_2)$th block entry $(t_1, t_2 = 1, \ldots, M_T$ for each $r_1, r_2 = 1, \ldots, M_K)$, where $E(h_i^{r_i}h_j^{r_j})$ is an $L \times L$ correlation matrix.

**Channel 1.** In this case the taps are uncorrelated with spatial independence. Based on spatial independence, $E(h_i^{r_i}h_j^{r_j})$ reduces to the 0 matrix when $r_1 \neq r_2$ or $t_1 \neq t_2$. Furthermore, from the uncorrelated tap assumption, $E(h_i^{r_i}h_j^{r_j})$ simplifies to a diagonal matrix with $\sigma^2(0), \ldots, \sigma^2(L - 1)$ along the diagonal. Then, in this case, the correlation matrix $K$ becomes a diagonal matrix, whose determinant is:

$$\det(K) = \left(\prod_{l=0}^{L-1} \sigma^2(l)\right)^{M_T M_K}$$  \hspace{1cm} (4.177)

For a specific code and **Channel 1** we want to know conditions for the best coding gain. Due to the arithmetic mean and geometric mean inequality and $\sum_{l=0}^{L-1} \sigma^2(l) = 1$, $\prod_{l=0}^{L-1} \sigma^2(l)$ will achieve the maximum $L^{-L}$ if and only if each $\sigma^2(l)$ is equal to each other, which implies the channel has uniform power delay profile.

**Channel 2.** In this case the taps are correlated with spatial independence. Again $E(h_i^{r_i}h_j^{r_j})$ reduces to a 0 matrix when $r_1 \neq r_2$ or $t_1 \neq t_2$. However, $K'_r = E(h_i^{r_i}h_j^{r_j})$ cannot be simplified to a diagonal matrix as in the case of **Channel 1**, but $K'_r$ still has exactly the same entries along the diagonal as those in **Channel 1**. More precisely, $K$ can be described as the matrix formed by arranging a set of non-diagonal matrices along the partition diagonal of a larger partitioned matrix with the appropriate zero padding. According to the property of the determinant of a partitioned matrix, we have:

$$\det(K) = \prod_{r=1}^{M_K} \prod_{t=1}^{M_T} \det(K'_r)$$  \hspace{1cm} (4.178)

According to Hadamard’s inequality that the determinant of a non-negative definite square matrix is not greater than the product of all its diagonal elements, we have:

$$\det(K'_r) \leq \prod_{l=0}^{L-1} \sigma^2(l)$$  \hspace{1cm} (4.179)
Combining Equation (4.178) and Inequality (4.179), results in:

$$\det(K) \leq \left(\prod_{l=0}^{L-1} \sigma^2(l)\right)^{M_T M_R} \tag{4.180}$$

Comparing Equation (4.177) and Inequality (4.180), we can see that with the spatial independence assumption, the coding gain of a specific STC for the uncorrelated tap assumption is always larger than or equal to that under a correlated tap assumption. In other words, the tap correlation will generally degrade the coding gain.

**Channel 3.** In this case the STC is applied to a spatially correlated frequency selective fading channel with correlated taps. Now, in general, none of the submatrices $E(h_{1t}^H h_{2t}^*)$ can be reduced to a $0$ matrix. In this case we will start from the following result in matrix calculus. Given three matrices $M_1(m \times m)$, $M_2(m \times n)$, $M_3(n \times n)$, if

$$M = \begin{bmatrix} M_1 & M_2 \\ M_2^H & M_3 \end{bmatrix}$$

is positive definite, then $\det(M) \leq \det(M_1)\det(M_3)$. This is known as Fischer’s inequality. The correlation matrix $K$ is a positive definite Hermitian matrix. So, by successively using Fischer’s inequality, we find:

$$\det(K) \leq \prod_{r=1}^{M_R} \prod_{t=1}^{M_T} \det(K_{rt}) \tag{4.181}$$

### 4.13.3 Space–time trellis code design

A systematic design procedure for space–time trellis codes (STTCs), presented in Section 4.2 for flat fading channels, will now be modified to handle frequency selective channels. The relationship between the symbols transmitted by real antennas and those transmitted by the virtual antennas is used. Define an $R$-bit binary input vector $a_k = (a_{k,1}, a_{k,2}, \ldots, a_{k,R})$ and concatenate this with a $(Q-1)R$-bit current state vector to get $\bar{a}_k = (a_{k}, a_{k-1}, \ldots, a_{k-Q+1})$. From Section 4.2 an STTC can be represented by:

$$\bar{x}_k = (x_{1k}, x_{2k}, \ldots, x_{MTk}) = \bar{a}_k G \tag{4.182}$$

where

$$G = \begin{bmatrix} g_{11} & g_{12} & \cdots & g_{1,M_T} \\ g_{21} & g_{22} & \cdots & g_{2,M_T} \\ \vdots & \vdots & \ddots & \vdots \\ g_{QR,1} & g_{QR,2} & \cdots & g_{QR,M_T} \end{bmatrix}$$

At each time slot, each component of the length $M_T$ output vector from Equation (4.182) is mapped into a constellation symbol and these symbols are transmitted simultaneously from $M_T$ antennas. In this case the $g_{ij}, i = 1, \ldots, QR, j = 1, \ldots, M_T$ can be taken from an alphabet whose size is equal to the constellation size $s$. The trellis encoder starts and ends in state zero at the beginning and end of each frame.

Consider the case with $M_T = 2$, $R = 1$, $Q = 5$, $s = 2$ (BPSK) and $L = 3$. This results in a 16-state 1 b/s/Hz BPSK STTC for a frequency selective channel with three taps. Let $\mathcal{L}$ denote the shortest length error event, as in Section 4.2, for all such codes. First, all codes with maximum diversity gain
Table 4.14 16-state BPSK STTCs with maximum $\eta$ and different $\bar{\eta}_{CP}(L)$ [113] © 2003, IEEE.

<table>
<thead>
<tr>
<th>No.</th>
<th>$G^T$</th>
<th>$\eta$</th>
<th>$\bar{\eta}_{CP}(L)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$G_{11}$</td>
<td>$\begin{bmatrix} 1 &amp; 1 &amp; 1 &amp; 0 &amp; 1 \ 1 &amp; 1 &amp; 0 &amp; 1 &amp; 1 \end{bmatrix}$</td>
<td>5.532</td>
<td>9.357</td>
</tr>
<tr>
<td>$G_{12}$</td>
<td>$\begin{bmatrix} 1 &amp; 1 &amp; 0 &amp; 1 &amp; 1 \ 1 &amp; 0 &amp; 1 &amp; 1 &amp; 1 \end{bmatrix}$</td>
<td>5.532</td>
<td>9.357</td>
</tr>
<tr>
<td>$G_{13}$</td>
<td>$\begin{bmatrix} 1 &amp; 1 &amp; 1 &amp; 1 &amp; 0 \ 1 &amp; 1 &amp; 1 &amp; 0 &amp; 1 \end{bmatrix}$</td>
<td>5.532</td>
<td>8.512</td>
</tr>
<tr>
<td>$G_{14}$</td>
<td>$\begin{bmatrix} 1 &amp; 1 &amp; 1 &amp; 0 &amp; 1 \ 1 &amp; 0 &amp; 0 &amp; 1 &amp; 0 \end{bmatrix}$</td>
<td>5.532</td>
<td>7.008</td>
</tr>
<tr>
<td>$G_{15}$</td>
<td>$\begin{bmatrix} 1 &amp; 1 &amp; 1 &amp; 0 &amp; 1 \ 0 &amp; 0 &amp; 1 &amp; 0 &amp; 1 \end{bmatrix}$</td>
<td>5.532</td>
<td>6.465</td>
</tr>
<tr>
<td>$G_{16}$</td>
<td>$\begin{bmatrix} 0 &amp; 0 &amp; 0 &amp; 1 &amp; 1 \ 1 &amp; 0 &amp; 0 &amp; 0 &amp; 1 \end{bmatrix}$</td>
<td>5.532</td>
<td>5.532</td>
</tr>
</tbody>
</table>

Figure 4.67 Performance comparison of best 16-state BPSK STTC and delay diversity code with two transmit and two receive antennas for a channel with uncorrelated power delay profile vector (1/3, 1/3, 1/3).

and maximum coding gain of $\eta = 5.532$ are found [113]. There are many such codes. Among them, those with larger $\bar{\eta}_{CP}(L)$ are chosen [113]. In this case, there are only two codes yielding maximum $\bar{\eta}_{CP}(L) = 9.357$. They are $G_{11} = (11101, 11011)^T$ and $G_{12} = (11011, 10111)^T$ respectively, as listed in Table 4.14. Monte Carlo simulation is used to evaluate the code performance. Figure 4.67

$^1 (11101, 11011)^T$ denotes $\begin{bmatrix} 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 \end{bmatrix}$ for simplicity.
Table 4.15 $q$-state BPSK STTCs for channels with two taps [113] © 2003, IEEE

<table>
<thead>
<tr>
<th>$q$</th>
<th>No.</th>
<th>$\mathbf{G}^T$</th>
<th>$\eta$</th>
<th>$\bar{\eta}_{ICP}(\mathcal{L})$</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>$\mathbf{G}_{21}$</td>
<td>$\begin{bmatrix} 1 &amp; 1 &amp; 1 \ 1 &amp; 0 &amp; 1 \end{bmatrix}$</td>
<td>4.000</td>
<td>5.968</td>
</tr>
<tr>
<td>8</td>
<td>$\mathbf{G}_{22}$</td>
<td>$\begin{bmatrix} 1 &amp; 1 &amp; 1 &amp; 1 \ 1 &amp; 0 &amp; 0 &amp; 1 \end{bmatrix}$</td>
<td>5.981</td>
<td>7.825</td>
</tr>
<tr>
<td>16</td>
<td>$\mathbf{G}_{23}$</td>
<td>$\begin{bmatrix} 1 &amp; 1 &amp; 0 &amp; 1 &amp; 1 \ 1 &amp; 0 &amp; 1 &amp; 0 &amp; 1 \end{bmatrix}$</td>
<td>7.445</td>
<td>9.973</td>
</tr>
<tr>
<td>32</td>
<td>$\mathbf{G}_{24}$</td>
<td>$\begin{bmatrix} 1 &amp; 1 &amp; 1 &amp; 0 &amp; 0 &amp; 1 \ 0 &amp; 0 &amp; 1 &amp; 1 &amp; 0 &amp; 1 \end{bmatrix}$</td>
<td>9.514</td>
<td>10.942</td>
</tr>
</tbody>
</table>

Figure 4.68 Transmission from three ($M$) transmit antennas to two ($N$) receive antennas [157].

The latest results in the field are presented in [115–156].

4.14 OPTIMIZATION OF A MIMO SYSTEM

4.14.1 The channel model

As before, the model consists of $M$ transmit and $N$ receive antenna elements. We assume that $M > N$. The antenna weights on the receive side are described as a column vector $\mathbf{V}$ with elements $V_1, V_2, \ldots$, and the vector is normalized so the norm is unity ($\mathbf{V}'\mathbf{V} = 1$). Similarly, $\mathbf{V}$ denotes the transmit weight vector. For notation, $^*$ signifies complex conjugation, and $'$ transpose and conjugate. $(M, N)$ refers to $M$ antennas at the transmitter end and $N$ antennas at the receiver end.

An example of a (3,2) system is shown in Figure 4.68. Similarly to Equation (4.151a) the transfer matrix from the transmit antennas to the receive antennas is described by transmission matrix $\mathbf{H}$ with
elements $H_{ik}$. They are random complex Gaussian quantities. A normalization:

$$E(|H_{ik}|^2) = 1$$  \hspace{1cm} (4.183)

is used. It is assumed that the angular spreads seen from both sides are so large that the antenna signals are spatially uncorrelated.

### 4.14.2 Gain optimization by singular value decomposition (SVD)

The matrix $H$ will, in general, be rectangular with $N$ rows and $M$ columns. An SVD expansion of $H$ can be represented as (see Appendix 5.1)

$$H = U_{\lambda} \cdot D \cdot V_{\lambda}'$$  \hspace{1cm} (4.184)

where $D$ is a diagonal matrix of real, non-negative singular values, the square roots of the eigenvalues of $G$, where $G = H' \cdot H$ is an $M \times M$ Hermitian matrix. The columns of the unitary matrices $U_{\lambda}$ and $V_{\lambda}$ are the corresponding singular vectors. Thus, Equation (4.184) is just a compact way of writing the set of independent channels [157]:

$$HV_1 = \sqrt{\lambda_1} U_1$$

$$HV_2 = \sqrt{\lambda_2} U_2$$

$$\vdots$$

$$HV_N = \sqrt{\lambda_N} U_N$$

The SVD is particularly useful for interpretation in the antenna context. For one particular eigenvalue, one can see that $V_i$ is the transmit weight factor for excitation of the singular value $\sqrt{\lambda_i}$.

A receive weight factor of $U'_i$, a conjugate match, gives the receive voltage $S_r$, and the square of that the received power, $P_r$:

$$S_r = U'_i U_i \sqrt{\lambda_i} = \sqrt{\lambda_i}$$

$$P_r = |S_r|^2 = \lambda_i$$  \hspace{1cm} (4.186)

This clearly shows that the matrix $H$ of transmission coefficients may be diagonalized, leading to a number of independent orthogonal modes of excitation, where the power gain of the $i$th mode or channel is $\lambda_i$. The weights applied to the arrays are given directly from the columns of the $U_{\lambda}$ and $V_{\lambda}$ matrices. Thus, the eigenvalues and their distributions are important properties of the arrays and the medium, and the maximum gain is of course given by the maximum eigenvalue. The number of non-zero eigenvalues may be shown to be the minimum value of $M$ and $N$. The situation is illustrated in Figure 4.68, where the total power is distributed among the $N$ parallel channels by weight factors $\alpha$. These coefficients are discussed later in more detail. An important parameter is the trace of $G$, i.e. the sum of the eigenvalues

$$\text{Trace} = \sum \lambda_i$$  \hspace{1cm} (4.187)

which may be shown to have a mean value of $MN$. We illustrate the above relations by an example [157].

#### 4.14.2.1 Example: $(2, 2)$ system

For the $(M, N) = (2, 2)$ example, matrix $G$ is given by:

$$G = \begin{bmatrix}
|H_{11}|^2 + |H_{12}|^2 & H_{11}H_{21}^* + H_{12}H_{22}^* \\
H_{11}^*H_{21} + H_{12}^*H_{22} & |H_{22}|^2 + |H_{21}|^2
\end{bmatrix}$$

$$= \begin{bmatrix}
a & c \\
c^* & b
\end{bmatrix}$$  \hspace{1cm} (4.188)
and the two eigenvalues are (see Appendix 5.1)

\[ \lambda_{\text{max}} = \frac{1}{2} (a + b + \sqrt{(a - b)^2 + 4|c|^2}) \]  

and

\[ \lambda_{\text{min}} = \frac{1}{2} (a + b - \sqrt{(a - b)^2 + 4|c|^2}) \]  

Note that

\[ \text{Trace} = \sum \lambda_i = a + b \]

\[ = |H_{11}|^2 + |H_{12}|^2 + |H_{21}|^2 + |H_{22}|^2 \]

so the sum of the eigenvalues displays the full fourth-order diversity.

The distribution of ordered eigenvalues may be found in [9], from which the distributions for \( \lambda_{\text{min}} \) and \( \lambda_{\text{max}} \) may be derived:

\[ p(\lambda_{\text{min}}) = 2e^{-2\lambda} \]  

\[ p(\lambda_{\text{max}}) = e^{-\lambda}(\lambda^2 - 2\lambda + 2) - 2e^{-2\lambda} \]

In this particular case, it may be shown that the mean values are:

\[ E(\lambda_{\text{max}}) = 3.5 \quad E(\lambda_{\text{min}}) = 0.5 \]

The minimum eigenvalue is Rayleigh distributed with mean power 0.5. The cumulative probability distribution for \( \lambda_{\text{max}} \) is:

\[ \Pr(\lambda_{\text{max}} < x) = 1 - e^{-x}(x^2 + 2) + e^{-2x} \approx x^4/12 \quad x \ll 1 \]

One can show that for the case of standard diversity \((M, N) = (1, 4)\):

\[ \Pr(P < x) = 1 - e^{-x}(1 + x + x^2/2 + x^3/6) \approx x^4/24 \quad x \ll 1 \]

so the \((2, 2)\) case displays full fourth-order diversity but with twice the cumulative probability for the same power level.

The cumulative probability distributions are shown in Figure 4.69, where the maximum eigenvalue (the array gain) follows the fourth-order maximum ratio diversity distribution quite closely.

One should be aware that in order to make full benefit of the maximum eigenvalue, the full knowledge of the channel at the transmitter is required, otherwise the eigenvectors cannot be found.

### 4.14.3 The general \((M, N)\) case

For the \((4, 4)\) case in Figure 4.70, the two arrays have 16 different uncorrelated transmission coefficients, so the diversity order is 16. In the asymptotic limit when \(N\) is large, it may be shown [158, 159] that the largest eigenvalue is bounded above by:

\[ \lambda_{\text{max}} < (\sqrt{c} + 1)^2N; \quad c = M/N \geq 1 \]

whereas the smallest eigenvalue is bounded below by:

\[ \lambda_{\text{min}} > (\sqrt{c} - 1)^2N \quad c \geq 1 \]

In the previous examples, \(c = 1\), and the upper asymptotic bound for this case is \(4N\). These bounds should not be understood as absolute bounds, but rather as limits approached as \(N\) tends to infinity for a fixed \(c\).
Figure 4.69 Cumulative probability distribution of eigenvalues for a \((T, R) = (2, 2)\) array with four uncorrelated paths. The maximum eigenvalue follows closely the fourth-order diversity with a shift of 0.75 dB.

Figure 4.70 Cumulative probability distribution of eigenvalues (power) for two arrays of four elements each, including the sum of eigenvalues corresponding to a \((1, 16)\) case.

The mean array gains (mean of the maximum eigenvalues) are shown in Figure 4.71, together with the upper bound and the gain for the correlated, free space case, \(N^2\). For \(N = 10\), the true mean gain is just 1 dB below the upper bound. For a partly correlated case, we can expect the gain to lie between the \(\rho = 0\) and the \(\rho = 1\) cases, where \(\rho\) is the spatial correlation coefficient between the elements.
The gain relative to one element of \((N, N)\) arrays in a correlated situation \((\rho = 1)\), and in an uncorrelated case \((\rho = 0)\). The upper bound equals \(4N\), and is the asymptotic upper bound for the maximum eigenvalue for \(N\) tending to infinity.

In some situations, it might be advantageous to have more antennas on one side than on the other, especially for asymmetric situations with heavy downloading of data from a base station. Again, the asymptotic upper bound for the largest eigenvalue is useful, Inequality (4.197). Introducing \(M = cN\) directly we find:

\[
G_{\text{upper bound}} = (\sqrt{M} + \sqrt{N})^2
\]

which, asymptotically, will approach \(M\) for large values of \(M\) and fixed \(N\). This clearly illustrates that the composite gain of the link cannot be factored into one belonging to the transmitter and one belonging to the receiver.

### 4.14.4 Gain optimization by iteration for a reciprocal channel

Since the channel is reciprocal, exactly the same weights may be used for transmission as for reception. Consider now the situation with transmission from \(M\) transmit antennas to \(N\) receive antennas (or vice versa). The iteration starts with an arbitrary \(V_1\), in the numerical calculations, chosen as a unit vector with equal elements. At the receive side, the weights are adjusted for maximum gain, and the same weights are then used for transmit since the channel is reciprocal. This may then be repeated a number of times. In principle, the process might converge to an eigenvalue different from the maximum one, but experience shows excellent performance [157].

An example of the convergence in the mean is shown for a \((3, 3)\) case in Figure 4.72. After a few iterations, the gain has converged to the steady state. This might actually be a computationally efficient way of finding the maximum gain solution in practice without going through the trouble of finding the eigenvectors. For details regarding eigenvalue decomposition see Appendix 5.1.
Figure 4.72 Convergence of gain by iterative transmissions between receiver and transmitter for a (3, 3) case. The gain values are mean values.

4.14.5 Spectral efficiency of parallel channels

From Figure 4.70 with four independent channels one can see that there are other options for using the eigenvalues than using the largest for maximum gain. Another option is to keep them as parallel channels with independent information, as discussed in the previous sections of this chapter. The knowledge about the distribution of the eigenvalues and the upper and lower bounds may now be used for evaluating bounds on the theoretical capacity of the link. Shannon’s capacity measure gives an upper bound on the realizable information rates through parallel channels, and how the power should be distributed over the channels to achieve maximum capacity through ’water filling’ [160].

From Equation (4.151b), the basic expression for the spectral efficiency measured in bits/s/Hz for one Gaussian channel is given by:

$$C = \log_2(1 + P) \text{ bits } / \text{s } / \text{Hz} \quad (4.200)$$

where $P$ is the signal to noise ratio, SNR, for one channel.

Assuming all noise powers to be the same, the ’water filling’ concept is the solution to the maximum capacity, where each channel is filled up to a common level $D$:

$$\frac{1}{\lambda_1} + P_1 = \frac{1}{\lambda_2} + P_2 = \frac{1}{\lambda_3} + P_3 = \cdots = D \quad (4.201)$$

Thus, the channel with the highest gain receives the largest share of the power. The constraint on the powers is that:

$$\sum_i P_i = P \quad (4.202)$$

The weight factors $\alpha_i$ in Figure 4.68 equal $P_i/P$. In the case where level $D$ drops below a certain $1/\lambda_i$, that power is set to zero. In the limit where the SNR is small ($P < 1/\lambda_2 - 1/\lambda_1$), only one eigenvalue, the largest, is left, and we are back to the maximum gain solution of the previous section.

For the case of $(M, N) = (2, 2)$ (Figure 4.69) for $P < 2 - 2/7 = 2.34$ dB, only the largest eigenvalue is active, using the mean values from Equation (4.192). The capacity equals

$$C = \sum_{N'} \log_2(1 + \lambda_i P_i) = \sum_{N'} \log_2(\lambda_i D) \quad (4.203)$$
where the summation is over all channels with non-zero powers. The water filling is of course
dependent on the knowledge of the channels on the transmit side. In the case where the channel is
unknown at the transmitter, the only reasonable division of power is a uniform distribution over the
antennas, i.e.

$$P_i = \frac{P}{M}$$  \hspace{1cm} (4.204)

$M$ being the number of transmit antennas [100]. It may also be argued that the transmit antenna ‘sees’
$M$ eigenvalues, not taking into account that there are only $N$ non-zero eigenvalues. Thus, power is
lost by allocating power to the zero-valued eigenvalues.

It also follows that when $M = N$, the difference between the capacity for known and unknown
channels is small for large $P$.

### 4.14.6 Capacity of the $(M, N)$ array

It follows from Inequalities (4.197) and (4.198) that the instantaneous eigenvalues are limited by:

$$\left(\sqrt{M} - \sqrt{N}\right)^2 < \lambda_i < \left(\sqrt{M} + \sqrt{N}\right)^2$$  \hspace{1cm} (4.205)

So, for $M$ much larger than $N$, all the eigenvalues tend to cluster around $M$. Furthermore, each of
them will be non-fading due to the high $MN$th order diversity. Thus, the uncorrelated asymmetric
channel with many antennas has a very large theoretical capacity of $N$ equal, constant channels with
high gains of $M$. The above illustrates in a mathematical sense the observation of Winters [161] that
$M$ should be of the order $2N$. In the limit of large $M$ and $N$, with $M$ much larger than $N$, the capacity
is easily found to be:

$$C = N \log_2 \left( 1 + \frac{P}{N M} \right)$$  \hspace{1cm} (4.206)

with the result that the theoretical capacity grows linearly with the number of elements $N$ [160, 161]
for $M/N$ fixed. The result may conveniently be interpreted as $N$ parallel channels, each with $1/N$ of
the power and each having a gain of $M$. Note that this capacity is higher than the one used in [160,
161], where the power is divided between the $M$ antennas instead of the $N$ channels, given as:

$$C_{\text{unknown}} = N \log_2 (1 + P)$$  \hspace{1cm} (4.207)

The numerical results shown in Figure 4.73 support this approximate analysis for the mean values.
It should be remembered that the potential gains are higher when a certain outage probability is studied
due to the high order diversity effects. This is illustrated in Figure 4.74, which shows the cumulative
probability distribution (on a log scale) of the capacity for the case of four receiving elements, and
four and twelve transmitting elements. The signal to noise ratio is 20 dB for the $(1, 1)$ case, and it
is worth emphasizing that the total power radiated remains constant. The improvement going from
four to twelve transmitting antennas is mainly due to the improved gain of the smallest eigenvalues
as indicated by Inequality (4.205). Using Equation (4.206) in the $(12, 4)$ case gives 32.9 b/s/Hz.

### 4.15 MIMO SYSTEMS WITH CONSTELLATION ROTATION

#### 4.15.1 System model

In this section we consider a system in which the base station transmitter has $L$ antennas that transmit
simultaneously. Each component of an $L$-dimensional signal point is transmitted on one antenna, and
the receiver makes a decision based on the entire $L$-dimensional received vector. The performance of
the system is optimized by rotating the baseline constellations in $L$ dimensions.
Figure 4.73 Mean capacity for two arrays of each $N$ elements. The capacity grows linearly with the number of elements and is approximately the same for the known and the unknown channel. The total transmitted power is constant.

Figure 4.74 The cumulative probability distribution of capacity on a log scale for the $(M, N) = (1, 1), (4, 4),$ and $(12, 4)$ cases. The basic signal to noise ratio is 20 dB. The total radiated power is the same in all cases.
The received signal with one antenna receiver at the mobile is:

\[ r(t) = \sum_{i=1}^{L} a_i m_i s_i(t) \cos(w_i t + \theta_i) + n(t), \quad 0 \leq t \leq T \]  

(4.208)

The signal from the transmitter’s \( i \)th antenna is a pulse-amplitude-modulated (PAM) signal with amplitude \( m_i \), the pulse shape \( s_i(t) \), the fading amplitude \( a_i \) and \( n(t) \) is a white Gaussian noise process with power spectral density \( N_0/2 \). It is assumed that the fading amplitude for a given link is constant over the signalling interval \([0, T]\) and that the receiver uses coherent detection.

The signals \( s_i(t), s_j(t), i \neq j \) are assumed to be orthogonal, and all of the energy of \( s_i(t), 1 \leq i \leq L \), is contained in \([0, T]\). The optimum receiver consists of a bank of \( L \) correlators. The output of the \( i \)th correlator is:

\[ y_i = 2 \int_{0}^{T} r(t)s_i(t) \cos(w_i t + \theta_i) dt = a_i m_i E_s + \eta_i, \quad 1 \leq i \leq L \]  

(4.209)

The received vector \( y = (y_1, y_2, \ldots, y_L) \) is fed into a decision device that estimates the transmitted vector \( m = (m_1, m_2, \ldots, m_L) \). To simplify notations in this section we do not use bold fonts for some vectors and matrices. It is assumed that the receiver can estimate the fading amplitudes \( a_i, 1 \leq i \leq L \). The receiver finds the \( L \)-dimensional constellation point in \( C \), with coordinates suitably amplified, which has the closest Euclidean distance to the received vector \( y \). That is, the receiver picks the symbol \( \hat{m} = (\hat{m}_1, \hat{m}_2, \ldots, \hat{m}_L) \in C \) that minimizes \( \sum_{i=1}^{L} (y_i/E_s - a_i(\hat{m}_i))^2 \). A symbol detection error occurs when \( \hat{m} \neq m \). The fading amplitudes \( a_i \) are modelled as independent and identically distributed Rayleigh random variables with the common probability density function.

### 4.15.2 Performance in a Rayleigh fading channel

Assuming that all points in an \( L \)-dimensional constellation \( C \) with \(|C|\) points, are transmitted with equal probability, an upper bound on the average probability of symbol detection error can be obtained from the union bound:

\[ P(\text{error}) \leq \frac{1}{|C|} \sum_{m} \sum_{\hat{m} \neq m} P(m \rightarrow \hat{m}) \]  

(4.210)

where \( P(m \rightarrow \hat{m}) \) is the probability that the received symbol is closer to the \( L \)-dimensional symbol \( \hat{m} = (\hat{m}_1, \hat{m}_2, \ldots, \hat{m}_L) \) than to \( m = (m_1, m_2, \ldots, m_L) \), given that \( m \) was transmitted.

If the receiver estimates the fading amplitudes in Equation (4.209) perfectly, and if a minimum distance decoding rule is used by the receiver, then the pairwise error probability, conditioned on the fading amplitude vector \( (a_1, a_2, \ldots, a_L) \), can be upper bounded using the Chernoff bound as follows:

\[ P(m \rightarrow \hat{m} | a_1, \ldots, a_L) \leq \prod_{i=1}^{L} \exp \left\{ -a_i^2 (m_i - \hat{m}_i)^2 E_s / (8N_0) \right\} \]  

(4.211)

Assuming that the fading amplitudes are independent, averaging over the probability density function for \( a_1 \) gives

\[ P(m \rightarrow \hat{m}) \leq \prod_{i=1}^{L} \frac{1}{1 + (m_i - \hat{m}_i)^2 E_s / (8N_0)} \leq \prod_{i=1}^{L} \frac{8N_0}{E_s (m_i - \hat{m}_i)^2} \]  

(4.212)

At sufficiently high SNR, Equation (4.210) is dominated by the largest \( P(m \rightarrow \hat{m}) \) term, and assuming \( m_i \neq \hat{m}_i, 1 \leq i \leq L \), then \( P(\text{error}) \propto 1 / (E_s / N_0)^{L} \). From Equation (4.212), the following
quantity, which we call the constellation gain in Rayleigh fading channel, gives an indication of the performance of a signal constellation at high SNR:

\[ c_{\text{Rayleigh}}(C) = \min_{m, \tilde{m} \in C, m \neq \tilde{m}} \prod_{i=1}^{L} (m_i - \tilde{m}_i)^2 \]  

(4.213)

In [162] this parameter is called the constellation figure of merit. In order not to degrade the performance of the constellation in the AWGN channel, we apply a transformation to the constellation that preserves the Euclidean distances between points, but improves the constellation’s gain in fading. If we represent the original constellation as a \(|C| \times L|\) matrix \(C\), where each row of the matrix corresponds to a point in the \(L\)-dimensional constellation, one possible distance-preserving transformation is to multiply this matrix by an orthogonal \(L \times L\) matrix \(A\). The optimal matrix \(A\) maximizes \(c_{\text{Rayleigh}}(CA)\). It is shown in Appendix 4.1 how an \(L \times L\) orthogonal matrix \(A\) can be written as the product of \(\binom{L}{2}\) rotation matrices and a reflection matrix. Such multiplication of the constellation matrix \(C\) by an arbitrary orthogonal matrix \(A\) has the following geometrical interpretation. The constellation is rotated with respect to the \((i, j)\) plane by an amount \(\theta_{ij}, 1 \leq i < L - 1, i + 1 \leq j \leq L\) and there are \(\binom{L}{2}\) such rotations.

Then the constellation is reflected in the \(i\)th axis, where the matrix \(\hat{I}\) has an \((i, i)\) entry equal to \(-1\) and the number of such reflections is equal to the number of \(-1\) elements on the main diagonal of \(\hat{I}\).

Writing \(A = Q\hat{I}\) where \(Q\) is the product of \(\binom{L}{2}\) rotation matrices in Equation (A4.1.1), we see that

\[ c_{\text{Rayleigh}}(CA) = c_{\text{Rayleigh}}(CQ\hat{I}) = c_{\text{Rayleigh}}(CQ) \]  

(4.214)

where the second identity in Equation (4.214) follows because the matrix \(CQ\hat{I}\) is just the matrix \(CQ\) with several of its columns negated, and negating the columns of a constellation matrix does not affect the \(c_{\text{Rayleigh}}\) of the constellation. Rather than look for an optimal constellation \(CA\) it is sufficient to look for an optimal constellation \(CQ\).

The method of obtaining an optimal constellation \(C_{\text{opt}} = CQ\), one with maximum \(c_{\text{Rayleigh}}(CQ)\), given a starting constellation \(C\), is to vary \(\binom{L}{2}\) rotation angles according to a numerical optimization or search algorithm. The constellation \(C\) is rotated with respect to the \((i, j)\) plane, \(1 \leq i \leq L - 1, i + 1 \leq j \leq L\).

For example, when \(L = 2\) the baseline constellation matrix is

\[
C = \begin{bmatrix}
1 & 1 \\
1 & -1 \\
-1 & 1 \\
-1 & -1
\end{bmatrix}.
\]  

(4.215)

In this case, maximization of \(c_{\text{Rayleigh}}\) is done by varying only one rotation angle. The optimal angle of rotation for this constellation can be found using an exhaustive search to be \(\theta_{\text{opt}} = 31.7^\circ\), assuming a discretization interval of 0.1°. The optimally rotated constellation is:

\[
C_{\text{opt}} = \begin{bmatrix}
-0.325 & -1.376 \\
-1.376 & 0.325 \\
1.376 & -0.325 \\
0.325 & 1.376
\end{bmatrix}
\]  

(4.216)

These two constellations are shown in Figure 4.75. Each row in the constellation matrix corresponds to a point \((m_1, m_2)\). With \(L = 3\), the baseline constellation \(C\) consists of the vertices of a three-dimensional cube, and optimization is done over three rotation angles. Using a discretization interval of 1° our search procedure produced the rotation vector \(\theta_{\text{opt}} = [\theta_{12}, \theta_{13}, \theta_{23}] = [24^\circ, 36^\circ, 66^\circ]\).
Both the baseline constellation and the rotated constellation are given as:

\[
C = \begin{bmatrix}
1 & 1 & 1 \\
1 & 1 & -1 \\
1 & -1 & 1 \\
1 & -1 & -1 \\
-1 & 1 & 1 \\
-1 & 1 & -1 \\
-1 & -1 & 1 \\
-1 & -1 & -1
\end{bmatrix}, \quad C_{\text{opt}} = \begin{bmatrix}
0.177 & 0.474 & -1.656 \\
-0.997 & -1.003 & -0.998 \\
-0.480 & 1.654 & -0.181 \\
-1.655 & 0.176 & 0.476 \\
1.655 & -0.176 & -0.476 \\
0.480 & -1.654 & 0.181 \\
0.997 & 1.003 & 0.998 \\
-0.177 & -0.474 & 1.656
\end{bmatrix}
\] (4.217)

For \(L \geq 4\) an exhaustive search over the \(\binom{L}{2}\) rotation angles in order to maximize \(c_{g_{\text{Rayleigh}}}\) for the \(L\) cube proved to be too time consuming. For these constellations, many rotation vectors could be picked at random, and a gradient-based approach can be used to vary the rotation angles so as to converge to a local maximum. For \(L = 4\), there are six degrees of freedom, and the rotation vector found was \(\theta_{\text{opt}} = [\theta_{12}, \theta_{13}, \theta_{14}, \theta_{23}, \theta_{24}, \theta_{34}] = [206^\circ, 15^\circ, 306^\circ, 42^\circ, 213^\circ, 31^\circ]\).

For \(L = 5\) there are ten degrees of freedom, and the rotation vector was found as \(\theta_{\text{opt}} = [\theta_{12}, \theta_{13}, \theta_{14}, \theta_{15}, \theta_{23}, \theta_{24}, \theta_{25}, \theta_{34}, \theta_{35}, \theta_{45}] = [294^\circ, 349^\circ, 18^\circ, 340^\circ, 103^\circ, 184^\circ, 114^\circ, 275^\circ, 212^\circ, 25^\circ]\).

Some results are shown in Figures 4.76 and 4.77.

### 4.16 DIAGонаL ALGebraic SPACE–TIмE BLOCK CODES

In the previous section we established the basic concept of space time system improvements based on constellation rotation. In this section, we present a new family of linear ST block codes by the use of rotated constellations and the Hadamard transform that will be referred to as diagonal algebraic ST (DAST) block codes. These codes have a normalized rate of 1 symbol/s and achieve the full diversity over \(n\) transmit and \(m\) receive antennas. They maintain their diversity and coding gains over all real or complex constellations carved from the ring of complex integers \(\mathbb{Z}[i]\), with \(i = \sqrt{-1}\), such as pulse–amplitude modulation (PAM) or quadrature–amplitude modulation (QAM). Due to the lattice structure of these codes, the ML decoding can be implemented by the sphere decoder at moderate complexity independent of the transmission rate (see Appendix 4.2) or [171, 172]. The DAST block codes outperform the ST codes from orthogonal design (see Section 4.3) for \(n > 2\).

#### 4.16.1 System model

Based on Equations (4.213) and (4.214), the minimum product distance of the constellation \(Q\) will be defined as

\[
d_{d,\text{min}} = \min_{x_1 \neq x_2 \in Q} \prod_{j=1}^{d} |y_j|
\] (4.218)
Figure 4.76 Probability of bit error for several transmission schemes. $L = 3$: (——) baseline, (-x-) optimal constellation, (—-) identical transmissions.

Figure 4.77 Probability of bit error for several transmission schemes. $L = 5$: (——) baseline, (-x-) optimal constellation, (—-) identical transmissions.

Some of the results for quasi-optimal rotations with the good values of the minimum product distance are reported in [171, 172]. In [171], the construction of rotations $M_d$ in dimension $d$ was done in an iterative manner in a ‘Hadamard’ way as follows:

$$M_d = \begin{bmatrix} M_{d/2}^1 & -M_{d/2}^2 \\ M_{d/2}^2 & M_{d/2}^1 \end{bmatrix}$$

(4.219)

where $M_{d/2}^1$ is the optimal real rotation in dimension $d/2$ and $M_{d/2}^2$ is an orthogonal transformation in dimension $d/2$ depending only on one parameter [171]. Then, one varies this parameter in order to choose the rotation that maximizes the minimum product distance. This method works very well for
the dimensions \( d = 2, 3, 4 \) and 6. It becomes less successful for \( d \geq 8 \), since too many parameters are excluded in the rotations in high dimensions. Table 4.16 presents the first row of the optimal real rotations found in [171] along with \( d_{d, \text{min}} \) for \( d = 2, 4 \) that is used to construct the DAST block codes in this section. The rest of the rotation matrix can easily be obtained from Equation (4.219). For the dimensions \( d = 2^q, q \geq 3 \), the rotations given in [172] are used, which are constructed on the real part of the cyclotomic number field of degree \( 4d \): \( Q(\cos(2\pi/8d)) \), which give relatively good values of the minimum product distance [172]:

\[
d_{d, \text{min}} = \sqrt{2}/(2d)^{d/2}
\]

(4.220)

Equation (4.221) shows the MATLAB program which generates the rotation matrix \( M_d \) of any dimension \( d = 2^q \) constructed on the number fields \( Q(\cos(2\pi/8d)) \). This method to generate full modulation diversity rotations is attractive, especially for large \( d \). Note that the best rotations in [171] give better \( d_{d, \text{min}} \) for \( d = 2, 4 \) while starting from \( d = 8 \), the rotations given in (4.221) are better.

\[
M = \text{sqrt}(2/d) \ast \cos(p/4 \ast d) \ast (4 \ast [1 : d]' - 1) \ast (2 \ast [1 : d] - 1)
\]

(4.221)

### 4.16.2 The DAST coding algorithm

If \( M_n \) is a rotation of dimensions \( n \times n \) (with, \( n = 1, 2 \) or \( n \) is a multiple of 4), which generates a full modulation diversity lattice, then DAST block code in dimensions \( n \times n \) is constructed as:

\[
\Xi_n \triangleq H_n \text{diag} (x_1, \ldots, x_n)
\]

(4.222)

where \( x = (x_1, \ldots, x_n)^T = M_n a \), and \( a = (a_1, \ldots, a_n)^T \) is the information symbol vector. In the sequel, we denote the entries of the Hadamard matrix \( H_n \) by \( w_{ij} \) (Walsh) in order to differentiate them from the entries of the channel transfer matrix \( h_{ij} \). The Hadamard transform is a real unitary transformation that exists for 1, 2, and all the dimensions multiple of 4. In dimension \( n \), the Hadamard transform \( H_n \) satisfies \( H_n H_n^T = n I_n \), with \( I_n \) the identity matrix in dimension \( n \).

As an example for \( n = 2 \) the corresponding DAST block code is:

\[
\Xi_2 \triangleq \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ 0 \end{bmatrix} = \begin{bmatrix} x_1 & x_2 \\ x_1 & -x_2 \end{bmatrix}
\]

(4.223)

where \( x = (x_1, x_2)^T = M_2 a \), and \( M_2 \) is the two-dimensional rotation matrix given in Table 4.16. For \( n = 4 \), the corresponding DAST block code is given by

\[
\Xi_4 \triangleq \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \begin{bmatrix} x_1 & 0 & 0 & 0 \\ 0 & x_2 & 0 & 0 \\ 0 & 0 & x_3 & 0 \\ 0 & 0 & 0 & x_4 \end{bmatrix} = \begin{bmatrix} x_1 & x_2 & x_3 & x_4 \\ x_1 & -x_2 & x_3 & -x_4 \\ x_1 & x_2 & -x_3 & -x_4 \\ x_1 & -x_2 & -x_3 & x_4 \end{bmatrix}
\]

(4.224)

where \( x = (x_1, x_2, x_3, x_4)^T = M_4 a \), and \( M_4 \) is the four-dimensional rotation matrix given in Table 4.16.

The DAST block code \( \Xi_n \) has a transmit diversity equal to \( n \) under quasi-static fading assumption. When \( n \) is a power of 2 and for the rotations given in Section 4.16.1, the coding gain of the DAST

<table>
<thead>
<tr>
<th>Dimension</th>
<th>Column</th>
<th>( d_{d, \text{min}} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>1–2</td>
<td>0.526</td>
</tr>
<tr>
<td>4</td>
<td>1–4</td>
<td>0.201</td>
</tr>
</tbody>
</table>
block code, equals:

\[ \delta = \begin{cases} 2^{n/2}, & \text{for } n = 2, 4 \\ \sqrt[2n-1]{n}, & \text{for } n \geq 8. \end{cases} \tag{4.225} \]

To prove it, let \( y = x - e = M_n(a - b) \) such that \( a \neq b \). We can write the DAST block code at \( y \) as Equation (4.222), \( \Xi_n = H_n \text{diag} (y_1, \ldots, y_n) \). Since \( M_n \) generates a full modulation diversity lattice, one has \( y_j \neq 0 \forall j = 1 \cdots n \) taken over all the vectors \( a \neq b \) in the considered constellation. It follows that the matrix \( \text{diag} (y_1, \ldots, y_n) \) is full rank, and also \( \Xi_n \) is full rank over all the differences of codewords. For the coding gain one computes (see Equations 4.25–4.30):

\[
\det \left( \Xi_n \Xi_n^H \right) = \det \left( H_n \text{diag} (y_1, \ldots, y_n) \text{ diag} (y_1^*, \ldots, y_n^*) H_n^T \right) = \det \left( n I_n \text{ diag} \left( |y_1|^2, \ldots, |y_n|^2 \right) \right) = n^n \prod_{j=1}^{n} |y_j|^2 \tag{4.226} \]

The coding gain is defined as the minimum of \( \det \left( \Xi_n \Xi_n^H \right)^{1/n} \), computed over all the differences between distinct codeword pair. By taking the minimum over \( y \) of the determinant above and then taking the \( n \)th root, one obtains the coding gain of the DAST block code:

\[ \delta_n = n \left( d_{n, \min} \right)^{2/n}. \tag{4.227} \]

From Table 4.16 one has \( d_{3, \min} = 1/\sqrt{3} \), which gives \( d_3 = 2/\sqrt{3} \), and \( d_{4, \min} = 1/40 \), which gives \( \delta_4 = \sqrt{2}/\sqrt{3} \). For \( n \geq 8 \), and for the rotations given in Section 4.16.1, replacing Equation (4.220) in Equation (4.227) gives \( \delta_8 = 1/2^{(n-1)/n} \). Note that the coding gain given in Equation (4.225) is greater than 0.5 and it approaches this value when \( n \) increases. For example, \( \delta_8 = 0.5453 \) and \( \delta_{32} = 0.5109 \). In Equation (4.227) there is a multiplicative factor in the coding gain expression because in the model we normalize the radiated power at a given SNR by the number of transmit antennas \( n \) by multiplying the noise variance by \( n \). If the normalization is done at the transmitter side then the coding gain will be \( \delta_n = (d_{n, \min})^{2/n} \).

### 4.16.3 The DAST decoding algorithm

During \( n \) periods of time, the received signal is given by an \( m \times n \) matrix:

\[ \mathbf{r} = H \left( H_n \text{diag} (x_1, \ldots, x_n) \right) + \mathbf{v} \tag{4.228} \]

where the \( m \times n \) complex matrix \( \mathbf{v} \) has independent Gaussian distributed random variables of variance per real dimension as entries. Equivalently, one has:

\[ \mathbf{r} = (HH_n) \text{ diag} (x_1, \ldots, x_n) + \mathbf{v} \]

\[ \text{vec} \left( \mathbf{r}^T \right) = \begin{bmatrix} \text{diag} (H_1 H_n) \\ \vdots \\ \text{diag} (H_m H_n) \end{bmatrix} x + \text{vec} \left( \mathbf{v}^T \right) \tag{4.229} \]

where \( H_j \) denotes the \( j \)th row of \( H \) (the MIMO channel matrix) representing the \( j \)th receive antenna, and \( \text{vec} (\mathbf{r}) \) arranges the matrix \( \mathbf{r} \) in a one column vector by putting its columns one after the other. If at each receive antenna \( 1 \leq j \leq m \) we write \( \text{diag} \left( H_j H_n \right) = \text{diag} (\alpha_1, \ldots, \alpha_j) \), then the received
signal is given by:

\[
\mathbf{r}_1 \triangleq \text{vec} \left( \mathbf{r}^T \right) = \begin{bmatrix}
\alpha_{11} & 0 & \ldots & 0 \\
0 & \alpha_{12} & \ddots & \vdots \\
\vdots & \ddots & \ddots & 0 \\
0 & \ldots & 0 & \alpha_{1n} \\
\alpha_{m1} & 0 & \ldots & 0 \\
0 & \alpha_{m2} & \ddots & \vdots \\
\vdots & \ddots & \ddots & 0 \\
0 & \ldots & 0 & \alpha_{mn}
\end{bmatrix} \mathbf{M}_n \mathbf{a} + \text{vec} \left( \mathbf{v}^T \right) \triangleq \mathbf{M}_n \mathbf{a} + \mathbf{v}_1.
\] (4.230)

Since the Hadamard transform is an orthogonal transformation, the variables \(\alpha_{ij}, i = 1 \cdots m, \ j = 1 \cdots n\) are independent and identically distributed (i.i.d.) complex Gaussian variables with variance \(n/2\) per real dimension.

From Equation (4.230) it is easier to understand how the DAST block codes exploit the transmit diversity. Using a DAST block code over \(n\) transmit antennas and \(m\) receive antennas is equivalent to sending the word \((x_1, \ldots, x_n)\) over one transmit antenna and \(m\) receive antennas during \(n\) periods of time, where the channel changes randomly every time instant (since the fading between each transmit–receive antenna pair is independent). The latter scheme has a diversity of \(mn\) since the lattice from which we transmit the words has a full modulation diversity [171].

A perfect CSI is assumed to be available at the receiver. First we perform maximum ratio combining of Equation (4.230). This yields:

\[
r_2 = \mathbf{A}^H \mathbf{r}_1 = \text{diag} \left( \sum_{j=1}^{m} |\alpha_{j1}|^2, \ldots, \sum_{j=1}^{m} |\alpha_{jn}|^2 \right) \mathbf{M}_n \mathbf{a} + \mathbf{v}_2
\] (4.231)

where \(\mathbf{v}_2\) is a colored Gaussian noise with covariance matrix \(\mathbf{E}[\mathbf{v}_2 \mathbf{v}_2^H] = 2\sigma^2 \mathbf{A}^H \mathbf{A}\). In order to whiten the noise, we multiply the received signal in Equation (4.231) by \((\mathbf{A}^H \mathbf{A})^{-1/2}\), giving:

\[
r_3 = (\mathbf{A}^H \mathbf{A})^{-1/2} r_2 = \text{diag} \left( \sqrt{\sum_{j=1}^{m} |\alpha_{j1}|^2}, \ldots, \sqrt{\sum_{j=1}^{m} |\alpha_{jn}|^2} \right) \mathbf{M}_n \mathbf{a} + \mathbf{v}_3
\] (4.232)

with \(\mathbf{v}_3\) an \(n \times 1\) additive white Gaussian noise. Then we apply the sphere decoder [168, 169] on the real and imaginary parts of Equation (4.232). The sphere decoder takes advantage of the lattice structure of the received signals and searches the closest lattice points to the received signal, which are enclosed in a sphere of radius \(C_0\) centered at the received signal. Each time a lattice point of a norm less than \(C_0\) is found, we reduce the sphere radius accordingly and restart the search until an empty sphere is reached. The choice of \(C_0\) depends on the considered lattice, which is generated by

\[
\text{diag} \left( \sqrt{\sum_{j=1}^{m} |\alpha_{j1}|^2}, \ldots, \sqrt{\sum_{j=1}^{m} |\alpha_{jn}|^2} \right) \mathbf{M}_n
\] in Equation (4.232), as well as on the additive noise level. Some results are shown in Figures 4.78–4.81. In Figure 4.78, we compare the Alamouti code \(\mathbf{G}_2\) with the code \(\Xi_2\) for one and two receive antennas with the 4-QAM modulation. At the same spectral efficiency of 2 b/s/Hz, the Alamouti scheme shows almost 1 dB of gain over the code \(\Xi_2\). For \(n = 2\) transmit antennas it seems difficult to outperform the Alamouti scheme since it is the unique complex orthogonal design transmitting at a normalized rate of 1 symbol/s. However, when \(n\) increases (Figures 4.79–4.81), the DAST block codes give better performances.
Figure 4.78 Average BER as a function of SNR, two transmit, one and two receive antennas.

Figure 4.79 Average BER as a function of SNR, four transmit and one receive antennas.

Figure 4.80 Average BER as a function of SNR, four transmit and two receive antennas.
Figure 4.81  Average BER as a function of SNR, four transmit and four receive antennas.

APPENDIX 4.1 QR FACTORIZATION

Orthogonal matrix triangularization (QR decomposition) reduces a real \((m, n)\) matrix \(A\) with \(m \geq n\) and full rank to a much simpler form. A suitably chosen orthogonal matrix \(Q\) with triangularize the given matrix.

\[
A = Q \begin{bmatrix} R \\ 0 \end{bmatrix}
\]

with the \((n, n)\) upper triangular matrix \(R\). One only has then to solve the triangular system \(Rx = Pb\), where \(P\) consists of the first \(n\) rows of \(Q\). The least squares problem \(Ax \approx b\) is easy to solve with \(A = QR\) and \(Q^TQ = I\). The solution \(x = (A^T A)^{-1}A^T b\) becomes \(x = (R^T Q^T QR)^{-1} R^T Q^T b = (R^T R)^{-1} R^T Q^T b = R^{-1} Q^T b\).

This is a matrix-vector multiplication \(Q^T b\), followed by the solution of the triangular system \(Rx = Q^T b\) by back-substitution.

Many different methods exist for the QR decomposition, e.g. the Householder transformation, the Givens rotation, or the Fram–Schmidt decomposition.

Householder transformation

The most frequently applied algorithm for QR decomposition uses the Householder transformation \(u = Hv\), where the Householder matrix \(H\) is a symmetric and orthogonal matrix of the form. \(H = I - 2xx^T\), with the identity matrix \(I\) and any normalized vector \(x\) with \(||x||_2^2 = x^T x = 1\). Householder transformations zero the \(m - 1\) elements of a column vector \(v\) below the first element.

\[
\begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix} \rightarrow \begin{bmatrix} c \\ 0 \\ \vdots \\ 0 \end{bmatrix}
\]

with \(c = \pm \left( \sum_{i=1}^{m} v_i^2 \right)^{1/2}\)
One can verify that:

\[ x = f \begin{bmatrix} v_1 - c \\ v_2 \\ \vdots \\ v_m \end{bmatrix} \]

with \( f = \frac{1}{\sqrt{2c(c - v_1)}} \)

fulfils \( x^T x = 1 \) and that with \( H = I - 2xx^T \) one obtains the vector \( [c, 0 \cdots 0]^T \). To perform the decomposition of the \((m, n)\) matrix \( A = QR \) (with \( m \geq n \)) we construct in this way an \((m, m)\) matrix \( H^{(1)} \) to zero the \( m - 1 \) elements of the first column. An \((m - 1, m - 1)\) matrix \( G^{(2)} \) will zero the \( m - 2 \) elements of the second column. With \( G^{(2)} \) we produce the \((m, m)\) matrix:

\[ H^{(2)} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 1 & \cdots \end{bmatrix}, \text{ etc} \]

After \( n (n - 1) \) for \( m = n \) such orthogonal transforms \( H^{(i)} \) we obtain:

\[ R = H^{(n)} \cdots H^{(2)} H^{(1)} A. \]

\( R \) is upper triangular all the orthogonal matrix \( Q \) becomes:

\[ Q = H^{(1)} H^{(2)} \cdots H^{(n)}. \]

In practise the \( H^{(i)} \) are never explicitly computed.

### Givens transformations

Given an \( n \times m \) matrix \( A \) with \( n \geq m \) there is an \( n \times m \) matrix \( Q \) with orthonormal columns and an upper triangular \( m \times m \) matrix \( R \) such that \( A = QR \). If \( m = n \) \( Q \) is orthogonal (e.g., see [164, p. 112]).

The method based on Givens transformations is of particular interest to us (e.g., see [165, p. 214]). If \( A \) is an \( n \times n \) matrix, then \( A = QR \Rightarrow Q^T A = R. \)

The QR factorization method based on Givens transformations gives us a method for writing \( Q^T \) as the product of \( \binom{n}{2} = n(n - 1)/2 \) Givens matrices:

\[ G(i, k, \theta) = \begin{bmatrix} 1 & \cdots & c & \cdots & s \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ -s & \cdots & 1 & \cdots & \ddots \\ \vdots & \ddots & \vdots & \cdots & \ddots \\ \end{bmatrix} \]

is an \( n \times n \) matrix called the Givens (or Jacobi) matrix, with \( c = \cos \theta, s = \sin \theta \) It consists of 1’s on the main diagonal, except for the two elements \( c \) in rows (and columns) \( i \) and \( k \). All off-diagonal elements are zero, except the two elements \( s \) and \(-s\) in rows (and columns) \( i \) and \( k \). Postmultiplication of a vector by \( G \) rotates the vector counterclockwise by \( \theta \) degrees with respect to the \((i, k)\) plane.

**Proposition 1:** If an \( n \times n \) matrix \( A \) is orthogonal, then a QR factorization algorithm yields \( Q = A \hat{I} \) where \( \hat{I} \) denotes a matrix in which each main diagonal element is either 1 or and all off-diagonal elements are zero.
Proof: $Q^T$ is orthogonal because $Q$ is orthogonal. If $A$ is orthogonal, then $Q^T A = R$ is also orthogonal since the product of two orthogonal matrices is an orthogonal matrix. Since $R$ is an upper triangular orthogonal matrix, each main diagonal element must be either 1 or $-1$ and all off-diagonal elements must be zero. We have $A = Q \hat{I} = A \hat{I}^{-1} = A \hat{I}$.

Proposition 2: Any orthogonal $n \times n$ matrix $A$ can be written as the product of $n(n-1)/2$ $n \times n$ Givens matrices and an $n \times n \hat{I}$-matrix.

Proof: The proof follows directly from the proof of the previous proposition. Given an arbitrary orthogonal $n \times n$ matrix $A$ we have a method for constructing $Q$, the product of $n(n-1)/2$ Givens matrices, such that $A = Q \hat{I}$. Note that the identity matrix is a Givens matrix as well as an $\hat{I}$(matrix).

For example, $2 \times 2$ all orthogonal matrices have the form:

$$A = \begin{bmatrix}
\cos \theta & \sin \theta \\
-\sin \theta & \cos \theta
\end{bmatrix}$$

or

$$A = \begin{bmatrix}
\cos \theta & \sin \theta \\
\sin \theta & -\cos \theta
\end{bmatrix}$$

This is a well-known fact (for example, see [166, p. 282]). The first form corresponds to $G(1, 2, \theta) \hat{I}_{2 \times 2}$ and the second form corresponds to $G(1, 2, \theta)(-\hat{I}_{2 \times 2})$. Proposition 2, together with the QR factorization method using Givens transformations, implies that all $3 \times 3$ orthogonal matrices have the form:

$$A = G(1, 2, \theta_{12})G(1, 3, \theta_{13})G(2, 3, \theta_{23})\hat{I}_{3 \times 3} =$$

$$= \begin{bmatrix}
\cos \theta_{12} & \sin \theta_{12} & 0 \\
-\sin \theta_{12} & \cos \theta_{12} & 0 \\
0 & 0 & 1
\end{bmatrix} \cdot \begin{bmatrix}
\cos \theta_{13} & 0 & \sin \theta_{13} \\
0 & 1 & 0 \\
-\sin \theta_{13} & 0 & \cos \theta_{13}
\end{bmatrix} \cdot \begin{bmatrix}
1 & 0 & 0 \\
0 & \cos \theta_{23} & \sin \theta_{23} \\
0 & -\sin \theta_{23} & \cos \theta_{23}
\end{bmatrix} \cdot \begin{bmatrix}
\pm 1 & 0 & 0 \\
0 & \pm 1 & 0 \\
0 & 0 & \pm 1
\end{bmatrix}$$

In general, any $L \times L$ orthogonal matrix $A$ can be factored into $L(L-1)/2 = (\binom{L}{2})$ rotation matrices and an $\hat{I}$ matrix as follows:

$$A = \left( \prod_{1 \leq i < j \leq L} G(i, j, \theta_{i,j}) \right) \cdot \hat{I}_{L \times L} \quad \text{(A4.1.1)}$$

for suitable $\theta_{i,j}$.

**APPENDIX 4.2 LATTICE CODE DECODER FOR SPACE–TIME CODES**

Consider the system of $M$ transmit and $N$ receive antennas, the single data stream in the input is demultiplexed into $M$ substreams, and each substream is modulated independently then transmitted by its dedicated antenna. It is assumed that the same constellation is used for all the substreams. The transmission is done by burst of length $l$ over a quasi-static Rayleigh fading channel changing randomly every $l$ symbol durations. The power launched by each transmitter is proportional to $1/M$ so that the total radiated power is constant and independent of $M$. The proximity of antennas presupposes the synchronization of the system.

Notation $\mathbb{Z}$ is the ring of integers. $\mathbb{R}$ is the field of real numbers. $\mathbb{Z}_{a+j}$ is the set of numbers $a + ib$ with $a, b \in \mathbb{Z}$, and $i = \sqrt{-1}$.
The received signal at each instant time is given by

\[ \mathbf{r} = \mathbf{H}\mathbf{u}^T + \mathbf{v} \quad \text{(A4.2.1)} \]

where \( \mathbf{a} = (a_1, a_2, \ldots, a_M) \) is the transmitted vector which belongs to the constellation QAM carved from \( \mathbb{Z}_M \), and \( \mathbf{v} \) is a \( N \times 1 \) complex vector AWGN component-wise independent with a variance \( \sigma^2 \) per dimension. Moreover, \( \mathbf{H} \) is an \( N \times M \) transfer matrix of the channel with entries \( h_{kj} \), where \( h_{kj} \) is the fading between transmitter \( j \) and receiver \( k \). In the sequel we set \( M = N \). One can write the system (A4.2.1) as

\[ \mathbf{r}' \triangleq \left[ \Re(\mathbf{r}^T) \Im(\mathbf{r}^T) \right] = \mathbf{u} \left[ \begin{bmatrix} \Re(\mathbf{H}^T) & \Im(\mathbf{H}^T) \end{bmatrix} \right] + \mathbf{v}' = \mathbf{u}\mathbf{M_H} + \mathbf{v}' \quad \text{(A4.2.2)} \]

where \( \mathbf{u} = [\Re(\mathbf{a}) \Im(\mathbf{a})] \in \mathbb{C}^{2M} \), and \( \mathbf{v}' = [\Re(\mathbf{v}^T) \Im(\mathbf{v}^T)] \in \mathbb{C}^{2M} \).

\( \Re(\mathbf{a}), \Im(\mathbf{a}) \) denote the real and imaginary part of \( \mathbf{a} \), respectively. Note that the rank of \( \mathbf{M_H} \) is \( 2M \) almost always, and its Gram matrix \( \mathbf{G}_{2M} = \mathbf{M_H}\mathbf{M}_H^T \) is positive definite. Hence, we can represent the multi-antenna environment by a lattice sphere packing [173, 174], and one can apply the universal lattice decoder in a multi-antenna system. The principle of the algorithm is to search the closest lattice point to the received signal within a sphere of radius \( \sqrt{C} \) cantered at the received signal (see [174] and references therein). The choice of \( C \) is very crucial to the speed of the algorithm. In practice, \( C \) can be adjusted according to the noise (and eventually the fading) variance. When a failure is detected, one can either declare an erasure on the detected symbol, or increase \( C \). The complexity of the algorithm is independent of the lattice constellation size, which is very useful for high data rate transmission. In [175], Fincke and Phost showed that if \( d^{-1} \) is a lower bound for the eigenvalues of the Gram matrix \( \mathbf{G}_{2M} \), then the number of arithmetical operations is

\[ O \left( n^2 \times \left( 1 + \frac{\eta - 1}{4dC} \right)^{4dC} \right) \quad \text{(A4.2.3)} \]

which, for a judicious choice of the radius \( C = d^{-1} \), is approximated by \( O(n^6) \) arithmetical operations, where \( n \) is the lattice dimension.

**APPENDIX 4.3 MIMO CHANNEL CAPACITY**

The transmitted signals in each symbol period are represented by an \( n_T \times 1 \) column matrix \( \mathbf{x} \), where the \( i \)th component \( x_i \) refers to the transmitted signal from antenna \( i \). We consider a Gaussian channel, for which, according to information theory, the optimum distribution of transmitted signals is also Gaussian. Thus, the elements of \( \mathbf{x} \) are considered to be zero mean independent identically distributed (i.i.d.) Gaussian variables. The covariance matrix of the transmitted signal is given by \( \mathbf{R}_{xx} = E\{\mathbf{x}\mathbf{x}^H\} \) where \( E\{\cdot\} \) stands for expectation and \( \mathbf{A}^H \) denotes the Hermitian of matrix \( \mathbf{A} \) (transpose and component-wise complex conjugate). The total transmitted power is constrained to \( P \), regardless of the number of transmit antennas \( n_T \). It can be represented as \( P = \text{tr}(\mathbf{R}_{xx}) \), where \( \text{tr}(\mathbf{A}) \) denotes the trace of matrix \( \mathbf{A} \) (sum of the diagonal elements of \( \mathbf{A} \)). If the channel is unknown at the transmitter, we assume that the signals transmitted from individual antenna elements have equal powers of \( P/n_T \). The covariance matrix of the transmitted signal is given by

\[ \mathbf{R}_{xx} = \frac{P}{n_T} \mathbf{I}_{n_T} \quad \text{(A4.3.1)} \]

where \( \mathbf{I}_{n_T} \) is the \( n_T \times n_T \) identity matrix. The transmitted signal bandwidth is narrow enough, so its frequency response can be considered as flat. In other words, we assume that the channel is memoryless. The channel is described by an \( n_R \times n_T \) complex matrix, denoted by \( \mathbf{H} \). The \( ij \)th component of the matrix \( \mathbf{H} \), denoted by \( h_{ij} \), represents the channel fading coefficient from the \( j \)th transmit to the \( i \)th receive antenna. For normalization purposes we assume that the received power
for each of \( n_R \) receive branches is equal to the total transmitted power. Physically, it means that we ignore signal attenuations and amplifications in the propagation process, including shadowing, antenna gains, etc. Thus we obtain the normalization constraint for the elements of \( \mathbf{H} \), on a channel with fixed coefficients, as:

\[
\sum_{j=1}^{n_T} |h_{ij}|^2 = n_T, \quad i = 1, 2, \ldots, n_R
\]  

(A4.3.2)

The noise at the receiver is described by an \( n_R \times 1 \) column matrix, denoted by \( \mathbf{n} \). Its components are statistically independent complex zero-mean Gaussian variables, with independent and equal variance real and imaginary parts. The covariance matrix of the receiver noise is given by \( \mathbf{R}_{nn} = E\{\mathbf{n}\mathbf{n}^H\} \). If there is no correlation between components of \( \mathbf{n} \), the covariance matrix is obtained as \( \mathbf{R}_{nn} = \sigma^2 \mathbf{I}_{n_R} \). Each of \( n_R \) receive branches has identical noise power of \( \sigma^2 \).

The receiver is based on a maximum likelihood principle operating jointly over \( n_R \) receive antennas. The received signals are represented by an \( n_R \times 1 \) column matrix, denoted by \( \mathbf{r} \), where each complex component refers to a receive antenna. We denote the average power at the output of each receive antenna by \( P_r \). The average signal-to-noise ratio (SNR) at each receive antenna is defined as:

\[
\gamma = \frac{P_r}{\sigma^2} \quad \text{(A4.3.3)}
\]

As it was assumed that the total received power per antenna is equal to the total transmitted power, the SNR is equal to the ratio of the total transmitted power and the noise power per receive antenna and it is independent of \( n_T \) and can be written as:

\[
\gamma = \frac{P}{\sigma^2} \quad \text{(A4.3.4)}
\]

By using the linear model the received vector can be represented as \( \mathbf{r} = \mathbf{Hx} + \mathbf{n} \). The received signal covariance matrix, \( E\{\mathbf{rr}^H\} \), is given by \( \mathbf{R}_{rr} = \mathbf{HR}_x \mathbf{H}^H \) while the total received signal power can be expressed as \( \text{tr} (\mathbf{R}_{rr}) \).

**MIMO system capacity:** The system capacity is defined as the maximum possible transmission rate such that the probability of error is arbitrarily small. Initially, we assume that the channel matrix is not known at the transmitter, while it is perfectly known at the receiver.

By the singular value decomposition (SVD) theorem [Appendix 5.1] any \( n_R \times n_T \) matrix \( \mathbf{H} \) can be written as \( \mathbf{H} = \mathbf{UDV}^H \), where \( \mathbf{D} \) is an \( n_R \times n_T \) non-negative and diagonal matrix, \( \mathbf{U} \) and \( \mathbf{V} \) are \( n_R \times n_R \) and \( n_T \times n_T \) unitary matrices, respectively. That is, \( \mathbf{UU}^H = \mathbf{I}_{n_R} \) and \( \mathbf{VV}^H = \mathbf{I}_{n_T} \), where \( \mathbf{I}_{n_R} \) and \( \mathbf{I}_{n_T} \) are \( n_R \times n_R \) and \( n_T \times n_T \) identity matrices, respectively. The diagonal entries of \( \mathbf{D} \) are the non-negative square roots of the eigenvalues of matrix \( \mathbf{HH}^H \). The eigenvalues of \( \mathbf{HH}^H \), denoted by \( \lambda \), are defined here as:

\[
\mathbf{HH}^H \mathbf{y} = \lambda \mathbf{y}, \quad \mathbf{y} \neq 0
\]  

(A4.3.5)

where \( \mathbf{y} \) is an \( n_R \times 1 \) vector associated with \( \lambda \), called an eigenvector.

The nonnegative square roots of the eigenvalues are also referred to as the singular values of \( \mathbf{H} \). Furthermore, the columns of \( \mathbf{U} \) are the eigenvectors of \( \mathbf{HH}^H \) and the columns of \( \mathbf{V} \) are the eigenvectors of \( \mathbf{H}^H \mathbf{H} \). We can write for the received vector \( \mathbf{r} \):

\[
\mathbf{r} = \mathbf{UDV}^H \mathbf{x} + \mathbf{n}
\]  

(A4.3.6)

In the sequel we will need the following transformations:

\[
\mathbf{r}' = \mathbf{U}^H \mathbf{r}
\]

\[
\mathbf{x}' = \mathbf{V}^H \mathbf{x}
\]  

(A4.3.7)

\[
\mathbf{n}' = \mathbf{U}^H \mathbf{n}
\]
as U and V are invertible. Clearly, multiplication of vectors $r$, $x$ and $n$ by the corresponding matrices as defined in (A4.3.6) has only a scaling effect. Vector $n'$ is a zero mean Gaussian random variable with i.i.d real and imaginary parts. Thus, the original channel is equivalent to the channel represented as:

$$r' = D x' + n'$$

(A4.3.8)

The number of nonzero eigenvalues of matrix $HH^H$ is equal to the rank of matrix $H$, denoted by $r$. For the $n_R \times n_T$ matrix $H$, the rank is at most $m = \min(n_R, n_T)$, which means that at most $m$ of its singular values are nonzero. Let us denote the singular values of $H$ by $\lambda_i, i = 1, 2, \ldots, r$. By substituting the entries $\sqrt{\lambda_i}$ in (A4.3.7), we get for the received signal components:

$$r'_i = \sqrt{\lambda_i} x'_i + n'_i, \quad i = 1, 2, \ldots, r$$

$$r'_i = n'_i, \quad i = r + 1, r + 2, \ldots, n_R$$

(A4.3.9)

As (A4.3.8) indicates, received components, $r'_i$, $i = r + 1, r + 2, \ldots, n_R$, do not depend on the transmitted signal, i.e. the channel gain is zero. On the other hand, received components $r'_i$ for $i = 1, 2, \ldots, r$ depend only on the transmitted component $x'_i$. Thus the equivalent MIMO channel from (A4.3.7) can be considered as consisting of $r$ uncoupled parallel subchannels. Each sub-channel is assigned to a singular value of matrix $H$, which corresponds to the amplitude channel gain. The channel power gain is thus equal to the eigenvalue of matrix $HH^H$. For example, if $n_T > n_R$, as the rank of $H$ cannot be higher than $n_R$.

Equation (A4.3.8) shows that there will be at most $n_R$ nonzero gain subchannels in the equivalent MIMO channel, as indicated in Section 4.14. On the other hand if $n_R > n_T$, there will be at most $n_T$ nonzero gain subchannels in the equivalent MIMO channel. The eigenvalue spectrum is a MIMO channel representation, which is suitable for evaluation of the best transmission paths.

The covariance matrices and their traces for signals $r'$, $x'$ and $n'$ can be derived from (A4.3.6) as:

$$R_{r'r'} = U^H R_{rr} U \quad \text{tr}(R_{r'r'}) = \text{tr}(R_{rr})$$

$$R_{x'x'} = V^H R_{xx} V \quad \text{tr}(R_{x'x'}) = \text{tr}(R_{xx})$$

$$R_{n'n'} = U^H R_{nn} U \quad \text{tr}(R_{n'n'}) = \text{tr}(R_{nn})$$

The above relationships show that the covariance matrices of $r'$, $x'$ and $n'$, have the same sum of the diagonal elements, and thus the same powers, as for the original signals, $r$, $x$ and $n$, respectively.

Note that in the equivalent MIMO channel model described by (A4.3.8), the subchannels are uncoupled and thus their capacities add up. Assuming that the transmit power from each antenna in the equivalent MIMO channel model is $P/n_T$, we can estimate the overall channel capacity, denoted by $C$, by using the Shannon capacity formula as in (4.203):

$$C = W \sum_{i=1}^r \log_2 \left(1 + \frac{P_{ri}}{\sigma^2} \right)$$

(A4.3.10)

where $W$ is the bandwidth of each sub-channel and $P_{ri}$ is the received signal power in the $i$th subchannel given by

$$P_{ri} = \frac{\lambda_i P}{n_T}$$

(A4.3.11)

In (A4.3.11) $\sqrt{\lambda_i}$ is the singular value of channel matrix $H$. Thus the channel capacity can be written as

$$C = W \sum_{i=1}^r \log_2 \left(1 + \frac{\lambda_i P}{n_T \sigma^2} \right) = W \log_2 \left( \prod_{i=1}^r \left(1 + \frac{\lambda_i P}{n_T \sigma^2} \right) \right)$$

(A4.3.12)

Now we will show how the channel capacity is related to the channel matrix $H$. Assuming that $m = \min(n_R, n_T)$, Equation (A4.3.5), defining the eigenvalue-eigenvector relationship, can be rewritten as:

$$(\lambda I_m - Q)y = 0, \quad y \neq 0$$

(A4.3.13)
where $Q$ is the Wishart matrix defined as

$$Q = \begin{cases} HH^H, & n_R < n_T \\ H^H H, & n_R < n_T \end{cases} \quad (A4.3.14)$$

That is, $\lambda$ is an eigenvalue of $Q$, if and only if $\lambda I_m - Q$ is a singular matrix. Thus the determinant of $\lambda I_m - Q$ must be zero

$$\det (\lambda I_m - Q) = 0 \quad (A4.3.15)$$

The singular values $\lambda$ of the channel matrix can be calculated by finding the roots of Equation (A4.3.15). We consider the characteristic polynomial $p(\lambda)$ from the left-hand side in Equation (A4.3.15):

$$p(\lambda) = \det (\lambda I_m - Q) \quad (A4.3.16)$$

It has degree equal to $m$, as each row of $\lambda I_m - Q$ contributes one and only one power of $\lambda$ in the Laplace expansion of $\det (\lambda I_m - Q)$ by minors. As a polynomial of degree $m$ with complex coefficients has exactly $m$ zeros, by counting multiplicities, we can write for the characteristic polynomial:

$$p(\lambda) = \prod_{i=1}^{m} (\lambda - \lambda_i) \quad (A4.3.17)$$

where $\lambda_i$ are the roots of the characteristic polynomial $p(\lambda)$, equal to the channel matrix singular values and we can write Equation (A4.3.15) as

$$\prod_{i=1}^{m} (\lambda - \lambda_i) = 0 \quad (A4.3.18)$$

Further we can equate the left-hand sides of (A4.3.15) and (A4.3.18):

$$\prod_{i=1}^{m} (\lambda - \lambda_i) = \det (\lambda I_m - Q) \quad (A4.3.19)$$

Substituting $-\frac{n_T \sigma^2}{P}$ for $\lambda$ in (1.28) we get:

$$\prod_{i=1}^{m} \left(1 + \frac{\lambda_i P}{n_T \sigma^2}\right) = \det \left(I_m + \frac{P}{n_T \sigma^2} Q\right) \quad (A4.3.20)$$

and Equation (A4.3.12) can be written as:

$$C = W \log_2 \det \left(I_m + \frac{P}{n_T \sigma^2} Q\right) \quad (A4.3.21)$$

As the nonzero eigenvalues of $HH^H$ and $H^H H$ are the same, the capacities of the channels with matrices $H$ and $H^H H$ are the same. Note that if the channel coefficients are random variables, formulas (A4.3.12) and (A4.3.21), represent instantaneous capacities or mutual information. The mean channel capacity can be obtained by averaging over all realizations of the channel coefficients as it was used in Chapter 14 (see Equation (14.54)).

**Water-filling principle**: Let us consider a MIMO channel where the channel parameters are known at the transmitter. The allocation of power to various transmitter antennas can be obtained by a ‘water-filling’ principle used in Equation (4.201). The ‘water-filling principle’ can be derived by maximizing the MIMO channel capacity under the power constraint:

$$\sum_{i=1}^{n_T} P_i = P \quad i = 1, 2, \ldots, n_T \quad (A4.3.22)$$

where $P_i$ is the power allocated to antenna $i$ and $P$ is the total power, which is kept constant. The normalized capacity of the MIMO channel is determined as:

$$C/W = \sum_{i=1}^{n_T} \log_2 \left[1 + \frac{P_i \lambda_i}{\sigma^2}\right] \quad (A4.3.23)$$
Following the method of Lagrange multipliers, we introduce the function:

\[ Z = \sum_{i=1}^{n_T} \log_2 \left( 1 + \frac{P_i \lambda_i}{\sigma^2} \right) + L \left( P - \sum_{i=1}^{n_T} P_i \right) \]  

where \( L \) is the Lagrange multiplier, \( \lambda_i \) is the \( i \)th channel matrix singular value and \( \sigma^2 \) is the noise variance. The unknown transmit powers \( P_i \) are determined by setting the partial derivatives of \( Z \) to zero:

\[ \frac{\delta Z}{\delta P_i} = 0 \]

\[ \frac{\delta Z}{\delta P_i} = \frac{1}{\ln 2} \frac{\lambda_i / \sigma^2}{1 + P_i \lambda_i / \sigma^2} - L = 0 \]  

(A4.3.25)

which gives for \( P_i \)

\[ P_i = \mu - \frac{\sigma^2}{\lambda_i} \]  

(A4.3.26)

where \( \mu \) is a constant, given by \( 1/L \ln 2 \). It can be determined from the power constraint of Equation (A4.3.22).

**MIMO channel capacity for adaptive transmit power allocation**: When the channel parameters are known at the transmitter, the capacity given by Equation (A4.3.21) can be increased by assigning the transmitted power to various antennas according to the ‘water-filling’ rule. It allocates more power when the channel is in good condition and less when the channel state gets worse. The power allocated to channel \( i \) is given by:

\[ P_i = \left( \mu - \frac{\sigma^2}{\lambda_i} \right)^+, \quad i = 1, 2, \ldots, r \]  

(A4.3.27)

where \( a^+ \) denotes \( \max(a, 0) \) and \( \mu \) is determined so that \( \sum_{i=1}^{r} P_i = P \).

We consider the singular value decomposition of channel matrix \( H \), then, the received power at subchannel \( i \) in the equivalent MIMO channel model is given by:

\[ P_{ri} = \left( \lambda_i \mu - \sigma^2 \right)^+ \]  

(A4.3.28)

The MIMO channel capacity is then:

\[ C = W \sum_{i=1}^{r} \log_2 \left( 1 + \frac{P_{ri}}{\sigma^2} \right) \]  

(A4.3.29)

Substituting the received signal power from Equation (A4.3.28) into Equation (A4.3.29) we get:

\[ C = W \sum_{i=1}^{r} \log_2 \left[ 1 + \frac{1}{\sigma^2} \left( \lambda_i \mu - \sigma^2 \right)^+ \right] \]  

(A4.3.30)

The covariance matrix of the transmitted signal is given by:

\[ R_{xx} = V \text{diag} \left( P_1, P_2, \ldots, P_{n_T} \right) V^H \]  

(A4.3.31)

**REFERENCES**


REFERENCES


REFERENCES


REFERENCES


Multiuser Communication

The basic principles of CDMA were discussed in Chapter 1. In this chapter, after brief discussion of code generation, we focus on multiuser detection. More details can be found in the recent book on WCDMA [1].

5.1 PSEUDORANDOM SEQUENCES

5.1.1 Binary shift register sequences

Let us define a polynomial

\[ h(x) = h_0 x^n + h_1 x^{n-1} + \cdots + h_{n-1} x + h_n \] (5.1)

in the discrete field with two elements \( h_i \in (0, 1) \) and \( h_0 = h_n = 1 \). An example polynomial could be \( x^4 + x + 1 \) or \( x^5 + x^2 + 1 \). The coefficients \( h_i \) of the polynomial can be represented by binary vectors 10011 and 100101, or in octal notation 23 and 45 (every group of three bits is represented by a number between 0 and 7). A binary sequence \( u \) is said to be a sequence generated by \( h(x) \) if, for all integers \( j \),

\[ h_0 u_j \oplus h_1 u_{j-1} \oplus h_2 u_{j-2} \oplus \cdots \oplus h_n u_{j-n} = 0 \] (5.2)

where \( \oplus = \) addition modulo 2.

If we formally change the variables

\[ j \rightarrow j + n, \quad \text{and} \quad h_0 = 1 \] (5.3)

then Equation (5.2) becomes:

\[ u_{j+n} = h_n u_j \oplus h_{n-1} u_{j+1} \oplus \cdots \oplus h_1 u_{j+n-1} \] (5.4)

In this notation, \( u_j \) is the \( j \)th bit (called chip) of the sequence \( u \). Equation (5.4) suggests that the sequence \( u \) can be generated by an \( n \)-stage binary linear feedback shift register which has a feedback tap connected to the \( i \)th cell if \( h_i = 1, 0 < i \leq n \). As an example, for \( n = 5 \), Equation (5.4) becomes:

\[ u_{j+5} = h_5 u_j \oplus h_4 u_{j+1} \oplus h_3 u_{j+2} \oplus h_2 u_{j+3} \oplus h_1 u_{j+4} \] (5.5)
For $x^5 + x^2 + 1$, octal representation (45), the coefficients $h_i$ are:

$$h_0 \ h_1 \ h_2 \ h_3 \ h_4 \ h_5 = 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ \text{(octal representation 45)}$$

and the block diagram of the circuit is shown in Figure 5.1.

Similarly, for the polynomial $x^5 + x^4 + x^3 + x^2 + 1$, the coefficients $h_i$ are given as:

$$h_0 \ h_1 \ h_2 \ h_3 \ h_4 \ h_5 = 1 \ 1 \ 1 \ 1 \ 0 \ 1 \ \text{(octal representation 75)}$$

and by using Equation (5.4) one can get the generator shown in Figure 5.2.

Some of the properties of these sequences and definitions are listed below. Details can be found in standard literature listed at the end of the book, especially [2–22].

If $u$ and $v$ are generated by $h(x)$, then so is $u \oplus v$, where $u \oplus v$ denotes the sequence whose $i$th element is $u_i \oplus v_i$. The all zero state of the shift register is not allowed because, for this initial state, Equation (5.5) would continue to generate zero chips. For this reason, the period of $u$ is at most $2^n - 1$, where $n$ is the number of cells in the shift register, or equivalently, the degree of $h(x)$. If $u$ denotes an arbitrary \{0, 1\} valued sequence, then $x(u)$ denotes the corresponding \{+1, −1\} valued sequence, where the $i$th element of $x(u)$ is just $x(u_i)$:

$$x(u_i) = (-1)^{u_i} \quad (5.6)$$

If $T^i$ is a delay operator (delay for $i$ chip periods) then we have:

$$T^i(x(u)) = x(T^iu) \quad \text{and} \quad \sum x(u) = x(u_0) + x(u_1) + \cdots + x(u_{N-1})$$

$$= N^+ - N^- = (N - N^-) - N^- \quad (5.7)$$

$$= N - 2N^- = N - 2wt(u)$$

where $wt(u)$ denotes the Hamming weight of unipolar sequence $u$, that is, the number of ones in $u$, $N$ is the sequence period and $N^+$ and $N^-$ are the number of positive and negative chips in bipolar
sequence \( x(u) \). The crosscorrelation function between two bipolar sequences can be represented as:

\[
\theta_{u,v}(l) \equiv \theta_{x(u),x(v)}(l) = \sum_{i=0}^{N-1} x(u_i)x(v_{i+l})
\]

\[
= \sum_{i=0}^{N-1} (-1)^{u_i}(-1)^{v_{i+l}} = \sum_{i=0}^{N-1} (-1)^{u_i \oplus v_{i+l}} = \sum_{i=0}^{N-1} x(u_i \oplus v_{i+l})
\]

(5.8)

By using Equation (5.7) we have:

\[
\theta_{u,v}(l) = N - 2wt(u \oplus T^iv)
\]

(5.9)

The periodic autocorrelation function \( \theta_u(\cdot) \) is just \( \theta_{u,u}(\cdot) \)

\[
\theta_u(l) = N - 2wt(u \oplus T^lu)
\]

\[
= N^+ - N^- \equiv (N - N^-) - N^- = N - 2N^-
\]

(5.10)

### 5.1.2 Properties of binary maximal length sequences

As was mentioned earlier the all zero state of the shift register is not allowed because, based on Equation (5.4), the generator could not get out of this state. Bear in mind that the number of possible states of the shift register is \( 2^n \). So, the period of a sequence \( u \) generated by the polynomial \( h(x) \) cannot exceed \( 2^n - 1 \), where \( n \) is the degree of \( h(x) \). If \( u \) has this maximal period, \( N = 2^n - 1 \), it is called a maximal length sequence or \( m \)-sequence. To get such a sequence, \( h(x) \) should be a primitive binary polynomial of degree \( n \). There are exactly \( N \) non-zero sequences generated by \( h(x) \), and they are just the \( N \) different phases of \( u, Tu, T^2u, \ldots, T^{N-1}u \). Given distinct integers \( i \) and \( j \), \( 0 \leq i, j < N \), there is a unique integer, \( k \), distinct from both \( i \) and \( j \), such that \( 0 \leq k < N \) and

\[
T^iu \oplus T^ju = T^ku
\]

(5.11)

From the above discussion on the number of ones and zeros, \( wt(u) = 2^{n-1} = 1/2(N + 1) \), so that from Equation (5.9) we have:

\[
\theta_u(l) = \begin{cases} 
N, & \text{if } l \equiv 0 \mod N \\
-1, & \text{if } l \neq 0 \mod N
\end{cases}
\]

(5.12)

From here on, \( \tilde{u} \) will be called a characteristic \( m \)-sequence, or the characteristic phase of the \( m \)-sequence \( u \) if \( \tilde{u}_i = \tilde{u}_{2i} \) for all \( i \in \mathbb{Z} \).

Let \( q \) denote a positive integer, and consider the sequence \( v \) formed by taking every \( q \)th bit of \( u \) (i.e. \( v_i = u_{qi} \) for all \( i \in \mathbb{Z} \)). The sequence \( v \) is said to be a decimation by \( q \) of \( u \), and will be denoted by \( u[q] \).

Assume that \( u[q] \) is not identically zero. Then, \( u[q] \) has period \( N/\gcd(N, q) \), and is generated by the polynomial whose roots are the \( q \)th powers of the roots of \( h(x) \), where \( \gcd(N, q) \) is the greatest common divisor of the integers \( N \) and \( q \). The tables of primitive polynomials are available in any book on coding theory. The reciprocal \( m \)-sequence \( v \) is generated by the reciprocal polynomial of \( h(x) \), that is,

\[
\hat{h}(x) = x^n h(x^{-1}) = h_n x^n + h_{n-1} x^{n-1} + \cdots + h_0
\]

(5.13)

### 5.1.3 Crosscorrelation spectra

Frequently, we do not need to know more than the set of crosscorrelation values together with the number of integers \( l(0 \leq l < N) \) for which \( \theta_{u,v}(l) = c \) for each \( c \) in this set. Let \( u \) and \( v \) denote \( m \)-sequences of period \( 2^n - 1 \). If \( v = u[q] \), where either \( q = 2^k + 1 \) or \( q = 2^{2k} - 2^k + 1 \), and if
\[ e = \gcd(n, k) \] is such that \( n/e \) is odd, then the spectrum of \( \theta_{u,v} \) is three-valued [23–26] as:

\[
\begin{align*}
-1 + 2^{(n+1)/2} & \quad \text{occurs } 2^{n-e-1} + 2^{(n-e-2)/2} \text{ times} \\
-1 & \quad \text{occurs } 2^n - 2^{n-e} - 1 \text{ times} \\
-1 - 2^{(n+1)/2} & \quad \text{occurs } 2^{n-e-1} - 2^{(n-e-2)/2} \text{ times}
\end{align*}
\]

(5.14)

The same spectrum is obtained if instead of \( v = u[q] \), we let \( u = v[q] \). Notice that if \( e \) is large, \( \theta_{u,v}(l) \) takes on large values but only very few times, while if \( e \) is small, \( \theta_{u,v}(l) \) takes on smaller values more frequently. In most instances, small values of \( e \) are desirable. If we wish to have \( e = 1 \), then clearly \( n \) must be odd in order that \( n/e \) be odd. When \( n \) is odd, we can take \( k = 1 \) or \( k = 2 \) (and possibly other values of \( k \) as well), and obtain \( \theta(u, u[3]), \theta(u, u[5]) \) and \( \theta(u, u[13]) \) all having the three-valued spectrum given by Expressions (5.14) (with \( e = 1 \)). Suppose next that \( n \equiv 2 \text{ mod } 4 \). Then, \( n/e \) is odd if \( e \) is even and a divisor of \( n \). Letting \( k = 2 \), we obtain that \( \theta(u, u[5]) \) and \( \theta(u, u[13]) \) both have the three-valued spectrum given by Expressions (5.14) (with \( e = 2 \)).

Let us define \( t(n) \) as:

\[ t(n) = 1 + 2[(n + 2)/2] \]

(5.15)

where \( [\alpha] \) denotes the integer part of the real number \( \alpha \). Then if \( n \not\equiv 0 \text{ mod } 4 \), there exist pairs of \( m \)-sequences with three-valued crosscorrelation functions, where the three values are \(-1, -t(n) \) and \( t(n) - 2 \). A crosscorrelation function taking on these values is called a preferred three-valued crosscorrelation function and the corresponding pair of \( m \)-sequences (polynomials) is called a preferred pair of \( m \)-sequences (polynomials).

Let \( u \) and \( v \) denote \( m \)-sequences of period \( 2^n-1 \), where \( n \) is a multiple of 4. If \( v = u[-1 + 2^{(n+2)/2}] = u[t(n) - 2] \), then \( \theta_{u,v} \) has a four-valued spectrum represented as:

\[
\begin{align*}
-1 + 2^{(n+2)/2} & \quad \text{occurs } (2^{n-1} - 2^{(n-2)/2})/3 \text{ times} \\
-1 + 2^{n/2} & \quad \text{occurs } 2^{n/2} \text{ times} \\
-1 & \quad \text{occurs } 2^{n-1} - 2^{(n-2)/2} - 1 \text{ times} \\
-1 - 2^{n/2} & \quad \text{occurs } (2^n - 2^{n/2})/3 \text{ times}
\end{align*}
\]

(5.16)

### 5.1.4 Maximal connected sets of \( m \)-sequences

The preferred pair of \( m \)-sequences is a pair of \( m \)-sequences of period \( N = 2^n - 1 \), which has the preferred three-valued crosscorrelation function. The values taken on by the preferred three-valued crosscorrelation functions are \(-1, -t(n) \) and \( t(n) - 2 \), where \( t(n) \) is given by Equation (5.15). The pair of primitive polynomials that generate a preferred pair of \( m \)-sequences is called a preferred pair of polynomials. A connected set of \( m \)-sequences is a collection of \( m \)-sequences which has the property that each pair in the collection is a preferred pair. A largest possible connected set is called a maximal connected set, and the size of such a set is denoted by \( M_n \). Some examples are given in Table 5.1.

### 5.1.5 Gold sequences

A set of Gold sequences of period \( N = 2^n - 1 \), consists of \( N + 2 \) sequences for which \( \theta_e = \theta_o = t(n) \). A set of Gold sequences can be constructed from appropriately selected \( m \)-sequences as described below. Suppose \( f(x) = h(x)\hat{h}(x) \), where \( h(x) \) and \( \hat{h}(x) \) have no factors in common. The set of all sequences generated by \( f(x) \) is of the form \( a \oplus b \), where \( a \) is some sequence generated by \( h(x) \), \( b \) is some sequence generated by \( \hat{h}(x) \) and we do not make the usual restriction that \( a \) and \( b \) are non-zero sequences. We represent such a set by:

\[ G(u, v) \triangleq \{u, v, u \oplus v, u \oplus T v, u \oplus T^2 v, \ldots, u \oplus T^{N-1} v\} \]

(5.17)

\( G(u, v) \) contains \( N + 2 = 2^n + 1 \) sequences of period \( N \). Let \( \{u, v\} \) denote a preferred pair of \( m \)-sequences of period \( N = 2^n - 1 \) generated by the primitive binary polynomials \( h(x) \) and \( \hat{h}(x) \).
respectively. Then set \(G(u, v)\) is called a set of Gold sequences. For \(y, z \in G(u, v)\), \(\theta_{y,z}(l) \in \{-1, -t(n), t(n) - 2\}\) for all integers \(l\) and \(\theta_{y}(l) \in \{-1, -t(n), t(n) - 2\}\) for all \(l \neq 0 \mod N\). Every sequence in \(G(u, v)\) can be generated by the polynomial \(f(x) = h(x)\hat{h}(x)\). Note that the non-maximal length sequences belonging to \(G(u, v)\) also can be generated by adding together (term by term, modulo 2) the outputs of the shift registers corresponding to \(h(x)\) and \(\hat{h}(x)\). The maximal length sequences belonging to \(G(u, v)\) are, of course, the outputs of the individual shift registers. Let us compare the parameter \(\theta_{\text{max}} = \max \{\theta_{u, \theta_{c}}, \theta_{v, \theta_{c}}\}\) for a set of Gold sequences to a bound due to Sidelnikov, which states that for any set of \(N\) or more binary sequences of period \(N\),

\[
\theta_{\text{max}} > (2N - 2)^{1/2}
\]  

For Gold sequences, they form an optimal set with respect to the bounds when \(n\) is odd. When \(n\) is even, Gold sequences are not optimal.

### 5.1.6 Gold-like and dual-BCH sequences

Let \(n\) be even and let \(q\) be an integer such that \(\gcd(q, 2^n - 1) = 3\). Let \(u\) denote an \(m\)-sequence of period \(N = 2^n - 1\) generated by \(h(x)\), and let \(v^{(k)}\), \(k = 0, 1, 2\), denote the result of decimating \(T^ku\) by \(q\). The \(v^{(k)}\) are sequences of period \(N' = N/3\) which are generated by the polynomial \(\hat{h}(x)\) whose roots are \(q\)th powers of the roots of \(h(x)\). Gold-like sequences are defined as:

\[
H_{q}(u) = \{u, u \oplus v^{(0)}, u \oplus Tv^{(0)}, \ldots, u \oplus T^{N'-1}v^{(0)},
\]

\[
\quad u \oplus v^{(1)}, u \oplus Tv^{(1)}, \ldots, u \oplus T^{N'-1}v^{(1)},
\]

\[
\quad u \oplus v^{(2)}, u \oplus Tv^{(2)}, \ldots, u \oplus T^{N'-1}v^{(2)}\}
\]  

Table 5.1 Set sizes and crosscorrelation bounds for the sets of all \(m\)-sequences and for maximal connected sets [2]

<table>
<thead>
<tr>
<th>(n)</th>
<th>(N = 2^n - 1)</th>
<th>Number of (m)-sequences</th>
<th>(\theta_{c}) for set of all (m)-sequences</th>
<th>(M_n)</th>
<th>(t(n))</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>7</td>
<td>2</td>
<td>5</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>4</td>
<td>15</td>
<td>2</td>
<td>9</td>
<td>0</td>
<td>9</td>
</tr>
<tr>
<td>5</td>
<td>31</td>
<td>6</td>
<td>11</td>
<td>3</td>
<td>9</td>
</tr>
<tr>
<td>6</td>
<td>63</td>
<td>6</td>
<td>23</td>
<td>2</td>
<td>17</td>
</tr>
<tr>
<td>7</td>
<td>127</td>
<td>18</td>
<td>41</td>
<td>6</td>
<td>17</td>
</tr>
<tr>
<td>8</td>
<td>255</td>
<td>16</td>
<td>95</td>
<td>0</td>
<td>33</td>
</tr>
<tr>
<td>9</td>
<td>511</td>
<td>48</td>
<td>113</td>
<td>2</td>
<td>33</td>
</tr>
<tr>
<td>10</td>
<td>1023</td>
<td>60</td>
<td>383</td>
<td>3</td>
<td>65</td>
</tr>
<tr>
<td>11</td>
<td>2047</td>
<td>176</td>
<td>287</td>
<td>4</td>
<td>65</td>
</tr>
<tr>
<td>12</td>
<td>4095</td>
<td>144</td>
<td>1407</td>
<td>0</td>
<td>129</td>
</tr>
<tr>
<td>13</td>
<td>8191</td>
<td>630</td>
<td>(\geq 703)</td>
<td>4</td>
<td>129</td>
</tr>
<tr>
<td>14</td>
<td>16383</td>
<td>756</td>
<td>(\geq 5631)</td>
<td>3</td>
<td>257</td>
</tr>
<tr>
<td>15</td>
<td>32767</td>
<td>1800</td>
<td>(\geq 2047)</td>
<td>2</td>
<td>257</td>
</tr>
<tr>
<td>16</td>
<td>65535</td>
<td>2048</td>
<td>(\geq 4095)</td>
<td>0</td>
<td>513</td>
</tr>
</tbody>
</table>

Note that \(H_{q}(u)\) contains \(N + 1 = 2^n\) sequences of period \(N\).

For \(n \equiv 0 \mod 4, \gcd(t(n), 2^n - 1) = 3\) vectors \(v^{(k)}\) are taken to be of length \(N\) rather than \(N/3\). Consequently, it can be shown that for the set \(H_{t(n)}(u)\), \(\theta_{\text{max}} = t(n)\). We call \(H_{t(n)}(u)\) a set of Gold-like sequences. The correlation functions for the sequences belonging to \(H_{t(n)}(u)\) take on values in the set
{-1, -t(n), t(n) - 2, -s(n), s(n) - 2} where s(n) is defined (for even n only) by:

\[ s(n) = 1 + 2^{n/2} = \frac{1}{2}(t(n) + 1) \]  

(5.20)

### 5.1.7 Kasami sequences

Let n be even and let u denote an m-sequence of period \( N = 2^n - 1 \) generated by \( h(x) \). Consider the sequence \( w = u[s(n)] = u[2^{n/2} + 1] \). \( w \) is a sequence of period \( 2^{n/2} + 1 \) which is generated by the polynomial \( h'(x) \), whose roots are the \( s(n) \)th powers of the roots of \( h(x) \). Furthermore, since \( h'(x) \) can be shown to be a polynomial of degree \( n/2 \), \( w \) is an m-sequence of period \( 2^{n/2} - 1 \). Consider the sequences generated by \( h(x)h'(x) \) of degree \( 3n/2 \). Any such sequence must be of one of the forms \( T^j u, T^j w, T^j u \oplus T^j w, 0 \leq i < 2^n - 1, 0 \leq j < 2^{n/2} - 1 \). Thus, any sequence \( y \) of period \( 2^n - 1 \) generated by \( h(x)h'(x) \) is some phase of some sequence in the set \( K_s(u) \) defined by:

\[ K_s(u) \Delta \{ u, u \oplus w, u \oplus T w, \ldots, u \oplus T^{2^{n/2}-2} w \} \]

(5.21)

This set of sequences is called a small set of Kasami sequences with:

\[ \theta = \{ -1, -s(n), s(n) - 2 \} \]

\[ \theta_{\text{max}} = s(n) = 1 + 2^{n/2} \]  

(5.22)

\( \theta_{\text{max}} \) for the set \( K_s(u) \) is approximately one half of the value of \( \theta_{\text{max}} \) achieved by the sets of sequences discussed previously. \( K_s(u) \) contains only \( 2^{n/2} = (N + 1)/2 \) sequences, while the sets discussed previously contain \( N + 1 \) or \( N + 2 \) sequences.

Let \( n \) be even and let \( h(x) \) denote a primitive binary polynomial of degree \( n \) that generates the m-sequence \( u \). Let \( w = u[s(n)] \) denote an m-sequence of period \( 2^{n/2} - 1 \) generated by the primitive polynomial \( h'(x) \) of degree \( n/2 \), and let \( h(x) \) denote the polynomial of degree \( n \) that generates \( u[t(n)] \). Then, the set of sequences of period \( N \) generated by \( h(x)h'(x) \), called the large set of Kasami sequences and denoted by \( K_L(u) \), is defined as follows:

1. If \( n \equiv 2 \pmod{4} \), then:

\[ K_L(u) = G(u, v) \bigcup \left[ \bigcup_{i=0}^{2^{n/2}-2} \{ T^i w \oplus G(u, v) \} \right] \]

(5.23)

where \( v = u[t(n)] \), and \( G(u, v) \) is defined in Equation (5.17).

2. If \( n \equiv 0 \pmod{4} \), then:

\[ K_L(u) = H_{t(n)}(u) \bigcup \left[ \bigcup_{i=0}^{2^{n/2}-2} \{ T^i w \oplus H_{t(n)}(u) \} \right] \]

\[ \times \bigcup \{ v^{(j)} \oplus T^k w : 0 \leq j \leq 2, 0 \leq k < (2^{n/2} - 1)/3 \} \]

(5.24)

where \( v^{(j)} \) is the result of decimating \( T^i u \) by \( t(n) \) and \( H_{t(n)}(u) \) is defined earlier by Equation (5.19). In either case, the correlation functions for \( K_L(u) \) take on values in the set \( \{ -1, -t(n), t(n) - 2, -s(n), s(n) - 2 \} \) and \( \theta_{\text{max}} = t(n) \). If \( n \equiv 2 \pmod{4} \), \( K_L(u) \) contains \( 2^{n/2}(2^n + 1) \) sequences, while if \( n \equiv 0 \pmod{4} \), \( K_L(u) \) contains \( 2^{n/2}(2^n + 1) - 1 \) sequences. The large set of Kasami sequences contains both the small set of Kasami sequences and a set of Gold (or Gold-like) sequences as subsets. More interestingly, the correlation bound \( \theta_{\text{max}} = t(n) \) is the same as that for the latter subsets. The previous discussion is summarized in Table 5.2 for some examples of codes.

<table>
<thead>
<tr>
<th>N</th>
<th>Polynomial</th>
<th>Construction</th>
<th>No.</th>
<th>Values taken on by the correlation functions</th>
</tr>
</thead>
<tbody>
<tr>
<td>31</td>
<td>3551</td>
<td>G</td>
<td>33</td>
<td>7   -1  -9</td>
</tr>
<tr>
<td>2373</td>
<td>G</td>
<td>33</td>
<td>11 7 3 -1 -5 -9</td>
<td></td>
</tr>
<tr>
<td>63</td>
<td>14551</td>
<td>G</td>
<td>65</td>
<td>15  -1  -9</td>
</tr>
<tr>
<td>14343</td>
<td>G</td>
<td>65</td>
<td>15 11 7 3 -1 -5 -9 -13</td>
<td></td>
</tr>
<tr>
<td>12471</td>
<td>H3</td>
<td>64</td>
<td>15 7 -1 -9 -17</td>
<td></td>
</tr>
<tr>
<td>1527</td>
<td>K3</td>
<td>8</td>
<td>7   -1  -9</td>
<td></td>
</tr>
<tr>
<td>133605</td>
<td>K3L</td>
<td>520</td>
<td>15 7 -1  -9 -17</td>
<td></td>
</tr>
<tr>
<td>65</td>
<td>10761</td>
<td>G</td>
<td>63</td>
<td>15 11 7 3 -1  -5 -9 -13</td>
</tr>
<tr>
<td>127</td>
<td>41567</td>
<td>G</td>
<td>129</td>
<td>15  -1  -9</td>
</tr>
<tr>
<td>255</td>
<td>231441</td>
<td>G</td>
<td>257</td>
<td>31  -1  -1</td>
</tr>
<tr>
<td>264455</td>
<td>G</td>
<td>257</td>
<td>31,15 7 3 -1  -5 -9 -13 -17,..., -29</td>
<td></td>
</tr>
<tr>
<td>326161</td>
<td>H3</td>
<td>256</td>
<td>31  -1  -1 -17 -33</td>
<td></td>
</tr>
<tr>
<td>267543</td>
<td>H3</td>
<td>256</td>
<td>31  -1  -1 -17 -33</td>
<td></td>
</tr>
<tr>
<td>11367</td>
<td>K3</td>
<td>16</td>
<td>15  -1  -1</td>
<td></td>
</tr>
<tr>
<td>6031603</td>
<td>K3L</td>
<td>4111</td>
<td>31  -1  -1 -17 -33</td>
<td></td>
</tr>
</tbody>
</table>

5.1.8 JPL sequences

These sequences are constructed by combining sequence $S_1(t, T_c)$ of length $L_1$ and $S_2(t, T_c)$ of length $L_2$ with $L_1, L_2$ prime, as $S = S(t, T_c) = S_1(t, T_c) \oplus S_2(t, T_c)$ of length $L = L_1 \times L_2$. If the composite sequence is delayed for $L_1$ chips

$$S(t - L_1T_c, T_c) = S_1(t - L_1T_c, T_c) \oplus S_2(t - L_1T_c, T_c) = S_1(t, T_c) \oplus S_2(t - L_1T_c, T_c)$$

(5.25)

and, summed up with its original version,

$$S(t, T_c) \oplus S(t - L_1T_c, T_c) = S_1(t, T_c) \oplus S_2(t, T_c) \oplus S_1(t - L_1T_c, T_c) \oplus S_2(t - L_1T_c, T_c) = S_1(t, T_c) \oplus S_2(t, T_c) \oplus S_1(t - L_1T_c, T_c) \oplus S_2(t - L_1T_c, T_c) = S_2(t - L_1T_c, T_c)$$

(5.26)

The result is only a component sequence $S_2$. In a similar way, by delaying the composite sequence for $L_2$ chips, a component sequence $S_1$ will be obtained. This can be used to synchronize sequence $S$ of length $L_1 \times L_2$ by synchronizing separately component sequences $S_1$ and $S_2$ of length $L_1$ and $L_2$, which can be done much faster. The acquisition time is proportional to $T_{acq}(S) \sim \max[T_{acq}(S_1), T_{acq}(S_2)] \sim \max[L_1, L_2]$.

5.1.9 Kronecker sequences

In this case the component sequences $S_1(t, T_{c1})$ of length $L_1$ and chip interval $T_{c1}$, and $S_2(t, T_{c2})$ with $L_2, T_{c2} = L_1 T_{c1}$ are combined as:

$$S(t, T_{c1}, T_{c2}) = S_1(t, T_{c1}) \oplus S_2(t, T_{c2})$$

(5.27)

The composite sequence, $S$, synchronization is now performed in cascade, first $S_1$ with a much faster chip rate and then $S_2$. Correlation of $S$ by $S_1$ gives:

$$F_2(S_1 \cdot S) = \rho_1 S_2$$

(5.28)

and after that, this result is correlated with sequence $S_2$. The acquisition time is proportional to $T_{acq}(S) \sim T_{acq}(S_1) + T_{acq}(S_2) \sim L_1 + L_2$. 


5.1.10 Walsh functions

A Walsh function of order $n$ can be defined recursively as follows:

$$ W(n) = \begin{bmatrix} W(n/2), & W(n/2) \\ W(n/2), & W'(n/2) \end{bmatrix} $$  \hspace{1cm} (5.29)

$W'$ denotes the logical complement of $W$, and $W(1) = [0]$. Thus,

$$ W(2) = \begin{bmatrix} 0, & 0 \\ 0, & 1 \end{bmatrix} \quad \text{and} \quad W(4) = \begin{bmatrix} 0, & 0, & 0, & 0 \\ 0, & 1, & 0, & 1 \\ 0, & 0, & 1, & 1 \\ 0, & 1, & 1, & 0 \end{bmatrix} $$  \hspace{1cm} (5.30)

$W(8)$ is as follows:

$$ W(8) = \begin{bmatrix} 0, & 0, & 0, & 0, & 0, & 0, & 0, & 0 \\ 0, & 1, & 0, & 1, & 0, & 1, & 0, & 1 \\ 0, & 0, & 1, & 1, & 0, & 0, & 1, & 1 \\ 0, & 1, & 1, & 0, & 0, & 1, & 1, & 0 \\ 0, & 0, & 0, & 0, & 1, & 1, & 1, & 1 \\ 0, & 1, & 0, & 1, & 1, & 0, & 1, & 0 \\ 0, & 0, & 1, & 1, & 1, & 1, & 0, & 0 \\ 0, & 1, & 1, & 0, & 1, & 0, & 0, & 1 \end{bmatrix} $$  \hspace{1cm} (5.31)

One can see that any two rows from the matrix

$$ w_k(n) = \{w_k, j(n)\}; \quad j = 1, \ldots, n $$
$$ w_m(n) = \{w_m, j(n)\} $$

represent the sequences whose bipolar versions have crosscorrelation equal to zero (orthogonal codes). This is valid as long as the codes are aligned as in the matrix.

A modification of the previous construction rule is shown in Figure 5.3, producing orthogonal variable speeding factor (OVSF) sequences. At each node of the graph, a code $w_k(n/2)$ of length $n/2$ is producing two new codes of length $n$ by a rule

$$ w_k(n/2) \rightarrow w_{2k-1}(n) = \{w_k(n/2), w_k(n/2)\} $$
$$ \rightarrow w_{2k}(n) = \{w_k(n/2), -w_k(n/2)\} $$

![Figure 5.3 Flow graph generating OVSF codes of length 4.](image-url)
5.1.11 Optimum PN sequences

If we represent the information bit stream as
\[ \{b_n\} = \ldots, b_{-1}, b_0, b_1, b_2, \ldots; b_k = \pm 1 \]  
and the sequence as a vector of chips
\[ y = (y_0, y_1, \ldots, y_{N-1}) \quad y_k = \pm 1 \]  
then the product of these two streams would create
\[ \hat{y}_i = \ldots; b_{-1} y_i; b_0 y_i; b_1 y_i; \ldots. \]

In other words, \( \hat{y} \) is the DSSS baseband signal which has as its \( i \)th element \( \hat{y}_i = b_n y_k \) for all \( i = nN + k \) for \( k \) in the range \( 0 \leq k \leq N - 1 \). A synchronous correlation receiver forms the inner product
\[ \langle \hat{y}_n, y \rangle = b_n \langle \hat{y}, y \rangle = b_n \theta_y(0) \]  
If the other signal is \( \hat{x} \) which is formed from the data sequence \( \{b'_n\} \) and the signature sequence \( x \) (generated by a binary vector \( x = (x_0, x_1, \ldots, x_{N-1}) \) in exactly the same manner as \( \hat{y} \) was formed from \( \{b_n\} \) and \( y \), then we have for the overall received signal:
\[ \hat{y} + T^{-l} \hat{x} \quad \text{where} \]
\[ \hat{x} = \ldots; b'_0 x; b'_1 x; \ldots \]

The output of a correlation receiver which is in synchronism with \( y \) is given by:
\[ z_n = \langle \hat{y}_n, y \rangle + \left[ b'_{n-1} \sum_{i=0}^{l-1} x_{N-l+i} + b'_n \sum_{i=l}^{N-1} x_i \right] \]

Having in mind the following relations:
\[ \sum_{i=0}^{l-1} x_{N-l+i} = \sum_{i=0}^{N-1+m} x_{i-m} \]
\[ \sum_{i=l}^{N-1} x_i = \sum_{j=0}^{N-1-l} x_j y_{j+l} \]

and the definition of aperiodic crosscorrelation function \( C_{x,y} \)
\[ C_{x,y}(l) = \begin{cases} 
    \sum_{j=0}^{N-1-l} x_j y_{j+l}, & 0 \leq l \leq N - 1 \\
    \sum_{j=0}^{N-1+l} x_{j-l} y_j, & 1 - N \leq l < 0 \\
    0, & |l| \geq N 
\end{cases} \]

Equation (5.37) becomes
\[ z_n = b_n \theta_y(0) + [b'_{n-1} C_{x,y}(l - N) + b'_n C_{x,y}(l)] \]

The optimum sequences should minimize the interfering term for all values of \( l \). Further details may be found in [14, 15, 27–40].

5.1.12 Golay code

As an appendix to this section, we will present Golay code [27] which is used in the WUMTS syncho channel due to its good aperiodic correlation properties. Additional information on the construction and implementation of these sequences can be found in [27–40].
In general for the two complementary sequences $a$ and $b$ of length $N$, with aperiodic autocorrelations $A(k)$ and $B(k)$ (delay $k$) we have:

$$A(k) + B(k) = 0, k \neq 0 \quad \text{and} \quad A(0) + B(0) = 2N.$$  

The primary synchronization code (PSC), $C_{psc}$ is constructed as a so-called generalized hierarchical Golay sequence. Define:

$$a = \langle x_1, x_2, x_3, \ldots, x_{16} \rangle = \langle 1, 1, 1, 1, 1, -1, -1, -1, -1, -1, -1, -1, 1 \rangle$$

The PSC is generated by repeating the sequence modulated by a Golay complementary sequence, and creating a complex-valued sequence with identical real and imaginary components. The PSC $C_{psc}$ is defined as:

$$C_{psc} = (1 + j) \times \langle a, a, -a, -a, a, a, -a, -a, a, a, -a, -a, a, a \rangle$$

where the leftmost chip in the sequence corresponds to the chip transmitted first in time. The 16 secondary synchronization codes (SSCs), $\{C_{ssc,1}, \ldots, C_{ssc,16}\}$, are complex-valued with identical real and imaginary components, and are constructed from position wise multiplication of a Hadamard sequence and a sequence $z$, defined as:

$$z = \langle b, b, -b, b, -b, b, -b, b, -b, b, -b, b, -b, b, -b, b \rangle, \quad \text{where}$$

$$b = \langle x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, -x_9, -x_{10}, -x_{11}, -x_{12}, -x_{13}, -x_{14}, -x_{15}, -x_{16} \rangle$$

and $x_1, x_2, \ldots, x_{15}, x_{16}$, are same as in the definition of the sequence $a$ above.

The Hadamard sequences (see also Equations (5.29–5.31)) are obtained as the rows in a matrix $H_8$ constructed recursively by:

$$H_8 = \begin{bmatrix} H_{k-1} & H_{k-1} \\ H_{k-1} & -H_{k-1} \end{bmatrix}, \quad k \geq 1$$

The rows are numbered from the top starting with row 0 (the all ones sequence). Denote the $n$th Hadamard sequence as a row of $H_8$ numbered from the top, $n = 0, 1, 2, \ldots, 255$, subsequently. Furthermore, let $h_n(i)$ and $z(i)$ denote the $i$th symbol of the sequence $h_n$ and $z$, respectively where $i = 0, 1, 2, \ldots, 255$ and $i = 0$ corresponds to the leftmost symbol. The $k$th SSC, $C_{ssc,k}, k = 1, 2, 3, \ldots, 16$ is then defined as:

$$-C_{ssc,k} = (1 + j) \times \langle h_m(0) \times z(0), h_m(1) \times z(1), h_m(2) \times z(2), \ldots, h_m(255) \times z(255) \rangle;$$

where $m = 16 \times (k - 1)$ and the leftmost chip in the sequence corresponds to the chip transmitted first in time.

### 5.1.12.1 Alternative generation

The generalized hierarchical Golay sequences for the PSC described above may also be viewed as being generated (in real-valued representation) by the following methods:

**Method 1:** The sequence $y$ is constructed from two constituent sequences $x_1$ and $x_2$ of length $n_1$ and $n_2$ respectively, using the following formula:

$$y(i) = x_2(i \mod n_2) \times x_1(i \div n_2), i = 0 \cdots (n_1 \times n_2) - 1$$

The constituent sequences $x_1$ and $x_2$ are chosen to be the length 16 (i.e. $n_1 = n_2 = 16$) sequences:

- $x_1$ is defined to be the length 16 ($N^{(1)} = 4$) Golay complementary sequence [27] obtained by the delay matrix $D^{(1)} = [8, 4, 1, 2]$ and weight matrix

$$W^{(1)} = [1, \ -1, \ 1, \ 1].$$
• $x_2$ is a generalized hierarchical sequence using the following formula, selecting $s = 2$ and using the two Golay complementary sequences $x_3$ and $x_4$ as constituent sequences. The lengths of the sequences $x_3$ and $x_4$ are called $n_3$ and $n_4$ respectively.

• $x_2(i) = x_4(i \mod s + s(i \div n_3))x_3((i \div s) \mod n_3), i = 0 \cdots (n_3 n_4) - 1$.

• $x_3$ and $x_4$ are defined to be identical and the length $4(N^{(3)} = N^{(4)} = 2)$ Golay complementary sequence obtained by the delay matrix $D^{(3)} = D^{(4)} = [1, 2]$ and weight matrix $W^{(3)} = W^{(4)} = [1, 1]$.

The Golay complementary sequences $x_1, x_3$ and $x_4$ are defined using the following recursive relation:

$$a_0(k) = \delta(k) \quad \text{and} \quad b_0(k) = \delta(k);$$
$$a_n(k) = a_{n-1}(k) + W_n^{(j)} \cdot b_{n-1}(k - D_n^{(j)});$$
$$b_n(k) = a_{n-1}(k) - W_n^{(j)} \cdot b_{n-1}(k - D_n^{(j)});$$

$k = 0, 1, 2, \ldots, 2^{N^{(j)} - 1}; n = 1, 2, \ldots, N^{(j)}$.

The desired Golay complementary sequence $x_j$ is defined by $a_n$ assuming $n = N^{(j)}$. The Kronecker delta function is described by $\delta; k, j$ and $n$ are integers.

Method 2: The sequence $y$ can be viewed as a pruned Golay complementary sequence, generated using the following parameters which apply to the generator equations for $a$ and $b$ above:

• Let $j = 0, N^{(0)} = 8$.
• $[D_0^{(1)}, D_2^{(1)}, D_3^{(1)}, D_4^{(1)}, D_5^{(1)}, D_6^{(1)}, D_7^{(1)}, D_8^{(1)}] = [128, 64, 16, 32, 8, 1, 4, 2]$.
• $[W_1^{(0)}, W_2^{(0)}, W_3^{(0)}, W_4^{(0)}, W_5^{(0)}, W_6^{(0)}, W_7^{(0)}, W_8^{(0)}] = [1, -1, 1, 1, 1, 1, 1, 1]$.
• For $n = 4, 6$, set $b_4(k) = a_4(k), b_6(k) = a_6(k)$.

5.2 MULTIUSER CDMA RECEIVERS

In this section we present a number of methods for CDMA multiple access interference cancellation. Multiple access interference is produced by the presence of the other users in the network which are located on the same bandwidth as our own signal. The common characteristic of all these schemes is some form of joint signal and parameter estimation for all signals present on the same bandwidth. It makes sense to implement this in a base station of a cellular system because all these signals are available there anyway. At the same time this concept will considerably increase the complexity of the receiver. Although very complex, these schemes are being standardized already because they offer significantly better performance.

If user $k$ transmits bit stream $b_k$, with bit interval $T$, using spreading sequence $s_k$, then the low pass equivalent of the overall signal received in the base station can be represented as [41, 42]

$$r_t = S_t(b) + \sigma n(t) \quad (5.41)$$

$$S_t(b) = \sum_{i=M}^{M} \sum_{k=1}^{K} b_k(i)s_k(t - iT - \tau_k) \quad (5.42)$$

where $K$ is the number of users, $b = (b_1, b_2, \cdots b_K)^T$ is the vector of bits of all users and the signal is observed in time interval $[-MT, MT]$. The noise component is represented by the second term of Equation (5.41) and $\tau_k$ is the delay of the signal from user $k$. 
5.2.1 Synchronous CDMA channels

If the signals from different users are received synchronously, Equation (5.41) becomes:

\[
r(t) = \sum_{k=1}^{K} b_k(j)s_k(t - jT) + \sigma n(t), \quad t \in [jT, jT + T]
\]  
(5.43)

If we use notation \( y_k \) for the output of the matched filter of user \( k \) then we have

\[
y_k = \int_0^T r(t)s_k(t)dt, \quad k = 1, \ldots, K
\]  
(5.44)

and we can write

\[
\begin{align*}
y_1 &= \sum_j b_j R_{1j} + n, \\
y_2 &= \sum_j b_j R_{2j} + n_2 \\
&\vdots \\
y_k &= \sum_j b_j R_{kj} + n_k
\end{align*}
\]  
(5.45)

The vector form of these outputs can be presented as:

\[
y = Rb + n
\]  
(5.46)

where \( R \) is the non-negative definite matrix of crosscorrelations between the assigned waveforms:

\[
R_{ij} = \int_0^T s_i(t)s_j(t)dt
\]  
(5.47)

Conventional single user detection can be represented as:

\[
\hat{b}_k = \text{sgn } y_k
\]  
(5.48)

The optimum multiuser detector becomes:

\[
\hat{b} \in \arg \min_{b \in \{-1, 1\}^K} \int_0^T \left[ r(t) - \sum_{k=1}^{K} b_k s_k(t) \right]^2 dt
\]

\[
= \arg \max_{b \in \{-1, 1\}^K} 2y^Tb - b^TRb
\]  
(5.49)

5.2.2 The decorrelating detector

In the absence of noise, the matched filter output vector is \( y = Rb \). This suggests that the detector should perform the following operation \( \hat{b} = \text{sgn } R^{-1}y \). Note that the noise components in \( R^{-1}y \) are correlated, and therefore \( \text{sgn } R^{-1}y \) does not result in optimum decisions. It is interesting to point out that this detector does not require knowledge of the energies of any of the active users.

5.2.3 The optimum linear multiuser detector

The linear detector which minimizes the probability of bit error will be referred to as the optimum linear multiuser detector. Its operation can be represented as:

\[
\hat{b} = \text{sgn}(Ty) = \text{sgn}(TRb + Tn)
\]  
(5.50)
We will consider the set \( I(\mathbf{R}) \) of generalized inverses of the crosscorrelation matrix \( \mathbf{R} \) and analyze the properties of the detector:

\[
\hat{\mathbf{b}} = \text{sgn} \mathbf{R}^l \mathbf{y}
\]  

(5.51)

in Section 5.3. The special case \( I(\mathbf{R}) = \mathbf{R}^{-1} \) is referred to as a decorrelating detector.

### 5.2.4 Multistage detection in asynchronous CDMA [43]

If the indexing of users is arranged in increasing order of their delays, then the output of the correlator of user \( k \) can be represented as:

\[
z^{(i)}(k)(0) = \int_{-\infty}^{\infty} r(t) s_i(t + iT - \tau_k) \, dt
= \eta^{(i)} + \sum_{l=k+1}^{K} R_{il}(1)b_l^{(i-1)} + \sum_{l=1}^{k} R_{il}(0)b_l^{(i)} + \sum_{l=1}^{K} R_{il}(-1)b_l^{(i+1)}
\]  

(5.52)

where \( \eta^{(i)} \) is the component of the statistic due to the additive channel noise. In vector notation, letting \( z^{(i)}(0) = [z_1^{(i)}(0), z_2^{(i)}(0), \ldots, z_K^{(i)}(0)]^T \), we have:

\[
z^{(i)}(0) = \eta^{(i)} + \mathbf{R}(1)\mathbf{b}^{(i-1)} + \mathbf{R}(0)\mathbf{b}^{(i)} + \mathbf{R}(-1)\mathbf{b}^{(i+1)}
\]  

(5.53)

The multistage detector recreates the interfering term for each user based on bit estimations in the previous stage (iteration), subtracts the estimated MAI and then makes the new estimate of data which can be represented as:

\[
\hat{b}_k^{(i)}(m + 1) = \text{sgn}[z_k^{(i)}(m)]
\]  

(5.54)

where

\[
z_k^{(i)}(m) = z_k^{(i)}(0) - \sum_{l=k+1}^{K} h_{il}(1)b_l^{(i-1)}(m) - \sum_{l \neq k} h_{il}(0)b_l^{(i)}(m) - \sum_{l=1}^{k-1} h_{il}(-1)b_l^{(i+1)}(m)
\]  

(5.55)

Examples of probability of error curves are shown in Figure 5.4. All parameters are shown in the figure itself. One can see that even a two-stage detector may significantly improve the system.

![Figure 5.4](image-url)
Figure 5.5 Error probability comparison for a two-user channel with $r_{12} = 0.7$ and signal to noise ratio of user 1 fixed at 8 dB [44] © 1991, IEEE.

Figure 5.6 Error probability comparison of the linear, two-stage and optimum detectors for a two-user channel with $r_{12} = 0.7$ and signal to noise ratio of user 1 fixed at 12 dB © 1991, IEEE.

performance. In order to further emphasize the role of MUD in the presence of the near–far effect, Figure 5.4 presents the BER for the case when the crosscorrelation is very high $r_{12} = 1/3$ (three chips long sequences). One can see that when the second user becomes stronger and stronger, the improvement compared with a conventional detector is more significant.

This conclusion becomes more and more relevant if either $r_{12}$ is increased, as in Figure 5.5, or signal to noise ratio is increased, as in Figure 5.6. Figure 5.7 demonstrates the same results for five users in the network.
5.2.5 Non-coherent detector

A conventional detector for differential phase keying signals is defined by the following equation:

$$ \hat{b}_m = \text{sgn} \left[ \text{Re} \left\{ z_m(-1)z_m(0) \right\} \right], $$

where $z_m(i) = \frac{1}{2} \int_{iT}^{(i+1)T} r(t) f_m(t - iT)dt$ (5.56)

where $f_m(t)$ is the signal matched filter function. In the trivial case it is the signal spreading code only.

In general, a non-coherent linear multiuser detector for the $m$th user, denoted by a non-zero transformation $h^{(m)} \in \mathbb{C}^K$, is defined by the decision:

$$ \hat{b}_m = \text{sgn} \left[ \text{Re} \left\{ \sum_{k=1}^{K} \bar{h}_k^{(m)} z_k(-1) \sum_{l=1}^{K} \bar{h}_l^{(m)} z_l(0) \right\} \right] $$

(5.57)

where $K$ is the length of the code. A non-coherent decorrelating detector for user $m$ is defined by the decision with the linear transformation $h = d$, where $d$ denotes the complex conjugate of the $m$th column of a generalized inverse $R'$ of $R$. If the $m$th user is linearly independent, it can be shown that $R\bar{d} = u_m$, the $m$th unit vector. If all the signature signals are linearly independent, the $R^{-1}$ exists and the decorrelating transformation $d$ is uniquely characterized as the complex conjugate of the $m$th column of the inverse of $R$. The receiver block diagram is shown in Figure 5.8.

5.2.6 Non-coherent detection in asynchronous multiuser channels [45]

The $z$-transform of Equation (5.53) gives:

$$ Z(z) = S(z) \cdot \hat{D}(z) + N(z) $$

(5.58)

where

$$ S(z) = R(-1)z + R(0) + R(1)z^{-1}, $$

(5.59)

and $Z(z)$, $\hat{D}(z)$ and $N(z)$ are the vector-valued $z$-transforms of the matched-filter output sequence, the sequence $\{\hat{d}(l) = A(l)d(l)\}$ and the noise sequence $\{n(l)\}$ at the output of the matched filters. If we
Figure 5.8 Linear multiuser DPSK detector.

Figure 5.9 Non-coherent decorrelating detector.

define

\[ G(z) = [S(z)]^{-1} = \frac{\text{adj} S(z)}{\det S(z)} \]  

(5.60)

then we have

\[ \hat{d}(z) = G(z)Z(z) \]  

(5.61)

and

\[ \hat{b}(i) = \text{sgn} \ Re[\hat{d}(i - 1) \otimes \hat{d}(i)] \]  

(5.62)

The system block diagram is shown in Figure 5.9.
Figure 5.10 Asynchronous CDMA flat Rayleigh fading channel model.

### 5.2.7 Multiuser detection in frequency non-selective Rayleigh fading channels

Topics covered in the previous chapter are now repeated for the fading channel. Previously described algorithms are extended to the fading channel by using as much analogy as possible in the process of deriving the system transfer functions. In frequency selective channels, decorrelators are combined with the RAKE-type receiver in order to further improve the system performance. A number of simulation results are presented in order to illustrate the effectiveness of these schemes. The concept of this chapter is based on proper understanding of the channel model, covered in Chapter 14. The overall system model, including the channel model for frequency non-selective fading, is shown in Figure 5.10.

Parameters $c_k(i)$ are, for fixed $i$, independent, zero mean, complex-valued Gaussian random variables, with variances $\frac{|c_k|^2}{2}$ with independent quadrature components. The time-varying nature of the channel is described via the spaced time correlation function of the $k$th channel $\Phi_k(\Delta t)$:

$$E\{c_k^*(i)c_k(j)\} = \Phi_k((j - i)T)$$ (5.63)

The received signal at the central receiver can be expressed as:

$$r(t) = S(t, b) + n(t)$$

$$S(t, b) = \sum_{i=-M}^{M} b_k(i)c_k(i)u_k(t - iT - \tau_k)$$ (5.64)

$$u_k(t) = \sqrt{E_k} x_k(t) e^{j\phi_k}$$

where $u_k(t)$ is referred to as the user $k$ signature sequence, and includes the signal amplitude (square root of signal energy), the code itself and the signal phase. By using proper notation, $r(t)$ can be represented as:

$$r(t) = b^T C u_t + n(t)$$ (5.65)

where

$$b^T [b_1(-M)b_2(-M)\cdots b_K(-M)\cdots b_1(M)b_2(M)\cdots b_K(M)]$$

$$u_t = [u^T(t + MT)\cdots u^T(t - MT)]^T$$

$$u(t) = [u_1(t - \tau_1)\cdots u_K(t - \tau_K)]^T$$ (5.66)

$$C = \text{diag}(C(-M)\cdots C(M))$$

$$C(i) = \text{diag}(c_1(i)\cdots c_K(i))$$
5.2.7.1 Multiuser maximum likelihood sequence detection

By using analogy from the previous section, the likelihood function in this case can be represented as:

\[ L(b) = 2 \text{Re}\{b^H y\} - b^H C^H R_u C b \]  

(5.67)

Upper index \(^H\) denotes the conjugate transpose and

\[ y = \int_{-\infty}^{+\infty} r(t) C^H u^*_t \, dt \]  

(5.68)

represents the vector of matched filter outputs. The correlation matrix \( R_u \) can be represented as

\[
R_u = \int_{-\infty}^{+\infty} u^*_t u_t^T \, dt = \begin{bmatrix}
R_u(0) & R_u(-1) & 0 & \cdots \\
R_u(1) & R_u(0) & R_u(-1) & \cdots \\
\vdots & \ddots & \ddots & \ddots \\
0 & \cdots & R_u(1) & R_u(0)
\end{bmatrix}
\]  

(5.69)

with block elements of dimension \( K \times K \)

\[ R_u(i - j) = \int_{-\infty}^{+\infty} u^*(t - iT)u^T(t - jT) \, dt \]  

(5.70)

and scalar elements

\[ [R_u(i - j)]_{mn} = \int_{-\infty}^{+\infty} u_m^*(t - iT - \tau_m)u_n(t - jT - \tau_n) \, dt \]  

(5.71)

5.2.7.2 Decorrelating detector

If we slightly modify the vector notation, Equation (5.65) becomes:

\[ r(t) = \sum_{i=-M}^{M} s^T(t - iT)E\Phi C(i)b(i) + n(t) \]  

(5.72)

with normalized signature waveform vector:

\[ s(t) = [s_1(t - \tau_1)s_2(t - \tau_2) \cdots s_K(t - \tau_K)]^T \]  

(5.73)

\( K \times K \) multichannel matrix

\[ C(i) = \text{diag} \left( c_1(i)c_2(i) \cdots c_K(i) \right) \]  

(5.74)

\[ E = \text{diag} \left( \sqrt{E_1} \sqrt{E_2} \cdots \sqrt{E_K} \right) \]

and matrix of carrier phases

\[ \Phi = \text{diag} \left( e^{j\phi_1} e^{j\phi_2} \cdots e^{j\phi_K} \right) \]  

(5.75)

The \( K \times K \) crosscorrelation matrix of normalized signature waveforms becomes

\[ R(\ell) = \int_{-\infty}^{+\infty} s^*(t) s^T(t + \ell T) \, dt, \]  

(5.76)

The asynchronous nature of the channel is evident from the matrix elements

\[ R_{mn}(\ell) = \int_{-\ell T + \tau_m}^{(\ell+1)T + \tau_m} s_m^*(t - \tau_m)s_n(t + \ell T - \tau_n) \, dt \]  

(5.77)

Since there is no inter-symbol interference, \( R(\ell) = 0, \forall |\ell| > 1 \) and \( R(-1) = R^H (1) \). Due to the ordering of the user \( R^H (1) \) is an upper triangular matrix with zero elements on the diagonal.
decorrelating detector front end consists of \( K \) filters matched to the normalized signature waveforms of the users. The output of this filter bank, sampled at the \( \ell\)-th bit epoch is:

\[
y(\ell) = \int_{-\infty}^{+\infty} r(t)s(t - \ell T) \, dt
\]

(5.78)

The vector of sufficient statistics can also be represented as:

\[
y(\ell) = R(-1)\Phi C(\ell + 1)b(\ell + 1) + R(0)\Phi C(\ell)b(\ell)
+ R(1)\Phi C(\ell - 1)b(\ell - 1) + n_y(\ell)
\]

(5.79)

The covariance matrix of the matched filter output noise vector sequence, \( \{n_y(\ell)\} \) is given by:

\[
E\left\{n_y^*(i)n_y^T(j)\right\} = \sigma^2 R^*(i - j)
\]

(5.80)

As in Equation (5.60) the decorrelator is a \( K \)-input \( K \)-output linear time-invariant filter with transfer function matrix:

\[
G(z) = [R(-1)z + R(0) + R(1)z^{-1}]^{-1} \Delta S^{-1}(z)
\]

(5.81)

The \( z \)-transform of the decorrelator output vector is:

\[
P(z) = E\Phi(Ch(z) + N_p(z)
\]

(5.82)

where \( N_p(z) \) is the \( z \)-transform of the output noise vector sequence having power spectral density

\[
\sigma^2 S^{-1}(z) = \sigma^2 \sum_{m=-\infty}^{\infty} D(m)z^{-m}
\]

(5.83)

The receiver block diagram for coherent reception is shown in Figure 5.11. Performance results for the detector are shown in Figure 5.12. Significant improvement in the BER is evident.

Figure 5.11 Coherent decorrelating multiuser detector.
Figure 5.12 Bit error rate of user 1 for the two-user case with Rayleigh faded paths (same average path strength) and Gold sequences of period $J = 127$ [46].

### 5.2.8 Multiuser detection in frequency selective Rayleigh fading channels

By using analogy with Equation (5.64) the received signal in this case can be represented as:

$$r(t) = S(t, b) + n(t)$$

$$S(t, b) = \sum_{i=1}^{M} \sum_{k=1}^{K} b_k(i) h_k(t - iT - \tau_k)$$

$$h_k(t) = c_k(t)^\ast u_k(t)$$

In Equation (5.84), $h_k(t)$ is the equivalent received symbol waveform of finite duration $[0, T_k]$ (convolution of equivalent low pass signature waveform $u_k(t)$ and the channel impulse response $c_k(t)$).

We define the memory of this channel as $v$, the smallest integer such that $h_k(t) = 0$ for $t > (v + 1)T$, and all $k = 1 \cdots K$. The impulse response of the $k$th user channel is given by:

$$c_k(t) = \sum_{\ell=0}^{L-1} c_{k,\ell}(t) \delta(t - \tau_{k,\ell})$$

When the signaling interval $T$ is much smaller than the coherence time of the channel, the channel is characterized as slow fading, implying that the channel characteristics can be measured accurately. Since the channel is assumed to be Rayleigh fading, the coefficients $c_{k,\ell}(t)$ are modeled as independent zero mean complex-valued Gaussian random processes. We will use the following notation:

$$h_k(t) = \sum_{\ell=0}^{L-1} c_{k,\ell}(t) u_k(t - \tau_{k,\ell}) = c_k^T(t) u_k(t)$$

For the single user vector of channel coefficients we use:

$$c_k(t) = [c_{k,0}(t), c_{k,1}(t) \cdots c_{k,L-1}(t)]^T$$

and for the signal vector of the delayed signature waveform:

$$u_k(t) = [u_k(t - \tau_{k,0}) u_k(t - \tau_{k,1}) \cdots u_k(t - \tau_{k,L-1})]^T$$
The equivalent low pass signature waveform is represented as
\[ u_k(t) = \sqrt{E_k} s_k(t) e^{i\phi_k} \]  
(5.89)
where \( E_k \) is the energy, \( s_k(t) \) is the real-valued, unit-energy signature waveform with period \( T \) and \( \phi_k \) is the carrier phase. In this case the received signal given by Equation (5.84) becomes:
\[ r(t) = S(t, b) + n(t) = b^T h_t + n(t) \]  
(5.90)
The equivalent data sequence is as in Equation (5.66):
\[ b = [b_1(M) \cdots b_K(M) \cdots b_1(−M) \cdots b_K(−M)]^T \]  
(5.91)
The equivalent waveform vector of \( NK \) elements is:
\[ h_t = [h_{1}(t + MT) \cdots h_{K}(t - MT)]^T \]  
(5.92)
with
\[ h(t) = [h_1(t - \tau_1) \cdots h_K(t - \tau_K)]^T = C^T(t) u(t) \]  
(5.93)
where
\[ C(t) = \begin{bmatrix} c_1(t) & 0 & 0 & \cdots \\
0 & c_2(t) & 0 & \cdots \\
\vdots & \ddots & \ddots & \ddots \\
0 & 0 & \cdots & c_K(t) \end{bmatrix} \]  
(5.94)
is a \( KL \times K \) multichannel matrix. \( KL \) is the total number of fading paths for all \( K \) users and:
\[ u(t) = [u_1(t - \tau_1) \cdots u_K(t - \tau_K)]^T \]  
(5.95)
is the equivalent signature vector of \( KL \) elements.

### 5.2.8.1 Multiuser maximum likelihood sequence detection

The log likelihood function in this case becomes:
\[ L(b) = 2 \text{Re}\{b^H y\} - b^H H b \]  
(5.96)
where superscript \( H \) denotes the conjugate transpose,
\[ y = \int_{-\infty}^{+\infty} r(t) h_t^* dt \]  
(5.97)
is the output of the bank of matched filters sampled at the bit epoch of the users. Matrix \( H \) is an \( N \times N \) block Toeplitz crosscorrelation waveform matrix with \( K \times K \) block elements,
\[ H(i - j) = \int_{-\infty}^{+\infty} h^*(t - iT) h^T(t - jT) dt \]  
(5.98)

### 5.2.8.2 Viterbi algorithm

Since every waveform \( h_k(t) \) is time limited to \([0, T_k], T_k < (v + 1)T\), it follows that \( H(l) = 0, \forall |l| > v + 1 \) and \( H(j) = H^T(j) \) for \( j = 1 \cdots v + 1 \). Due to the ordering of the users \( H^T(v + 1) \) is an upper triangular matrix with zero elements on the diagonal. Provided that knowledge of a channel is available, the MLS detector may be implemented as a dynamic programming algorithm of the Viterbi type. The vector Viterbi algorithm is the modification of the one introduced for \( M \)-input \( M \)-output linear channels where the dimensionality of the state space is \( 2^{(v + 1)K} \). As in the case of the AWGN channel, a more efficient decomposition of the likelihood function results in an algorithm with a state space of dimension \( 2^{(v + 1)K - 1} \).
Frequency selective fading is described by the wide-sense stationary uncorrelated scattering model. The bandwidth of each signature waveform is much larger than the coherence bandwidth of the channel, $B_w \gg (\Delta f)_c$. The time varying frequency selective channel for each user can be represented as a tapped delay line with tap spacing $1/B_w$, so that Equation (5.86) becomes:

$$h_k(t) = \sum_{i=0}^{L-1} c_{k,i}(t) u_k \left( t - \frac{i}{B_w} \right)$$

$$= s_T^T(t) E_k \Phi_k c_k(t)$$

The signature waveform vector may be described as:

$$s_k(t) = \left[ s_k(t), s_k \left( t - \frac{i}{B_w} \right), \ldots, s_k \left( t - \frac{L-1}{B_w} \right) \right]^T$$

and

$$E_k = \sqrt{T_k} I_L$$

$$\Phi_k = e^{j\phi_k} I_L$$

For a data symbol duration much longer than the multipath delay spread, $T \gg T_m$, any intersymbol interference due to channel dispersion can be neglected. Based on the above discussion, the channel model is presented in Figure 5.13. If we use notation

$$b(i) = [b_1(i) b_2(i) \cdots b_K(i)]^T, \quad i = -M \cdots M$$

$$s(t) = \left[ s_1^T(t-t_1) s_2^T(t-t_2) \cdots s_K^T(t-t_K) \right]^T$$

$$E = \text{diag}(E_1, E_2, \cdots E_K)$$

$$\Phi = \text{diag}(\Phi_1, \Phi_2, \cdots \Phi_K)$$

$$h^T(t) = [h_1(t-t_1) \cdots h_K(t-t_K)] = s^T(t) E \Phi C(t)$$

Figure 5.13 A synchronous CDMA frequency selective Rayleigh fading channel model.
Equation (5.84) becomes:

\[
r(t) = \sum_{i=-M}^{M} h^T(t - iT)b(i) + n(t) = \sum_{i=-M}^{M} s^T(t)EC(t)b(i) + n(t) \tag{5.103}
\]

We define a \( KL \times KL \) crosscorrelation matrix of normalized signature waveforms,

\[
R(l) = \int_{-\infty}^{+\infty} s(t) s^T(t + lT) dt \tag{5.104}
\]

The asynchronous mode is evident from the structure of the \( L \times L \) crosscorrelation matrix between the users \( m \) and \( n \),

\[
R_{mn}(l) = \int_{IT + \tau_m}^{(l+1)T + \tau_m} s_m(t - \tau_m) s_n^T(t + IT - \tau_n) dt \tag{5.105}
\]

Since there is no inter-symbol interference, \( R(l) = 0, \forall |l| > 1 \) and \( R(-1) = R^H(1) \). Due to the ordering of the users, \( R^H(1) \) is an upper triangular matrix with zero elements on the diagonal.

The front end of the multiuser detector consists of \( KL \) filters matched to the normalized properly delayed signature waveforms of the users, as shown in Figure 5.14. The output of this filter bank sampled at the bit epochs is given by the vector

\[
y(l) = \int_{-\infty}^{+\infty} r(t)s(t - lT) dt \tag{5.106}
\]

The vector of sufficient statistics can also be expressed as:

\[
y(l) = R(-1)EC(l + 1)b(l + 1) + R(0)EC(l)b(l) + R(1)EC(l - 1)b(l - 1) + n(l) \tag{5.107}
\]

The covariance matrix of the matched filter output noise vector is given by:

\[
E\{n^*(i)n^T(j)\} = \sigma^2 R(i - j)
\]

Figure 5.14 Multipath decorrelation.
Taking the $z$-transform gives
\[ Y(z) = S(z) (E \Phi Cb)(z) + N(z) \] (5.108)

where $(E \Phi Cb)(z)$ is the transform of sequence
\[ \{ \sqrt{E_1 e^{j\phi_1} c_{1,0}(i)} b_1(i) \cdots \sqrt{E_L e^{j\phi_L} c_{1,L-1}(i)} b_1(i) \cdots \sqrt{E_K e^{j\phi_K} c_{K,L-1}(i)} b_K(i) \}^T \] (5.109)

$S(z)$ is the equivalent transfer function of the CDMA multipath channel which depends only on the signature waveforms of the users. The multipath decorrelating (MD) filter is a $KL$-input $KL$-output linear time-invariant filter with transfer function matrix:
\[ G(z) \triangleq \left[ S(z) \right]^{-1} = \frac{\text{adj} S(z)}{\det S(z)} = [R(-1)z + R(0) + R(1)z^{-1}]^{-1} \] (5.110)

The necessary and sufficient condition for the existence of a stable, but non-causal realization of the decorrelating filter is
\[ \det[ R(-1)e^{jw} + R(0) + R(1)e^{-jw} ]^{-1} \neq 0 \ \forall w \in [0, 2\pi] \] (5.111)

The $z$-transform of the decorrelating detector outputs is
\[ P(z) = (E \Phi Cb)(z) + N_p(z) \] (5.112)

$N_p(z)$ is the $z$-transform of a stationary, filtered Gaussian noise vector sequence. The $z$-transform of the noise covariance matrix sequence is equal to
\[ \sigma^2 [S(z)]^{-1} = \sigma^2 \sum_{m=-\infty}^{\infty} D(m)z^{-m} \] (5.113)

The output of the decorrelating detector containing $L$ signal replicas of user $k$ may be expressed as:
\[ p_k(l) = c_k \sqrt{E_k} e^{j\phi_k} b_k(l) + n_k(l) \] (5.114)

The noise covariance matrix is given by:
\[ \sigma^2[D(0)]_{kk} = \sigma^2 \frac{1}{2\pi} \int_0^{2\pi} [S(e^{-jw})]_{kk}^{-1} dw \] (5.115)

### 5.2.8.3 Coherent reception with maximal ratio combining

Since the front end of the coherent multiuser detector contains the decorrelating filter, the noise components in the $L$ branches of the $k$th user are correlated. The usual approach prior to combining is to introduce the whitening operation, where whitening filter $(T^H)^{-1}$ is obtained by Cholesky decomposition $[D(0)]_{kk} = T^T T^*$. So, the output of the user of interest is given by
\[ p_{kw} = f \sqrt{E_k} e^{j\phi_k} b_k + n_{kw} \] (5.116)

where
\[ f = (T^H)^{-1} c_k \] (5.117)

and $n_{kw}$ is a zero mean Gaussian white noise vector with covariance matrix $\sigma^2 I_L$. The optimal combiner in this situation is the maximal ratio combiner (MRC). The receiver block diagram is shown in Figure 5.15. The output of the maximal ratio combiner can be represented as:
\[ \hat{b}_k = \text{sgn} (p_{kw} \cdot \hat{p}_{kw}^*) \] (5.118)
Figure 5.15 Maximal ratio combining after multipath decorrelation for coherent reception.

Figure 5.16 Error probability for a coherent RAKE and MD-MRC multiuser receiver for different multipath diversity order in a mobile radio channel using Gold signature sequences of length $J = 127$, average $i = 20$ dB [46].

For illustration purposes, a CDMA cellular mobile radio system with 1.25 MHz bandwidth and 9600 bps data rate is used. The multipath intensity profile of the mobile radio channel is given by:

$$r(\tau) = \frac{P}{T_m} e^{-\frac{\tau}{T_m}}$$  \hspace{1cm} (5.119)

where $P$ is the total average received power and $T_m$ is the multipath delay spread. Typical values of the multipath delay spread are $T_m = 0.5 \mu s$ for the suburban environment and $T_m = 3 \mu s$ for an urban environment. Therefore, we expect the multipath diversity reception with two branches in suburban areas and four to five branches in an urban setting. For the given parameters, inter-symbol interference is negligible, and the mobile radio channel can be described as a discrete multipath Rayleigh fading channel with mean square value of the path coefficients given by:

$$c_{k,l}^2 = \frac{1}{B_w} r \left( \frac{l}{B_w} \right) \hspace{1cm} l = 0 \cdots L - 1$$  \hspace{1cm} (5.120)

The BER versus the number of users is shown in Figures 5.16 and 5.17. One can see that for a large product $KL$, multiuser detector performance starts to degrade due to noise enhancement caused by matrix inversion.
More details on multiuser detection can be found in [47–57]. The latest results in this field including the systems using multiple antennas, can be found in [58–80], and [81–124].

5.3 MINIMUM MEAN SQUARE ERROR (MMSE) LINEAR MULTIUSER DETECTION

If the amplitude of user $k$’s signal in Equation (5.43) is $A_k$, then the vector of matched filter outputs $y$ in Equation (5.46) can be represented as:

$$y = RAb + n$$

(5.121)

where $A$ is a diagonal matrix with elements $A_k$

$$A = \text{diag} \| A_k \|$$

(5.122)

If the multiuser detector transfer function is denoted as $M$ then the minimum mean square error (MMSE) detector is defined as

$$\min_{M \in \mathbb{R}^{K \times K}} E[\| b - M y \|^2]$$

(5.123)

One can show that the MMSE linear detector outputs the following decisions [84, 85, 88]:

$$\hat{b}_k = \text{sgn} \left( \frac{1}{A_k} ( [R + \sigma^2 A^{-2}]^{-1} y)_k \right)$$

(5.124)

$$= \text{sgn} ([R + \sigma^2 A^{-2}]^{-1} y)_k$$

Therefore, the MMSE linear detector replaces the transformation $R^{-1}$ of the decorrelating detector by

$$[R + \sigma^2 A^{-2}]^{-1}$$

(5.125)
MINIMUM MEAN SQUARE ERROR (MMSE) LINEAR MULTIUSER DETECTION

217

\[ \sigma^2 A^{-2} = \text{diag}\left\{ \frac{\sigma^2}{A_1^2}, \ldots, \frac{\sigma^2}{A_K^2} \right\} \] (5.126)

As an illustration, for the two users case we have:

\[ [\mathbf{R} + \sigma^2 \mathbf{A}^{-2}]^{-1} = \left[ \left( 1 + \frac{\sigma^2}{A_1^2} \right) \left( 1 + \frac{\sigma^2}{A_2^2} \right) - \rho^2 \right]^{-1} \left[ \begin{array}{cc} 1 + \frac{\sigma^2}{A_2^2} & -\rho \\ -\rho & 1 + \frac{\sigma^2}{A_1^2} \end{array} \right] \] (5.127)

In the asynchronous case, similarly to the solution in Section 5.3 the MMSE linear detector is a \( K \)-input, \( K \)-output, linear, time-invariant filter with transfer function

\[ [\mathbf{R}^T]z + \mathbf{R}[0] + \sigma^2 \mathbf{A}^{-2} + \mathbf{R}[1]z^{-1}]^{-1} \] (5.128)

In Figure 5.18, the BER is presented versus the near–far ratio for different detectors. One can see that MMSE shows better performance than the decorrelator. In the figure, the signal to noise ratio of the desired user is equal to 10 dB.

5.3.1 System model in multipath fading channels

In this section the channel impulse response and the received signal will be presented as

\[ c_k(t) = \sum_{l=1}^{L_k} c_{k,l}^{(n)} \delta(t - \tau_{k,l}) \] (5.129)
The received discrete time signal over a data block of $N_b$ symbols is:

$$
r = SCAb + n \in C^{SGN_b}$$

where

$$
r = \left[ r^{T(0)}, \ldots, r^{T(Nb-1)} \right]^T \in C^{SGN_b}$$

is the input sample vector with

$$
r^{T(n)} = [r(T_s(nSG + 1)), \ldots, r(T_s(n + 1)SG) \in C^{SG}$$

$$
S = [S^{(0)}, S^{(1)}, \ldots, S^{(Nb-1)}] \in R^{SGN_b \times KLN_b}
$$

is the sampled spreading sequence matrix and $D = [(T + T_m)/T]$. In a single path channel, $D = 1$ due to the asynchronicity of users. In multipath channels, $D \geq 2$ due to the multipath spread. The code matrix is defined with several components ($S^{(n)}(0), \ldots, S^{(n)}(D)$) for each symbol interval to simplify the presentation of the crosscorrelation matrix components. $T_m$ is the maximum delay spread,

$$
S^{(n)} = \left[ s^{(n)}_{1,1}, \ldots, s^{(n)}_{1,L}, \ldots, s^{(n)}_{K,L} \right] \in R^{SGN_b \times KL}
$$

where

$$
s^{(n)}_{k,l} = \begin{cases}
0_{SGN_b \times 1} & n = 0, \tau_{k,l} = 0 \\
[\underbrace{s_{k}(T_s(SG - \tau_{k,l} + 1)), \ldots, s_{k}(T_s(SG))]^T, 0_{SGN_b - \tau_{k,l} \times 1}]^T & n = 0, \tau_{k,l} > 0 \\
[\underbrace{0_{((n-1)SG + \tau_{k,l}) \times 1}, s_{k}^T, 0_{SGN_b - n - \tau_{k,l} \times 1}]}^T & 0 < n < N_b - 1 \\
[\underbrace{0_{SGN_b - n - \tau_{k,l} \times 1}, 0_{SGN_b - n - \tau_{k,l} \times 1}]}^T] & n = N_b - 1
\end{cases}
$$

$$
s_k = \left[ s_k(T_s), \ldots, s_k(T_s(SG)) \right]^T \in R^{SG}
$$

is the sampled signature sequence of the $k$th user. By analogy with Equation (5.94):

$$
C = \text{diag}[C^{(0)}, \ldots, C^{(N_b-1)}] \in C^{KLN_b \times KN_b}
$$

is the channel coefficient matrix with

$$
C^{(n)} = \text{diag}[c^{(n)}_1, \ldots, c^{(n)}_K] \in C^{KL \times K}
$$
and
\[ c_k^{(n)} = \left[ c_k^{(n)}(1), \ldots, c_k^{(n)}(L) \right]^T \in C^L \]  
(5.140)

Equation (5.122) now becomes:
\[ A = \text{diag}\left[ A^{(0)}, \ldots, A^{(N_b-1)} \right] \in R^{K N_b \times K N_b} \]  
(5.141)

the matrix of total received average amplitudes with
\[ A^{(n)} = \text{diag}[A_1, \ldots, A_K] \in R^{K \times K} \]  
(5.142)

The bit vector from Equation (5.91) becomes
\[ b = \left[ b^T(0), \ldots, b^T(N_b-1) \right]^T \in \mathbb{N}^{K N_b} \]  
(5.143)

with the modulation symbol alphabet \( \mathbb{N} \) (with BPSK \( \mathbb{N} = \{-1, 1\} \)) and
\[ b^{(n)} = \left[ b_1^{(n)}, \ldots, b_K^{(n)} \right] \in \mathbb{N}^K \]  
(5.144)

and \( n \in C_{S G N_b} \) is the channel noise vector. It is assumed that the data bits are independent identically distributed random variables independent from the channel coefficients and the noise process.

The crosscorrelation matrix from Equation (5.104) for the spreading sequences can be formed as:
\[ R = S^T S \in R^{K L N_b \times K L N_b} \]  
(5.145)

where
\[ R^{(n,n-j)} = \sum_{i=0}^{D-j} S^{(i)}(i) S^{(n-j)}(i + j), \quad j \in \{0, \ldots, D\} \]  
(5.146)

and \( R^{(n-j,n)} = R^{T(i,n-j)} \). The elements of the correlation matrix can be written as
\[ R^{(n,n')} = \begin{bmatrix} R_{1,1}^{(n,n')} & \cdots & R_{1,K}^{(n,n')} \\ \vdots & \ddots & \vdots \\ R_{K,1}^{(n,n')} & \cdots & R_{K,K}^{(n,n')} \end{bmatrix} \in R^{K L \times K L} \]  
(5.147)

and
\[ R^{(n,n')}_{k,k'} = \begin{bmatrix} R_{k,k'}^{(n,n')} & \cdots & R_{k,k}^{(n,n')} \\ \vdots & \ddots & \vdots \\ R_{K,k}^{(n,n')} & \cdots & R_{K,K}^{(n,n')} \end{bmatrix} \in R^{L \times L} \]  
(5.148)

with
\[ R_{k,k'}^{(n,n')} = \sum_{j=\tau_k,l}^{SG-1+\tau_k,j} s_k(T_s(j - \tau_k,l)) s_{k'}^*(T_s(j - \tau_k,l' + (n' - n)SG)) = s_k^{T(i)} s_{k'}^{T(i)} \]  
(5.149)
which represents the correlation between users \( k \) and \( k' \), the \( h \)th and \( l' \)th paths, and between their \( n \)th and \( n' \)th symbol intervals.

### 5.3.2 MMSE detector structures

One of the conclusions in Section 5.2 was that noise enhancement in linear MUD causes system performance degradation for large products \( KL \). In this section we consider a possibility for reducing the size of the matrix to be inverted by using multipath combining prior to MUD. The structure is called a postcombining detector and the basic block diagram of the receiver is shown in Figure 5.19.

The starting point in the derivation of the receiver structure is the cost function \( E(\|b - \hat{b}\|^2) \) where

\[
\hat{b} = L_{\text{post}}^H r
\]

(5.150)

The detector linear transform matrix is given as

\[
L_{\text{post}} = SCA (AC^H RCA + \sigma^2 I)^{-1} \in \mathbb{C}^{S \times K N_b}
\]

(5.151)

This result is obtained by minimizing the cost function and derivation details may be found in any standard textbook on signal processing. Here \( R = S^T S \) is the signature sequence crosscorrelation matrix defined by Equation (5.145). The output of the postcombining LMMSE receiver is

\[
y_{\text{post}} = (AC^H RCA + \sigma^2 I)^{-1} (SCA)^H r \in \mathbb{C}^K
\]

(5.152)

where \((SCA)^H r\) is the multipath (MR) combined matched filter bank output. For non-fading AWGN:

\[
L_{\text{post}} = S(R + \sigma^2 (A^H A)^{-1})^{-1}
\]

(5.153)

The postcombining LMMSE receiver in fading channels depends on the channel complex coefficients of all users and paths. If the channel is changing rapidly, the optimal LMMSE receiver changes continuously. The adaptive versions of LMMSE receivers have increasing convergence problems as the fading rate increases. The dependence on the fading channel state can be removed by applying a precombining interference suppression type of receiver. The receiver block diagram in this case is shown in Figure 5.20.

![Figure 5.19 Postcombining interference suppression receiver.](image-url)
Figure 5.20 Precombining interference suppression receiver.

The transfer function of the detector is obtained by minimizing each element of the cost function

$$E(|h - \hat{h}|^2)$$  \hspace{1cm} (5.154)

where

$$h = CAb$$  \hspace{1cm} (5.155)

and

$$\hat{h} = L_{[\text{pre}]}^T r$$  \hspace{1cm} (5.156)

is estimated.

The solution of this minimization is [125]:

$$L_{[\text{pre}]} = S\left(R + \sigma^2 R_h^{-1}\right)^{-1} \in R^{SGN_b \times KLN_b}$$  \hspace{1cm} (5.157)

$$R_h = \text{diag}\left[A_1^2 R_{c_1}, \ldots, A_K^2 R_{c_k}\right] \in R^{KLN_b \times KLN_b}$$  \hspace{1cm} (5.158)

$$R_{c_k} = \text{diag}[E[|c_{k,1}|^2], \ldots, E[|c_{k,L}|^2]] \in R^{L \times L}$$  \hspace{1cm} (5.159)

$$y_{[\text{pre}]} = \left(R + \sigma^2 R_h^{-1}\right)^{-1} S^T r \in C^{KL}$$  \hspace{1cm} (5.160)

The two detectors are compared in Figure 5.21. The postcombining scheme performs better. The illustration of LMMSE–RAKE receiver performance in the near–far environment is shown in Figure 5.22 [126]. Considerable improvement compared to conventional RAKE is evident.

### 5.3.3 Spatial processing

When combined with multiple receive antennas the receiver structures may have one of the forms shown in Figure 5.23 [125–129].

The channel impulse response for the $k$th user’s $i$th sensor can now be written as:

$$c_{k,i}(t) = \sum_{l=1}^{L_k} c_{k,l}^{(n)} e^{j2\pi l^{-1}(\phi_{k,l}+\tau_i)} \delta(t - (\tau_{k,l,i}))$$  \hspace{1cm} (5.161)
Figure 5.21 Bit error probabilities as a function of the number of users for the postcombining and precombining LMMSE detectors in an asynchronous two-path fixed channel with different SNRs and bit rate 16 kbit/s, Gold code of length 31, $td/T = 4.63 \times 10^{-3}$, maximum delay spread 10 chips [126].

Figure 5.22 Bit error probabilities as a function of the near–far ratio for the conventional RAKE receiver and the precombining LMMSE (LMMSE–RAKE) receiver with different spreading factors ($G$) in a two-path Rayleigh fading channel with maximum delay spreads of 2 $\mu$s for $G = 4$, and 7 $\mu$s for other spreading factors. The average signal to noise ratio is 20 dB, the data modulation is BPSK, the number of users is 2, the other user has 20 dB higher power. Data rates vary from 128 kbit/s to 2.048 Mbit/s; no channel coding is assumed [126].
Figure 5.23  (a) The spatial–temporal multiuser (STM) receiver; (b) the TMS receive post combining interference suppression receiver with spatial signal processing (c) the SMT receiver; (d) the MST receiver. A precombining interference suppression receiver with spatial signal processing.
Figure 5.23 (Cont.).

where $L_k$ is the number of propagation paths (assumed to be the same for all users for simplicity; $L_k = L, \forall k$), $c_{k,l}^{(n)}$ is the complex attenuation factor of the $k$th user’s $l$th path, $\tau_{k,l,i}$ is the propagation delay for the $i$th sensor, $\varepsilon_i$ is the position vector of the $i$th sensor with respect to some arbitrarily chosen reference point, $\lambda$ is the wavelength of the carrier, $e(\phi_{k,l})$ is a unit vector pointing to direction $\phi_{k,l}$ (direction of arrival), and $\langle .., .. \rangle$ indicates the inner product.

Assuming that the number of propagation paths is the same for all users, the channel impulse response can be written as:

$$c_{k,l}(t) = \sum_{l=1}^{L} c_{k,l}^{(n)} e^{j2\pi \frac{\lambda}{\lambda} \angle(e(\phi_{k,l}), \varepsilon_i)} \delta(t - \tau_{k,l})$$  (5.162)

The channel matrix for the $i$th sensor consists of two components

$$C_i = C \circ \Phi_i \in \mathbb{C}^{KN_b \times KN_b}$$  (5.163)
where \( \mathbf{C} \) is the channel matrix defined in Equation (5.139). \( \circ \) is the Schur product defined as \( \mathbf{Z} = \mathbf{X} \circ \mathbf{Y} \in \mathbb{C}^{x \times y} \), i.e. all components of the matrix \( \mathbf{X} \in \mathbb{C}^{x \times y} \) are multiplied elementwise by the matrix \( \mathbf{Y} \in \mathbb{C}^{x \times y} \). \( \Phi = \text{diag}(\phi_1, \ldots, \phi_K) \), \( \Phi_i = \text{diag}(\phi_1, \ldots, \phi_{K_i}) \), is the matrix of the direction vectors

\[
\Phi_i = \left[ e^{j2\pi \lambda^{-1}\phi_{i1}}, \ldots, e^{j2\pi \lambda^{-1}\phi_{iL}} \right]^T \in \mathbb{C}^{KL}
\]  

(5.164)

By using the previous notation one can show that the equivalent detector transform matrices are given as [125–128].

\[
\begin{align*}
\mathbf{L}_{\text{STM}} &= \sum_{i=1}^I \mathbf{S}(\mathbf{C} \circ \Phi_i) \cdot \left( \sum_{i=1}^I A_{ij}^H(\Phi_i^H \circ \mathbf{C}) \mathbf{R}(\mathbf{C} \circ \Phi_i) \mathbf{A} + \sigma^2 \mathbf{I} \right)^{-1} \\
\mathbf{L}_{\text{SMT}} &= \sum_{i=1}^I \mathbf{S} \Phi_i \left( \sum_{i=1}^I \Phi_i^H \Phi_i + \sigma^2 \mathbf{R}_h^{-1} \right)^{-1} \\
\mathbf{L}_{\text{TST}} &= \mathbf{S} (\mathbf{R} + \sigma^2 \mathbf{R}_h^{-1})^{-1} \\
\mathbf{L}_{\text{TMS}} &= \text{SCA} (\mathbf{C}^H \mathbf{RCA} + \sigma^2 \mathbf{I})^{-1}
\end{align*}
\]  

5.4 SINGLE USER LMMSE RECEIVERS FOR FREQUENCY SELECTIVE FADING CHANNELS

5.4.1 Adaptive precombining LMMSE receivers

In this case the MSE criterion \( E[|\mathbf{h} - \hat{\mathbf{h}}|^2] \) requires that the reference signal \( \mathbf{h} = \mathbf{CAb} \) is available in adaptive implementations. For adaptive single user receivers, the optimization criterion is presented for each path separately, i.e.

\[
J_{k,l} = E[|\mathbf{h}_{k,l} - (\hat{\mathbf{h}})_{k,l}|^2]
\]  

(5.166)

The receiver block diagram is given in Figure 5.24 [89–94].

By using notation

\[
\begin{align*}
\mathbf{r}^{(n)} &= \left[ \mathbf{r}^{(n-D)}, \ldots, \mathbf{r}^{(n)} \right]^T \in \mathbb{C}^{\text{MSG}} \\
\mathbf{w}^{(n)} &= \left[ w_{k,l}^{(n)}(0), \ldots, w_{k,l}^{(n)}(\text{MSG} - 1) \right]^T \in \mathbb{C}^{\text{MSG}} \\
\mathbf{y}^{(n)} &= \mathbf{w}^{(n)} \mathbf{r}^{(n)}
\end{align*}
\]  

(5.167)

the bit estimation is defined as

\[
\hat{\mathbf{e}}^{(n)} = \text{sgn} \left( \sum_{l=1}^L \mathbf{z}^{(n)} \mathbf{y}^{(n)}_{k,l} \right)
\]  

(5.168)

The filter coefficients \( \mathbf{w} \) are derived using the MSE criterion \( (E[|e_{k,l}^{(n)}|^2]) \). This leads to the optimal filter coefficients \( \mathbf{w}_{\text{MSE}}^{(n)} = \mathbf{R}^{-1} \mathbf{R}_{d_{k,l}} \), where \( \mathbf{R}_{d_{k,l}} \) is the crosscorrelation vector between the input vector \( \mathbf{r} \) and the desired response \( d_{k,l} \), and \( \mathbf{R} \) is the input signal crosscorrelation matrix. Adaptive filtering can be implemented by using a number of algorithms.

5.4.1.1 The steepest descent algorithm

In this case we have

\[
\mathbf{w}^{(n+1)}_{k,l} = \mathbf{w}^{(n)}_{k,l} - \mu \nabla_{k,l}
\]  

(5.169)
Figure 5.24 Block diagram of the adaptive LMMSE–RAKE receiver.

where $\nabla$ is the gradient of

$$J_{k,l} = E\left\{ \left| c_{k,l} A_k b_k - w_{k,l}^H r \right|^2 \right\}$$

(5.170)

This can be represented as

$$\nabla_{k,l} = \frac{\partial J_{k,l}}{\partial \text{Re}(w_{k,l})} + j \frac{\partial J_{k,l}}{\partial \text{Im}(w_{k,l})} = 2 \frac{\partial J_{k,l}}{\partial w_{k,l}^*}$$

(5.171)

If the processing window $M = 1$ we have $\mathbf{r}^{(n)} = \mathbf{r}^{(n)} \approx \mathbf{r}$ and

$$\nabla_{k,l} = -2E[\mathbf{r}(c_{k,l} A_k b_k)^*] + 2E[\mathbf{r}\mathbf{r}^H] w_{k,l}$$

$$= -2 \mathbf{R}_{d_{k,l}} + 2 \mathbf{R}_w w_{k,l}$$

(5.172)

where $d_{k,l} = c_{k,l} A_k b_k$. If we assume that $A_k = 1$, $\forall k$

$$w_{k,l}^{(n+1)} = w_{k,l}^{(n)} - 2\mu \left( \mathbf{R}_{d_{k,l}} - \mathbf{R}_w w_{k,l}^{(n)} \right)$$

(5.173)

As a stochastic approximation, Equation (5.172) can be represented as

$$\nabla_{k,l} \approx -2\mathbf{r}(c_{k,l} b_k)^* + 2\mathbf{r}\mathbf{r}^H w_{k,l}^{(n)} = -2\mathbf{r}(c_{k,l} b_k)^* + 2\mathbf{r} y_{k,l}^*$$
From this equation, and assuming that $M > 1$, the LMS algorithm for updating the filter coefficients results in

$$ w_{k,l}^{(n+1)} = w_{k,l}^{(n)} + 2\mu (c_{k,l}^{(n)} b_{k}^{(n)} - y_{k,l}^{(n)})^{*} \in C^{\text{MSG}} $$

(5.174)

We decompose Equation (5.174) into adaptive and fixed components as:

$$ w_{k,l}^{(n)} = \bar{s}_{k,l} + x_{k,l}^{(n)} \in C^{\text{MSG}} $$

(5.175)

where $x_{k,l}^{(n)}$ is the adaptive filter component and

$$ \bar{s}_{k,l} = \left[ 0_{(DSG+\tau_{k,l})\times1}^{T}, \bar{s}_{k,l}^{T}, \ 0_{(DSG-\tau_{k,l})\times1}^{T} \right]^{T} $$

(5.176)

is the fixed spreading sequence of the $k$th user with the delay $\tau_{k,l}$. In this case every branch from Figure 5.24 can be represented as shown in Figure 5.25.

In this case, Equation (5.47) gives:

$$ x_{k,l}^{(n+1)} = x_{k,l}^{(n)} - 2\mu_{k,l}^{(n)} (c_{k,l}^{(n)} b_{k}^{(n)} - y_{k,l}^{(n)})^{*} \bar{r}^{(n)} = x_{k,l}^{(n)} - 2\mu_{k,l}^{(n)} \epsilon_{k,l}^{(n)} \bar{r}^{(n)} $$

(5.177)

The reference signal is

$$ d_{k,l}^{(n)} = c_{k,l}^{(n)} b_{k}^{(n)} \text{ or } \hat{d}_{k,l}^{(n)} = \hat{c}_{k,l}^{(n)} \hat{b}_{k}^{(n)} $$

(5.178)

and the channel estimator is using a pilot channel

$$ \hat{c}_{k,l}^{(n)} = \frac{1}{2N+1} \sum_{i=-N}^{N} s_{p,l}^{(n-i)} \bar{r}^{(n-i)} $$

(5.179)

To illustrate the system operation the following example is used [126]: carrier frequency 2.0 GHz, symbol rate 16 kbit/s, 31 chip Gold code and rectangular chip waveform. A synchronous downlink with equal energy two-path ($L = 2$) Rayleigh fading channel with vehicle speeds of 40 km/h (which results in the maximum normalized Doppler shift of $4.36 \times 10^{-3}$) and maximum delay spread of ten chip intervals. The number of users examined was 1–30 including the unmodulated pilot channel. The average energy was the same for the pilot channel and user data channels. A simple moving average
Figure 5.26 Simulated bit error rates as a function of the average SNR for the conventional RAKE and the adaptive LMMSE–RAKE in a two-path fading channel for vehicle speeds of 40 km/h with different numbers of users [126].

A smoother of length eleven symbols was used in a conventional channel estimator. Perfect channel estimation and ideal truncated precombining LMMSE receivers were used in the analysis to obtain the bit error probability lower bounds. The receiver processing window is three symbols ($M = 3$) unless otherwise stated. The adaptive algorithm used in the simulations was normalized LMS with

$$\mu_k^{(n)} = \frac{1}{100 \cdot (2D + 1)} SG \left( \mathbf{r}_{k,l}^{(n)} \mathbf{f}_{k,l}^{(n)} \right)^{-1}.$$ (5.180)

The simulation results were produced by averaging over the BERs of randomly selected users with different delay spreads.

The simulation results are shown in Figure 5.26. In general, one can notice that the improvement gains are lower than in the case of multiuser detectors. In the presence of a strong near–far effect, the improvements should be more evident.

### 5.4.1.2 Blind adaptive LMMSE–RAKE

In this case, in Equation (5.170) we use estimates of bits $\hat{b}_{k,l}$ instead of $b_{k,l}$ [95–98]:

$$x_{k,l}^{(n+1)} = x_{k,l}^{(n)} + 2\mu_k^{(n)} \left( c_k^{(n)} \mathbf{b}_{k,l}^{(n)} - y_{k,l}^{(n)} \right) \mathbf{f}^{(n)}$$ (5.181)

The MSE criterion now gives

$$w_{[MSE],k,l} = R_f^{-1} R_{d,k,l} = R_f^{-1} \tilde{s}_{k,l} E \left[ \left| c_{k,l} \right|^2 \right]$$ (5.182)

Similarly, the minimum output energy criteria defined as

$$MOE( E \left[ \left| y_{k,l} \right|^2 \right] )$$ (5.183)

gives

$$w_{[MOE],k,l} = R_f^{-1} \tilde{s}_{k,l} \left( \mathbf{f}_{k,l}^T R_f^{-1} \tilde{s}_{k,l} \right)$$ (5.184)
An implementation example can be seen in [130]. The stochastic approximation of the gradient of Equation (5.172) for the MOE criterion gives

$$\nabla_{k,l} = r^{(n)}F^{H(n)}w_{k,l}$$  \hspace{1cm} (5.185)

If we want to keep the useful signal autocorrelation unchanged, Equation (5.184) should be constrained to satisfy \( \bar{s}_{k,l}^{T}x_{k,l}^{(n)} = 0 \). The orthogonality condition is maintained at each step of the algorithm by projecting the gradient onto the linear subspace orthogonal to \( \bar{s}_{k,l}^{T} \). In practice, this is accomplished by subtracting an estimate of the desired signal component from the received signal vector. An implementation can be seen in [131]. So, we have:

$$x_{k,l}^{(n+1)} = x_{k,l}^{(n)} - 2\mu_{k,l}^{(n)}F^{H(n)}\left(\bar{s}_{k,l} + x_{k,l}^{(n)}\right)\left(r^{(n)} - F_{k,l}^{T}(F_{k,l}^{T}r^{(n)})\right)$$  \hspace{1cm} (5.186)

where

$$F_{k,l} = \begin{bmatrix} 0_{T_{k,l}\times 1}, & s_{l}^{T} & 0_{T_{k,l}(DSG - T_{k,l})\times 1} \\ 0_{T_{k,l}(SG - T_{k,l})\times 1}, & s_{l}^{T} & 0_{T_{k,l}(DSG + T_{k,l})\times 1} \\ 0_{T_{k,l}((2DSG + T_{k,l})\times 1)} \end{bmatrix} \in \mathbb{R}^{MSG \times M}$$  \hspace{1cm} (5.187)

is a block diagonal matrix of sampled spreading sequence vectors. Effectively, \( M \) separate filters are adapted.

### 5.4.1.3 Griffiths’ algorithm

In this case instead of assuming that vector \( R_{c_{k,l}} \) is known, the instantaneous estimate for the covariance is used, i.e.

$$R_{c} \approx \bar{r}^{(n)}r^{H(n)}$$  \hspace{1cm} (5.188)

In this case the crosscorrelation is \( R_{c_{k,l}} = E[|c_{k,l}|^2]s_{k,l} \), and Griffiths’ algorithm results in:

$$x_{k,l}^{(n+1)} = x_{k,l}^{(n)} + 2\mu_{k,l}^{(n)}E[|c_{k,l}|^2]F_{k,l}^{T}R_{c} = x_{k,l}^{(n)} + 2\mu_{k,l}^{(n)}\left(\bar{s}_{k,l} + x_{k,l}^{(n)}\right)F_{k,l}^{T}\bar{r}^{(n)}$$  \hspace{1cm} (5.189)

In practice, the energy of multipath components \( E[|c_{k,l}|^2] \) is not known and must be estimated.

### 5.4.1.4 Constant modulus algorithm

In this case the optimization criterion is \( E[\|y_{k,l}\|^2 - \omega^2] \) where \( \omega \) is the so-called constant modulus (CM), set according to the received signal power, i.e. \( \omega = E[|c_{k,l}|^2] \) or \( \omega^{(n)} = |c_{k,l}^{(n)}|^2 \). By using the CM algorithm, it is possible to avoid the use of the data decisions in the reference signal in the adaptive LMMSE–RAKE receiver by taking the absolute value of the estimated channel coefficients \( |\hat{c}_{k,l}^{(n)}| \) in adapting the receiver. In the precombining LMMSE receiver framework, the cost function for the BPSK data modulation is

$$E[|\hat{h}|^2 - |h|^2]$$  \hspace{1cm} (5.190)

The stochastic approximation of the gradient for the CM criterion is

$$\nabla_{k,l}^{(n+1)} = \left(\|y_{k,l}^{(n)}\|^2 - \|\hat{c}_{k,l}^{(n)}\|^2\right)F^{(n)}F^{H(n)}w_{k,l}$$  \hspace{1cm} (5.191)

Hence, the constant modulus algorithm can be expressed as:

$$x_{k,l}^{(n+1)} = x_{k,l}^{(n)} - 2\mu_{k,l}^{(n)}x_{k,l}^{(n)}\left(\|y_{k,l}^{(n)}\|^2 - \|\hat{c}_{k,l}^{(n)}\|^2\right)F^{(n)}$$  \hspace{1cm} (5.192)
5.4.1.5 Constrained LMMSE–RAKE, Griffiths’s and constant modulus algorithms

The adaptive LMMSE–RAKE, the Griffith’s, and the constant modulus algorithm contain no constraints. By applying the orthogonality constraint $\mathbf{s}_{s,k,l}^\top \mathbf{x}_{k,l}^{(n)} = 0$ to each of these algorithms, an additional term $\mathbf{s}_{s,k,l}^\top \mathbf{x}_{k,l}^{(n)} \mathbf{s}_{k,l}$ is subtracted from the new $\mathbf{x}_{k,l}^{(n+1)}$ update at every iteration. The constrained LMMSE–RAKE receiver becomes [28, 34]:

$$\mathbf{x}_{k,l}^{(n+1)} = \mathbf{x}_{k,l}^{(n)} + 2 \mu_{k,l} \left( \mathbf{c}_{k,l}^\top \mathbf{y}_{k,l} - \mathbf{w}_{k,l}^\top \mathbf{r}^{(n)} \right) \mathbf{r}^{(n)} - \mathbf{s}_{s,k,l}^\top \mathbf{s}_{k,l} \mathbf{x}_{k,l}^{(n)}$$  \hspace{1cm} (5.193)

The Griffith’s and the constant modulus algorithms can also be defined in a similar way.

5.4.2 Blind least squares receivers

All blind adaptive algorithms described in the previous section are based on the gradient of the cost function. In practical adaptive algorithms, the gradient is estimated, i.e. the expectation in the optimization criterion is not taken but is replaced in most cases by some stochastic approximation. This results in rather slow convergence, which may be intolerable in practical applications.

Another drawback with the blind adaptive receivers presented above is the delay estimation. Those receiver structures as such support only conventional delay estimation based on matched filtering (MF). The MF-based delay estimation is sufficient for the downlink receivers in systems with an unmodulated pilot channel, since the zero mean MAI can be averaged out if the rate of fading is low enough. If CDMA systems do not have the pilot channel, it would be beneficial to use some near–far resistant delay estimators.

5.4.3 Least squares (LS) receiver

One possible solution to both the convergence and the synchronization problems is based on blind linear least squares (LS) receivers. The cost function in this case is:

$$J_{[LS]} = \sum_{j=n-N+1}^{n} \left( \mathbf{c}_{k,l}^\top \mathbf{y}_{k,l} - \mathbf{w}_{k,l}^\top \mathbf{r}^{(j)} \right)^2$$  \hspace{1cm} (5.194)

where $N$ is the observation window in symbol intervals. Filter weights are given as

$$\mathbf{w}_{k,l}^{(n)} = \hat{\mathbf{R}}_f^{−1(n)} \hat{\mathbf{s}}_{k,l}$$  \hspace{1cm} (5.195)

$\hat{\mathbf{R}}_f^{−1(n)}$ denotes the estimated covariance matrix over a finite data block called the sample covariance matrix. This matrix can be expressed as

$$\hat{\mathbf{R}}_f^{(n)} = \sum_{j=n-N+1}^{n} \mathbf{r}^{(j)} \mathbf{r}^\top^{(j)}$$  \hspace{1cm} (5.196)

Analogous to the MOE criterion, the LS criterion can be modified as:

$$J_{[LS']} = \sum_{j=n-N+1}^{n} \left( \mathbf{w}_{k,l}^{(n)} \mathbf{r}^{(j)} \right)^2, \text{ subject to } \mathbf{w}_{k,l}^\top \hat{\mathbf{s}}_{k,l} = 1$$  \hspace{1cm} (5.197)

which results in

$$\mathbf{w}_{k,l}^{(n)} = \frac{\hat{\mathbf{R}}_f^{−1(n)} \hat{\mathbf{s}}_{k,l}}{\hat{\mathbf{s}}_{k,l}^\top \hat{\mathbf{R}}_f^{−1(n)} \hat{\mathbf{s}}_{k,l}}$$  \hspace{1cm} (5.198)

The adaptation of the blind LS receiver means updating the inverse of the sample covariance. The blind adaptive LS receiver is significantly more complex than the stochastic gradient based blind adaptive receivers. Recursive methods, such as the recursive least squares (RLS) algorithm, for
updating the inverse and iteratively finding the filter weights are known. Also, the methods based on eigendecomposition of the covariance matrix have been proposed to avoid explicit matrix inversion.

### 5.4.4 Method based on the matrix inversion lemma

The general relation

$$(A + BCD)^{-1} = A^{-1} - A^{-1}B(DA^{-1}B + C^{-1})^{-1}DA^{-1} \quad (5.199)$$

becomes

$$\hat{R}_{\tau}^{-1(n)} = \left(\hat{R}_{\tau}^{-1(n-1)} + \hat{r}^{H(n)}\hat{r}(n)\right)^{-1} \quad (5.200)$$

In time-variant channels, the old values of the inverses must be weighted by the so-called forgetting factor ($0 < \gamma < 1$), which results in:

$$\hat{R}_{\tau}^{-1(n)} = \frac{1}{\gamma} \left(\hat{R}_{\tau}^{-1(n-1)} - \hat{R}_{\tau}^{-1(n-1)}\hat{r}^{H(n)}\hat{r}(n)\hat{R}_{\tau}^{-1(n-1)}\right) \quad (5.201)$$

It is sufficient to initialize the algorithm as $\hat{R}_{\tau}^{-1(0)} = I$. For illustration purposes a numerical example is shown in Figure 5.27 [126] and Table 5.3. System parameters are shown in the figure. In general, one can see that the blind algorithms are inferior when compared with LMMSE–RAKE using pilot symbols.

More information on the topic can be found in [99–106] and especially in the late publications that include MIMO channels [107–113].

![Figure 5.27](image-url)

**Figure 5.27** Excess mean squared error as a function of the number of iterations for different blind adaptive receivers in a two-path fading channel with vehicle speeds of 40 km/h, the number of active users $K = 10$, SNR = 20 dB, $\mu = 10^{-1}$ [126].
Table 5.3 The BERs of different blind adaptive receivers at an SNR of 20 dB in a two-path Rayleigh fading channel at vehicle speeds of 40 km/h. The acronyms used are: adaptive LMMSE–RAKE (LR), adaptive MOE (MOE), Griffiths’s algorithm (GRA), constant modulus algorithm with average channel tap powers (CMA2), constrained adaptive LMMSE–RAKE (C-LR), constrained constant modulus algorithm (C-CMA), constrained Griffiths’s algorithm (C-GRA), constrained constant modulus algorithm with average channel tap powers (C-CMA2) and conventional RAKE (RAKE) [126]

<table>
<thead>
<tr>
<th>Adaptive receiver</th>
<th>( K = 30 )</th>
<th>( K = 15 )</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>( \mu = 100^{-1} )</td>
<td>( \mu = 10^{-1} )</td>
</tr>
<tr>
<td>LR</td>
<td>( 4.5 \times 10^{-2} )</td>
<td>( 6.3 \times 10^{-4} )</td>
</tr>
<tr>
<td>MOE</td>
<td>( 2.8 \times 10^{-2} )</td>
<td>( 6.0 \times 10^{-4} )</td>
</tr>
<tr>
<td>GRA</td>
<td>( 2.8 \times 10^{-2} )</td>
<td>( 6.4 \times 10^{-4} )</td>
</tr>
<tr>
<td>CMA</td>
<td>( 3.9 \times 10^{-2} )</td>
<td>( 1.2 \times 10^{-3} )</td>
</tr>
<tr>
<td>CMA2</td>
<td>( 3.3 \times 10^{-2} )</td>
<td>( 1.8 \times 10^{-3} )</td>
</tr>
<tr>
<td>C-LR</td>
<td>( 3.2 \times 10^{-2} )</td>
<td>( 6.3 \times 10^{-4} )</td>
</tr>
<tr>
<td>C-CMA</td>
<td>( 3.3 \times 10^{-2} )</td>
<td>( 6.1 \times 10^{-4} )</td>
</tr>
<tr>
<td>C-GRA</td>
<td>( 2.8 \times 10^{-2} )</td>
<td>( 6.1 \times 10^{-4} )</td>
</tr>
<tr>
<td>C-CMA2</td>
<td>( 2.9 \times 10^{-2} )</td>
<td>( 7.7 \times 10^{-4} )</td>
</tr>
<tr>
<td>RAKE</td>
<td>( 3.1 \times 10^{-2} )</td>
<td>( 7.1 \times 10^{-3} )</td>
</tr>
</tbody>
</table>

5.5 SIGNAL SUBSPACE-BASED CHANNEL ESTIMATION FOR CDMA SYSTEMS

The practical CDMA systems use the pilot signals for channel estimation enabling even the use of a smoother for these purposes. While the solution is simple, it is very inefficient in the presence of Doppler or the near–far effect. The advanced channel estimation algorithms in CDMA systems are based either on Kalman-type [114, 120–122] estimators for channels with high dynamics, or a signal subspace-based approach in the presence of high levels of multiple access interference [116, 132–141]. The best performance is obtained if a joint detection of data and channel is used [117, 119]. Additional results including blind estimation are given in [123, 124, 142]. In this section we present a multiuser channel estimation problem through a signal subspace-based approach [116]. For these purposes the received signal for \( K \) users will be presented as:

\[
r(t) = \sum_{k=1}^{K} r_k(t) + \eta, \quad -\infty < t < \infty \tag{5.202}
\]

If the channel impulse response for user \( k \) is \( h_k(t, \tau) \) we have

\[
r_k(t) = h_k(t, \tau)^* s_k(t) = \int_{-\infty}^{\infty} h_k(t, \alpha) s_k(\alpha) d\alpha \tag{5.203}
\]

If phase shift keying (PSK) is used to modulate the data, then the baseband complex envelope representation of the \( k \)th user’s transmitted signal is given by

\[
s_k(t) = \sqrt{2P_k} e^{j\phi_k} \sum_i e^{j(2\pi Mm_i^{(k)})} a_k(t - iT) \tag{5.204}
\]
where \( P_k \) is the transmitted power, \( \phi_k \) is the carrier phase relative to the local oscillator at the receiver, \( M \) is the size of the symbol alphabet, \( m_k^{(i)} \in \{0, 1, \ldots, M-1\} \) is the transmitted symbol, \( a_k(t) \) is the spreading waveform, and \( T \) is the symbol duration. The spreading waveform is given by

\[
a_k(t) = \sum_{n=0}^{N-1} \Pi_{T_c}(t-nT_c) a_k^{(n)}(t)
\]  

(5.205)

where \( \Pi_{T_c}(t) \) is a rectangular pulse, \( T_c \) is the chip duration \( (T_c = T/N) \), and \( \{a_k^{(n)}\} \) for \( n = 0, 1, \ldots, N-1 \) is a signature sequence (possibly complex-valued since the signature alphabet need not be binary). The chip matched filter can be implemented as an integrate-and-dump circuit, and the discrete time signal is given by

\[
r[n] = \frac{1}{T_c} \int_{(n+1)T_c}^{nT_c} r(t) \, dt
\]  

(5.206)

Thus, the received signal can be converted into a sequence of WSS random vectors by buffering \( r[n] \) into blocks of length \( N \)

\[
y_i = [r[iN] \, r[iN+1] \cdots r[iN+N-1]]^T \in \mathbb{C}^N
\]  

(5.207)

where the \( n \)th element of the \( i \)th observation vector is given by \( y_{i,n} = r[n+iN] \). Although each observation vector corresponds to one symbol interval, this buffering was done without regard to the actual symbol intervals of the users. Since the system is asynchronous, each observation vector will contain at least the end of the previous symbol (left) and the beginning of the current symbol (right) for each user. The factors due to the power, phase and transmitted symbols of the \( k \)th user may be collected into a single complex constant \( c_k^{(i)} \), e.g. some constant times \( \sqrt{2T_c} e^{j(\phi_k+2\pi i/M m_k^{(i)})} \), and Equation (5.207) becomes:

\[
y_i = \sum_{k=1}^{K} \left[ c_k^{(i-1)} u_k^i + c_k^{(i)} u_k^i \right] + \eta_i = A c_i + \eta_i
\]  

(5.208)

where \( \eta_i = [\eta_{i,0}, \ldots, \eta_{i,N-1}]^T \in \mathbb{C}^N \) is a Gaussian random vector. Its elements are zero mean with variance \( \sigma^2 = N_0/2T_c \) and are mutually independent. Vectors \( u_k^i \) and \( u_k^i \) are the right side of the \( k \)th user’s code vector followed by zeros, and zeros followed by the left side of the \( k \)th user’s code vector, respectively. In addition, we have defined \( c_i = [c_i^{(i-1)} c_i^{(i)} \cdots c_i^{(i-1)} c_i^{(i)}]^T \in \mathbb{C}^{2K} \) and the signal matrix \( A = [u_1^i u_2^i \cdots u_K^i u_K^i] \in \mathbb{C}^{N \times 2K} \). We will start with the assumption that each user’s signal goes through a single propagation path with an associated attenuation factor and propagation delay. We assume that these parameters vary slowly with time, so that for sufficiently short intervals the channel is approximately a linear time-invariant (LTI) system. The baseband channel impulse response can then be represented by a Dirac delta function as \( h_k(t, \tau) = h_k(t) = \alpha_k \delta(t - \tau_k), \forall \tau, \) where \( \alpha_k \) is a complex-valued attenuation weight and \( \tau_k \) is the propagation delay. Since there is just a single path, we assume that \( \alpha_k \) is incorporated into \( c_k^{(i)} \) and concentrate solely on the delay.

Let us define \( v \in \{0, \ldots, N-1\} \) and \( \gamma \in [0, 1) \) such that \( (\tau_k/T_c) \mod N = v + \gamma \). If \( \gamma = 0 \), i.e. the received signal is precisely aligned with the chip matched filter and only one chip will contribute to each sample, the signal vectors become:

\[
u_k^i = a_k^i(v) \equiv \begin{bmatrix} a_k^{(N-v)} & \cdots & a_k^{(N-1)} & 0 & \cdots & 0 \end{bmatrix}^T
\]  

(5.209)

\[
u_k^i = a_k^i(v) \equiv \begin{bmatrix} 0 & \cdots & 0 & a_k^{(0)} & \cdots & a_k^{(N-v-1)} \end{bmatrix}^T
\]  

(5.210)

Since the chip matched filter is just an integrator, the samples for a non-zero \( \gamma \) can be represented as:

\[
u_k^i = (1-\gamma)a_k^i(v) + \gamma a_k^i(v+1)
\]  

(5.210)
For the more general case of a multipath transmission channel with \( L \) distinct propagation paths, the impulse response becomes a series of delta functions

\[
h_k(t, \tau) = h_k(t) = \sum_{p=1}^{L} \alpha_{k,p} \delta(t - \tau_{k,p})
\]  

(5.211)

The signal vectors can be represented as

\[
u^* = \sum_{p=1}^{L} \alpha_{k,p} \left[ (1 - \gamma_{k,p}) \mathbf{a}_k^* (v_{k,p}) + \gamma_{k,p} \mathbf{a}_k^* (v_{k,p} + 1) \right] 
\]

\[
u'_k = \sum_{p=1}^{L} \alpha_{k,p} \left[ (1 - \gamma_{k,p}) \mathbf{a}_k^* (v_{k,p}) + \gamma_{k,p} \mathbf{a}_k^* (v_{k,p} + 1) \right] 
\]

(5.212)

If we introduce the following notation

\[
\begin{align*}
U^*_k &= \left[ \mathbf{a}_k^* (0) \cdots \mathbf{a}_k^* (N-1) \right] \in \mathbb{C}^{N \times N} \\
U'_k &= \left[ \mathbf{a}_k^* (0) \cdots \mathbf{a}_k^* (N-1) \right] \in \mathbb{C}^{N \times N}
\end{align*}
\]

(5.213)

where the \( \mathbf{a}_k \) are as defined in Equation (5.209), then the signal vectors may be expressed as a linear combination of the columns of these matrices

\[
u^*_k = U^*_k \mathbf{h}_k \\
\nu'_k = U'_k \mathbf{h}_k
\]

(5.214)

where \( \mathbf{h}_k \) is the composite impulse response of the channel and the receiver front end, evaluated modulo the symbol period. Thus, the \( n \)th element of the impulse response is given by

\[
h_{k,n} = \sum_{j=0}^{\infty} \frac{1}{T_c} \int_{jT+\pi T_c}^{(j+1)T_c} h_k(t) \prod_{T_c(t)} dt
\]

(5.215)

For delay spread \( T_m < T/2 \), at most two terms in the summation will be non-zero.

### 5.5.1 Estimating the signal subspace

The correlation matrix of the observation vectors is given by:

\[
\mathbf{R} = \mathbb{E} [\mathbf{y}_i \mathbf{y}_i^H] = \mathbf{A} \mathbf{C} \mathbf{A}^H + \sigma^2 \mathbf{I}
\]

(5.216)

where \( \mathbf{C} = \mathbb{E} [\mathbf{c}_i \mathbf{c}_i^H] \in \mathbb{C}^{2K \times 2K} \) is diagonal. The correlation matrix can also be expressed in terms of its eigenvector decomposition:

\[
\mathbf{R} = \mathbf{V} \mathbf{D} \mathbf{V}^H
\]

(5.217)

where the columns of \( \mathbf{V} \in \mathbb{C}^{N \times N} \) are the eigenvectors of \( \mathbf{R} \), and \( \mathbf{D} \) is a diagonal matrix of the corresponding eigenvalues \( \lambda_n \). Details of eigenvector decomposition are given in Appendix 5.1. Furthermore,

\[
\lambda_n = \begin{cases} 
\sigma_n + \sigma^2, & \text{if } n \leq 2K \\
\sigma^2, & \text{otherwise}
\end{cases}
\]

(5.218)

where \( \sigma_n \) is the variance of the signal vectors along the \( n \)th eigenvector and we assume that \( 2K < N \). Since the \( 2K \) largest eigenvalues of \( \mathbf{R} \) correspond to the signal subspace, \( \mathbf{V} \) can be partitioned as \( \mathbf{V} = [\mathbf{V}_S \mathbf{V}_N] \), where the columns of \( \mathbf{V}_S = [\mathbf{v}_{S,1} \cdots \mathbf{v}_{S,2K}] \in \mathbb{C}^{N \times 2K} \) form a basis for the signal subspace \( \mathcal{S}_T \), and \( \mathbf{V}_N = [\mathbf{v}_{N,1} \cdots \mathbf{v}_{N,N-2K}] \in \mathbb{C}^{N \times N-2K} \) spans the noise subspace \( \mathcal{N}_T \). Readers less familiar with eigenvalue decomposition are referred to Appendix 5.1. Since we would like to track
slowly varying parameters, we form a moving average or a Bartlett estimate of the correlation matrix based on the $J$ most recent observations:

$$\hat{R}_i = \frac{1}{J} \sum_{j=i-J+1}^{i} y_j y_j^H$$  \hspace{1cm} (5.219)$$

It is well known [143] that the maximum likelihood estimate of the eigenvalues and associated eigenvectors of $R$ is just the eigenvector decomposition of $\hat{R}_i$. Thus, we perform an eigenvalue decomposition of $\hat{R}_i$ and select the eigenvectors corresponding to the $2K$ largest eigenvalues as a basis for $\hat{SY}$.

### 5.5.2 Channel estimation

Consider the projection of a given user’s signal vectors into the estimated noise subspace

$$\hat{e}_k^r = (u_k^r \hat{V}_N)^T$$

$$\hat{e}_k^l = (u_k^l \hat{V}_N)^T$$

If $u_k^r$ and $u_k^l$ both lie in the signal subspace, then their sum $u_k = u_k^r + u_k^l$ must also be contained in $V_S$. The projection of $u_k$ into the estimated noise subspace

$$\tilde{e}_k = (u_k^r \hat{V}_N)^T$$

is a Gaussian random vector [115] and thus has probability density function

$$p_k(\tilde{e}_k) = \frac{1}{\det[\pi K]} \exp\{-\tilde{e}_k^H K^{-1} \tilde{e}_k\}$$  \hspace{1cm} (5.222)$$

The covariance matrix $K$ is a scalar multiple of the identity given by

$$K = \frac{1}{J} u_k^H Qu_k I$$  \hspace{1cm} (5.223)$$

and

$$Q = \sigma^2 \left[ \sum_{k=1}^{2K} \frac{\lambda_k}{(\sigma^2 - \lambda_k)} v_{S,k} v_{S,k}^H \right]$$

Therefore, within an additive constant, the log likelihood function of $\tilde{e}_k$ is:

$$\Lambda(\tilde{e}_k) = -(N - 2K) \ln(u_k^H Qu_k) - J \frac{\tilde{e}_k^H \tilde{e}_k}{u_k^H Qu_k}$$

$$= -(N - 2K) \ln(u_k^H Qu_k) - J \frac{u_k^H \hat{V}_N V_N^H u_k}{u_k^H Qu_k}$$  \hspace{1cm} (5.225)$$

The exact $V_N$ and $Q$ are unknown, but we may replace them with their estimates. The best estimates will minimize $\tilde{e}_k$, which will result in the maximum of the likelihood function.

Unfortunately, maximizing this likelihood function is prohibitively complex for a general multipath channel, so we will consider only a single propagation path. In this case, the vector $u_k$ is a function of only one unknown parameter: the delay $\tau_k$. To form the timing estimate, we must solve

$$\hat{\tau}_k = \arg \max_{\tau \in [0, T]} \Lambda(u_k)$$  \hspace{1cm} (5.226)$$

Ideally, we would like to differentiate the log likelihood function with respect to $\tau$. However, the desired user’s delay lies within an uncertainty region, $\tau_k \in [0, T]$, and $u_k(\tau)$ is only piecewise continuous on this interval. To deal with these problems, we divide the uncertainty region into $N$ cells
of width $T_c$ and consider a single cell, $c_v \equiv [vT_c, (v + 1)T_c)$. We again define $v \in \{0, \ldots, N - 1\}$ and $\gamma \in [0, 1)$ such that $(\tau/T_c) \mod N = v + \gamma$, and for $\tau \in c_v$ the desired user’s signal vector becomes
\[ u_k(\tau) = (1 - \gamma)u_c(v) + \gamma u_k(v + 1) \] (5.227)
and
\[ \frac{d}{d\tau} u_k(\tau) = u_k(v + 1) - u_k(v) \] (5.228)

Thus, within a given cell, we can differentiate the log likelihood function and solve for the maximum in closed form. We then choose whichever of the $N$ solutions yields the largest value for Equation (5.226). Details can be found in [116].

Under certain conditions, it may be possible to simplify this algorithm. Note that maximizing the log likelihood function (5.226) is equivalent to maximizing
\[ \Lambda(\hat{e}_k) = -\frac{N - 2K}{J} \ln(u_k^H Q u_k) - \frac{u_k^H V N V^H N u_k}{u_k^H Q u_k} \] (5.229)
As $J \to \infty$, the leading term goes to zero; thus, for large observation windows, we can use the following approximation:
\[ \Lambda(\hat{e}_k) \approx -\frac{u_k^H V N V^H N u_k}{u_k^H Q u_k} \] (5.230)
This yields a much simpler expression for the stationary points [116]. The MUSIC (Multiple Signal Classification) algorithm is equivalent to Equation (5.29) when one only maximizes the numerator and ignores the denominator, i.e. one assumes $u_k^H Q u_k$ is equal to one in Equation (5.28) or Equation (5.29). This yields an even simpler approximation for the log likelihood function
\[ \Lambda(\hat{e}_k) \approx -u_k^H V N V^H N u_k \] (5.231)
which further simplifies the solution for the stationary points [116].

For illustration purposes, the simulation results for five users with length 31 Gold codes are presented in Figures 5.28–5.30. A single desired user was acquired and tracked in the presence of strong multiple access interference (MAI). The power ratio between each of the four interfering users and the desired user is designated the MAI level.

We first compare the true log likelihood estimate, Equation (5.225), with the large observation window approximation, Equation (5.29), and the MUSIC algorithm, Equation (5.30). This is done for a window size of 200 symbols and with a varying SNR. Figure 5.28(a) shows the probability of acquisition for each method, where acquisition is defined as $|\hat{\tau}_k - \tau_k| < 1/2T_c$. Using the approximate log likelihood function results in almost no drop in performance. Furthermore, when the SNR is poor, both probabilistic approaches considerably outperform the MUSIC algorithm. In Figure 5.28(b), we compare the RMSE of the delay estimate once acquisition has occurred, i.e. after processing enough symbols to reach within half of one chip. The approximate log-likelihood function experiences a slight increase in error at low SNR, but again both probabilistic methods do better than MUSIC.

The same parameters as a function of the window size are shown in Figure 5.29. One can say that for $J > 100$ the performance curve settles down to steady state values. The RMSE versus MAI and SNR are shown in Figure 5.30. One can see that for an extremely wide ranging near–far effect the performance is good.

### 5.6 Iterative Receivers for Layered Space–Time Coding

Space–time trellis codes have a potential drawback in that the maximum likelihood decoder complexity grows exponentially with the number of bits per symbol, thus limiting achievable data rates. In Chapter 4.10 we discussed a layered space–time (LST) architecture that can attain a tight lower bound
Figure 5.28 (a) Probability of acquisition for the maximum likelihood estimator, the approximate ML and the MUSIC algorithm \([K = 5, N = 31, J = 200, \text{MAI} = 20 \text{ dB}]\); (b) Root mean squared error (RMSE) of the delay estimate in chips for the maximum likelihood (ML) estimator, the approximate ML and the MUSIC algorithm \([K = 5, N = 31, J = 200, \text{MAI} = 20 \text{ dB}]\).

There is a number of various LST architectures, depending on the way the modulated symbols are assigned to transmit antennas. In uncoded LST structure, known as vertical layered space–time (VLST) or vertical Bell Laboratories layered space–time (VBLAST), the input information sequence, denoted by \(c\), is first demultiplexed into \(n_T\) sub-streams and each of them is subsequently modulated by an \(M\)-level modulation scheme and transmitted from a transmit antenna. The signal processing chain related to an individual sub-stream is referred to as a layer. The modulated symbols are arranged into a transmission matrix, denoted by \(X\), which consists of \(n_T\) rows and \(L\) columns, where \(L\) is the transmission block length. The \(t\)th column of the transmission matrix, denoted by \(x_t\), consists of the
Tracking errors of the order of 0.1 chips can be achieved with windows of less than 200 bits, indicating that the algorithm can be used for tracking of slowly time-varying parameters.

Figure 5.29 (a) Probability of acquisition and (b) root mean squared error (RMSE) of timing estimate in chips of the subspace-based maximum likelihood estimator for varying window size \( N = 31 \), \( \text{SNR} = 8 \text{ dB} \), \( K = 5 \), \( \text{MAI} = 20 \text{ dB} \) [116] © 1996, IEEE.

Figure 5.30 RMSE of the subspace-based maximum likelihood estimator for varying MAI level \( K = 5 \), \( N = 31 \), \( J = 200 \), \( \text{SNR} = 8 \text{ dB} \).
Figure 5.31 LST transmitter architectures with error control coding: (a) an HLST architecture with a single code; (b) an HLST architecture with separate codes in each layer; (c) DLST and TLST architectures.

modulated symbols $x_1^t, x_2^t, \ldots, x_n^t$, where $t = 1, 2, \ldots, L$. At a given time $t$, the transmitter sends the $t$th column from the transmission matrix, one symbol from each antenna. That is, a transmission matrix entry $x_i^t$ is transmitted from antenna $i$ at time $t$. Vertical structuring refers to transmitting a sequence of matrix columns in the space–time domain. This simple transmission process can be combined with conventional block or convolutional one-dimension codes, to improve the performance of the system. The term ‘one-dimensional’ refers to the space domain, while these codes can be multidimensional in the time domain. The block diagrams of various LST architectures with error control coding are shown in Figure 5.31.

In the horizontal layered space–time (HLST) architecture, shown in Figure 5.31(a), the information sequence is first encoded by a channel code and subsequently demultiplexed into $n_T$ sub-streams. Each sub-stream is modulated, interleaved and assigned to a transmit antenna.

If the modulator output symbols are denoted by $x_i^t$, where $i$ represents the layer index and $t$ is the time interval, the transmission matrix, formed from the modulator outputs, denoted by $X$, is given by

$$X = [x_i^t]$$

(5.232)
In a system with three transmit antennas, the transmission matrix $X$ is given by

$$X = \begin{bmatrix} x_1^1 & x_1^2 & x_1^3 & x_1^4 & \cdots \\ x_2^1 & x_2^2 & x_2^3 & x_2^4 & \cdots \\ x_3^1 & x_3^2 & x_3^3 & x_3^4 & \cdots \end{bmatrix}$$ \hspace{1cm} (5.233)

The sequence $x_1, x_2, x_3, \ldots$ is transmitted from antenna 1, the sequence $x_2^1, x_2^2, x_2^3, \ldots$ from antenna 2 and the sequence $x_3^1, x_3^2, x_3^3, \ldots$ transmitted from antenna 3.

An HLST architecture can also be implemented by splitting the information sequence into $n_T$ sub-streams, as shown in Figure 5.31(b). After that, each sub-stream is encoded independently by a channel encoder, interleaved, modulated and then transmitted by a particular transmit antenna. We assume that channel encoders for various layers are identical although different coding in each sub-stream can be used.

A better performance is achieved by a diagonal layered space–time (DLST) architecture as discussed in Section 4.1.6, in which a modulated codeword for each encoder is distributed among the $n_T$ antennas along the diagonal of the transmission array. For example, the DLST transmission matrix, for a system with three antennas, is formed from matrix $X$ in (5.233), by delaying the $i$th row entries by $(i - 1)$ time units, so that the first nonzero entries lie on a diagonal in $X$. The entries below the diagonal are padded by zeros. Then the first diagonal is transmitted from the first antenna, the second diagonal from the second antenna, the third diagonal from the third antenna, and then the fourth diagonal from the first antenna, etc. Hence the codeword symbols of each encoder are transmitted over different antennas.

This can be represented by introducing a spatial interleaving $SI$ after the modulators, as shown in Figure 5.31(c). The spatial interleaving operation on for the DLST scheme can be represented as:

$$\begin{bmatrix} x_1^1 & x_1^2 & x_1^3 & x_1^4 & \cdots \\ 0 & x_2^2 & x_2^3 & x_2^4 & \cdots \\ 0 & 0 & x_3^3 & x_3^4 & \cdots \end{bmatrix} \rightarrow \begin{bmatrix} x_1^1 & x_1^2 & x_1^3 & x_1^4 & \cdots \\ 0 & x_2^2 & x_2^3 & x_2^4 & \cdots \\ 0 & 0 & x_3^3 & x_3^4 & \cdots \end{bmatrix}$$ \hspace{1cm} (5.234)

The rows of the matrix on the right-hand side of Equation (5.234) are obtained by concatenating the corresponding diagonals of the matrix on the left-hand side. The first row of this matrix is transmitted from the first antenna, the second row from the second antenna and the third row from the third antenna. The diagonal layering introduces space diversity and thus achieves a better performance than the horizontal one. It is important to note that there is a spectral efficiency loss in DLSV, since a portion of the transmission matrix on the left-hand side of Equation (5.234) is padded with zeros.

A threaded layered space–time (TLST) structure [144] is obtained from the HLSI by introducing a spatial interleaver $SI$ prior to the time interleavers, as shown in Figure 5.31(c). In a system with $n_T = 3$, the operation of $SI$ can be expressed as

$$\begin{bmatrix} x_1^1 & x_1^2 & x_1^3 & x_1^4 & \cdots \\ x_2^2 & x_2^3 & x_2^4 & \cdots \\ x_3^3 & x_3^4 & \cdots \end{bmatrix} \rightarrow \begin{bmatrix} x_1^1 & x_1^3 & x_1^4 & \cdots \\ x_2^2 & x_3^3 & x_4^4 & \cdots \\ x_3^3 & x_3^4 & \cdots \end{bmatrix}$$ \hspace{1cm} (5.235)

in which an element of the modulation matrix, shown on the left-hand side of Equation (5.235) denoted by $x_i^t$, represents the modulated symbol of layer $i$ at time $t$. The matrix $X$ on the right-hand side of Equation (5.235), is the TLST transmission matrix. The modulated symbols $x_1^1, x_2^2, x_3^3, x_4^4, \ldots$ generated by modulators in layers 1, 3, 2 and 1, respectively, are transmitted from antenna 1.

The spatial interleaver of the TLST can be represented by a cyclic-shift interleaver as follows. If we denote the left-hand side matrix in Equation (5.235) by $X$, the first column of the transmission matrix $X'$ is identical to the first column of the modulated matrix $X$. The second column of $X'$ is obtained by a cyclic shift of the second column of $X$ by one position from the top to the bottom. The
third column of $X'$ is obtained by a cyclic shift of the third column of $X$ by two positions, while the fourth column of $X'$ is identical to the fourth column of $X$, etc. In general, if we denote the entries of $X'$ by $x'_i$, the mapping of $x'_i$ to $x'_i$ can be expressed as $x'_i = x'_i, i' = [(i + t - 2) \mod n_T] + 1$.

The spectral efficiency HLST and TLST schemes is $Rmn_T$, where $R$ is the code rate and $m$ is the number of bits in a modulated symbol, while the spectral efficiency of the DLST is slightly reduced due to zero padding in the transmission matrix.

5.6.2 LST receivers

In this section we discuss receiver structures for layered space–time architectures. For simplicity, horizontal layering with binary channel codes and BPSK modulation are assumed. Extension to nonbinary codes and to multilevel modulation schemes is straightforward.

The transmit diversity introduces spatial interference. The signals transmitted from various antennas propagate over independently scattered paths and interfere with each other upon reception at receiver. This interference can be represented by the following matrix operation:

$$r_t = Hx_t + n_t$$

where $r_t$ is an $n_R$ component column matrix of the received signals across the $n_R$ receive antennas, $x_t$ is the $t$th column in the transmission matrix $X$ and $n_t$ is an $n_R$-component column matrix of the AWGN noise signals from the receive antennas. The noise variance per receive antenna is $\sigma^2$. In a structure with spatial interleaving, vector $x_t$ is the $t$th column of the matrix at the output of the spatial interleaver, denoted by $X'$. In order to simplify the notation, we omit the subscripts in vectors $r_t, x_t$ and $n_t$ and refer to them as $r, x, and n$, respectively.

In accordance with the discussion presented so far in this chapter, an LST structure can be viewed as a synchronous CDMA in which the number of transmit antennas is equal to the number of users. Similarly, the interference between antennas is equivalent to multiple access interference (MAI) in CDMA systems, while the complex fading coefficients correspond to the spreading sequences. This analogy can be further extended to receiver strategies, so that multiuser receiver structures derived for CDMA in Sections 5.2–5.4 can be directly applied to LST systems. Under this scenario, the optimum receiver for an uncoded LST system is a maximum likelihood (ML) multiuser detector operating on a trellis. It computes $ML$ statistics (see Equation (5.49)) as in the Viterbi algorithm. The complexity of this detection algorithm is exponential in the number of the transmit antennas.

For coded LST schemes, the optimum receiver performs joint detection and decoding on an overall trellis obtained by combining the trellises of the layered space–time code and the channel code. The complexity of the receiver is an exponential function of the product of the number of the transmit antennas and the code memory order. In this section we will examine a number of less complex receiver structures that have good performance/complexity trade-offs.

The original VLST receiver, described in Section 4.10, is based on a combination of interference suppression and cancellation. Conceptually, each transmitted sub-stream is considered to be the desired symbol and the remainder are treated as interferers. In [145] these interferers are suppressed by a zero forcing (ZF) approach which corresponds to decorrelating detector discussed in Section 5.2.2. This detection algorithm produces ZF-based decision statistics for a desired sub-stream from the received signal vector $r$, which contains a residual interference from other transmitted sub-streams. Subsequently, a decision on the desired sub-stream is made from the decision statistics and its interference contribution is regenerated and subtracted out from the received vector $r$. Thus $r$ contains a lower level of interference and this will increase the probability of correct detection of other sub-streams.

This operation is illustrated in Figure 5.32 where, the first detected sub-stream is $n_T$. The detected symbol is subtracted from all other layers. These operations are repeated for the lower layers, until layer 1 is detected. Assuming that all symbols at previous layers have been detected correctly, the decision statistics for layer 1 will be free from interference. The soft decision statistics (see Section 2.3) from the detector at each layer are passed to a decision making device in a VBLAST system. In
coded LST schemes, the decision statistics are passed to the channel decoder, which makes the hard decision on the transmitted symbol in this sub-stream. The hard symbol estimate is used to reconstruct the interference from this sub-stream, which then fed back to cancel its contribution while decoding the next sub-stream.

The ZF strategy is only possible if the number of receive antennas at least as large as the number of transmit antennas. Another drawback of this approach is that achievable diversity depends on a particular layer. If the ZF strategy is used in removing interference and if \( n_R \) receive antennas are available, it is possible to remove \( n_i = n_R - d_o \) interferences with diversity order of \( d_o \) \[146\]. The diversity order can be expressed as \( d_o = n_R - n_i \).

If the interference suppression starts at layer \( n_T \), then at this layer \((n_T - 1)\) interferers need to be suppressed. Assuming that \( n_R = n_T \), the diversity order in this layer 1. In the first layer, there are no interference to be suppressed, so the diversity order is \( n_R = n_T \). As different layers have different diversity orders, diagonal layering is required to achieve equal performance of various encoded streams.

Apart from the original BLAST receivers we will consider minimum mean square error (MMSE) detectors, discussed in Section 5.3 and iterative receivers. The iterative receiver \[147, 148\] based on the turbo processing principle introduced in Section 2.3.1, is the architecture with the best complexity/performance trade-off. Its complexity grows linearly with the number of transmit antennas and transmission rate.

### 5.6.3 QR Decomposition/SIC detector

As discussed in Appendix 4.1, any \( n_R \times n_T \) matrix \( H \), where \( n_R \geq n_T \), can be decomposed as \( H = U_R \cdot R \), where \( U_R \) is an \( n_R \times n_T \) unitary matrix and \( R \) is an \( n_T \times n_T \) upper triangular matrix, with entries \((R_{i,j})_t = 0\) for \( i > j, i, j = 1, 2, \ldots, n_T \), represented as:

\[
R = \begin{bmatrix}
(R_{1,1}), & (R_{1,2}), & \cdots & (R_{1,n_T}), \\
0 & (R_{2,2}), & \cdots & (R_{2,n_T}), \\
0 & 0 & \cdots & (R_{3,n_T}), \\
\vdots & \vdots & \vdots & \vdots \\
0 & 0 & \cdots & (R_{n_T,n_T})
\end{bmatrix}
\]  

(5.237)
If we introduce an \( n_T \)-component column matrix \( y \) obtained by multiplying from the left the receive vector \( r \), given by Equation (5.237), by \( U_R^T \) then we have \( y = U_R^T r \) or \( y = U_R^T H x + U_R^T n \). Substituting the QR decomposition of \( H \), we get \( y = R x + n' \), where \( n' = U_R^T n \) is an \( n_T \)-component column matrix of i.i.d AWGN noise signals. As \( R \) is upper-triangular, the \( i \)th component in \( y \) depends only on the \( i \)th and higher layer transmitted symbols at time \( t \), as follows:

\[
y_i^t = (R_{i,i})_t x_i^t + n_i^t + \sum_{j=i+1}^{n_T} (R_{i,j})_t x_j^t
\]

(5.238)

Consider \( x_i^t \) as the current desired detected signal. Equation (5.238) shows that \( y_i^t \) contains a lower level of interference than in the received signal \( r_t \), as the interference from \( x_j^t \), for \( l < i \), are suppressed. The third term in Equation (5.238) represents contributions from other interferers \( x_{i+1}^t, x_{i+2}^t, \ldots, x_{n_T}^t \), which can be cancelled by using the available decisions \( \hat{x}_{i+1}^t, \hat{x}_{i+2}^t, \ldots, \hat{x}_{n_T}^t \), that have been detected. The decision statistics on \( x_i^t \), denoted by \( y_i^t \), can be rewritten as:

\[
y_i^t = \sum_{j=i}^{n_T} (R_{i,j})_t x_j^t + n_i^t \quad i = 1, 2, \ldots, n_T
\]

(5.239)

The estimate on the transmitted symbol \( x_i^t \) is given by

\[
x_i^t = q \left( \frac{y_i^t - \sum_{j=i+1}^{n_T} (R_{i,j})_t \hat{x}_j^t}{(R_{i,i})_t} \right) i = 1, 2, \ldots, n_T
\]

(5.240)

where \( q(x) \) denotes the hard decision on \( x \). A QR factorization algorithms are discussed in Appendix 4.1.

For a system with three transmit antennas, the decision statistics for various layers can be expressed as

\[
\begin{align*}
y_1^t &= (R_{1,1})_t x_1^t + (R_{1,2})_t x_2^t + (R_{1,3})_t x_3^t + n_1^t \\
y_2^t &= (R_{2,1})_t x_1^t + (R_{2,2})_t x_2^t + n_2^t \\
y_3^t &= (R_{3,1})_t x_1^t + n_3^t
\end{align*}
\]

The estimate on the transmitted symbol \( x_1^t \), denoted by \( \hat{x}_1^t \), can be obtained from \( y_1^t \) as

\[
\hat{x}_1^t = q \left( \frac{y_1^t}{(R_{1,1})_t} \right)
\]

The contribution of \( \hat{x}_1^t \) is cancelled from \( y_2^t \) and the estimate on \( x_2^t \) is obtained as

\[
\hat{x}_2^t = q \left( \frac{y_2^t - (R_{2,1})_t \hat{x}_1^t}{(R_{2,2})_t} \right)
\]

Finally, after cancelling out \( \hat{x}_1^t \) and \( \hat{x}_2^t \), we obtain for

\[
\begin{align*}
\hat{x}_1^1 &= q \left( \frac{y_1^1 - (R_{1,1})_t \hat{x}_1^t}{(R_{1,1})_t} \right) \\
\hat{x}_1^2 &= q \left( \frac{y_2^1 - (R_{2,1})_t \hat{x}_1^t}{(R_{2,1})_t} \right)
\end{align*}
\]

The described algorithm applies to VBLAST. In coded LST schemes, the soft decision statistics on \( x_i^t \), given by the arguments in the \( q(\cdot) \) expressions are passed to the channel decoder, which estimates \( \hat{x}_i^t \).

In the above example the decision statistic \( y_i^{n_T} \) is computed first, then \( y_i^{n_T-1} \), and so on. The performance can be improved if the layer with the maximum SNR is detected first, followed by the one with the next largest SNR and so on [149].
5.6.4 MMSE/SIC detector

In the MMSE detection algorithm introduced in Section 5.3, the expected value of the mean square error between the transmitted vector \( \mathbf{x} \) and a linear combination of the received vector \( \mathbf{W}^H \mathbf{r} \) is minimized. With this notation Equation (5.123) becomes:

\[
\min E \left\{ (\mathbf{x} - \mathbf{W}^H \mathbf{r})^2 \right\}
\]  

(5.241)

where \( \mathbf{W} \) is an \( n_R \times n_T \) matrix of linear combination coefficients given by:

\[
\mathbf{W}^H = \left[ \mathbf{H}^H \mathbf{H} + \sigma^2 \mathbf{I}_{n_T} \right]^{-1} \mathbf{H}^H
\]  

(5.242)

\( \sigma^2 \) is the noise variance and \( \mathbf{I}_{n_T} \) is an \( n_T \times n_T \) identity matrix. The decision statistics for the symbol sent from antenna \( i \) at time \( t \) is obtained as:

\[
y_i^t = \mathbf{W}_i^H \mathbf{r}
\]  

(5.243)

where \( \mathbf{W}_i^H \) is the \( i \)th row of \( \mathbf{W}^H \) consisting of \( n_R \) components. The estimate of the symbol sent by antenna \( i \), denoted by \( \hat{x}_i^t \), is obtained by making a hard decision on \( y_i^t \):

\[
\hat{x}_i^t = q \left( y_i^t \right)
\]  

(5.244)

In an algorithm with interference suppression only, the detector calculates the hard decisions estimates by using Equations (5.243) and (5.244) for all transmit antennas. In a combined interference suppression and interference cancellation, the receiver starts from antenna \( n_T \) and computes its signal estimate by using Equations (5.243) and (5.244). The received signal \( \mathbf{r} \) in this level is denoted by \( \mathbf{r}^{n_T} \). For calculation of the next antenna signal \( (n_T - 1) \), the interference contribution of the hard estimate \( \hat{x}_i^{n_T} \) is subtracted from the received signal \( \mathbf{r}^{n_T} \) and this modified received signal, denoted by \( \mathbf{r}^{n_T-1} \), is used in computing the decision statistics for antenna \( (n_T - 1) \) in Equation (5.243) and its hard estimate from Equation (5.244). In the next level, corresponding to antenna \( (n_T - 2) \), the interference from \( n_T - 1 \) is subtracted from the received signal \( \mathbf{r}^{n_T-1} \) and this signal is used to calculate the decision statistics in Equation (5.243) for antenna \( (n_T - 2) \). This process continues for all other levels up to the first antenna. After detection of level \( i \), the hard estimate \( \hat{x}_i^t \) is subtracted from the received signal to remove its interference contribution, giving the received signal for level \( i - 1 \):

\[
\mathbf{r}^{i-1} = \mathbf{r}^i - \hat{x}_i^t \mathbf{h}_i
\]  

(5.245)

where \( \mathbf{h}_i \) is the \( i \)th column in the channel matrix \( \mathbf{H} \), corresponding to the path attenuations from antenna \( i \). The operation \( \hat{x}_i^t \mathbf{h}_i \) in Equation (5.245) replicates the interference contribution caused by \( \hat{x}_i^t \) in the received vector. \( \mathbf{r}^{i-1} \) is the received vector free from interference coming from \( \hat{x}_{i-1}^{n_T}, \hat{x}_{i-2}^{n_T}, \ldots, \hat{x}_i^t \). For estimation of the next antenna signal \( x_{i-1}^{n_T} \), this signal \( \mathbf{r}^{i-1} \) is used in Equation (5.243) instead of \( \mathbf{r} \). Finally, a deflated version of the channel matrix is calculated, denoted by \( \mathbf{H}_d^{i-1} \), by deleting a column from \( \mathbf{H}_d^i \). The deflated matrix \( \mathbf{H}_d^{i-1} \) at the \( (n_T - i + 1) \)th cancellation step is given by:

\[
\mathbf{H}_d^{i-1} = \begin{bmatrix}
h_{1,1} & h_{1,2} & \cdots & h_{1,i-1} \\
h_{2,1} & h_{2,2} & \cdots & h_{2,i-1} \\
\vdots & \vdots & \ddots & \vdots \\
h_{n_R,1} & h_{n_R,2} & \cdots & h_{n_R,i-1}
\end{bmatrix}
\]  

(5.246)

This definition is needed as the interference associated with the current symbol has been removed. This deflated matrix \( \mathbf{H}_d^{i-1} \) is used in Equation (5.242) for computing the MMSE coefficients and the signal estimate from antenna \( i - 1 \). Once the symbols from each antenna have been estimated,
the receiver repeats the process on the vector $r_{t+1}$ received at time $(t + 1)$. The algorithm can be summarized by the following pseudo code

```
MMSE/SIC Algorithm
Set $i = n_T$
and $r^{n_T} = r$
while $i \geq 1$
{
    $W^H = [H^H H + \sigma^2 I_{n_T}]^{-1} H^H$
    $y_i^t = W^H_i r^i$
    $\hat{x}_i^t = q(y_i^t)$
    $r^{i-1} = r^i - \hat{x}_i^t h_i$
    Compute $H_{d_i}^{-1}$ by deleting column $i$ from $H_i$.
    $H = H_{d_i}^{-1}$
    $i = i - 1$
}
```

The receiver can be implemented without the interference cancellation step of Equation (5.245), which will reduce system performance but some computational cost can be saved. Using cancellation requires that MMSE coefficients be recalculated at each iteration, as $H$ is deflated. With no cancellation, the MMSE coefficients are only computed once, as $n$ remains unchanged. The most computationally intensive operation detection algorithm is the computation of the MMSE coefficients. A direct calculation of the MMSE coefficients based on Equation (5.242), has a complexity polynomial in the number of transmit antennas. However, on slow fading channels, it is possible to implement adaptive MMSE receivers with the complexity being linear in the number of transmit antennas.

The described algorithm is for uncoded LST systems. The same detector can be applied to coded systems. The receiver consists of the MMSE interference suppressor/canceller (MMSE/SIC) followed by the decoder. The decision statistics $y_i^t$, from Equation (5.243), is passed to the decoder which makes the decision on the symbol estimate $\hat{x}_i^t$.

The performance of a QR decomposition receiver (QR), the linear MMSE (LMMSE) detector and the performance of the last detected layer in MMSE/SIC are shown for a VBLAST structure with $n_T = 4$, $n_R = 4$ and BPSK modulation on a slow Rayleigh fading channel in Figure 5.33. The figure also shows the interference free (single layer) BER which is given by [150]

$$P_b = \frac{1}{2} \left( 1 - \mu \right)^{n_R} \sum_{k=0}^{n_R-1} \left[ \frac{1}{2} \left( 1 + \mu \right) \right]^k$$

(5.247)

where $\mu = \sqrt{\frac{\gamma_b}{n_R + \gamma_b}}$ and $\gamma_b = \frac{E_b}{N_0}$.

One of the disadvantages of the MMSE scheme with successive interference cancellation is that the first desired detected signal to be processed sees all the interference from the remaining $(n_T - 1)$ signals, whereas each antenna signal to be processed later sees less and less interference as the cancellation progresses. This problem can be alleviated either by ordering the layers to be processed in decreasing signal power or by assigning power to the transmitted signals according to the processing order. Another disadvantage of the successive scheme is that a delay of $n_T$ computation stages is required to carry out the cancellation process.

The complexity of the LST receiver can be further reduced by replacing the MMSE interference suppressor by a matched filter, resulting in interference cancellation only.
Figure 5.33 V-BLAST example, $n_T = 4$, $n_R = 4$, with QR decomposition, MMSE interference suppression and MMSE/SIC.

5.6.5 Iterative LST Receivers

The objective in the detection of space–time signals is to design a low-complexity detector, which can efficiently remove multilayer interference and approach the interference free bound. The iterative processing principle, as applied in turbo coding and discussed in Section 2.3.1, has been successfully extended to joint detection and decoding [151–158]. This receiver can be applied only in coded LST systems. Block diagrams of the iterative receivers for LST(a)–(c) architectures are shown in Figure 5.34. In all three receivers, the detector provides joint soft-decision estimates of the $n_T$ transmitted symbol sequences. In LST(a) the detected sequence is decoded by a single decoder with soft inputs/soft outputs, while in LST (b) each of the detected sequences is decoded by a separate channel decoder with soft inputs/soft outputs. At each iteration, the decoder soft outputs are used to update the a priori probabilities of the transmitted signals. These updated probabilities are then used to calculate the symbol estimate in the detector. Note that each of the coded streams is independently interleaved to enable receiver convergence. In LST (c), apart from time interleaving/deinterleaving, there is space interleaving/deinterleaving across transmit antennas.

5.6.5.1 Iterative receiver (IR)/PIC

A block diagram of the standard iterative receiver (IR) with a parallel interference canceller (PIC-STD) is shown in Figure 5.35 as IR/PIC structure. For simplicity we assume that an HLST architecture with separate error control coding in each layer is used. In addition, the same convolutional codes with BPSK modulation are selected in each layer. In the first iteration, the PIC detectors are equivalent to a bank of matched filters. The detectors provide decision statistics of the $n_T$ transmitted symbol sequences. The decision statistics in the first iteration, for antenna $i$ and time $t$, denoted by $y^{i,1}_t$, is determined as $y^{i,1}_t = h^H_i r$, where $h^H_i$ is the $i$th row of matrix $H^H$. These decision statistics are passed to the respective decoders, which generate soft estimates on the transmitted symbols. In the second and later iterations, the decoder soft output is used to update the PIC detector decision statistics.

The decision statistics in the $k$th iteration at time $t$, for transmit antenna $i$ denoted by $y^{i,k}_t$, is given by $y^{i,k}_t = h^H_i (r - H\hat{X}^{k-1}_i)$ where $\hat{X}^{k-1}_i$ is an $n_T \times 1$ column matrix with the symbol estimates from
Figure 5.34 Block diagrams of iterative LSTC receivers; (LST-a) HLST with a single decoder; (LST-b) HLST with separate decoders; (LST-c) DLST and TLST receivers.

Figure 5.35 Iterative receiver with PIC-STD (IR/PIC-STD).
the \((k-1)\)th iteration a \(s\) elements, except for the \(i\)th element which is set to zero. It can be written as:

\[
\hat{x}_t^{k-1} = (\hat{x}_t^{1,k-1}, \ldots, \hat{x}_t^{i-1,k-1}, 0, \hat{x}_t^{i+1,k-1}, \ldots, \hat{x}_t^{nT,k-1})^T
\]

The detection outputs for layer \(i\) for a whole block of transmitted symbols form a vector, \(y^{i,k}\), which is interleaved and then passed to the \(i\)th decoder.

The decoder in the \(k\)th iteration calculates the log-likelihood ratios (LLR) for antenna \(i\) at time \(t\), denoted by \(\lambda_t^{i,k}\) and given by:

\[
\lambda_t^{i,k} = \log \frac{P(x_t^{i,k} = 1 | y_t^{i,k})}{P(x_t^{i,k} = -1 | y_t^{i,k})}
\]

where \(P(x_t^{i,k} = j | y_t^{i,k}), j = 1, -1\), are the symbol \(a posteriori\) probabilities (APP). The LLR can be calculated by the iterative MAP algorithm (Appendix 2.1). The symbol \(a posteriori\) probabilities \(P(x_t^{i,k} = j | y_t^{i,k}), j = 1, -1\), can be expressed as:

\[
P(x_t^{i,k} = 1 | y_t^{i,k}) = \frac{e^{\lambda_t^{i,k}}}{1 + e^{\lambda_t^{i,k}}}; \quad P(x_t^{i,k} = j | y_t^{i,k}) = \frac{1}{1 + e^{\lambda_t^{i,k}}}
\]

The estimates of the transmitted symbols are calculated by finding their mean:

\[
\hat{x}_t^{i,k} = 1 \cdot P(x_t^{i,k} = 1 | y_t^{i,k}) + (-1) \cdot P(x_t^{i,k} = -1 | y_t^{i,k})
\]

By combining the above equations we express the symbol estimates as functions of the LLR:

\[
\hat{x}_t^{i,k} = \frac{e^{\lambda_t^{i,k}} - 1}{e^{\lambda_t^{i,k}} + 1}
\]

When the LLR is calculated on the basis of the \(a posteriori\) probabilities, it is obtained as

\[
\lambda_t^{i,k} = \log \frac{\sum_{m,m'=M_t-1}^{m,m'=M_t-1} \alpha_j - 1 (m') p_t(x_t^i = 1) \exp \left( -\sum_{l=0}^{n} (y_t^{j,k}-\mu_t^i)^2 \frac{1}{2(\sigma_t^{i,k})^2} \right) \beta_j(m)}{\sum_{m,m'=0}^{m,m'=0}^{M_t-1} \alpha_j - 1 (m') p_t(x_t^i = -1) \exp \left( -\sum_{l=0}^{n} (y_t^{j,k}-\mu_t^i)^2 \frac{1}{2(\sigma_t^{i,k})^2} \right) \beta_j(m)}
\]

where \(\lambda_t^{i,k}\) denotes the LLR ratio for the \(p\)th symbol within the \(j\)th codeword transmitted at time \(t = (j-1)n + p\), and \(n\) is the code symbol length, \(m'\) and \(m\) are the pair of states connected in the trellis, \(x_t^i\) is the \(r\)th BPSK modulated symbol in a code symbol connecting the states \(m'\) and \(m\), \(y_t^{j,k}\) is the detector output in iteration \(k\) for antenna \(i\) at time \(t\), \((\sigma_t^{i,k})^2\) is the noise variance for layer \(i\), and iteration \(k\), \(M_t\) is the number of states in the trellis, and \(a(m')\) and \(\beta(m)\) are the feed-forward and the feedback recursive variables, defined as for the LLR (Appendix 2.1).

In computing the LLR value in Equation (5.250) the decoder uses two inputs. The first input is the decision statistics, \(y_t^{j,k}\), which depends on the transmitted signal \(x_t^i\). The second input is the \(a priori\) probability on the transmitted signal \(x_t^i\), computed as:

\[
P_t(x_t^i = l) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(\mu_t^i - \mu_t^l)^2}{2\sigma^2}} \quad l = 1, -1
\]

where \(\mu_t^l\) is the mean of the received amplitude after matched filtering, given by \(\mu_t^l = h_h^i x_t^i\).

As \(p_t(x_t^i = l)\) in Equation (5.252) depends also on \(x_t^i\), the inputs to the decoder in iteration \(k\), where \(k > 1\), are correlated. This causes the decision statistics mean value, conditional on \(x_t^i\), to be biased [141, 159]. The bias always has a sign opposite to \(x_t^i\). Thus, the bias reduces the useful signal term and degrades the system performance. This bias is particularly significant for a large number of interferers. The bias effect can be eliminated by estimating the mean of the transmitted symbols based on the \(a posteriori\) extrinsic information ratio (EIR) instead of the LLR [147]. The
Extrinsic information represents the information on the coded bit of interest calculated from the a priori information on the other coded bits and the code constraints. The EIR does not include the metric for the symbol $x_i^t$ that is being estimated. That is:

$$
\lambda_{i,t}^{j,k} = \log \frac{\sum_{m,m' = 0}^{m_0-1} \alpha_j - 1 (m') p_t \left( x_i^t = 1 \right) \exp \left( -\frac{\sum_{l=1}^{L} \left( y_{i,l}^t - y_{l}^t \right)^2}{2(\sigma_i^t)^2} \right) \beta_j (m)}{\sum_{m,m' = 0}^{m_0-1} \alpha_j - 1 (m') p_t \left( x_i^t = -1 \right) \exp \left( -\frac{\sum_{l=1}^{L} \left( y_{i,l}^t - y_{l}^t \right)^2}{2(\sigma_i^t)^2} \right) \beta_j (m)}
$$

(5.252)

where $\lambda_{i,t}^{j,k}$ denotes the EIR for the $p$th symbol within the $j$th codeword transmitted at time $t = (j - 1)n + p$, $y_{i}^{t}$ is the detector output iteration $k$ for antenna $i$, $\alpha(m')$ and $\beta(m')$ are defined as for the LLR (Appendix 2.1). However, excluding the contribution of the bit of interest reduces the extrinsic information SNR, which leads to a degraded system performance.

A decision statistics combining (DSC) method is effective in minimizing these effects. In the iterative parallel interference canceller with decision statistics combining (PIC-DSC) [147], shown in Figure 5.36, a DSC module is added to the PIC-DSC structure. The decision statistics of the PIC-DSC is generated as a weighted sum of the current PIC output and the DSC output from the previous operation. In each stage, except in the first one, the PIC output is passed to the DSC module. The DSC node performs recursive linear combining of the detector output in iteration $k$ for layer $i$, denoted by $y_{i}^{t,k}$, with the DSC output from the previous iteration for the same layer, denoted by $y_{i}^{t,k-1}$. The output of the decision statistics combiner, in iteration $k$ and for layer $i$, denoted by $y_{i}^{t,k}$, is given by $y_{i}^{t,k} = p_1^{i,k} y_{i}^{t,k-1} + p_2^{i,k} y_{i}^{t,k-1}$ where $p_1^{i,k}$ and $p_2^{i,k}$ are the DSC weighting coefficients in stage $k$, respectively. They are estimated by maximizing the signal-to-noise plus interference ratio (SINR) at the output of DSC in iteration $k$ under the assumption that $y_{i}^{t,k}$ and $y_{i}^{t,k-1}$ are Gaussian random variables with the conditional means $\mu_{i}^{t,k}$ and $\mu_{i}^{t,k-1}$, given that $x_i$ is the transmitted symbol for antenna $i$, and variances $(\sigma_{i}^{t,k})^2$ and $(\sigma_{i}^{t,k-1})^2$, respectively.

Figure 5.36 Iterative receiver with PIC-DSC (IR/PIC-DSC).
Coefficients $p_{i,k}^{i,k}$ and $p_{c}^{i,k}$ can be normalized in the following way: $E[y_{c}^{i,k}] = p_{i,k}^{i,k} \mu_{c}^{i,k} + p_{c}^{i,k} \mu_{c}^{i,k-1} + 1$. The SINR at the output of the DSC for layer $i$ and in iteration $k$ is then given by:

$$\text{SINR}^{i,k} = \frac{1}{(p_{i}^{i,k})^2 (\sigma_{i}^{i,k})^2 + 2p_{i}^{i,k} (1 - p_{i}^{i,k} \mu_{c}^{i,k}) \rho_{k,k-1}^{i} + \left( \frac{1 - p_{i}^{i,k} \mu_{c}^{i,k}}{\mu_{c}^{i,k-1}} \right)^2 (\sigma_{c}^{i,k-1})^2}$$

where $\rho_{k,k-1}^{i}$ is the correlation coefficient for layer $i$, between the detector output in the $k$th and $(k-1)$th iterations defined as:

$$\rho_{k,k-1}^{i} = \frac{E \{ (y_{c}^{i,k} - \mu_{c}^{i,k} x_{i}^{k}) (y_{c}^{i,k-1} - \mu_{c}^{i,k-1} x_{i}^{k}) \} | x_{i}^{k} \}$$

The optimal combining coefficient is given by

$$p_{i}^{i,k}_{\text{opt}} = \frac{\mu_{c}^{i,k} \left( \sigma_{c}^{i,k-1} \right)^2 - \frac{1}{\mu_{c}^{i,k-1} \rho_{k,k-1}^{i} \sigma_{c}^{i,k-1}} \left( \sigma_{i}^{i,k} \right)^2 \left( \sigma_{c}^{i,k-1} \right)^2}{\left( \sigma_{i}^{i,k} \right)^2 - 2 \frac{\mu_{c}^{i,k} \left( \rho_{k,k-1}^{i} \sigma_{i}^{i,k} \sigma_{c}^{i,k-1} \right) + \left( \frac{\mu_{c}^{i,k}}{\mu_{c}^{i,k-1}} \right)^2 \left( \sigma_{c}^{i,k-1} \right)^2}{\left( \sigma_{i}^{i,k} \right)^2 + \left( \sigma_{c}^{i,k} \right)^2}}$$

In the derivation of the optimal coefficients we assume that $\mu_{c}^{i,k}, \mu_{c}^{i,k-1}, (\sigma_{c}^{i,k})^2$ and $(\sigma_{c}^{i,k-1})^2$ are the true conditional means and the true variances of the detector outputs.

The parameters required for the calculation of the optimal combining coefficients in Equation (5.255) are difficult to estimate, apart from the signal variances. However, in a system with a large number of interferers, which happens when the number of transmit antennas is large relative to the number of receive antennas, and for the APP based symbol estimates, the DSC inputs in the first few iterations are low correlated. Thus, it is possible to combine them, in a way similar to receiving diversity maximum ratio combining. Under these conditions, the weighting coefficient in this receiver can be obtained from Equation (5.255) by assuming that the correlation coefficient is zero and neglecting the reduction of the received signal conditional mean caused by interference. The DSC coefficients are then given by:

$$p_{i}^{i,k} = \frac{(\sigma_{c}^{i,k-1})^2}{(\sigma_{c}^{i,k-1})^2 + (\sigma_{i}^{i,k})^2}$$

The DSC output, in the second and higher iterations, with coefficients from Equation (5.256) can be expressed as:

$$y_{c}^{i,k} = \frac{(\sigma_{c}^{i,k-1})^2 1}{(\sigma_{c}^{i,k-1})^2 + (\sigma_{i}^{i,k})^2} y_{c}^{i,k} + \frac{(\sigma_{c}^{i,k})^2}{(\sigma_{i}^{i,k})^2 + (\sigma_{c}^{i,k-1})^2} y_{c}^{i,k-1}$$

The complexity of both PIC-STD and PIC-DSC is linear in the number of transmit antennas. We demonstrate the performance of an HLST with separate $R = 1/2$, 4-state convolutional component encoders, the frame size of $L = 206$ symbols and BPSK modulation. In simulations decoding is performed by a MAP algorithm. The HLSTC with $n_T$ transmit and $n_R$ receive antennas is denoted as an $(n_T, n_R)$ HI-STC. The channel is modelled as a frequency flat slow Rayleigh fading channel and the results are shown in the form of the frame error rate (FER) versus $E_b/N_0$. The SNR is related to $E_b/N_0$ as $SNR = \eta E_b/N_0$, where $\eta = Rmn_{\text{T}}$ is the spectral efficiency and $m$ is the number of bits per modulation symbol. Figure 5.37 compares the performance of the PIC-STD with EIR and LLR based symbol estimates and the PIC-DSC for a (6,2) HLSTC. The spectral efficiency of the HI-STC is $\eta = 3$ bits/s/Hz. The results show that for the PIC-STD with LLR based symbol estimates the error floor is higher than for the other two schemes. With $l = 8$ iterations, the error floor for the PIC-STD(LLR) appears at FER of 0.1, while for the PIC-STD (EIR) the error floor is about 0.04. However, the PIC-DSC receiver has an error floor below 0.007.
ITERATIVE RECEIVERS FOR LAYERED SPACE–TIME CODING

Figure 5.37 PER performance of $n_T = 6$, $n_R = 2$, $R = 1/2$, BPSK, a PIC-STD and PIC-DSC detection on a slow Rayleigh fading channel.

Figure 5.38 Block diagram of an iterative F/B MMSE receiver.

5.6.5.2 Iterative F/B MMSE receiver

We consider an iterative receiver with a multiuser detector consisting of a feed-forward (F) module which performs interference suppression followed by a feedback (B) module which performs parallel interference cancellation, as proposed in [144]. We refer to this receiver structure, presented in Figure 5.38, as an iterative F/B MMSE receiver. The decision statistics vector obtained at the output of
the feedback module the $k$th iteration at time $t$, for layer $i$, denoted by $y_{i,k}^{i,k}$, is given by $y_{i,k}^{i,k} = (w_f^{i,k})^H r + w_b^{i,k}$, where $w_f^{i,k}$ is an $n_R \times 1$ optimized feed-forward coefficients column matrix and $w_b^{i,k}$ is a single coefficient which represents cancellation term.

The coefficients $w_f^{i,k}$ and $w_b^{i,k}$ are calculated by minimizing the mean square error between the transmitted symbol and its estimate, given by:

$$ e = E \left\{ \left| (w_f^{i,k})^H r + w_b^{i,k} - x_i^t \right|^2 \right\} $$.  

(5.258)

Let us denote by $\mathbf{x}^t$ a column matrix with $(n_T - 1)$ components, consisting of the transmitted symbols from all antennas except antenna $i$ ($\mathbf{x}^t = (x_1^t, x_2^t, \ldots, x_i^{t-1}, x_i^{t+1}, \ldots, x_{nT}^t)$).

Similarly, we define a vector $\hat{\mathbf{x}}_i^{t,k}$ of the symbol estimates from other antennas in the $k$th iteration ($\hat{\mathbf{x}}_i^{t,k})^T = (\hat{x}_1^{t,k}, \hat{x}_2^{t,k}, \ldots, \hat{x}_i^{k-1,t}, \hat{x}_i^{k+1,t}, \ldots, \hat{x}_{nT}^{k,t})$) The decoder calculates the LLRs for the transmitted symbols at a particular time instant for each transmit antenna. These LLR values are used to calculate the transmitted symbol estimates $\hat{x}_i^{t,k}$, $l = 1, 2, \ldots, i - 1, i + 1, \ldots, n_T$ in $(\hat{\mathbf{x}}_i^{k})^T$ as before.

Let us denote by $h$, the $i$th of the channel matrix $H$, representing a column matrix with $n_R$ complex channel gains for the $i$th transmit antenna, and by $\mathbf{H}$ an $n_R \times (n_T - 1)$ matrix composed of the complex channel gains for the other ($n_T - 1$) transmit antennas. To simplify the notation, we define the following matrices:

$$ A = h \cdot h^H $$

$$ B = \mathbf{H} \left[ \mathbf{I}_{n_T - 1} - \text{diag}((\hat{\mathbf{x}}_i^{t,k})^H (\hat{\mathbf{x}}_i^{t,k})^H + \mathbf{S}_i^{t,k} (\mathbf{S}_i^{t,k})^H) \right] (\mathbf{H})^H $$

(5.259)

where $\mathbf{I}_{n_T - 1}$ is an $(n_T - 1) \times (n_T - 1)$ identity matrix, and

$$ D = \mathbf{H} \hat{\mathbf{x}}_i^{t,k} \mathbf{R}_n = \sigma^2 \mathbf{I}_{n_R} $$

(5.260)

where $\sigma^2$ is the noise variance. The optimum feed-forward and feedback coefficients are given by

$$ (w_f^{i,k})^H = h^H (A + B + \mathbf{R}_n - DD^H) $$

$$ w_b^{i,k} = - (w_f^{i,k})^H D $$

(5.261)

The MMSE coefficients were derived assuming perfect interleaving and feedback symbol estimates based on the extrinsic information ratio (EIR) [158]. In the first iteration, since the $a priori$ probabilities of the transmitted symbols are the same, the symbol estimates $\hat{x}_i^{t,1}$ are zeros. Thus in the first iteration the feed-forward coefficients $W_f^{i,1}$ are obtained in a similar way as in Equation (5.242) and the feedback coefficient $w_b^{i,1}$ = 0. In the second and higher iterations, the symbol estimates, computed by the decoder as in Equation (5.249), are used to recalculate the new set of feed-forward and feedback coefficients as described above. In the case of hard decision decoding $(\hat{x}_i^{t,k})^2 = 1$ for all $i$ and $k > 1$. The iterative MMSE receiver, which employs hard decision decoders, is equivalent to the receiver that performs linear MMSE suppression in the first iteration and parallel interference cancellation in the following iterations. This filter would be optimal in the MMSE sense if perfect symbol estimates were fed back.

The $a priori$ probability on the transmitted signal $x_i^t$, used in the decoder, is computed as:

$$ p_i (x_i^t = 1) = \frac{1}{\sqrt{2 \pi \sigma}} e^{- \frac{(y_i^t - l)^2}{2 \sigma^2}} \quad l = 1, -1 $$

(5.262)

It has been observed that the iterative MMSE receiver performs better if LLRs are used for symbol estimation instead of EIRs, though the MMSE filter coefficients were derived assuming EIR symbol estimation and uncorrelated decoder outputs. If LLRs are used there is a bias between symbol estimates. However, the bias effect is less relevant in the iterative MMSE receiver than in the iterative PIC receiver, since the MMSE detector performs interference suppression as well as cancellation. Thus the use of DSC in iterative MMSE receivers is less effective than for iterative PIC receivers.

Figure 5.39 compares the iterative MMSE and iterative PIC-DSC performance for an (8,2) HLSTC. The results show that the iterative PIC-DSC outperforms the iterative MMSE in terms of the achieved
FER after two iterations. The error floor in the FBR performance of the MMSE-STD appears at $E_b/N_0 = 13$ dB and for $FER = 0.1$, while for the PIC-DSC receiver the error floor appears at $E_b/N_0 = 15$ dB and for $FER = 0.03$.

5.7 APPENDIX 5.1 LINEAR AND MATRIX ALGEBRA

5.7.1 Definitions

Consider an $m \times n$ matrix $R$ with elements $r_{ij}, i = 1, 2, \ldots, m; j = 1, 2, \ldots, n$. A shorthand notation for describing $R$ is

$[R]_{ij} = r_{ij}$

The transpose of $R$, which is denoted by $R^T$, is defined as the $n \times m$ matrix with elements $r_{ji}$, or

$[R^T]_{ij} = r_{ji}$

A square matrix is one for which $m = n$. A square matrix is symmetric if $R^T = R$. The rank of a matrix is the number of linearly independent rows or columns, whichever is less. The inverse of a square $n \times n$ matrix is the square $n \times n$ matrix $R^{-1}$ for which

$R^{-1}R = RR^{-1} = I$

where $I$ is the $n \times n$ identity matrix. The inverse will exist if and only if the rank of $R$ is $n$. If the inverse does not exist, then $R$ is singular. The determinant of a square $n \times n$ matrix is denoted by $\det(R)$. It is computed as

$\det(R) = \sum_{j=1}^{n} r_{ij} C_{ij}$
where

\[ C_{ij} = (-1)^{i+j} M_{ij} \]

\( M_{ij} \) is the determinant of the submatrix of \( R \) obtained by deleting the \( i \)th row and \( j \)th column and is termed the \textit{minor} of \( r_{ij} \). \( C_{ij} \) is the cofactor of \( r_{ij} \). Note that any choice of \( i \) for \( i = 1, 2, \ldots, n \) will yield the same value for \( \det(R) \).

A \textit{quadratic form} \( Q \) is defined as

\[ Q = \sum_{i=1}^{n} \sum_{j=1}^{n} r_{ij} x_i x_j \]

In defining the quadratic form it is assumed that \( r_{ji} = r_{ij} \). This entails no loss in generality since any quadratic function may be expressed in this manner. \( Q \) may also be expressed as

\[ Q = x^T R x \]

where \( x = [x_1, x_2, \ldots, x_n]^T \) and \( R \) is a square \( n \times n \) matrix with \( r_{ji} = r_{ij} \) or \( R \) is a symmetric matrix.

A square \( n \times n \) matrix \( R \) is \textit{positive semidefinite} if \( R \) is symmetric and \( x^T R x \geq 0 \) for all \( x \neq 0 \). If the quadratic form is strictly positive, then \( R \) is \textit{positive definite}. When referring to a matrix as \textit{positive definite} or positive semidefinite, it is always assumed that the matrix is symmetric.

The \textit{trace} of a square \( n \times n \) matrix is the sum of its diagonal elements or

\[ \text{tr}(R) = \sum_{i=1}^{n} r_{ii} \]

A \textit{partitioned} \( m \times n \) matrix \( R \) is one that is expressed in terms of its submatrices. An example is the \( 2 \times 2 \) partitioning

\[ R = \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix} \]

Each ‘element’ \( R_{ij} \) is a submatrix of \( R \). The dimensions of the partitions are given as

\[ \begin{bmatrix} k \times l & k \times (n-l) \\ (m-k) \times l & (m-k) \times (n-l) \end{bmatrix} \]

### 5.7.2 Special Matrices

A \textit{diagonal} matrix is a square \( n \times n \) matrix with \( r_{ij} = 0 \) for \( i \neq j \), in other words, all elements off the principal diagonal are zero. A diagonal matrix appears as:

\[ R = \begin{bmatrix} r_{11} & 0 & \cdots & 0 \\ 0 & r_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & r_{nn} \end{bmatrix} \]

A diagonal matrix will sometimes be denoted by \( \text{diag}(r_{11}, r_{22}, \ldots, r_{nn}) \). The inverse of a diagonal matrix is found by simply inverting each element on the principal diagonal. A generalization of the diagonal matrix is the square \( n \times n \) block diagonal matrix

\[ R = \begin{bmatrix} R_{11} & 0 & \cdots & 0 \\ 0 & R_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & R_{kk} \end{bmatrix} \]
in which all submatrices \( \mathbf{R}_{ii} \) are square and the other submatrices are identically zero. The dimensions of the submatrices need not be identical. For instance, if \( k = 2 \), \( \mathbf{R}_{11} \) might have dimension \( 2 \times 2 \) while \( \mathbf{R}_{22} \) might be a scalar. If all \( \mathbf{R}_{ii} \) are nonsingular, then the inverse is easily found as:

\[
\mathbf{R}^{-1} = \begin{bmatrix}
\mathbf{R}^{-1}_{11} & 0 & \ldots & 0 \\
0 & \mathbf{R}^{-1}_{22} & \ldots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \ldots & \mathbf{R}^{-1}_{kk}
\end{bmatrix}
\]

Also, the determinant is:

\[
\det(\mathbf{R}) = \prod_{i=1}^{n} \det(\mathbf{R}_{ii})
\]

A square \( n \times n \) matrix is orthogonal if

\[
\mathbf{R}^{-1} = \mathbf{R}^T
\]

For a matrix to be orthogonal the columns (and rows) must be orthonormal or, if

\[
\mathbf{R} = [\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_n]
\]

where \( \mathbf{r}_i \) denotes the \( i \)th column, the conditions

\[
\mathbf{r}_i^T \mathbf{r}_j = \begin{cases} 
0 & \text{for } i \neq j \\
1 & \text{for } i = j
\end{cases}
\]

must be satisfied.

An idempotent matrix is a square \( n \times n \) matrix which satisfies

\[
\mathbf{R}^2 = \mathbf{R}
\]

This condition implies that \( \mathbf{R}^l = \mathbf{R} \) for \( l \geq 1 \). An example is the projection matrix

\[
\mathbf{R} = \mathbf{H} (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T
\]

where \( \mathbf{H} \) is an \( m \times n \) full rank matrix with \( m > n \).

A square \( n \times n \) Toeplitz matrix is defined as

\[
[R]_{ij} = r_{i-j}
\]

or

\[
\mathbf{R} = \begin{bmatrix}
r_0 & r_{-1} & r_{-2} & \cdots & r_{-(n-1)} \\
r_1 & r_0 & r_{-1} & \cdots & r_{-(n-2)} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
r_{n-2} & r_{n-3} & \cdots & r_0 & r_{-1} \\
r_{n-1} & r_{n-2} & r_{n-3} & \cdots & r_0
\end{bmatrix}
\]

Each element along a northwest–southeast diagonal is the same. If, in addition, \( r_{-k} = r_k \), then \( \mathbf{R} \) is symmetric Toeplitz.

### 5.7.3 Matrix manipulation and formulas

Some useful formulas for the algebraic manipulation of matrices are summarized in this section. For \( n \times n \) matrices \( \mathbf{R} \) and \( \mathbf{P} \) the following relationships are useful.

\[
(\mathbf{RP})^T = \mathbf{P}^T \mathbf{R}^T
\]

\[
(\mathbf{R}^T)^{-1} = (\mathbf{R}^{-1})^T
\]

\[
(\mathbf{RP})^{-1} = \mathbf{P}^{-1} \mathbf{R}^{-1}
\]

\[
\det(\mathbf{R}^T) = \det(\mathbf{R})
\]
The extension of these properties to arbitrary partitioning is straightforward. Determination of the inverses and determinants of partitioned matrices is facilitated by employing the following formulas. 

\[
\det(cR) = c^n \det(R) \quad (c \text{ a scalar})
\]

\[
\det(RP) = \det(R) \det(P)
\]

\[
\det(R^{-1}) = \frac{1}{\det(R)}
\]

\[
\text{tr}(RP) = \text{tr}(PR)
\]

\[
\text{tr}(R^T P) = \sum_{i=1}^{n} \sum_{j=1}^{n} [R]_{ij} [P]_{ij}
\]

For vectors \( x \) and \( y \) we have

\[
y^T x = \text{tr}(xy^T)
\]

It is frequently necessary to determine the inverse of a matrix analytically. To do so one can make use of the following formula. The inverse of a square \( n \times n \) matrix is

\[
R^{-1} = \frac{C^T}{\det(R)}
\]

where \( C \) is the square \( n \times n \) matrix of cofactors \( R \). The cofactor matrix is defined by

\[
[C]_{ij} = (-1)^{i+j} M_{ij}
\]

where \( M_{ij} \) is the minor of \( r_{ij} \) obtained by deleting the \( i \)th row and \( j \)th column of \( R \).

Another formula which is quite useful is the matrix inversion lemma

\[
(R + PCD)^{-1} = R^{-1} - R^{-1}P(DR^{-1}P + C^{-1})^{-1}DR^{-1}
\]

where it is assumed that \( R \) is \( n \times n \), \( P \) is \( n \times m \), \( C \) is \( m \times m \), and \( D \) is \( m \times n \) and that the indicated inverses exist. A special case known as Woodbury’s identity results when \( P \) is an \( n \times 1 \) column vector \( u \), \( C \) a scalar of unity, and \( D \) a \( 1 \times n \) row vector \( u^T \). Then

\[
(R + uu^T)^{-1} = R^{-1} - \frac{R^{-1}uu^TR^{-1}}{1 + uu^TR^{-1}u}
\]

Partitioned matrices may be manipulated according to the usual rules of matrix algebra by considering each submatrix as an element. For multiplication of partitioned matrices, the submatrices which are multiplied together must be conformable. As an illustration, for \( 2 \times 2 \) partitioned matrices

\[
RP = \begin{bmatrix}
R_{11} & R_{12} \\
R_{21} & R_{22}
\end{bmatrix}
\begin{bmatrix}
P_{11} & P_{12} \\
P_{21} & P_{22}
\end{bmatrix}
= \begin{bmatrix}
R_{11}P_{11} + R_{12}P_{21} & R_{11}P_{12} + R_{12}P_{22} \\
R_{21}P_{11} + R_{22}P_{21} & R_{21}P_{12} + R_{22}P_{22}
\end{bmatrix}
\]

The transposition of a partitioned matrix is formed by transposing the submatrices of the matrix and applying \( T \) to each submatrix. For a \( 2 \times 2 \) partitioned matrix

\[
\begin{bmatrix}
R_{11} & R_{12} \\
R_{21} & R_{22}
\end{bmatrix}^T
= \begin{bmatrix}
P_{11}^T & P_{12}^T \\
P_{21}^T & P_{22}^T
\end{bmatrix}
\]
and

\[
\text{det}(R) = \text{det}(R_{22}) \text{det}(R_{11} - R_{12} R_{22}^{-1} R_{21}) \\
= \text{det}(R_{11}) \text{det}(R_{22} - R_{21} R_{11}^{-1} R_{12})
\]

where the inverses of \(R_{11}\) and \(R_{22}\) are assumed to exist.

### 5.7.4 Theorems

Some important theorems are summarized in this section.

1. A square \(n \times n\) matrix \(R\) is invertible (non-singular) if and only if its columns (or rows) are linearly independent or, equivalently, if its determinant is non-zero. In such a case, \(R\) is full rank. Otherwise, it is singular.

2. A square \(n \times n\) matrix \(R\) is positive definite if and only if
   - it can be written as
     \[
     R = CC^T
     \]
   where \(C\) is also \(n \times n\) and is full rank and hence invertible, or
   - the principal minors are all positive. (The \(i^{th}\) principal minor is the determinant of the submatrix formed by deleting all rows and columns with an index greater than \(i\)). If \(R\) can be written as in the previous equation, but \(C\) is not full rank or the principal minors are only non-negative, then \(R\) is positive semidefinite.

3. If \(R\) is positive definite, then the inverse exists and may be found from the previous equation as
   \[
   R^{-1} = (C^{-1})^T (C^{-1})
   \]

4. Let \(R\) be positive definite. If \(P\) is an \(m \times n\) matrix of full rank with \(m \leq n\), then \(PRP^T\) is also positive definite.

5. If \(R\) is positive definite (positive semidefinite), then
   - the diagonal elements are positive (non-negative)
   - the determinant of \(R\), which is a principal minor, is positive (non-negative).

### 5.7.5 Eigendecomposition of matrices

An eigenvector of a square \(n \times n\) matrix \(R\) is an \(n \times 1\) vector \(v\) satisfying

\[
Rv = \lambda v
\]

for some scalar \(\lambda\), which may be complex. \(\lambda\) is the eigenvalue of \(R\) corresponding to the eigenvector \(v\). It is assumed that the eigenvector is normalized to have unit length or \(v^T v = 1\). If \(R\) is symmetric, then one can always find \(n\) linearly independent eigenvectors, although they will not in general be unique. An example is the identity matrix for which any vector is an eigenvector with eigenvalue 1. If \(R\) is symmetric, then the eigenvectors corresponding to distinct eigenvalues are orthonormal or \(v_i^T v_j = \delta_{ij}\) and the eigenvalues are real. If, furthermore, the matrix is positive definite (positive semidefinite), then the eigenvalues are positive (non-negative). For a positive semidefinite matrix the rank is equal to the number of non-zero eigenvalues.

The defining previous relation can also be written as

\[
R[v_1 \ v_2 \ \cdots \ v_n] = [\lambda_1 v_1 \ \lambda_2 v_2 \ \cdots \ \lambda_n v_n]
\]

or

\[
RV = V\Lambda
\]
where

\[
V = [v_1 \ v_2 \ \cdots \ v_n]
\]

\[
\Lambda = \text{diag}(\lambda_1, \lambda_2, \cdots, \lambda_n)
\]

If \( R \) is symmetric so that the eigenvectors corresponding to distinct eigenvalues are orthonormal and the remaining eigenvectors are chosen to yield an orthonormal eigenvector set, \( V \) is an orthonormal matrix. As such, its inverse is \( V^T \), so that the previous equation becomes

\[
R = V \Lambda V^T = \sum_{i=1}^{n} \lambda_i v_i v_i^T
\]

Also, the inverse is easily determined as

\[
R^{-1} = V^{-1} \Lambda^{-1} V^{-1} = \sum_{i=1}^{n} \frac{1}{\lambda_i} v_i v_i^T
\]

A final useful relationship follows as

\[
\det (R) = \det (V) \det(\Lambda) \det(V^{-1}) = \det(\Lambda) = \prod_{i=1}^{n} \lambda_i
\]

### 5.7.6 Calculation of eigenvalues and eigenvectors

We can write Equation (A5.1) as

\[
Rv - \lambda v = 0 \Rightarrow (R - \lambda I)v = 0
\]

These are \( n \) linear algebraic equations in the \( n \) unknowns \( v_1, v_2, v_3, \ldots v_n \) (the components of \( v \)). For these equations to have a solution \( v \neq 0 \), the determinant of the coefficient matrix \( R - \lambda I \) must be zero. As an example, if \( n = 2 \), we have

\[
\begin{bmatrix}
    r_{11} - \lambda & r_{12} \\
    r_{21} & r_{22} - \lambda
\end{bmatrix}
\begin{bmatrix}
v_1 \\
v_2
\end{bmatrix}
= \begin{bmatrix}
0 \\
0
\end{bmatrix}
\]

which can be written as

\[
(r_{11} - \lambda)v_1 + r_{12}v_2 = 0 \\
r_{21}v_1 + (r_{22} - \lambda)v_2 = 0
\]

(A5.2)

Now, \( R - \lambda I \) is singular if and only if its determinant \( \det(R - \lambda I) \) is zero. This can be written as

\[
\det(R - \lambda I) = \begin{vmatrix}
    r_{11} - \lambda & r_{12} \\
    r_{21}r_{22} - \lambda
\end{vmatrix} = (r_{11} - \lambda)(r_{22} - \lambda) - r_{12}r_{21}
\]

\[
= \lambda^2 - (r_{11} + r_{22})\lambda + r_{11}r_{22} - r_{12}r_{21} = 0
\]

This quadratic equation in \( \lambda \) is called the characteristic equation of \( R \). Its solutions are the eigenvalues \( \lambda_1 \) and \( \lambda_2 \) of \( R \). First determine these. Then use Equation (A5.2) with \( \lambda = \lambda_1 \) to determine the eigenvector \( v_1 \) of \( R \) corresponding to \( \lambda_1 \). Finally, use Equation (A5.2) with \( \lambda = \lambda_2 \) to find an eigenvector \( v_2 \) of \( R \) corresponding to \( \lambda_2 \). Note that if \( v \) is an eigenvector of \( R \) so is \( k v \) for any \( k \neq 0 \). As an example, suppose that

\[
R = \begin{bmatrix}
-4.0 & 4.0 \\
1.6 & 1.2
\end{bmatrix}
\]

then we have

\[
\det(R - \lambda I) = \begin{vmatrix}
    -4 - \lambda & 4 \\
    -1.6 & 1.2 - \lambda
\end{vmatrix} = \lambda^2 + 2.8\lambda + 1.6 = 0
\]
It has the solutions $\lambda_1 = -2$ and $\lambda_2 = -0.8$. These are the eigenvalues of $R$. Eigenvectors are obtained from Equation (A5.2). For $\lambda = \lambda_1 = -2$ we have, from Equation (A5.2):

\[
\begin{align*}
(-4.0 + 2.0)v_1 + 4.0v_2 &= 0 \\
-1.6v_1 + (1.2 + 2.0)v_2 &= 0
\end{align*}
\]

A solution of the equation is $v_1 = 2$, $v_2 = 1$. Hence, an eigenvector of $R$ corresponding to $\lambda = \lambda_1 = -2$ is

\[
v_1 = \begin{bmatrix} 2 \\ 1 \end{bmatrix}
\]

Similarly, for $\lambda = \lambda_2 = -0.8$

\[
v_2 = \begin{bmatrix} 1 \\ 0.8 \end{bmatrix}
\]

So, for this example:

\[
V = [v_1 \ v_2] = \begin{bmatrix} 2 & 1 \\ 1 & 0.8 \end{bmatrix}.
\]

REFERENCES


3. Mezger, K. and Bouwens, R. J. (1972) An Ordered Table of Primitive Polynomials over GF(2) of Degrees 2 Through 19 for Use with Linear Maximal Sequence Generators, TM107, Cooley Electronics Laboratory, University of Michigan, Ann Arbor (AD 746876).


REFERENCES


REFERENCES


REFERENCES


REFERENCES


Channel Estimation and Equalization

The basic concept of time division multiple access (TDMA) has been discussed in Chapter 1. Within this chapter we cover the basic enabling technologies for TDMA. Coding and modulation are covered in Chapters 2, 3 and 4 so that in this chapter we focus on the remaining topics, mainly TDMA-specific channel estimation and equalization. MIMO channel equalization will be discussed in Chapter 9 within the general problem of linear precoding.

6.1 EQUALIZATION IN THE DIGITAL DATA TRANSMISSION SYSTEM

6.1.1 Zero-forcing equalizers

The basic problem of channel equalization is illustrated in Figure 6.1. Figure 6.1(a) presents the transmitted pulse \( p(t) \), 6.1(b) shows the pulse after propagation through the channel \( p_c(t) \), while Figure 6.1(c) shows the difference between the received \( p_c(t) \) and the equalized pulse \( p_{eq}(t) \approx p(t) \).

Figure 6.2 illustrates the mutual impact (inter-symbol interference) of the non-equalized and equalized signals. Finally, Figure 6.3 illustrates a general structure and function of the equalizer. To summarize, the transmitted signal for full response signaling is created in such a way that the pulse goes through zero at the time instances \( \pm kT \), \( k \neq 0 \) from the pulse maximum (Nyquist signaling) \( p(\pm kT) = 0 \), \( k \neq 0 \). That way the adjacent symbols sampled at those instances will not be affected. The degradation caused by the channel will result in the pulse \( p_c(\pm kT) \neq 0 \), \( k \neq 0 \) which produces inter-symbol interference. The equalizer is supposed to compensate for this degradation by regenerating a pulse \( p_{eq}(\pm kT) \equiv p(\pm kT) = 0 \), \( k \neq 0 \).

This is represented by

\[
p_{eq}(t) = \sum_{n=-N}^{N} C_n p_c(t - nT)
\]
Figure 6.1 (a) Transmitted; (b) received; and (c) equalized pulses.
Figure 6.2 (a), (b) Transmitted signals and (c), (d) received signals.
Figure 6.2 (Cont.)

Figure 6.3 Transversal filter equalizer.

for $t = mT + \Delta t$

$$p_{eq}(mT + \Delta t) = \sum_{n=-N}^{N} C_n p_c((m-n)T + \Delta t)$$

where $t = \Delta t$ is the sampling time for which $p_{eq}(t)$ is maximal. Equation (6.1) is implemented by the structure shown in Figure 6.3 with the basic building block shown in Figure 6.4. From time to time, throughout the rest of the book, we will explicitly emphasize the importance of this basic building
Figure 6.4 Transversal filter equalizer basic building block.

block in order to be able to motivate some basic approaches in building up a common reconfigurable platform for different technologies.

Equation (6.1) for $m = 0, \pm 1, \pm 2, \ldots, \pm N$ can be written in matrix form as $P_{eq} = P_c C$, where $P_{eq}$ and $C$ are vectors or column matrices given by

$$P_{eq} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \\ \vdots \\ 0 \\ \vdots \\ 0 \\ N \text{ zeros} \end{bmatrix}, \quad C = \begin{bmatrix} C_{-N} \\ C_{-N+1} \\ \vdots \\ C_{0} \\ C_{1} \\ \vdots \\ C_{N} \end{bmatrix}$$

(6.2)

and $P_c$ is the $(2N + 1) \times (2N + 1)$ matrix of channel responses of the form

$$P_c = \begin{bmatrix} p_c(0) & p_c(-1) & \ldots & p_c(-2N) \\ p_c(1) & p_c(0) & \ldots & p_c(-2N + 1) \\ p_c(2) & p_c(1) & \ldots & p_c(-2N + 2) \\ \vdots & \vdots & \vdots & \vdots \\ p_c(2N) & p_c(2N - 1) & \ldots & p_c(0) \end{bmatrix}$$

(6.3)

The solution for the equalizer coefficients is

$$C = P_c^{-1} P_{eq}$$

(6.4)

given $P_c^{-1} \neq 0$. This way we force the $P_{eq}(\pm kT) = 0$, hence the name zero-forcing equalizer. One should be aware that in a channel with noise $P_c \Rightarrow P_c + P_n$, where $P_n$ is the noise matrix given by (6.3) where channel samples are replaced with corresponding noise samples $p_c(\cdot) \Rightarrow p_n(\cdot)$. Equation (6.4) now results in

$$P_{eq} = P_c C + P_n C = P_c C + N_{eq}$$

(6.5)

$$C = P_c^{-1} P_{eq} - P_c^{-1} N_{eq}$$

(6.6)

The second term represents the noise enhancement factor and is a limiting factor in achieving good performance.
6.1.1.1 Example

Suppose that the sample values for the channel response are

\[
p_c(-5) = 0.01 \quad p_c(-4) = -0.02 \quad p_c(-3) = 0.05 \\
p_c(-2) = -0.1 \quad p_c(-1) = 0.2 \quad p_c(0) = 1 \\
p_c(1) = -0.1 \quad p_c(2) = 0.1 \quad p_c(3) = -0.05 \\
p_c(4) = 0.02 \quad p_c(5) = 0.005
\]

The channel response matrix is

\[
P_c = \begin{bmatrix} 1.0 & 0.2 & -0.1 & 0.05 & -0.02 \\ -0.1 & 1.0 & 0.2 & -0.1 & 0.05 \\ 0.1 & -0.1 & 1.0 & 0.2 & -0.1 \\ -0.05 & 0.1 & -0.1 & 1.0 & 0.2 \\ 0.02 & -0.05 & 0.1 & -0.1 & 1.0 \end{bmatrix}
\]

and its inverse

\[
P_c^{-1} = \begin{bmatrix} 0.996 & -0.170 & 0.117 & -0.083 & 0.056 \\ 0.118 & 0.945 & -0.158 & 0.112 & -0.083 \\ -0.091 & 0.133 & 0.937 & -0.158 & 0.117 \\ 0.028 & -0.095 & 0.133 & 0.945 & -0.170 \\ -0.002 & 0.028 & -0.091 & 0.118 & 0.966 \end{bmatrix}
\]

The coefficient vector is the center column of \(P_c^{-1}\). Therefore,

\[
C_{-2} = 0.117 \quad C_{-1} = -0.158 \quad C_0 = 0.937 \\
C_1 = 0.133 \quad C_2 = -0.091
\]

The sample values of the equalized pulse response are given as

\[
p_{eq}(m) = \sum_{n=-2}^{2} C_n p_c(m - n)
\]

where \(\Delta t = 0\). For this example we have,

\[
p_{eq}(0) = (0.117)(0.1) + (-0.158)(-0.1) + (0.937)(1) \\
\quad \quad \quad + (0.133)(0.2) + (-0.091)(-0.1) \\
= 1.0
\]

which checks with the desired value of unity. Similarly, it can be verified that \(p_{eq}(-2) = p_{eq}(-1) = p_{eq}(1) = p_{eq}(2) = 0\).

Values of \(p_{eq}(n)\) for \(n < -2\) or \(n < 2\) are not zero. For example,

\[
p_{eq}(3) = (0.117)(0.005) + (-0.158)(0.02) + (0.937)(-0.05) \\
\quad \quad \quad + (0.133)(0.1) + (-0.091)(-0.1) \\
= -0.027
\]

\[
p_{eq}(-3) = (0.117)(0.2) + (-0.158)(-0.1) + (0.937)(0.05) \\
\quad \quad \quad + (0.133)(-0.02) + (-0.091)(0.01) \\
= 0.082
\]
6.2 LMS EQUALIZER

Let us now, in the next iteration, revisit the same problem of the channel equalization by introducing more details. We will be dealing with a QPSK signal and equalizer, as shown in Figure 6.5.

6.2.1 Signal model

The transmitted QPSK signal has the form

\[ s_{\text{tr}}(t) = d_1(t) \cos \omega_0 t - d_2(t) \sin \omega_0 t \]  

(6.7)

The received signal can be represented as

\[ y(t) = s_{\text{rec}}(t) + n(t) \]

\[ = s_\text{tr}(t) + \beta s_{\text{tr}}(t - \tau_m) + n_c(t) \cos(\omega_0 t + \alpha) - n_s(t) \sin(\omega_0 t + \alpha) \]  

(6.7a)

We define the desired non-distorted complex signal at the receiver as

\[ D(t) = (1 + \beta) [d_1(t) + j d_2(t)] \]  

(6.8)

The cost function for the MMSE equalizer is defined as

\[ C = E[|d_c(t) - D(t)|^2] \]  

(6.9)

If we define the complex vectors of equalizer coefficients \( W \) and LPF output signal samples \( Z \) (see Figure 6.5)

\[ W = \begin{bmatrix} W_1 \\ W_2 \\ \vdots \\ W_N \end{bmatrix} \quad Z = \begin{bmatrix} z(t) \\ z(t - \Delta) \\ \vdots \\ z(t - (N-1)\Delta) \end{bmatrix} \]  

(6.10)

then the cost function becomes

\[ C = E[|W^T Z - D(t)|^2] \]  

(6.11)
By setting the gradient to zero we get
\[ \mathbf{W}_{\text{opt}} = \left[ \mathbf{E} \left( \mathbf{Z} \mathbf{Z}^* \right) \right]^{-1} \mathbf{E} \left( \mathbf{Z} \mathbf{D}^*(t) \right) \]

A sample of performance results is shown in Figure 6.6.

### 6.2.2 Adaptive weight adjustment

Setting the tap coefficients of the zero-forcing and LMS equalizers involves the solutions of a set of simultaneous equations. In the case of the zero-forcing equalizer, adjustment of the tap coefficients involves measuring the channel filter output at \( T \) second spaced sampling times in response to a test pulse, and solving for the tap gains. In the case of the LMS equalizer, these equations involve data, noise and multipath dependent parameters, which may be difficult to determine or may not be known at all. Known data assumes a kind of preamble (midamble) embedded into the data.

### 6.2.3 Automatic systems

So-called preset algorithms, use a training sequence. In adaptive algorithms, adjustment of the coefficients is performed continuously during data transmission.

A zero-forcing equalizer can be solved iteratively for the coefficient vector \( \mathbf{C} \). Let \( \mathbf{C} \) at the \( k \)th iteration be \( \mathbf{C}^{(k)} \), then the error in the solution is
\[ \mathbf{E}^{(k)} = \mathbf{P}_c \mathbf{C}^{(k)} - \mathbf{P}_{eq} \]  
\[ (6.12) \]

where \( \mathbf{E}^{(k)} \) is a vector with \( 2N + 1 \) components, each of which represent the error in a component of \( \mathbf{C}^{(k)} \). Each component of \( \mathbf{C}^{(k)} \) can be adjusted in accordance with the error in it.
6.2.4 Iterative algorithm

If \( A \) is a small positive constant, the adjustment algorithm for the \( j \)th component of \( C^{(k)} \) is

\[
C_j^{(k+1)} = C_j^{(k)} - Asgn\left(E_j^{(k)}\right)
\]

(6.13)

The iteration process is continued until \( C_j^{(k+1)} \) and \( C_j^{(k)} \) differ by some suitable small increment. The algorithm converges under fairly broad restrictions.

6.2.5 The LMS algorithm

The gradient of the mean square error with respect to the \( j \)th tap gain is twice the correlation between the equalizer output and the error between actual and desired outputs. Two problems arise in applying it to adaptive weight adjustment. First, it requires the expectation or average to be taken. Since this is not available, the unbiased but noisy estimate \( z^*(t - j\Delta) e(t) \) can be used. Secondly, the undistorted output \( D(t) \) is not available unless a known data sequence is transmitted.

An alternative to sending a known data sequence is to assume that the detected data is correct (which is true even in a fairly bad channel giving an error probability of only \( 10^{-2} \)) and using the detected data to reconstruct an estimate of the undistorted output \( D(t) \). Equalizers using this method of data estimation are called decision-directed.

A suitable decision-directed algorithm for weight adjustment of the LMS equalizer is

\[
W(n + 1) = W(n) - A Z^T \left[ d_e(t_n - \Delta) - D_d(t_n) \right]
\]

(6.14)

More details on LMS-based equalizers can be found in [1–7].

6.2.6 Decision feedback equalizer (DFE)

Non-linear equalizer structures may provide better performance under many circumstances. A simple non-linear equalizer is the decision feedback equalizer (DFE) which uses feedback of decisions on symbols already received to cancel the interference from symbols which have already been detected.

The basic idea is that, assuming past decisions are correct, the ISI contributed by these symbols can be canceled exactly by subtracting appropriately weighted past symbol values from the equalizer output. This is the purpose of the delay line with weights \( b_1, b_2, \ldots, b_M \) shown in Figure 6.7. The forward transversal filter, with weights \( c_0, c_1, \ldots, c_{N-1} \), then compensates for ISI over a smaller

![Figure 6.7 Decision feedback equalizer.](image-url)
portion of the ISI-contaminated received signal. Both the feedback and feedforward coefficients can be adjusted simultaneously to minimize the mean square error.

Decision feedback equalizers are discussed in [8–41].

### 6.2.7 Blind equalizers

What we need for designing a blind equalizer is to recover the transmitted message \((a_t)\) from the received one \((x_t)\) only, without any preamble for identification of the unknown channel.

The equivalent system models are given in Figure 6.8 and Figure 6.9. The cost function defined by Equation (6.11) is now modified in such a way that it does not require knowledge of data. Options are presented by Equations (6.15) and (6.16). The remaining details are the same as in the previous discussion.

\[
J(W) = E \left( \frac{1}{2} c_t^2(W) - \alpha |c_t(W)| \right) \tag{6.15}
\]

\[
\alpha = \frac{E a_t^2}{E |a_t|}
\]

For complex signals, parameters \(a_t, x_t, c_t, S\) and \(W\) in Figure 6.9 are complex. \(\text{Re} \ a_t\) and \(\text{Im} \ a_t\) are independent and the cost function is defined as

\[
J(W) = E(\psi(\text{Re} \ c_t(W)) + \psi(\text{Im} \ c_t(W)))
\]

\[
\psi(x) = \frac{1}{2} x^2 - \alpha |x| \quad \text{(Sato function)} \tag{6.16}
\]

\[
\alpha = \left(\int x^2 v(dx) \right) \left(\int |x| v(dx) \right)^{-1}
\]

Blind equalizers are discussed in [42–64].

![Figure 6.8 System models.](image1)

![Figure 6.9 Equivalent model of the system from Figure 6.8.](image2)
6.3 DETECTION FOR A STATISTICALLY KNOWN, TIME VARYING CHANNEL

The system model is given in Figure 6.10. Parameters $h_T(\tau), c(t, \tau)$ and $h_R(\tau)$ represent transmitter filter, channel and receive filter impulse responses respectively. The overall system pulse response is a convolution $f(\tau; t)$ of the three and is given by Equation (6.17).

$$f(\tau; t) = h_T(\tau) * c(\tau; t) * h_R(\tau)$$

$$r(kT_s) = r_k = \sum_n a_n f(kT_s - nT; kT_s) + w_k$$  \hspace{1cm} (6.17)

Equation (6.17) also represents the samples of the overall received signal at sampling points $kT_s$. The sampling interval is $T_s$ and the symbol interval is $T$.

6.3.1 Signal model

The convolution can also be presented as:

$$r_k = \sum_{n=0}^L a_{l-n} f_i(k) + w_k$$  \hspace{1cm} (6.18)

For the $T$-spaced sampling (TS) case, we have $l = k$, and $i = n$; for the fractional spaced sampling (FS) case, $l = [k/2]$, and $i = 2n + m$, with $m = (k + 1)$ mod 2, i.e. $m = 1$ for $k$ even, and $m = 0$ for $k$ odd. Tapped delay line system models for TS and FS are given in Figures 6.11 and 6.12.

6.3.2 Channel model

For the channel model we assume:

- The worst case channel in terms of $T_M$ and $f_D$.
- For the worst case value of $T_M$ we use 20 $\mu$s.
- With the IS-54 symbol duration of $T = 1/24000 \approx 41.7 \mu$s, this corresponds to roughly $T_M = T/2$, i.e. the echo delay is half a symbol duration.
- For the impulse response shape, we use a ‘double-spike’ spaced by $T_M : c(\tau; t) = c_0(t)\delta(\tau) + c_1(t)\delta(\tau - T_M)$ where $E[|c_0(t)|^2] = E[|c_1(t)|^2] = 0.5$ so that the average energy of the channel is normalized to one.
- The response $f(\tau; t)$ is then

$$f(\tau; t) = c_0(t)h(\tau) + c_1(t)h(\tau - T_M)$$
Figure 6.11 Tapped delay line model of the equivalent discrete time channel for the TS case. The blocks containing ‘T’ denote delays $T_s$.

Figure 6.12 Tapped delay line model of the equivalent discrete time channel for the FS case and a channel impulse response length of $3T$. The blocks containing ‘$T/2$’ denote delays of $T/2$ s, $l = [k/2]$, and $[x]$ denotes the smallest integer $\geq x$.

- $h(\tau)$ is the full raised cosine response, equal to the convolution of $h_T(\tau)$ and $h_R(\tau)$.
- We set $t$ equal to $nT_s$ and $t$ equal to $kT_s$ (assuming the sampling phase $t_0 = 0$).
- For the TS case, we have, with $T_s = T$,

$$f_{0}^{TS}(k) = c_0(k)h(0) + c_1(k)h(T/2)$$
$$f_{n}^{TS}(k) = c_1(k)h((2n-1)T/2), \quad n = 1, 2, \ldots$$

- The symmetry of $h(\tau)$ was used.
- For the TS case, we truncate the number of taps to $L + 1 = 3$. The normalized average tap energies are then $E[|f_0|^2] = 0.7717$, $E[|f_1|^2] = 0.2136$, and $E[|f_2|^2] = 0.0147$. 
In the FS case, with \( t = nT/2 \) and \( k = kT/2 \)

\[
\begin{align*}
  f_0^{FS}(k) &= c_0(k)h(0) + c_1(k)h(T/2) \\
  f_1^{FS}(k) &= c_1(k)h(T/2) + c_1(k)h(0) \\
  f_{2n}^{FS}(k) &= c_1(k)h((2n - 1)T/2) \quad n = \pm 1, \pm 2, \ldots \\
  f_{2n+1}^{FS}(k) &= c_0(k)h((2n + 1)T/2) \quad n = \pm 1, \pm 2, \ldots 
\end{align*}
\]

- We retain the most significant taps, yielding \( 2(L + 1) = 6 \).
- The average (normalized) tap energies here are \( E[|f_{-1}|^2] = 0.00735 \), \( E[|f_3|^2] = 0.1068 \), and \( E[|f_1|^2] = 0.38585 \).
- For this model, for a vehicle speed of 30 m/s (67.1 mph) and a carrier frequency of 900 MHz, the maximum Doppler shift \( f_D \approx 90 \) Hz.

For simulation purposes, the autocorrelation is approximated by something more easily synthesized, namely, the inverse Fourier transform of a Chebyshev Type I magnitude squared frequency response.

For time separations \( \tau \leq 50T \), the Chebyshev filter yields a very good approximation to the desired autocorrelation.

### 6.3.3 Statistical description of the received sequence

The following assumptions are made

- the received sequence \( r_N = (r_1, r_2, \ldots, r_N)^T \)
- the transmitted sequence \( a_N = (a_1, a_2, \ldots, a_N)^T \)
- the pdf is complex Gaussian

\[
p(r_N|a_N) = \frac{1}{\pi \det[C(a_N)]} \cdot \exp \left( -r_N^H C^{-1}(a_N) r_N \right) \quad (6.19)
\]

- \( C(a_N) \) denotes the covariance matrix of \( r_N \) given \( a_N \).
- The superscript \( H \) denotes Hermitian (conjugate transpose).
- The elements of \( C(a_N) \), abbreviated \( C_a \), are obtained by taking the expectation \( E[r_N \cdot r_N^H|a_N] \):

\[
c_{ij} = \sum_{n=0}^{L} \sum_{m=0}^{L} a_i \cdot a_j^* \text{ } E[f_n(i)f_m^*(j)] + \sigma_w^2 \delta_{ij} \quad (6.20)
\]

The assumption that the channel is statistically known means that \( E[f_n(i)f_m^*(j)] \) are known.

### 6.3.4 The ML sequence (block) estimator for a statistically known channel

The maximum likelihood sequence estimator (MLSE) \( \hat{a} \) is defined as

\[
\hat{a} = \arg \min_{a} \Lambda(a, r) \quad (6.21)
\]
Figure 6.13  Plots of union upper bounds on the average sequence error probability versus $E_b/N_0$ for the TS and FS SKC detectors, and for the TS and FS known channel detector, for block length $N = 4$, using binary PSK modulation. Channel parameters are $f_D = 90$ Hz, $T_M = T/2$, and $t_0 = 0$.

Figure 6.14  Plots of the union upper bounds and simulated average sequence error probability versus $E_b/N_0$ for the TS and FS SKC detectors, for block length $N = 4$, and binary PSK modulation. The channel parameters are $f_D = 90$ Hz, $T_M = T/2$, and $t_0 = 0$.

where $\Lambda(\cdot)$ is the logarithm of Equation (6.19). The sequence metrics are

$$
\Lambda(a, r) = \ln |C_a| + r^H C_a^{-1} r
$$

(6.22)

Matrix inversion $C^{-1}$ can be calculated by Cholesky decomposition (factorization). For a given vector of samples $r$ and the known channel, the algorithm will search for $a$ that minimizes Equation (6.21). Some results are given in Figures 6.13–6.17.
Figure 6.15 Plots of simulated average symbol error probability $P_s$ versus $E_b/N_0$ for the FS SKC (solid lines) and FS KC (dashed lines) detectors, for block length $N = 3, 4, 5$ and binary PSK modulation. The channel parameters are $f_D = 90$ Hz, $T_M = T/2$, and $t_0 = 0$.

Figure 6.16 Plots of simulated average BEP $P_b$ (after differential decoding) versus $E_b/N_0$ for the TS (solid lines) and FS (dashed lines) SKC detectors, for block length $N = 2 – 5$, and binary PSK modulation. The channel parameters are $f_D = 90$ Hz, $T_M = T/2$, and $t_0 = 0$. 
Figure 6.17 Plots of simulated average symbol error probability $\bar{P}_s$ versus $E_b/N_0$ for the TS and FS SKC detectors, for block length $N = 4$ and binary PSK modulation, showing the effect of mismatch between the estimated delay spread, $\hat{T}_M$, and the actual delay spread $T_M$. For all cases, $\hat{T}_M = T/2$, but the actual $T_M$ is zero, $T/4$ and $T/2$. The other channel parameters are $f_D = 90$ Hz and $t_0 = 0$. Solid curves are TS results, dashed curves are FS results.

Figure 6.18 System model.

### 6.4 LMS-ADAPTIVE MLSE EQUALIZATION ON MULTIPATH FADING CHANNELS

Within this section we are going to drop the assumption that the channel is statistically known and deal with the problem where both data and channel have to be estimated simultaneously.

#### 6.4.1 System and channel models

The system model is given in Figure 6.18.

The received signal can be represented as

$$r_k = \sum_{i=0}^{L} x_{k-i} g_{k,i}^* + \eta_k$$

$$r_k = g_k^H x_k + \eta_k$$

where

$$g_k = [g_{k,0}, g_{k,1}, \ldots, g_{k,L}]^T, \quad x_k = [x_k, x_{k-1}, \ldots, x_{k-L}]^T$$

$$\frac{1}{2} E\{g_k^* g_{k-l,j}^*\} = R_g(l, i, j)$$
Parameter $g_{k,i}$ represents the samples of the equivalent channel impulse response. The operation of the system is presented in Figure 6.19. For data estimation by the Viterbi algorithm, the system uses the channel estimates. The Viterbi algorithm delay is $DT$. This class of algorithms is known as delayed decision-directed equalization.

The metric used in the Viterbi detector is $|r_k - \sum_{i=0}^{L} x_{k-i} \hat{g}_{k,i}^*|^2$

### 6.4.2 Adaptive channel estimator and LMS estimator model

Assuming that the feedback decisions are correct, i.e. $\hat{x}_{k-D} = x_{k-D}$ at the input of the LMS estimator, and the order of the LMS estimator is $L + 1$ the following step is implemented: the LMS estimator updates $\tilde{g}_{k-D}$ each time by:

$$\tilde{g}_{k-D} = \tilde{g}_{k-D-1} + \mu x_{k-D} \alpha_{k-D}$$

where

$$\alpha_{k-D} = r_{k-D} - \frac{1}{\tilde{g}_{k-D-1}} x_{k-D}$$ (6.24)

### 6.4.3 The channel prediction algorithm

The linear prediction generates

$$\hat{g}_k = \sum_{j=0}^{q} \alpha_j \tilde{g}_{k-D-j}$$ (6.25)

where $\alpha_0, \ldots, \alpha_q$ are constants and $q$ is a positive integer. Due to delay $DT$ of the Viterbi algorithm, the straight line extrapolator is used

$$\hat{g}_k = \tilde{g}_{k-D} + \frac{p}{q} (\tilde{g}_{k-D} - \tilde{g}_{k-D-q})$$ (6.26)

where $p$ is the prediction step. Although there exist many other channel predictions from simulation results and comparisons, the straight line extrapolator gives fair performance and very simple implementation. If the autocorrelation function $R_g(l, i, j)$ is known by the receiver, the prediction coefficients $\alpha_j$ can be obtained from the Wiener solution. Some performance results are given in Figures 6.20–6.24.

MLSE equalizers are discussed in [65–75].
Figure 6.20 Analytical and simulation results for adaptive MLSE employing straight line extrapolation prediction on a two-tap Rayleigh fading channel with $f_D = 6.7$, 24.3, 50 and 84 Hz.

Figure 6.21 Analytical and simulation results for adaptive MLSE employing linear channel prediction on a two-tap Rayleigh fading channel with $f_D = 6.7$, 24.3, 50 and 84 Hz.
Figure 6.22 Analytical and simulation results for adaptive MLSE without channel prediction on a two-tap Rayleigh fading channel with $f_D = 6.7, 24.3, 50$ and $84$ Hz. (Note that the analytical bound of the case of $f_D = 84$ Hz is greater than 1 and hence not shown.)

Figure 6.23 Analytical and simulation results obtained by varying the step size parameter $\mu$ at $E_b/N_0 = 35$ dB.
6.5 ADAPTIVE CHANNEL IDENTIFICATION AND DATA DEMODULATION

The estimation technique presented in Section 6.4 is rather inefficient when detection delay \( D \) or Doppler \( f_D \) increase. In such examples, a joint estimation of both data and channel would give better results. Unfortunately, pure joint estimation would be too complex. In this section we present an algorithm where the joint ML function is maximized by alternating the maximization process with respect to data (given the channel) and channel (given data from the previous iteration).

### 6.5.1 System model

The received signal is given as

\[
    r_k = \sum_{l=0}^{L} h_l(k)a_{k-l} + n_k, \quad k = 1, 2, \ldots, N
\]

The memory \( L \) of the channel is determined by the time spread \( T_{\text{max}} \) of the actual channel and the symbol period \( T_s \):

\[
    L = \left\lceil \frac{T_{\text{max}}}{T_s} \right\rceil + 1
\]

### 6.5.2 Joint channel and data estimation

The approach used in this section is based on [76]. The process is described by the following steps.

#### 6.5.2.1 Block sequence estimation (BSE)

- The received sequence is fed into the metric computation unit in blocks of \( N_b \) symbols at a time.
- The data demodulation is also performed in blocks, rather than symbol by symbol as in the conventional VA.

Figure 6.24 Analytical results obtained by varying the step size parameter \( \mu \) at \( E_b/N_0 = 35 \) dB and 40 dB.
At time $k$ the receiver processes a block of $N_b$ data symbols ($r_k, \ldots, r_{k+N_b-1}$).

After computing the path metrics at time $k+N_b - 1$, the survivor paths of each state are traced backwards in the trellis in order to detect a merge.

If a merge occurs within the block, at the time $k + N_b - 1 - \delta (\delta < N_b)$, decisions are made on the data sequence $\{a_k, \ldots, a_{k+N_b-1-\delta}\}$.

The initial point of the next block is set at time $k + N_b - 1 - \delta$ and the next $N_b$ data symbols are processed.

This means that the portion of the received symbols ($r_{k+N_b-\delta}, \ldots, r_{k+N_b-1}$) is processed twice.

The state metrics at time $k + N_b - 1 - \delta$ are reset to a large number except for the metric of the state at which the merge occurred. This metric is set to zero. This way the chances of errors associated with illegal state transitions at the beginning of the new block are reduced. It is equivalent to restarting the VA from a known initial state.

If a merge is not detected within the block ($\delta < N_b$), the state with the minimum metric at the end of the block (at the time $k + N_b - 1$) is chosen and its survivor path is traced backwards in the trellis.

A merge is assumed at time $k + N_b - \delta$ ($\delta = \lceil N_b/2 \rceil$) on the best survivor path.

The state metrics at time $k + N_b - \delta$ are reset to a large number, except for the metric of the best survivor. This metric is set to zero.

### 6.5.2.2 Block adaptive channel estimation

In the derivation of the adaptive schemes for the CIR estimation we will temporarily assume that the channel is unknown but static. The discrete time index $k$ will be dropped from Equation (6.27) to simplify the notation. We will revisit the time varying CIR scenario in subsequent sections. In vector notation, Equation (6.27) can be written as:

$$ r = Ah + n $$

where

$$ r = [r_1, \ldots, r_N]^T, \ h = [h_0, \ldots, h_L]^T, \ n = [n_1, \ldots, n_N]^T, \text{ and} $$

$$ A = \begin{bmatrix} a_1 & a_0 & \cdots & a_{1-L} \\ a_2 & a_1 & \cdots & a_{2-L} \\ \vdots & \vdots & \ddots & \vdots \\ a_N & a_{N-1} & \cdots & a_{N-L} \end{bmatrix} $$

Let us use $a = [a_{1-L}, \ldots, a_N]^T$ to denote the transmitted data vector.

### 6.5.2.3 Maximum likelihood (ML)

ML estimates of the channel $h_{ML}$ and data $a_{ML}$ are those that maximize the conditional probability density function (pdf) $p(r|a, h)$ or, equivalently, minimize $|r-Ah|^2$. With respect to $A$ and $h$

$$ (a_{ML}, h_{ML}) = \arg \min_{a \in \mathbb{A}^{N-L+1}, h \in \mathbb{R}^L} |r-Ah|^2 $$

(6.30)

The joint minimization over $a$ and $h$ cannot be computed in closed form. When either $a$ or $h$ is fixed, this minimization is a well known problem. When the channel $h$ is fixed, the ML estimate of $a$ is
computed via the VA. When the data vector $a$ is given, the minimization of $C(A, h) = |r - Ah|^2$ over $h$ is a standard least squares problem which results in

$$h_{ML} = (A^HA)^{-1}A^Hr$$  \hspace{1cm} (6.31)

Thus, the joint minimization of $C(A, h)$ over $A$ and $h$ can be viewed as an alternative minimization type of problem.

### 6.5.2.4 Iterative procedure

The above algorithms can be summarized as:

1. Start with an initial estimate $\hat{h}(0)$ of the channel.
2. Minimize $C(\hat{A}, \hat{h})$ with respect to $\hat{A}$ via the VA to obtain $\hat{A}(l)$.
3. Minimize $C(\hat{A}, \hat{h})$ w.r.t. $\hat{h}(l)$ to obtain
   $$\hat{h}(l + 1) = [\hat{A}^H(l)\hat{A}(l)]^{-1}\hat{A}^H(l)r$$  \hspace{1cm} (6.32)
4. Repeat Steps 2 and 3 until the algorithm converges.

Simulation studies have shown that this scheme can converge very fast to a global minimum (in approximately 5–6 iterations).

The above algorithm pays a heavy price as a tradeoff for its optimum performance. The receiver has to process a long received sequence (in the order of a few thousands of symbols) several times before it can provide the user with any reliable decisions about the transmitted data. This implies a significant decision delay. The algorithm is impractical for real-time implementation, especially when the channel is time varying. This is a strong motivation for constructing algorithms to perform the joint channel and data estimation recursively in time, which provides the user with reliable data estimates quickly enough to be able to adopt to channel variations too.

### 6.5.2.5 Iterative CIR estimator

In this case we have

$$\hat{h}(l + 1) = \hat{h}(l) + \mu \frac{\partial \ln p(r|\hat{A}, \hat{h})}{\partial \hat{h}} \bigg|_{\hat{h} = \hat{h}(l)}$$

$$\hat{h}(l + 1) = \hat{h}(l) + \mu \hat{A}^H[l - \hat{A}\hat{h}(l)]$$  \hspace{1cm} (6.33)

If we can achieve perfect convergence, then

$$\hat{h}(l + 1) = \hat{h}(l) \iff \hat{A}^H[l - \hat{A}\hat{h}(l)] = 0 \iff h(l)$$

$$\hat{h}(l + 1) = (\hat{A}^H\hat{A})^{-1}\hat{A}^Hl$$  \hspace{1cm} (6.34)

which is exactly the ML estimate of the channel. In Equation (6.32), $\hat{A}$ and $r$ extend over the whole data record. If, instead, we consider only a portion of data, i.e. a block of $N_b$ symbols, then each time we update the CIR estimate we can perform the joint ML estimation of data and channel recursively in time. Notice that Equation (6.33) has the form of a block least mean square (BLMS) equation. This is a consequence of the quadratic exponent of the Gaussian probability distribution of the noise vector. Given the CIR estimate $\hat{h}(l)$ at the $l$th recursion, the corresponding block of data estimates can be provided by the BSE, as was developed earlier. Starting from an initial guess, $\hat{h}(0)$, for the CIR we construct a joint data and channel estimation scheme which operates on successive blocks of data in a time recursive manner.
6.5.2.6 BSE/BLMS algorithm

In this case:

1. Start with an initial estimate \( \hat{\mathbf{h}}(0) \) of the CIR. Initialize the block counter \((l \rightarrow 1)\) and the symbol counter \((k \rightarrow 1)\).
2. Receive the \( l \)th block of data \( \mathbf{r}(l) = [r_k, \ldots, r_{k+N_b-1}]^T \), where \( N_b \) is the block length.
3. Apply the BSE on \( \mathbf{r}(l) \), using \( \hat{\mathbf{h}}(l-1) \) for metric computations, to obtain the \( l \)th block of data estimates:

\[
\hat{\mathbf{a}}(l) = [\hat{a}_k, \ldots, \hat{a}_{k+N_b-1-\delta(l)}]^T
\]

where \( \delta(l) \) is the merging delay of the BSE for the \( l \)th block.
4. Using the data estimates \( \hat{\mathbf{a}}(l) \) and the first \( N_b - \delta(l) \) entries of the \( l \)th block of received symbols \( \mathbf{r}_a(l) = [r_k, \ldots, r_{k+N_b-1-\delta(l)}]^T \), update the CIR estimate as follows:

\[
\hat{\mathbf{h}}(l) = \hat{\mathbf{h}}(l-1) + \mu \hat{\mathbf{A}}(l)[\mathbf{r}_a(l) - \hat{\mathbf{A}}(l)\hat{\mathbf{h}}(l-1)]
\]

(6.35)

The matrix \( \hat{\mathbf{A}}(l) \) is given by:

\[
\hat{\mathbf{A}}(l) = \begin{bmatrix}
\hat{a}_k & \hat{a}_{k-1} & \cdots & \hat{a}_{k-L} \\
\hat{a}_{k+1} & \hat{a}_k & \cdots & \hat{a}_{k+1-L} \\
\vdots & \vdots & \ddots & \vdots \\
\hat{a}_{k+N_b-1-\delta(l)} & \hat{a}_{k+N_b-2-\delta(l)} & \cdots & \hat{a}_{k+N_b-1-\delta(l)-L}
\end{bmatrix}
\]

(6.36)

and \( \mu \) is a step size parameter.

5. Reset the block and symbol counters to \( l \rightarrow l + 1 \) and \( k \rightarrow k + N_b - \delta(l) \) and go to Step 2 until all data have been processed. An issue is the choice of the step size parameter \( \mu \). A large value of \( \mu \) would provide faster adaptation but could lead to divergence of the channel estimate. On the other hand, if \( \mu \) is too small, we will lose a lot of data while trying to acquire the channel.

6.5.2.7 Recursive least squares (RLS) channel estimation

In this case the CIR estimation is based on the minimization of the weighted sum of the squared errors

\[
J(k) = \sum_{l=0}^{k} \lambda^{k-l} e^2(l) = \sum_{l=0}^{k} \lambda^{k-l} |r_l - \hat{\mathbf{h}}^T(k) \cdot \hat{\mathbf{a}}(l)|^2
\]

(6.37)

where \( \hat{\mathbf{h}}(k) = [\hat{h}_0(k), \ldots, \hat{h}_L(k)]^T \) is the CIR estimate at time \( k \), \( \hat{\mathbf{a}}(l) = [\hat{a}_l, \ldots, \hat{a}_{L-l}]^T \) is the vector of the estimated data at times \( l, \ldots, l-L \) and \( r_l \) is the received signal at time \( l \). The parameter \( \lambda \), \((0 < \lambda \leq 1)\) gives more weight to recent errors. Thus, we allow for time varying channels. The RLS algorithm for the CIR estimation is described as [77–79]:

\[
g(k) = \frac{P(k)\hat{\mathbf{a}}^*(k)}{\lambda + \hat{\mathbf{a}}^T(k)P(k)\hat{\mathbf{a}}(k)}
\]

\[
P(k + 1) = \frac{1}{\lambda} [P(k) - g(k)\hat{\mathbf{a}}^T(k)P(k)]
\]

\[
e(k) = r_k - \hat{\mathbf{h}}^T(k)\hat{\mathbf{a}}(k)
\]

\[
\hat{\mathbf{h}}(k + 1) = \hat{\mathbf{h}}(k) + e(k)g(k)
\]

\[
k = 1, 2, \ldots
\]

(6.38)

To initialize the algorithm we set \( P(0) = \varepsilon^{-1}\mathbf{I} \), where \( \varepsilon \) is a small positive constant.
6.5.2.8 BSE/RLS algorithm

Incorporating the CIR update into the recursive joint channel and data estimation algorithm gives:

1. Start with an initial estimate \( \hat{h}(0) \) of the CIR. Initialize the block counter \( l \rightarrow 1 \) and the symbol counter \( k \rightarrow 1 \).

2. Receive the \( l \)th block of data \( r(l) = [r_k, \ldots, r_{k+N_b-1}]^T \), where \( N_b \) is the block length.

3. Apply the BSE on \( r(l) \), using \( \hat{h}(l-1) \) for metric computations, to obtain the \( l \)th block of data estimates:
   \[
   \hat{a}(l) = [\hat{a}_k, \ldots, \hat{a}_{k+N_b-1-\delta(l)}]^T,
   \]
   where \( \delta(l) \) is the merging delay of the BSE for the \( l \)th block.

4. Using the data estimates \( \hat{a}(l) \) and the first \( N_b - \delta(l) \) entries of the \( l \)th block of received symbols \( r_d(l) = [r_k, \ldots, r_{k+N_b-1-\delta(l)}]^T \), obtain the new CIR estimate \( \hat{h}(l) \) by executing the RLS recursion \( N_b - \delta(l) \) times.

5. Reset the block and symbol counters to \( l \rightarrow l + 1 \) and \( k \rightarrow k + N_b - \delta(l) \) and go to Step 2 until all data have been processed.

A schematic diagram of adaptive BSE (BSE/BLMS or BSE/RLS) is shown in Figure 6.25.

The adaptive BSE derives its strength from the fact that, due to the variable decision delay, the data estimates used for CIR tracking are more likely to be taken from the ML path. The block processing of the data contributes to the averaging of the effects of symbol errors. Thus, the CIR estimator is fed with better data estimates, which improves its tracking capability compared to the conventional adaptive MLSE.

The possible data errors towards the end of each block, caused by computing the metrics for the whole block using the same CIR estimate, are alleviated by the fact that the last symbols of each block are processed again with the updated CIR estimate. If the fading rate gets higher, the block length \( N_b \) must be chosen carefully so that the CIR estimate is not outdated. A choice of \( N_b \approx 5L \) is a good rule of thumb.

6.5.3 Data estimation and tracking for a fading channel

In a static channel environment, the BSE/BLMS and BSE/RLS as described so far, operate starting from an initial guess \( \hat{h}(0) \) of the CIR. For a fading channel, the task of CIR acquisition is much heavier, especially when the fading rate becomes high. In such environments, the CIR acquisition
is accomplished via a training sequence, which is periodically sent to the receiver as a portion of a fixed size data packet. This format is used in mobile communications. An example of such a data structure is the time division multiple access (TDMA) slot of the IS-54 North American Digital Cellular standard described briefly in Chapter 1.

6.5.3.1 Training

The BSE/BLMS, BSE/RLS can be applied in this signaling format with an appropriate adjustment in the CIR acquisition step. This is accomplished by feeding the above described adaptive algorithms (BLMS or LMS, RLS) with the training symbols at the header of the TDMA slot, producing the CIR estimate $\hat{h}(0)$. As soon as the acquisition stage is over, the BSE/BLMS or BSE/RLS is activated for joint data estimation and channel tracking, as described in the previous subsections, for the duration of the entire TDMA slot. The performance of this scheme, obtained by simulation, is now presented.

6.5.3.2 Performance and computational complexity

In the evaluation of the system performance the same assumptions are used as in [76]:

- Static and fading channel environments.
- To cover both the static and time varying CIR, the SNR is defined as

$$\text{SNR} = 10 \log \left( \frac{\sigma^2_a}{\sigma^2} \sum_{i=0}^{L} E(|h_i|^2) \right)$$

where $\sigma^2_a$ and $\sigma^2$ are the variances of the input data and additive noise, respectively.
- The above definition implies independence of the CIR variations from the transmitted data sequence, which is quite a reasonable assumption.

6.5.4 The static channel environment

- CIR is assumed constant for the duration of the entire received data record.
- In the simulation there is no training sequence, i.e. the BSE operates in a blind fashion.
- The modulation format is binary phase shift keying (BPSK).
- No coding is assumed.
- The input alphabet is $\{-1, 1\}$, i.e. the transmitted data sequence $\{a_t\}$ consists of real numbers.
- Two particular channel examples are used whose impulse responses are given by the vectors $h_1 = [0.407, 0.815, 0.407]$ and $h_2 = [0.1897, 0.5097, 0.6847, 0.46, 0.1545]$, respectively.
- Both channels exhibit deep nulls in their magnitude response within the frequency band of interest.
- Channel 2 has non-linear phase characteristics.
- As a result, linear equalization methods exhibit very poor performance on these channels.
- The MLSE for the corresponding known channel is used in Figure 6.26 for comparison.

6.5.4.1 Simulations/BSE/BLMS

- The error rates are steady state after the channel acquisition has been completed.
- The acquisition (convergence) time is described independently.
In both cases the block length is chosen to be $N_b = 8L$ and the step size parameter $\mu = 0.01$.

The initial guesses for Channels 1 and 2 are $\hat{h}(0) = [0, 1, 0]$ and $\hat{h}(0) = [0, 0, 1, 0, 0]$, respectively.

The probability of error is estimated over a data record of 100 000 symbols.

The performance is very close to that of the known CIR environment.

For comparison purposes, a linear blind equalizer (Sato’s algorithm) described by Equation (6.16) is used. As expected, it fails to converge at all for this type of channel.

### 6.5.4.2 The BSE/RLS

- Is simulated for Channels 1 and 2, starting with the same initial guesses for the channels as in BSE/BMLS.

- For the initialization of the BSE/RLS, the parameters $\lambda = 1$ and $\varepsilon = 0.001$ are used in $P(0)$.

- The performance of the BSE/RLS in Figure 6.26 is also very close to that of the known channel.

- An improvement is observed for Channel 2 compared to the BSE/BLMS, especially at high SNR. This is no surprise, since acquisition capability (convergence speed) is better than the LMS at high SNR.

- The tradeoff for the improved performance of the BSE/RLS, however, is its increased computational complexity.

![Figure 6.26 Probability of symbol error versus SNR using the BPSK waveform for the BSE/BLMS and the BSE/RLS algorithms over Channels 1 and 2.](image-url)
6.5.4.3 The effect of the block length $N_b$

BSE/BLMS is simulated on Channel 1 for $N_b = 8L$, $5L$, $4L$ and $3L$ symbols and the results are shown in Figure 6.27.

- The performance degradation for smaller $N_b$ must be attributed to the BSE algorithm.
- As the block length becomes smaller, the number of blocks in which no merge occurs increases.
- Bad data decisions are forced, reflecting upon the channel estimate too.

6.5.4.4 The convergence properties

- 300 independent Monte Carlo runs of BSE/BLMS are performed and the average squared error $|\hat{h} - h|^2$ as the number of processed data varies between 100 and 1000 symbols is computed.
- Simulations are performed for block sizes $N_b = 5L$, $10L$ and $15L$ points, and for SNR = 6 and 12 dB.
- The results are shown in Figure 6.28.
- For Channel 1, independent of the block size, we approach the steady state mean squared error within the first few hundred symbol periods.
- For SNR = 6 dB the steady state is reached after processing about 700 symbols, while for SNR = 12 dB we need approximately 400 symbols. This allows us to specify the acquisition time for the BSE/BLMS at about 400 symbol periods.
6.5.5 The time varying channel environment

In this segment a TDMA with the following parameters is simulated.

- The carrier frequency is assumed to be 900 MHz and the bit rate 48.6 kb/s.
- For QPSK modulation, that translates to a baud rate of 24.3 ksymbols/s, or a symbol period $T_s = 41 \mu$s.
- The mobile radio channel is assumed to be wideband with $L + 1$ taps.
- Each element of the CIR $\{h_i(k)\}_{i=0}^L$ is modeled as an independent low pass zero mean complex Gaussian random process with Rayleigh distributed amplitude and uniformly distributed phase in the interval $[-\pi, \pi]$.
- Each of the $h_i$s is generated by passing a zero mean white complex Gaussian noise sequence through a digital second order Butterworth filter whose cutoff frequency $f_d$ is determined by

$$f_d = T_s \left( f_c \frac{v}{v_c} \right)$$

where $f_d$ is the normalized Doppler shift, $f_c$ is the carrier frequency, and $v$ and $v_c$ are the vehicle speed and speed of light, respectively. This simulation procedure is widely used in the literature.

- For these system specifications the performance of all algorithms is investigated for vehicle speeds ranging from $v = 60$ mph (100 km/h) up to 150 mph (248 km/h).
- The corresponding normalized Doppler shift ranges from $f_d = 0.0034$ to 0.0085.
- A three-tap CIR ($L = 2$) is used. For the specifications of the IS-54 standard ($T_s = 41 \mu$s), this channel model covers a maximum time spread of $T_{\text{max}} = 123 \mu$s.
The CIR elements \( h_0(k), h_1(k), h_2(k) \) are constructed so as to have equal power \( \sigma^2 = 1/3 \) and to be independent of each other.

This corresponds to a fading scenario where no line of sight is present (Rayleigh fading).

Figure 6.29 compares the performance of all algorithms tested: the BSE; the conventional adaptive MLSE; a variable \( D \) (delay in the CIR estimation loop) version of the conventional adaptive MLSE; and the Per Survivor Processing (PSP) known CIR environment. PSP estimates the channel only for the surviving trajectories in VA, and for those trajectories data is known. This corresponds to an ideal receiver capable of estimating the CIR perfectly, with zero delay, which is unrealistic when the channel is varying rapidly.

The adaptive \( D \) version of the conventional MLSE is derived by detecting a merge within \( N_b \) epochs from the current symbol and setting \( D \) equal to the merging delay. If no merge is detected, \( D \) is set to \( N_b - 1 \). The symbol error rate is computed as the fraction of data symbols that are in error over a time interval of 5000 consecutive TDMA slots, each packet of which has the structure shown in Chapter 1. For convenience, only the initial training symbols and the information symbols inside the TDMA slot were considered.

For LMS channel acquisition and tracking, the optimum value of the step size parameter was found to be \( \mu = 0.06 \) for \( f_d = 0.0034 \) and \( \mu = 0.12 \) for \( f_d = 0.0085 \), at SNR = 20 dB. The optimum value of the weight parameter was found to be \( \lambda = 0.73 \) for \( f_d = 0.0085 \) at SNR = 20 dB. For all of the simulated algorithms presented, the initial CIR guess at the beginning of the first TDMA slot was set at \( \hat{h}(0) = [0, \ldots, 0] \). The last CIR estimate computed at the end of each subsequent slot is utilized to initialize the LMS or RLS at the beginning of the next slot.

### 6.5.5.1 Performance

In Figure 6.29, the performance of all the adaptive MLSE algorithms is presented for \( f_d = 0.0034 \) (82.6 Hz), which corresponds to a vehicle speed of 60 mph. The following parameters were used.
• An LMS update was used for the CIR acquisition and tracking. The step size parameter was set at \( \mu = 0.06 \) for all algorithms.

• For the conventional adaptive MLSE, the optimum performance was achieved for a delay in the CIR estimation loop equal to \( D = 4 \).

• The decision delay was set at \( \delta = 5L \) for both the PSP, the conventional MLSE and the adaptive \( D \) MLSE.

• For the BSE/LMS a block length \( N_b = 11 \) was found to yield the best performance.

• Both the PSP and the BSE/LMS exhibit a clear advantage over the conventional adaptive MLSE at SNR above 20 dB.

• The adaptive \( D \) MLSE also exhibits a performance advantage over the conventional MLSE and gets closer to the BSE at high SNR.

• For comparison purposes a receiver which performs no tracking of the CIR after the initial acquisition mode was also simulated. It reaches a floor of about \( 2 \times 10^{-1} \) at SNR = 20 dB. This clearly enforces the necessity of the use of CIR tracking algorithms even at this fading rate.

Figure 6.30 presents the performance of all algorithms for \( f_d = 0.0085 \) (206.5 Hz), and a speed of 150 mph. Both the LMS and RLS were used for CIR acquisition and tracking. For the LMS, the step size parameter was set at \( \mu = 0.12 \) and for the RLS, the weight parameter was set at \( \lambda = 0.73 \), for all algorithms. The optimum parameters were again found to be \( D = 4 \) and \( N_b = 11 \). The PSP and the BSE exhibit a clear advantage over the conventional and adaptive \( D \) MLSE at high SNR.

A much poorer performance was exhibited by all algorithms compared to the smaller vehicle speed in Figure 6.29. The performance degradation must be attributed to the higher fading rate making it very difficult to track the channel variations. The RLS exhibited slightly better performance at high SNR, which was anticipated. Absence of CIR tracking yields a floor of 0.5 probability of symbol error at SNR = 10 dB.

Figure 6.31 presents the error rates for the information symbols inside the TDMA slot for \( f_d = 0.0085 \) at SNR = 20 dB with LMS update. In the figure, only every other information symbol is

![Figure 6.30 Symbol error rate performance of the various adaptive MLSE algorithms at a vehicle speed of 150 mph. The results from the use of either the LMS or the RLS channel estimates are shown.](image-url)
Figure 6.31 Data symbol error rate versus symbol position in the TDMA slot at SNR = 20 dB using the LMS channel acquisition and tracking algorithms (a) $v = 60$ mph; (b) $v = 150$ mph.
plotted. The error rates tend to increase toward the end of the slot, even with CIR tracking algorithms. If no CIR tracking is employed, the error rates become unacceptable after the first 10–20 symbols within the TDMA slot. For BSE/LMS at 60 mph and error rate $10^{-3}$ there is a 10 dB worse performance from the lower bound of a known channel. PSP has manifested a similar performance. At 150 mph the error rate of all algorithms levels off at residual error rates larger than $10^{-2}$, even at 30 dB. All algorithms fail at a velocity of 150 mph.

### 6.5.5.2 Computational complexity and reconfiguration efficiency

Computational complexity is measured by the required number of multiplications, $N_{\text{mul}}$, and additions, $N_{\text{add}}$, per data sample. The complexity is computed for a CIR of length $L + 1$ and an alphabet size $M$. The exact number of required operations per data symbol for the BSE cannot be expressed in closed form because of the variable merging delay inside each block. If $\bar{\delta}$ denotes the average merging delay over all blocks processed, then the required operations for metric computations per data symbol must be multiplied by a factor

$$f_{\text{ex}} = \frac{1}{[1 - \bar{\delta}/N_b]}$$

to account for processing the last part of each block twice. Parameter $\bar{\delta}$ is obtained from simulations.

The three-tap fading channel ($L = 2$) of the previous section is used and 5000 frames of PSK data ($M = 4$) are processed, amounting to 82953 processed blocks, each of length $N_b = 11$ symbols. The fading rate is set at $f_d = 0.0034$ and the SNR is 30 dB. $\bar{\delta} = 2.9453$ symbols is found, which gives $f_{\text{ex}} = 1.365662$.

Table 6.1 presents the number of complex operations required per data symbol for metric calculations in the adaptive MLSE algorithm. Table 6.2 presents the required number of complex operations per symbol for this specific example, with both LMS and RLS updates for all channel estimators. For the Sato algorithm, discussed in Section 6.2, a tapped delay line equalizer with $L_{\text{sa}} = 16$ taps is assumed. From Tables 6.1 and 6.2, the conventional adaptive MLSE requires the smallest number of operations of all MLSE-based techniques, but significantly more operations than linear equalizers such as Sato’s algorithm. The BSE requires an increased number of operations for the metric update compared to the conventional adaptive MLSE due to reprocessing the last part of each data block. A hardware implementation of the BSE would be more difficult due to the necessary checks for the detection of a merge inside each block. For the CIR estimation part, the PSP requires considerably more operations than all other techniques due to the multiple CIR estimates that it preserves (one for each state). This overhead becomes even larger if the RLS estimation algorithm is used for CIR tracking.

Table 6.1 Complex operations per data symbol for adaptive MLSE algorithms (metric calculations)

<table>
<thead>
<tr>
<th></th>
<th>Conventional adaptive MLSE</th>
<th>BSE</th>
<th>PSP</th>
<th>Sato</th>
</tr>
</thead>
<tbody>
<tr>
<td>Metric Update</td>
<td>$N_{\text{mul}}$</td>
<td>$M^{L+1}(L + 1)$</td>
<td>$M^{L+1}(L + 1)f_{\text{ex}}$</td>
<td>$M^{L+1}(L + 1)$</td>
</tr>
<tr>
<td></td>
<td>$N_{\text{add}}$</td>
<td>$M^{L+1}(L + 1)$</td>
<td>$M^{L+1}(L + 1)f_{\text{ex}}$</td>
<td>$M^{L+1}(L + 1)$</td>
</tr>
<tr>
<td>CIR update</td>
<td>$N_{\text{mul}}$</td>
<td>$2(L + 1)$</td>
<td>$2(L + 1)$</td>
<td>$M^{L+1}(L + 1)$</td>
</tr>
<tr>
<td>(LMS)</td>
<td>$N_{\text{add}}$</td>
<td>$2(L + 1)$</td>
<td>$2(L + 1)$</td>
<td>$M^{L+1}(L + 1)$</td>
</tr>
<tr>
<td>CIR update</td>
<td>$N_{\text{mul}}$</td>
<td>$4(L^2 + L)$</td>
<td>$4(L^2 + L)$</td>
<td>$M^{L+1}(L^2 + L)$</td>
</tr>
<tr>
<td>(RLS)</td>
<td>$N_{\text{add}}$</td>
<td>$3(L^2 + L)$</td>
<td>$3(L^2 + L)$</td>
<td>$M^{L+1}(L^2 + L)$</td>
</tr>
</tbody>
</table>
Table 6.2 Complex operations per data symbol for \( M = 4 \) and \( L = 2 \) (channel estimations)

<table>
<thead>
<tr>
<th></th>
<th>Conventional</th>
<th>adaptive</th>
<th>MLSE</th>
<th>BSE</th>
<th>PSP</th>
<th>Sato</th>
</tr>
</thead>
<tbody>
<tr>
<td>Metric</td>
<td>( N_{\text{mul}} )</td>
<td>192</td>
<td>262</td>
<td>192</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Update</td>
<td>( N_{\text{add}} )</td>
<td>192</td>
<td>262</td>
<td>192</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>CIR update</td>
<td>( N_{\text{mul}} )</td>
<td>6</td>
<td>6</td>
<td>96</td>
<td>34</td>
<td></td>
</tr>
<tr>
<td>(LMS)</td>
<td>( N_{\text{add}} )</td>
<td>6</td>
<td>6</td>
<td>96</td>
<td>34</td>
<td></td>
</tr>
<tr>
<td>CIR update</td>
<td>( N_{\text{mul}} )</td>
<td>24</td>
<td>24</td>
<td>384</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(RLS)</td>
<td>( N_{\text{add}} )</td>
<td>24</td>
<td>24</td>
<td>384</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

For the calculation of reconfiguration efficiency, relations defined in Chapters 2 and 3 are still valid. For practical applications the relative complexity \( D_r \) can be calculated by using data directly from Tables 6.1 and 6.2.

More details on the topic can be found in [80–93].

6.6 TURBO EQUALIZATION

In this section we discuss how the problem from Section 6.5 can be solved by using the turbo principle discussed in Chapter 2.

6.6.1 Signal format

The transmitted signal \( s(t) \) is provided by the output of a filter whose impulse response is \( h_c(t) \). The signal emitted can be expressed in the form:

\[
s(t) = A \sum_k c_k h_c(t - kT) \exp(j2\pi f_0 t + \phi_0)
\]  \hspace{1cm} (6.39)

In a multipath channel, the received signal \( y(t) \), can be written as follows:

\[
y(t) = \sum_{m=0}^{M-1} A_m(t) \sum_k c_k h_c(t - \tau_m - kT) \exp(j2\pi f_0 t + \phi_0) + w(t)
\]  \hspace{1cm} (6.40)

where \( A_m(t) \) are complex-valued independent multiplicative noise processes. The receiver matched filter output

\[
R_n = R(nT) = \sum_{m=0}^{M-1} A_m(n) \sum_k c_{n-k} h_c(kT - \tau_m) + w_n
\]  \hspace{1cm} (6.41)

where \( A_m(n) \) equals \( A_m(nT) \) by definition, \( w_n \) denotes the response of the receiving matched filter to the noise \( w(t) \), sampled at time \( nT \). \( h_c(t) \) is defined by \( h_c(t) = h_c(t) \times h_c^*(-t) \) and satisfies the Nyquist criterion.

\[
\Gamma_c(n) = \sum_{m=0}^{M-1} A_m(n) h_c(kT - \tau_m)
\]  \hspace{1cm} (6.42)
Let us suppose that the ISI is limited to \((L_1 + L_2)\) symbols. Equation (6.41) may be written in the form:

\[
R_n = \sum_{k=-L_2}^{L_1} \Gamma_k(n) c_{n-k} + w_n
\]  

(6.43)

### 6.6.2 Equivalent discrete time channel model

Quantities \(\Gamma_k(n)\) are expressed as a linear combination of the multiplicative noises \(A_m(n)\). Therefore, they are Gaussian in the case of a Rayleigh-type channel and constant in the case of a Gauss-type channel. Consequently, the set of modules made up of the modulator, the transmission channel and the demodulator can be represented by an equivalent discrete time channel as shown in Figure 6.32.

### 6.6.3 Equivalent system state representations

By new indexing in Equation (6.43) we have:

\[
R_n = \sum_{k=0}^{L_1+L_2} \Gamma_{k-L_2}(n)c_{n+L_2-k} + b_n
\]  

(6.44)

If we denote \(S_n = (c_{n+L_2}, \ldots, c_{n-L_1+1})\) the state of the equivalent discrete time channel at time \(nT\) sample \(R_n\) depends on the channel state \(S_{n-1}\) and on the symbol \(c_{n+L_2}\). Therefore, the equivalent discrete time channel can be modeled as a Markov chain and its behavior can be represented by the trellis diagram shown in Figure 6.33.

### 6.6.4 Turbo equalization

In order to use a soft-input channel decoder, the symbol detector has to provide information about the reliability of the symbols estimated. This information may be obtained by using a soft-output Viterbi algorithm (SOVA) (the name often used for the algorithm described in Appendix 2.1), that associates an estimation of the logarithm of its likelihood ratio (LLR), \(\Lambda_1(c_n)\), to each symbol \(c_n\) detected:

\[
\Lambda_1(c_n) = \log \frac{\Pr\{c_n = +1|\mathbf{R}\}}{\Pr\{c_n = -1|\mathbf{R}\}}
\]  

(6.45)
where $R$ denotes the vector of samples that constitutes the observation. After deinterleaving, the SOVA decoder provides a new LLR value of $c_k$, $\Lambda_2(c_k)$, that may be derived by analogy with the calculations used in Chapter 2 and expressed in the form:

$$\Lambda_2(c_k) = \Lambda_1(c_k) + z_k \quad (6.46)$$

where $z_k$ is the extrinsic information associated with symbol $c_k$ and provided by the channel decoder (Figure 6.34).

The extrinsic information $z_k$ is another estimation of the LLR of symbol $c_k$ conditioned on the decoding step:

$$z_k = \log \frac{\Pr(c_k = +1 \mid \text{decoding})}{\Pr(c_k = -1 \mid \text{decoding})} \quad (6.47)$$

Hence, $z_k$ may be used through a feedback loop by the symbol detector after interleaving. This is the basis of the turbo equalization principle.

### 6.6.5 Viterbi algorithm

To evaluate the LLR of symbol $c_{n-L_1}$, the Viterbi algorithm used in the detector has to calculate a metric

$$\lambda_i = |R_n - r_n^i|^2 - 2\sigma_w^2 \log \Pr(c_{n-L_1} = i) \quad i = \pm 1 \quad (6.48)$$

where:

$$r_n^i = \sum_{k=0}^{L_1+L_2-1} \hat{g}_{k-L_2}(n) \cdot c_{n-L_2-k} + \hat{g}_{L_1}(n) \cdot i \quad i = \pm 1 \quad (6.49)$$
$\hat{r}_{k-L_2}(n)$, $0 \leq k \leq L_1 + L_2$ represents an estimation of quantity $\Gamma_k-L_2(n)$ and $\sigma^2_w$ denotes the variance of noise $w_n$, that is $\sigma^2_w = E[|w_n|^2]$.

The *a priori* probabilities $\Pr\{c_{n-L_1} = i\}$ used in Equation (6.48) may be estimated from the extrinsic information $z_{n-L_1}$, if we assume that

$$z_{n-L_1} = \log \frac{\Pr\{c_{n-L_1} = +1\}}{\Pr\{c_{n-L_1} = -1\}} \quad (6.50)$$

From Equation (6.50)

$$\Pr\{c_{n-L_1} = +1\} \approx \frac{\exp z_{n-L_1}}{1 + \exp z_{n-L_1}} \quad (6.51)$$
$$\Pr\{c_{n-L_1} = -1\} \approx \frac{1}{1 + \exp z_{n-L_1}}$$

Using Equations (6.51) and (6.48), metrics $\lambda_i^n$ are equal to:

$$\lambda_i^{+1} = \left| R_n - r_n^{+1} \right|^2 - \gamma z_{n-L_1} \quad (6.52)$$
$$\lambda_i^{-1} = \left| R_n - r_n^{-1} \right|^2$$

Note that the common term $\log(1 + \exp z_{n-L_1})$ has been suppressed in Equation (6.52). Coefficient $\gamma$ is a weight introduced to take into account variance $\sigma^2_w$ and the fact that the extrinsic information is only an *estimation* of the *a priori* probability. Its value depends on the signal to noise ratio, that is to say the reliability of the extrinsic information.

### 6.6.6 Iterative implementation of turbo equalization

The different processing stages in the turbo equalizer present a non-zero internal delay, so turbo equalizing can only be implemented in an iterative way. At each iteration $q$, a new value of extrinsic information is calculated and used by the symbol detector at the next iteration. Therefore, the turbo equalizer can be implemented in a modular pipelined structure, where each module is associated with one iteration. Then, performance in bit error rate (BER) terms is a function of the number of chained modules.

When extrinsic information is used by the symbol detector, it can be proved that, at iteration $q$, the LLR of symbol $c_n$, $\Lambda^q_1(c_n)$, may be expressed as:

$$\Lambda^q_1(c_n) = \hat{\Lambda}^q_1(c_n) + \gamma^q z_{n-L_1}^{-1} \quad (6.53)$$

where $\hat{\Lambda}^q_1$ is a term depending on the samples of observation $R$ and on $z_{n-L_1}^{-1}$, $k \neq n$, and $z_{n-L_1}^{-1}$ denotes the extrinsic information of symbol $c_n$ determined at iteration $q - 1$. If we apply the same approach as in turbo decoding described in Chapter 2, the quantity $\gamma^q z_{n-L_1}^{-1}$ provided by the channel decoder at the previous iteration has to be subtracted from $\Lambda^q_1(c_n)$ (see Equation (6.52)), as illustrated in Figure 6.35. Hence, after deinterleaving, the channel decoder input is in fact equal to:

$$\tilde{\Lambda}^q_1(c_n) = \Lambda^q_1(c_n) \bigg|_{z_{n-L_1}^{-1} = 0} \quad (6.54)$$

At the channel decoder output, the extrinsic information $z^q_k$ may also be written as follows, using Equation (6.46):

$$z^q_k = \Lambda^q_k(c_n) \bigg|_{\tilde{\Lambda}^q_1(c_n) = 0} \quad (6.55)$$

### 6.6.7 Performance

Performance of this device has been evaluated for a rate $R = 1/2$ recursive systematic encoder with constraint length $K = 5$ and generators $G_1 = 23$, $G_2 = 35$. Bits were interleaved in a non-uniform...
matrix whose dimensions are 64 by 64. The modulation used was a BPSK modulation, with a Nyquist filter whose transfer function \( H_s(f) \) was a raised cosine with a rolloff \( \alpha = 1 \), on both Gaussian and Rayleigh channels.

For both channels, \( M = 5 \) independent paths were considered, each with a mean power \( P_m = E[|A_m(n)|^2] \), so that the total mean power was normalized: \( \sum_{m=0}^{M-1} P_m = 1 \). The delays \( \tau_m \) were chosen as multiples of \( T (\tau_m = mT) \), since \( h_s[(k - m)T] = \delta_{k-m,0} \). The coefficients for the Gaussian channel were chosen equal to:

\[
\begin{align*}
\Gamma_0(n) &= \sqrt{0.45}, & \Gamma_1(n) &= \sqrt{0.25}, & \Gamma_2(n) &= \sqrt{0.15}, \\
\Gamma_3(n) &= \sqrt{0.10}, & \Gamma_4(n) &= \sqrt{0.05}
\end{align*}
\]

For the Rayleigh channel, the five paths had equal mean power (\( P_i = 1/M, \forall i \in [1, M] \)). A parameter \( BT \), which is the product of the Doppler bandwidth and the symbol duration, fixes the variation velocity of the channel: the smaller \( BT \) is, the more slowly the channel parameters vary during a time interval symbol.

The discrete time equivalent channel was modeled by a 16-state trellis, and the symbol detector was working on the SOVA algorithm. The channel coefficients \( \Gamma_k(n) \) were supposed perfectly known. After deinterleaving, the soft estimations provided by the SOVA detector were used by the decoder, which also worked on a 16-state trellis and the SOVA algorithm. The extrinsic information extracted from the decoder was used by the symbol detector according to the principle depicted in Figure 6.35.

The BER was computed as a function of signal to noise ratio \( E_b/N_0 \), where \( E_b \) is the mean energy received per information bit \( d_k \) and \( N_0 \) is the noise power bilateral spectral density. The signal to noise ratio \( E_b/N_0 \) may be expressed as:

\[
\frac{E_b}{N_0} = \frac{\sum_{m=0}^{M-1} P_m}{\sigma_b^2}
\]  

The results are shown in Figures 6.36–6.38 for different numbers of iterations \( n \). One can see that even for three iterations the performance curves approach the BER curve with no ISI.

More details on turbo equalization can be found in [94–103].

6.7 KALMAN FILTER BASED JOINT CHANNEL ESTIMATION AND DATA DETECTION OVER FADING CHANNELS

In this section we consider the problem of joint channel and data estimation in a fading channel based on using a Kalman-type estimator. The general system block diagram is shown in Figure 6.39. More details can be found in [104–107]. The presentation in this section is based on [104]. The channel model generator is shown in Figure 6.40.
Figure 6.36 Performance of turbo equalization over a Gaussian channel (convolutional encoding with $K = 5$).

Figure 6.37 Performance of turbo equalization over a Rayleigh channel with $BT = 0.1$ (convolutional encoding with $K = 5$).
Figure 6.38 Performance of turbo equalization over a Rayleigh channel with $BT = 0.001$ (convolutional encoding with $K = 5$).

Figure 6.39 The signal model for the baseband communication system.

Figure 6.40 The fading channel model.
6.7.1 Channel model

If the filter $P(z)$ in Figure 6.40 is modeled as

$$P(z) = \frac{D}{1 - Az^{-1} - Bz^{-2} - Cz^{-3}} \quad (6.57)$$

then at sampling time $k$, the CIR, $h_k$, a complex Gaussian vector, is

$$h_k = (h_{k,0}, h_{k,1}, \ldots, h_{k,\beta})^T \quad (6.58)$$

truncated to a finite length of $(\beta + 1)$. By considering the third-order approximation of Equation (6.57), an autoregressive (AR) representation for the CIR can be introduced as

$$h_k = Ai h_{k-1} + Bi h_{k-2} + Ci h_{k-3} + Di w_k \quad (6.59)$$

where $I$ is the identity matrix, $w_k$ is a zero mean white complex circularly symmetric Gaussian process with the covariance matrix defined as $E(w_k w_k^T) = Q \delta_{kl}$ and $w_k^T$ is the conjugate transpose of $w_k$. CIR at time $k$ depends on its three consecutive previous values.

The state of such a system is a vector composed of three consecutive channel impulse responses as

$$x_k = (h_k^T, h_{k-1}^T, h_{k-2}^T)^T \quad (6.60)$$

Using Equations (6.60) and (6.59) gives

$$x_{k+1} = \begin{bmatrix} A & B & C \\ I & 0 & 0 \\ 0 & I & 0 \end{bmatrix} x_k + \begin{bmatrix} D & 0 \\ 0 & 0 \end{bmatrix} w_k \quad (6.61)$$

$$x_{k+1} = Fx_k + Gw_k \quad (6.62)$$

where $F$ and $G$ are $3(\beta + 1) \times 3(\beta + 1)$ and $3(\beta + 1) \times (\beta + 1)$ matrices given in Equation (6.61). $F$ is called the state transition matrix and $G$ is the process noise coupling matrix.

6.7.2 The received signal

If we introduce a $1 \times (\beta + 1)$ vector $H_k$

$$H_k = (a_k, a_{k-1}, a_{k-2}, \ldots, a_{k-\beta}, 0, \ldots, 0) \quad (6.63)$$

where $a_k$ is the transmitted data sequence, then the received signal $z_k$ becomes

$$z_k = H_k x_k + n_k \quad (6.64)$$

The Kalman filter is optimum for minimizing the mean square estimation error [108], in the above linear time varying system. The algorithm is complex and, in practice, suboptimal methods are more advantageous due to their implementation simplicity.

6.7.3 Channel estimation alternatives

In this segment the same assumptions are used as in [104]:

- To avoid the decision delay, the PSP method [109] is used. In this method, there is a channel estimate for every possible sequence.
- The estimated channel impulse response will be used to compute the branch metrics in the trellis diagram of the Viterbi algorithm.
- The number of required estimators is limited to the number of survivor branches (or the number of states) in the Viterbi algorithm trellis diagram.
In PSP each surviving path keeps and updates its own channel estimate. This method eliminates the problem of decision delay, and in order to employ the best available information for data detection the data sequence of the shortest path is used for channel estimation along the same path.

- The data communication system is based on the IS-136 standard presented in Chapter 1.
- The modulation is QPSK with four possible symbols \((-1 \pm j)\) and a symbol rate of 25 ksymbols/s.
- The differentially encoded data sequence is arranged into 162 symbol frames.
- The first 14 symbols of each frame make up a training preamble sequence to help the adaptation of the channel estimator.

For the shaping filter at the transmitter, a finite impulse response (FIR) filter approximates a raised cosine frequency response with an excess bandwidth of 25\% (slightly different from the 35\% selected in IS-136).

A two-ray fading channel model as described in Figure 6.40, where one ray has a fixed delay equal to one symbol period.

- The multiplicative coefficients of $\alpha_0$ and $\alpha_1$ are produced at the output of two fading filters, where the inputs are two independent zero mean complex Gaussian processes with equal variances.

- The length of the discrete impulse response of the shaping filter is set equal to the symbol interval so that the ISI at the receiver is only due to the multipath nature of the channel.

- The total length of the CIR is two symbol intervals, i.e. $\beta + 1 = 2$ if there is one sample per symbol interval.

- Therefore, there is ISI between two neighboring symbols and there are four possible states in the trellis diagram.

- The LMS algorithm, RLS algorithm or the Kalman filter can be used to estimate the channel impulse response.

### 6.7.4 Implementing the estimator

#### 6.7.4.1 The Kalman filter algorithm [108]

The measurement update equations are:

\[
\hat{x}_{k+1} = \hat{x}_k + K_k(z_k - H_k \hat{x}_k)
\]

\[
K_k = P_k H_k^T R_k^{-1}
\]

\[
R_k = H_k P_k H_k^T + N_0
\]

\[
P_{k+1} = P_k - K_k H_k P_k
\]

The time update equations are:

\[
\hat{x}_{k+1} = F \hat{x}_{k|k}
\]

\[
P_{k+1} = F P_{k|k} F^T + G Q G^T
\]

#### 6.7.4.2 The RLS algorithm [110]

\[
\hat{x}_{k+1} = \hat{x}_k + K_k(z_k - H_k \hat{x}_k)
\]

\[
K_k = P_k H_k^T R_k^{-1}
\]

\[
R_k = H_k P_k H_k^T + \lambda
\]

\[
P_{k+1} = \lambda^{-1}(P_k - K_k H_k P_k)
\]
In the above equations the measurement updated estimate $\hat{x}_{k|k}$ is the linear least squares estimate of $x_k$ given observations $\{z_0, z_1, \ldots, z_k\}$. $\hat{x}_{k+1}$ is the time updated estimate of $x_k$ given observations $\{z_0, z_1, \ldots, z_k\}$. The corresponding error covariance matrices of these estimations are

$$P_{k|k} = E[(x_k - \hat{x}_{k|k})(x_k - \hat{x}_{k|k})^T]$$

$$P_k = E[(x_k - \hat{x}_k)(x_k - \hat{x}_k)^T]$$

(6.68)

In the RLS algorithm, $\lambda$ is a parameter called the forgetting factor.

### 6.7.5 The Kalman filter

The estimator is based on two parts: measurement update equations and time update equations. The RLS algorithm is essentially identical to the measurement update equations of the Kalman filter. The Kalman filter can be used for channel estimation when some a priori information about the channel is available at the receiver (i.e. the $F$ and $G$ matrices). The RLS algorithm, which is a suboptimal method, does not require this a priori information and its computational complexity is less than the Kalman filter.

### 6.7.6 Implementation issues

At the same precision, mathematically equivalent implementations can have different numerical stabilities, and some methods of implementation are more robust against roundoff errors.

In the Kalman filter and the RLS algorithm the estimation depends on the correct computation of the error covariance matrix. In an ill-conditioned problem, the solution will not be equal to the covariance matrix of the actual estimation uncertainty. Factors contributing to this problem are: large matrix dimensions, a growing number of arithmetic operations and poor machine precision. Solutions to these problems are factorization methods and square root filtering [110, 111].

### 6.7.6.1 Square root filtering

Some implementations are more robust against roundoff errors and ill-conditioned problems. The so-called square root filter implementations have generally better error propagation bounds than the conventional Kalman filter equations [112]. In the square root forms of the Kalman filter, matrices are factorized and triangular square roots are propagated in the recursive algorithm to preserve the symmetry of the covariance (information) matrices in the presence of roundoff errors.

Different techniques are used for changing the dependent variable of the recursive estimation algorithm to factors of the covariance matrix. A Cholesky factor of a symmetric non-negative definite matrix $M$ is a matrix $C$ such that $CC^T = M$. Cholesky decomposition algorithms solve for a diagonal factor and either a lower triangular factor $L$ or an upper triangular factor $U$ such that $M = UD_UU^T = LD_LD_L^T$, where $D_L$ and $D_U$ are diagonal factors with non-negative diagonal elements.

The square root methods propagate the L-D or U-D factors of the covariance matrix rather than the covariance matrix. The propagation of square root matrices implicitly preserves the Hermitian symmetry and non-negative definiteness of the computed covariance matrix.

The condition number $\kappa(P) = [\text{eigenvalue}_{\text{max}}(P)/\text{eigenvalue}_{\text{min}}(P)]$ of the covariance matrix $P$ can be written as

$$\kappa(P) = \kappa(LDL^T) = \kappa(DD^T) = [\kappa(B)]^2$$

where $B = LD_1^{1/2}$. The condition number of $B$ used in the square root method is much smaller than the condition number of $P$ and this leads to improved numerical robustness of the algorithm.

In the square root method, the dynamic range of the numbers entering into computations will be reduced. Loosely speaking, we can say that the computations which involve numbers ranging between $2^{-N}$ to $2^+N$ will be reduced to ranges between $2^{-N/2}$ to $2^+N/2$, which would halve the
length of required mantissa used in signal processing. All of these will directly affect the accuracy of computer computations.

Performance results are shown in Figures 6.41 and 6.42. Different Kalman algorithms will demonstrate superior performance only if the precision in the calculation of Equation (6.65) is high enough. From Figures 6.41 and 6.42, this means if the quantization is precise enough, requiring a mantissa length of 20 bits or higher.

### 6.8 EQUALIZATION USING HIGHER ORDER SIGNAL STATISTICS

In order to speed up the algorithms, equalization based on higher order signal statistics may be used [113–119]. In this section we discuss a group of such algorithms derived from the cost function defined as average entropy of the set of the signal samples [113]. Maximization of the average entropy is obtained by minimizing the mutual information of the set of signal samples which result in minimum ISI. The block diagram of the equalizer based on joint entropy minimization (JEM) principles is shown in Figure 6.43.

#### 6.8.1 Problem statement

For input data symbols \( s(n) \), discrete time channel output \( x(n) \) and the channel response \( c(n) \), the received baseband symbol can be represented as

\[
x(n) = \sum_{k=0}^{N} c(k)s(n - k)
\]  

(6.69)
Figure 6.42 The effects of changing the word length on BER ($E_b/N_0 = 15\, \text{dB}$). Kalman 1 represents the WGS method, Kalman 2 is for the correlation method and Kalman 3 is the direct method, the PSP method is employed for detection.

Figure 6.43 Block diagram of channel and JEM-DFE structure.
We will assume that input symbols are independently identically distributed (i.i.d.) and zero mean, which is true in a digital communication system, and that $c(0) = 1$.

Signal $y(n)$ at the output of the equalizer (see Figure 6.43) can be presented as

$$y(n) = x(n) + \sum_{k=1}^{N} b(k)r(n - k)$$

$$= s(n) + \sum_{k=1}^{N} c(k)s(n - k) + \sum_{k=1}^{N} b(k)r(n - k)$$

$$= s(n) + \sum_{k=1}^{N} c(k)s(n - k) + \sum_{k=1}^{N} b(k)g(y(n - k)) \quad (6.70)$$

Non-linearity $g(\cdot)$ is a strictly monotone (increasing or decreasing) differentiable function. A more general DFE structure usually consists of a feedforward finite impulse response (FIR) filter followed by the feedback FIR filter given in Figure 6.43. As discussed earlier in the chapter, the main purpose of the feedforward filter is to eliminate the precursor ISI and the feedback filter cancels the postcursor ISI. For simplicity, only the feedback part is considered. This is not a hurdle since it is possible to separate the adaptation of the feedback part from the feedforward part.

### 6.8.2 Signal model

By introducing auxiliary inputs and outputs we have

$$y(n) = B \cdot x(n) \quad (6.71)$$

where

$$x(n) = [x(n) \quad r(n - 1) \ldots r(n - N)]^T$$

$$y(n) = [y(n) \quad r(n - 1) \ldots r(n - N)]^T \quad (6.72)$$

and

$$B = \begin{bmatrix}
1 & b(1) & \ldots & b(N) \\
0 & 1 & \ddots & 0 \\
\vdots & \ddots & \ddots & \ddots \\
0 & \ddots & \ddots & 1
\end{bmatrix} \quad (6.73)$$

In the presence of ISI (i.e. dependence), the entropy of the received symbols is smaller than when they are independent. Thus, maximizing the joint entropy of the equalizer outputs will result in removing the ISI.

### 6.8.3 Derivation of algorithms for DFE

So, the algorithm is based on maximizing the joint entropy of the equalizer output $r(n)$. The joint entropy of $r(n) = g(y(n))$ denoted by $H[r_1(n), \ldots, r_{N+1}(n)]$ is defined as [78, 120 (Chapter 6)]:

$$H[r_1(n), \ldots, r_{N+1}(n)] \triangleq - E \{ \ln f_r(r(n)) \}$$

$$= E \{ \ln |J| \} - E \{ \ln f_y(y(n)) \}$$

$$= E \{ \ln |J| \} - E \{ \ln |B|^{-1} f_x(x(n)) \} \quad (6.74)$$

$$= E \{ \ln |J| \} - E \{ \ln f_x(x(n)) \}$$
where \( |J| \) is the absolute value of the Jacobian of the transformation

\[
J = \det \begin{bmatrix}
\frac{\partial r_1(n)}{\partial y_1(n)} & \cdots & \frac{\partial r_1(n)}{\partial y_{N+1}(n)} \\
\vdots & \ddots & \vdots \\
\frac{\partial r_{N+1}(n)}{\partial y_1(n)} & \cdots & \frac{\partial r_{N+1}(n)}{\partial y_{N+1}(n)}
\end{bmatrix}
\]

(6.75)

\[
J = \det \begin{bmatrix}
\frac{\partial r(n)}{\partial y(n)} & \cdots & \frac{\partial r(n)}{\partial y(n) - N} \\
\vdots & \ddots & \vdots \\
\frac{\partial r(n - N)}{\partial y(n)} & \cdots & \frac{\partial r(n - N)}{\partial y(n) - N}
\end{bmatrix} = \frac{\partial r(n)}{\partial y(n)}
\]

(6.76)

The quantity \( r_i(n)[x_i(n)] \) is the \( i \)th component of the vector \( r(n)[x(n)] \) and \( f_i(r(n))[f_i(x(n))] \) is the joint density function of the input vector \( r(n)[x(n)] \).

Assume that the previous decisions are correct, i.e.

Assumption: \( r(n - k) = s(n - k), \ k = 1, \ldots, N \) \hspace{1cm} (6.77)

Under Assumption (6.77), \( r(n - i) \) are independent of \( r(n - j), \ i > j \) and thus Equation (6.75) follows from Equation (6.75). The joint entropy can also be expressed as [120, Chapter 15]:

\[
H[r_1(n), \ldots, r_{N+1}(n)] = \sum_{i=1}^{N+1} H[r_i(n)] - I[r_1(n), \ldots, r_{N+1}(n)]
\]

(6.78)

Maximizing the joint entropy of \( r(n) \) is the same as maximizing the first term in Equation (6.78) while minimizing the mutual information \( I[r_1(n), \ldots, r_{N+1}(n)] \). Since the previous decisions that the outputs of the non-linear function \( g(\cdot) \) are the same as the transmitted symbols \( s(n - k), \ k = 1, \ldots, N, \ E[\ln f_i(x(n))] \) in Equation (6.74) can be considered to be a constant with respect to the feedback filter coefficients \( b(k), \ k = 1, \ldots, N \). Then, maximizing \( E[\ln |J|] \) is equivalent to maximizing \( H[r_1(n), \ldots, r_{N+1}(n)] \). In doing so, the statistical dependence between the current output of the non-linear function \( r(n) \) and the previous outputs \( r(n - k), \ k = 1, \ldots, N \) can be reduced, which leads to ISI suppression.

### 6.8.4 The equalizer coefficients

The joint entropy \( H[r_1(n), \ldots, r_{N+1}(n)] \) is a non-linear function of the unknown equalizer coefficients and does not lead easily to a closed form solution. A gradient descent algorithm is used for maximizing Equation (6.74). The equalizer coefficients are then updated iteratively

\[
b^{n+1}(k) = b^n(k) + \mu E \left\{ \frac{\partial \ln |J|}{\partial b^n(k)} \right\}
\]

(6.79)

where \( \mu \) is the positive step size. This equation can be further simplified based on the choice of non-linearity used in the decision device.
6.8.5 Stochastic gradient DFE adaptive algorithms

Despite its superiority over the linear equalizer and its popularity, a major drawback of the DFE is that it suffers from error propagation. Any previous decision errors at the output of the decision device will produce the symbol estimate $y(n)$, whose ISI is not completely eliminated by the feedback filter. This error in turn will affect future decisions. An intuitive choice is one where we replace the hard limiter with a softer function. It is hoped that the errors in $r(n)$ can be reduced by using soft decisions.

The update equation given in (6.79) depends on the mapping used in the soft decision device. Various non-linear functions may be used for the soft decision device (function $g(\cdot)$ shown in Figure 6.43). We use two different functions for $g(\cdot)$ to derive new blind algorithms for DFE.

6.8.5.1 Equivalence of JEM and ISIC

First choice for the non-linear function is

$$g(n) = \alpha \cdot \tanh[\beta \cdot n] \quad (6.80)$$

resulting in the following

$$\frac{\partial \ln |J|}{\partial b(k)} = \frac{\partial \ln |\partial r(n) / \partial y(n)|}{\partial b(k)}$$

$$= -2 \beta \tanh[\beta \cdot y(n)] r(n - k) \quad (6.81)$$

$$= -\frac{2 \beta}{\alpha} r(n) r(n - k)$$

The gradient is a function of the non-linearity used in the decision device that provides a tool via which the algorithm characteristics can be varied. A new adaptive blind algorithm for DFE based on JEM can be obtained as follows:

JEM-1:

$$b^{n+1}(k) = b^n(k) - \mu r(n) r(n - k) \quad (6.82)$$

Taylor series (TS) expansion of $r(n)$ leads to

$$r(n) = \alpha \cdot \tanh[\beta \cdot y(n)]$$

$$= \alpha \beta y(n) - \frac{\alpha \beta^3}{3} y^3(n) + \frac{2 \alpha \beta^5}{15} y^5(n) + O(y^6(n)) \quad (6.83)$$

For simplicity only the first few terms of the expansion are considered. Further variations of JEM-1 can be obtained by substituting Equation (6.83) into Equation (6.82). By using only the first term in Equation (6.83) with $\alpha = 3$ and $\beta = 1/3$, a simpler update equation for the feedback filter coefficients is

JEM-2:

$$b^{n+1}(k) = b^n(k) - \mu y(n) r(n - k) \quad (6.84)$$

JEM-2 is exactly the same as the ISIC algorithm [121] which uses soft decision feedback. Here, the same update equation is arrived at via the motivation to maximize the entropy of the observations at the output of a soft decision device.

6.8.5.2 Equivalence of JEM and decorrelation criterion

Another possible variation of Equation (6.82) is to use the TS approximation for both $r(n)$ and $r(n - k)$.

$$r(n) r(n - k) = \alpha^2 \beta^2 y(n) y(n - k) - \frac{\alpha^2 \beta^4}{3} [y^3(n) y(n - k)$$

$$+ y^3(n - k) y(n)] + \cdots \quad (6.85)$$
By using only the first term in Equation (6.85) with $\alpha = 3$ and $\beta = 1/3$, we have

$$JEM-3: \quad b^{n+1}(k) = b^n(k) - \mu y(n)y(n-k)$$  \hspace{1cm} (6.86)$$

JEM-3 is exactly the same as that for the adaptive blind algorithm based on the decorrelation criterion (DECA) [122].

### 6.8.5.3 JEM and CMA-DFE

Another function that has transfer characteristics similar to the hyperbolic tangent is the function with a cubic non-linearity

$$g(y(n)) = \alpha \cdot y(n) + \beta \cdot y^3(n)$$  \hspace{1cm} (6.87)$$

With the above mapping we have

$$\frac{\partial \ln |J|}{\partial b(k)} = \frac{6\beta y(n)}{\alpha + 3\beta y^2(n)} r(n-k)$$  \hspace{1cm} (6.88)$$

$$\frac{\partial \ln |J|}{\partial b(k)} = \left[ \frac{6\beta}{\alpha} y(n) \left\{ 1 - \frac{3\beta}{\alpha} y^2(n) + \frac{9\beta^2}{\alpha} y^4(n) \right\} + O(y^7(n)) \right] r(n-k)$$  \hspace{1cm} (6.89)$$

Dropping terms of order greater than three gives:

$$JEM-4: \quad b^{n+1}(k) = b^n(k) + \mu y(n) \left\{ 1 - \frac{3\beta}{\alpha} y^2(n) \right\} \cdot r(n-k)$$  \hspace{1cm} (6.90)$$

With $\alpha = 3$ and $\beta = 1$, Equation (6.90) coincides with the CMA-DFE algorithm, provided the soft decisions given by Equation (6.87) are used in the CM algorithm for a DFE. The original CMA-DFE uses hard decisions [123, 124].

### 6.8.6 Convergence analysis

#### 6.8.6.1 Alternative JEM-DFE scheme and Bussgang-type algorithm

Since soft decisions can possibly smooth the error surface and allow the algorithm to escape from local minima, the authors in [8] suggest the use of soft decisions at the start of equalization. These can then be replaced with a hard decision device after a few iterations. It is difficult to decide when one can switch from soft to hard decisions. A simple technique based on a modified JEM-DFE scheme is suggested to overcome this difficulty (Figure 6.44).

![Figure 6.44 Block diagram of channel and alternative JEM-DFE structure.](image)
For the update equations (for example, (6.82) and (6.84)), the soft decision device outputs, \( r(n) \) and \( r(n-k) \), are used. On the other hand, the hard decisions \( \hat{r}(n-k) \) in Figure 6.43 are used as inputs to the feedback filter instead of \( r(n-k) \); see Figure 6.43. In this way, soft and hard decision modes can be combined. A structure similar to JEM-DFE (see Figure 6.44) was used in [124] and will be used in what follows.

6.8.6.2 JEM-DFE structure in Figure 6.44 and classical Bussgang-type DFE

DFE can be considered as a linear equalizer [124]. The usual DFE structure consists of the feedforward filter followed by the feedback filter. The symbol estimate \( y(n) \) can be expressed as:

\[
y(n) = \sum_{k=-N_f}^{0} f(k)x(n-k) + \sum_{k=1}^{N_b} b(k)\hat{r}(n-k)
\]

\[
= \sum_{k=-N_f}^{N_b} w(k)\hat{x}(n-k)
\]

(6.91)

The input to the equalizer is

\[
\hat{x}(n-k) = \begin{cases} 
  x(n-k), & \text{when } -N_f \leq k \leq 0 \\
  \hat{r}(n-k), & \text{when } 1 \leq k \leq N_b
\end{cases}
\]

(6.92)

The coefficients of the equalizer are

\[
w(n-k) = \begin{cases} 
  f(n-k), & \text{when } -N_f \leq k \leq 0 \\
  b(n-k), & \text{when } 1 \leq k \leq N_b
\end{cases}
\]

(6.93)

and \( N_f \) and \( N_b \) are the feedforward and feedback filter lengths, respectively.

6.8.6.3 Bussgang-type algorithm [125 (Chapter 2)]

The algorithm is defined by

\[
w^{n+1}(k) = w^n(k) + \mu \{\hat{g}(y(n)) - y(n)\} \hat{x}(n-k)
\]

(6.94)

where \( \hat{g}(y(n)) \) is some non-linear function of \( y(n) \). Provided that \( N_f \) and \( N_b \) in Equation (6.94) are large enough, the symbol estimate \( \{y(n)\} \) is Bussgang (see [126, Chapter 18], [127]). A stochastic process \( \{y(n)\} \) is said to be a Bussgang process if it satisfies the condition \( E\{y(n)y(n+k)\} = E\{y(n)g(y(n+k))\} \) where the function \( g(\cdot) \) is a zero memory non-linearity.

Since the feedforward and feedback parts can be updated separately, consider only the feedback part of Equation (6.94). Then, the Bussgang-type DFE algorithm for the feedback part is

\[
b^{n+1}(k) = b^n(k) + \mu \{\hat{g}(y(n)) - y(n)\} r(n-k)
\]

(6.95)
Comparing Equation (6.95) to Equations (6.82) and (6.90), \(-r(n)\) in Equation (6.82), \(-y(n)\) in Equation (6.84) and \(y(n)\{1 - (3\beta/\alpha)y^2(n)\}\) in Equation (6.90) may be viewed to be equivalent to the error term \(\{\hat{g}(y(n)) - y(n)\}\) in Equation (6.95). Thus, JEM-1, JEM-2 and JEM-4 can be considered to be special cases of the Bussgang-type DFE. By comparing Equation (6.95) to Equation (6.86), however, we note that JEM-3 cannot be categorized as a Bussgang-type algorithm.

### 6.8.6.4 Convergence of JEM algorithms

JEM-1 and JEM-2 are guaranteed to converge for super-Gaussian inputs but not for sub-Gaussian. The input \(\{s(n)\}\) is sub-Gaussian (super-Gaussian) if the kurtosis \(\gamma_s = E\{s(n)^4\} - 3[E\{s(n)^2\}]^2\) is less than zero (greater than zero). For example, for a BPSK signal, \(\gamma_s = -2\) and hence it is sub-Gaussian.

Extensive simulations show, however, that these algorithms converge to the right solution in almost all the cases. Convergence of JEM-3 is not considered here since it cannot be categorized as a Bussgang-type algorithm. JEM-4 is guaranteed to converge. When \(\alpha = \infty\) and \(\beta = 1/\alpha = 0\), JEM-2 coincides with JEM-3.

### 6.8.7 Kurtosis-based algorithm

The JEM approach cannot guarantee that the maximization of the joint entropy leads to minimization of the mutual information. To compute the mutual information, the probability density function of \(r(n)\), the output of the decision device, is required. The approximation of the mutual information (or contrast function) was derived by Comon (see [128]). Maximization of the contrast function, or minimization of the mutual information of the equalizer outputs, is equivalent to minimization of the sum of the autocumulants squared. This result was obtained by using the Edgeworth expansion to separate the constant mixtures of the independent input signals [128].

#### 6.8.7.1 Modified signal model

Under Assumption (6.77), Equation (6.72) can be rewritten as:

\[
x(n) = [x(n) \ s(n-1) \ \cdots \ s(n-N)]^T
\]

\[
y(n) = [y(n) \ s(n-1) \ \cdots \ s(n-N)]^T
\]

The new equivalent channel model is

\[
x(n) = C \cdot s(n)
\]

where

\[
s(n) = [s(n) \ s(n-1) \ \cdots \ s(n-N)]^T
\]

and

\[
C = \begin{bmatrix}
1 & c(1) & \cdots & c(N) \\
0 & 1 & \ddots & 0 \\
\vdots & \ddots & \ddots & \ddots \\
0 & \ddots & \ddots & 1
\end{bmatrix}
\]

The matrix \(C\) is full rank so there exists a unique solution to Equation (6.97). Here, the mixture matrix \(C\) and the source vector \(s(n)\) are partially unknown. Equalization is attained by estimating the matrix \(B\) in Equation (6.71). It can be easily shown that \(B\) is the inverse matrix of \(C\) when \(c(n-k) = -b(n-k), \ k = 1, \ldots, N\). To find the solution \(B\), several contrast functions can be used (see [128] and [129]).
6.8.7.2 **Contrast function proposed in [129]**

In this case we have

\[ J = \sum_{k=1}^{N+1} C_{4y} (0, 0, 0, k) = C_{4y} (0, 0, 0) + \text{Constant} \]  

(6.100)

where \( C_{4y} (0, 0, 0) \) is the kurtosis (all zero lag of the fourth-order cumulant) of \( y(n) \). A criterion similar to one defined in Equation (6.100) for a single output FIR channel was originally proposed for multiinput multioutput (MIMO) channel equalization with MIMO-DFEs in [130]. The cost function for the present simplified case with only a single channel is

\[ J_{AC} = C_{4y} (0, 0, 0) \]  

(6.101)

The update equation for the feedback filter coefficients based on the criterion in Equation (6.101) is:

\[
b_{n+1}(k) = b_n(k) + \mu \text{sgn} \{ \gamma \} \cdot \left[ \hat{E} \{ y(n)^3 r(n - k) \} - 3 \hat{E} \{ y(n)^2 \} \right. \\
\left. \cdot \hat{E} \{ y(n) r(n - k) \} \right]
\]  

(6.102)

where \( \hat{E} \{ \cdot \} \) is the estimate of \( E \{ \cdot \} \). Note that the contrast-based cost function of Equation (6.100) is similar to the autocumulant (AC) criterion of Equation (6.101). Since the preprocessing stage (prewhitening of the observations) is missing in the AC approach, however, the proposed criterion can only be thought of as approximately minimizing the mutual information. Simulation examples indicate that the equalizer based on the AC criterion converges much faster than most existing algorithms.

6.8.7.3 **Performance examples [113]**

**Example 1**

An FIR non-minimum phase channel with impulse response given by

\[ c(n) = \delta[n] + 2\delta[n - 1] \]

is used. The channel input \( s(n) \) is a sequence of 2000 independent BPSK symbols, uniformly distributed over \{1, -1\}. The received symbols \( x(n) \) consist of the distorted symbols \( s(n) \) and additive noise at 20 dB. This data is used to update the one-tap feedback filter (\( b(1) \)), which is initialized with a zero. The results are shown in Figures 6.45 and 6.46.

**Example 2**

The performance of the adaptive blind algorithms was investigated with a multipath fading channel with impulse response

\[ c(t) = \sum_{i=1}^{L_d} \epsilon_i p(t - \tau_i) \]

where \( p(t) \) is the pulse shape (raised cosine pulse). The delay \{\( \tau_i \)\} is statistically independent and uniformly distributed in \([0, 3T]\) where \( T \) is the symbol interval. The attenuations \{\( \epsilon_i \)\} are independent zero mean Gaussian variables and the number of paths is six.

The equivalent discrete time baseband channel has a length of four symbol intervals (\( c(k), k = 0, \ldots, 3 \)). The channel coefficients \( c(k) \) were normalized by \( c(0) \). An array of three hundred sets of channels drawn from channel parameter distributions was generated. 10,000 independent uniformly distributed input symbol sequences \{\( s(n) \)\} over \{1, -1\} were used for each channel. For the sake of fair comparison, the step size was chosen to be the same for all the algorithms.
Figure 6.45 MSE comparison of ISIC, JEM-2, CMA-DFE and JEM-4: \( g(\cdot) = 3 \cdot \tanh[1/3y(n)] \) for JEM-2, \( g(y(n)) = 3 \cdot y(n) + y^3(n) \) for JEM-4, \( \mu_{\text{ISIC}} = \mu_{\text{JEM-2}} = \mu_{\text{CMA-DFE}} = 0.01 \), \( \mu_{\text{JEM-4}} = 0.001 \) averaged over 100 Monte Carlo runs. SNR = 20 dB [113] © 1998, IEEE.

Figure 6.46 MSE comparison of AC, decorrelation algorithm (DECA) and JEM-1: \( g(\cdot) = 3 \cdot \tanh[1/3y(n)] \), \( \mu_{\text{AC}} = \mu_{\text{DECA}} = \mu_{\text{JEM-1}} = 0.01 \), averaged over 100 Monte Carlo runs. SNR = 20 dB [113] © 1998, IEEE.
Table 6.3  SIR Comparison of (blind) DFE algorithms: $g(\cdot) = 3 \tanh \left( \frac{1}{3} y(n) \right)$ for JEM-1 and JEM-2, $g(\cdot) = 3 y(n) + y^3(n)$ for JEM-4, and $\mu = 1e^{-4}$ for all algorithms, averaged over 185 simulated FIR channels: noise-free case [113] © 1998, IEEE

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>SIR (dB)</th>
<th>Number of successes among a set of 300 channels</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unequalized</td>
<td>0.80</td>
<td>300</td>
</tr>
<tr>
<td>ISIC</td>
<td>4.87</td>
<td>300</td>
</tr>
<tr>
<td>JEM-2</td>
<td>5.70</td>
<td>300</td>
</tr>
<tr>
<td>CMA-DFE</td>
<td>10.88</td>
<td>295</td>
</tr>
<tr>
<td>JEM-4</td>
<td>58.92</td>
<td>185</td>
</tr>
<tr>
<td>DECA</td>
<td>6.16</td>
<td>300</td>
</tr>
<tr>
<td>AC</td>
<td>22.14</td>
<td>292</td>
</tr>
<tr>
<td>JEM-1</td>
<td>5.07</td>
<td>300</td>
</tr>
</tbody>
</table>

### 6.8.8 Performance results

Table 6.3 shows the results. Non-JEM-type algorithms yield about 5 dB–22 dB improvement in SIR. JEM-type algorithms exhibit an 11 dB–59 dB improvement. The JEM-4 algorithm is sensitive to step size. The third column of Table 6.3 shows the number of successful convergences. The JEM-4 algorithm became unstable with 115 simulated channels among 300. When the robustness and performance are considered simultaneously, JEM-1, JEM-2 or DECA are winners among the algorithms.

### REFERENCES


322 CHANNEL ESTIMATION AND EQUALIZATION


The basic principles of the orthogonal frequency division multiplexing (OFDM) concept are presented in Chapter 1. Within this chapter we introduce further details with the main focus on synchronization, channel estimation, space–time frequency coding, pick to average power ratio (PAPR) and some comparisons of efficiency in dealing with multipath propagation impacts by using equalization and TDMA or OFDM signal formats.

7.1 TIMING AND FREQUENCY OFFSET IN OFDM

As already indicated in Chapter 1, the transmitted OFDM signal can be represented as

$$x(t) = \sum_{k=-k_1}^{N+k_2+1} \sum_{n=0}^{N-1} D_k e^{j 2\pi \frac{n}{N} w\left(t - \frac{k}{f_s}\right)}$$

for $$\frac{-k_1}{f_s} < t < (N+k_2)/f_s$$, where $$k_1$$ and $$k_2$$ are pre- and post-fix lengths and $$w(t)$$ is the time domain window function. The received signal $$r(t)$$ is filtered and sampled at the rate, or multiples of, $$1/T$$.

The sampled signal at the output of the receiver FFT with ideal channel can be represented as a convolution:

$$y_n = \left[ \sum_{k=-\infty}^{\infty} X_c\left(f + \frac{Nk}{T}\right) \right] \otimes W(fT) \bigg|_{f=\frac{n}{T}}$$

where $$X_c(f)$$ is the Fourier transform of the periodically repeated analog equivalent of the signal generated by the transmitter’s IFFT.

$$X_c(f)$$ is a line spectrum at $$k/T$$ and $$W(f)$$ is the Fourier transform of window function $$w(t)$$. 
Assuming that the sampling time has a relative phase offset of $\tau$ and that the offset does not change during one OFDM symbol, the sampled received signal for a non-dispersive channel can be simplified to:

$$y_k = \sum_{n=0}^{N-1} D_n e^{j\varphi} e^{j2\pi \frac{\alpha}{f_s} k \tau} e^{jx}$$  (7.3)

where $\varphi$ represents envelope delay distortion. After the Fourier transform at the receiver we have

$$\tilde{D}_m = \frac{1}{N} \sum_{k=0}^{N-1} \sum_{n=0}^{N-1} D_n e^{j2\pi \frac{\alpha}{f_s} k \tau} e^{j\psi+2\pi \frac{k}{f_s}} e^{-j2\pi \frac{\alpha}{f_s} k}$$

$$= \sum_{n=0}^{N-1} D_n e^{j(\psi+2\pi \frac{n}{f_s} \tau)} \sum_{k=0}^{N-1} e^{-j2\pi \frac{\alpha}{f_s} (n-m) k} = \begin{cases} 0, & n \neq m \\ D_m e^{j(\psi+2\pi \frac{m}{f_s} \tau)}, & n = m \end{cases}$$  (7.4)

The timing phase (error), or envelope delay distortion, does not violate the orthogonality of the subcarriers and the effect of the timing offset is a phase rotation which linearly changes with subcarriers’ orders. On the other hand, envelope delay results in the same amount of rotation for all subcarriers.

It is straightforward to show that in a more general case, with pulse shaping filter with rolloff factor $\alpha$ and dispersive channel with impulse response $h(t)$, the detected data is attenuated and phase rotated such that $\tilde{D}_m = \gamma_m (\tau) D_m$, where:

$$\gamma_m (\tau) = \begin{cases} H \left( \frac{m}{NT} \right) e^{j2\pi \frac{\alpha}{f_s} \frac{m}{NT} \tau}, & 0 \leq \frac{m}{N} \leq \frac{1-\alpha}{2} \\ H \left( \frac{m}{NT} \right) e^{j2\pi \frac{\alpha}{f_s} \frac{m}{NT} \tau} + H \left( \frac{m-N}{NT} \right) e^{j2\pi \frac{\alpha}{f_s} \frac{m-N}{NT} \tau}, & \frac{1-\alpha}{2} \leq \frac{m}{N} \leq \frac{1+\alpha}{2} \\ H \left( \frac{m-N}{NT} \right) e^{j2\pi \frac{\alpha}{f_s} \frac{m-N}{NT} \tau}, & \frac{1+\alpha}{2} \leq \frac{m}{N} \leq 1 \end{cases}$$  (7.5)

where $H(m/NT)$ is the Fourier transform of $h(t)$ at frequency $m/NT$. The estimation of $\tau$ and $\varphi$ will be discussed later in this section.

We will now see that frequency offset amounts to inter-channel interference which is similar to inter-symbol interference of a single carrier signal due to a timing jitter. In a non-dispersive channel with rectangular pulse shaping, the interference caused by frequency offset could be too constraining. The sampled signal is:

$$y_k = \sum_{n=0}^{N-1} D_k e^{j2\pi \left( \frac{\alpha}{f_s} + \delta \right) k \tau} \bigg|_{\tau = \frac{\Delta f}{f_s}} = \sum_{n=0}^{N-1} D_k e^{j2\pi \left( \frac{\alpha}{f_s} \frac{m+n}{N} \right) k}$$  (7.6)

After DFT we have

$$\tilde{D} = D_m \left( e^{j2\pi \frac{\Delta f}{f_s}} - 1 \right) + \sum_{n=0}^{N-1} D_n \sum_{k=0}^{N-1} e^{j2\pi \frac{\alpha}{f_s} \left( n-m+\Delta f \right)} + N_m$$  (7.7)

where $\Delta f = n\delta/f_s$. So, due to the frequency offset we have attenuation of the desired signal (the first term of Equation (7.7)), and an interference between different symbols of several subcarriers (the second term of Equation (7.7)).

In order to avoid frequency offset, the window function

$$w_n = w(t)|_{t=nT}$$  (7.8)

must be such that zero crossings of its Fourier transform are at multiples of symbol frequency

$$W_m = W(\omega)|_{\omega=2\pi m f_s} = \delta_m$$  (7.9)
A generalized sinc function of the form
\[ \frac{\sin \omega n}{\omega n} \times g(n) \] (7.10)
for any differentiable function, \( g(t) \) satisfies the condition.

Another desired property of a sinc function is its low rate of change in the vicinity of in-frequency sampling points. A typical example is the raised cosine function in time:
\[
\begin{align*}
\frac{T}{\pi} \sin \frac{\beta}{2} \pi \frac{t}{T} & \times \frac{1}{1 - 4\beta^2 \frac{t^2}{T^2}} & 0 \leq |t| \leq \frac{1 - \beta}{2T} \\
0 & \text{elsewhere}
\end{align*}
\] (7.11)
where \( \beta \) is the rolloff factor for time domain pulse shaping. A higher rolloff factor requires a longer cyclic extension and a guard interval, which consumes higher bandwidth. The results for adjacent channel interference (ACI) caused by frequency offset in systems with different windowing functions are shown in Figure 7.1.

### 7.1.1 Robust frequency and timing synchronization for OFDM

#### 7.1.1.1 Symbol timing estimation algorithm

The symbol timing recovery relies on searching for a training symbol with two identical halves in the time. Consider the first training symbol where the first half is identical to the second half (in time order), except for a phase shift caused by the carrier frequency offset. If the conjugate of a sample from the first half is multiplied by the corresponding sample from the second half (\( T/2 \) seconds later), the effect of the channel should cancel, and the result will have a phase of approximately \( \phi = \pi T \Delta f \). At the start of the frame, the products of each of these pairs of samples will have approximately the same phase, so the magnitude of the sum will be a large value. Let us use \( L \) complex samples in one half of the first training symbol (excluding the cyclic prefix), and let the sum of the pairs of products be
\[
P(d) = \sum_{m=0}^{L-1} (r^*_d + m r_{d+m+L})
\] (7.12)
This can be implemented with the iterative formula

\[ P(d + 1) = P(d) + (r_{d+L}^* r_{d+2L}) - (r_d^* r_{d+L}) \]  \hspace{1cm} (7.13)

where \( d \) is a time index corresponding to the first sample in a window of \( 2L \) samples. This window slides along in time as the receiver searches for the first training symbol. The received energy for the second half symbol is defined by

\[ R(d) = \sum_{m=0}^{L-1} |r_{d+m+L}|^2 \]  \hspace{1cm} (7.14)

This can also be calculated iteratively. \( R(d) \) may be used as part of an automatic gain control (AGC) loop. A timing metric can be defined as

\[ M(d) = \frac{|P(d)|^2}{(R(d))^2} \]  \hspace{1cm} (7.15)

Equation (7.15) is shown in Figure 7.2 and Figure 7.3.

![Figure 7.2](image1.png)  
Figure 7.2 The timing metric for the AWGN channel (SNR = 10).

![Figure 7.3](image2.png)  
Figure 7.3 Expected value of timing metric with \( L = 512 \). Dashed lines indicate three standard deviations.
For the results in these figures, OFDM symbols are generated with 1000 frequencies, \(-500\) to 499, and slightly oversampled at a rate of 1024 samples for the useful part of each symbol. In an actual hardware implementation, the ratio of the sampling rate to the number of frequencies would be higher to ease filtering requirements. The guard interval is set to about 10% of the useful part, which is 102 samples.

### 7.1.1.2 Carrier frequency offset estimation algorithm

The main difference between the two halves of the first training symbol will be a phase difference of

\[ \phi = \pi \frac{T}{\Delta f} \]  

(7.16)

which can be estimated by

\[ \hat{\phi} = \text{angle}(P(d)) \]  

(7.17)

The second training symbol contains a PN sequence on the odd frequencies to measure these sub-channels, and another PN sequence on the even frequencies to help determine frequency offset. If |\hat{\phi}| can be guaranteed to be less than \(\pi\), then the frequency offset estimate is

\[ \Delta \hat{f} = \frac{\hat{\phi}}{\pi \frac{T}{\Delta f}} \]  

(7.18)

and the even PN frequencies on the second training symbol would not be needed. Otherwise, the actual frequency offset would be

\[ \frac{\phi}{\pi \frac{T}{\Delta f}} + \frac{2z}{T} \]  

(7.19)

where \(z\) is an integer. By partially correcting the frequency offset, adjacent carrier interference (ACI) can be avoided, and then the remaining offset of \(2z/T\) can be found. After the two training symbols are frequency corrected by \(\frac{\hat{\phi}}{\pi \frac{T}{\Delta f}}\) (by multiplying the samples by \(e^{-j2t\hat{\phi}/T}\)), let their FFTs be \(x_{1,k}\) and \(x_{2,k}\), and let the differentially-modulated PN sequence on the even frequencies of the second training symbol be \(v_k\). The PN sequence \(v_k\) will appear at the output except it will be shifted by \(2z\) positions because of the uncompensated frequency shift of \(2z/T\). Note that because there is a guard interval and there is still a frequency offset, even if there were no differential modulation between training symbols 1 and 2 (\(v_k = 1\)) there would still be a phase shift between \(x_{1,k}\) and \(x_{2,k}\) of \(2\pi(T + T_g)2z/T\). Since at this point the integer \(z\) is unknown, this additional phase shift is unknown. Since the phase shift is the same for each pair of frequencies, a metric similar to the previous one can be used.

Let \(X\) be the set of indices for the even frequency components, \(X = \{-W, -W + 2, \ldots, -4, -2, 2, 4, \ldots, W - 2, W\}\). The number of even positions shifted can be calculated by finding \(\hat{g}\) to maximize

\[ B(g) = \frac{\sum_{k \in X} x_{1,k} x_{1,k+2g} v_k^* x_{2,k+2g}}{2\left(\sum_{k \in X} |x_{2,k}|^2\right)^2} \]  

(7.20)

with integer \(g\) spanning the range of possible frequency offsets and \(W\) being the number of even frequencies with the PN sequence.

Then the frequency offset estimate would be:

\[ \Delta \hat{f} = \left[\hat{\phi}/(\pi \frac{T}{\Delta f})\right] + (2\hat{g}/T) \]  

(7.21)

### 7.1.1.3 Variance of carrier frequency offset estimator

From Equations (7.18) and (7.20) we have:

\[ \text{var}[\hat{\phi}/\pi] = \frac{1}{\pi^2 \cdot L \cdot \text{SNR}} \]  

(7.22)
One should keep in mind the Cramer–Rao bound is

\[
\text{var}[\hat{\phi}/\pi] \geq \frac{1}{\pi^2 \cdot L \cdot \text{SNR}} \tag{7.23}
\]

At the correct frequency offset, all the signal products

\[s_{1,k+g_{\text{correct}}}^* h \cdot v_{\text{correct}} s_{2,k+g_{\text{correct}}}^*\]

have the same phase and

\[
\mu_B = E[B(g_{\text{correct}})] = \frac{\sigma_s^4}{(\sigma_s^2 + \sigma_n^2)^2} \tag{7.24}
\]

\[
\text{var}[B(g_{\text{correct}})] = \frac{\sigma_s^4}{W} \frac{(2 + 2\mu_B)\sigma_s^2 \sigma_n^2 + (1 + 4\mu_B)\sigma_n^4}{\left(\sigma_s^2 + \sigma_n^2\right)^4}
\]

At an incorrect frequency offset the signal products no longer add phase, and \(B(g_{\text{incorrect}})\) has a chi-squared distribution with two degrees of freedom with

\[
E[B(g_{\text{correct}})] = \frac{1}{2W} \left(1 + \frac{\sigma_s^2}{\sigma_s^2 + \sigma_n^2}\right) < \frac{1}{W} \tag{7.25}
\]

\[
\text{var}[B(g_{\text{correct}})] = \frac{1}{4W^2} \left(1 + \frac{3\sigma_s^4 + 2\sigma_s^2\sigma_n^2}{\left(\sigma_s^2 + \sigma_n^2\right)^2}\right) < \frac{1}{W^2}
\]

### 7.2 FADING CHANNEL ESTIMATION FOR OFDM SYSTEMS

#### 7.2.1 Statistics of mobile radio channels

The channel impulse response will be represented as

\[h(t, \tau) = \sum_k \gamma_k(t) \delta(\tau - \tau_k) \tag{7.26}\]

Assume that \(\gamma_k(t)\) has the same normalized time correlation function \(r_t(\Delta t)\) for all \(k\), and power spectrum \(p_t(\Omega)\).

\[r_{y_k}(\Delta t) \overset{\Delta}{=} E[\gamma_k(t + \Delta t)\gamma_k^*(t)] = \sigma_k^2 r_t(\Delta t) \tag{7.27}\]

where \(\sigma_k^2\) is the average power of the \(k\)th path.

The frequency response of the time-varying radio channel at time \(t\) is

\[H(t, f) \overset{\Delta}{=} \int_{-\infty}^{\infty} h(t, \tau)e^{-j2\pi f \tau} d\tau = \sum_k \gamma_k(t)e^{-j2\pi f \tau_k} \tag{7.28}\]

The time–frequency correlation is defined as

\[r_H(\Delta t, \Delta f) \overset{\Delta}{=} E[H(t + \Delta t, f + \Delta f)H^*(t, f)] = \sum_k r_{y_k}(\Delta t)e^{-j2\pi \Delta f \tau_k} \Delta f \tau_k = r_t(\Delta t) \left(\sum_k \sigma_k^2 e^{-j2\pi \Delta f \tau_k}\right) = \sigma_H^2 r_t(\Delta t) r_f(\Delta f) \tag{7.29}\]
where $\sigma_H^2 = \sum_k \sigma_k^2$ is the total average power of the channel impulse response. The frequency correlation is defined as

$$r_f(\Delta f) = \sum_k \frac{\sigma_k^2}{\sigma_H^2} e^{-j2\pi \Delta f \tau_k}$$  \hspace{1cm} (7.30)$$

where $r_f(0) = r_f(0) = 1$. Without loss of generality, we also assume that $\sigma_H^2 = 1$, so that it can be omitted. For block length $T_f$ and tone spacing (subchannel spacing) $\Delta f$, the correlation function for different blocks and tones is

$$r_H[n, k] = r_f(k)$$  \hspace{1cm} (7.31)$$

From Jakes’s model of the channel we have:

$$r_f[n] = J_0(n\omega_d)$$  \hspace{1cm} (7.32)$$

As an example, for carrier frequency $f_c = 2$ GHz, $f_d = 184$ Hz when the user is moving at 60 mph.

7.2.2 Diversity receiver

In the case of space (antenna) diversity, the signal from the $m$th antenna at the $k$th tone and the $n$th block can be represented as

$$x_m[n, k] = H_m[n, k] a[n, k] + \omega_m[n, k]$$  \hspace{1cm} (7.33)$$

where $\omega_m[n, k]$ is additive Gaussian noise from the $m$th antenna at the $k$th tone and the $n$th block, with zero mean and variance $\rho$. Let us assume that $\omega_m[n, k]$ is independent for different $ns$, $ks$, or $ms$. $H_m[n, k]$, the frequency response at the $k$th tone and the $n$th block corresponding to the $m$th antenna, is assumed independent for different $ms$, but with the same statistics. $a[n, k]$ is the signal modulating the $k$th tone during the $n$th block and is assumed to have unit variance and be independent for different $ks$ and $ns$.

With knowledge of the channel parameters, $a[n, k]$ can be estimated as $y[n, k]$ by an MMSE combiner

$$y[n, k] = \frac{\sum_{m=1}^p H_m^*[n, k] x_m[n, k]}{\sum_{m=1}^p |H_m[n, k]|^2}$$  \hspace{1cm} (7.34)$$

The transceiver (transmitter/receiver) block diagram is given in Figure 7.4.

7.2.3 MMSE channel estimation

If the reference $a[n, k]$ is ideal (pilot symbols), a temporal estimation of $H[n, k]$ is

$$\tilde{H}[n, k] = x[n, k] a^*[n, k] \equiv H[n, k] + \omega[n, k] a^*[n, k]$$  \hspace{1cm} (7.35)$$
where \(^*\) denotes the complex conjugate. \(\hat{H} [n, k]s\) for different \(n_s\) and \(k_s\) are correlated; therefore, an MMSE channel estimator can be constructed as follows:

\[
\hat{H} [n, k] = \sum_l \sum_{m=\infty}^0 c[m, l, k] \tilde{H} [n-m, k-l]
\]  

(7.36)

where \(c[m, l, k]s\) are selected to minimize

\[
\text{MSE}(\{c[m, l, k]\}) = E|\hat{H}[n, k] - H[n, k]|^2
\]  

(7.37)

If \(K\) is the number of tones in each OFDM block, then we will be using the following notation

\[
c[m, k] \triangleq \begin{pmatrix} c[m, k - 1, k] \\ \vdots \\ c[m, 0, k] \\ \vdots \\ c[m, -K + k, k] \end{pmatrix} \quad \quad \text{and} \quad \quad \text{M}_l(\omega) = \sum_{n=0}^{\infty} \gamma[n] e^{-j\omega n}
\]  

(7.38)

Starting from the fact that the projection of the estimation error on \(H[n, k]\) is zero, \(E((\hat{H}[n, k] - H[n, k])\hat{H}^*[n-m, k-l]) = 0\) (for orthogonality principles see Chapter 5), the estimation coefficients are given by [1]:

\[
C(\omega) = U^H \Phi(\omega) U
\]  

(7.39)

where \(\Phi(\omega)\) is a diagonal matrix with the \(l\)th element

\[
\Phi_l(\omega) = 1 - \frac{1}{M_l(-\omega)\gamma_l[0]}
\]  

(7.40)

and \(M_l(\omega)\) is a stable one-sided FT

\[
M_l(\omega) = \sum_{n=0}^{\infty} \gamma[n] e^{-j\omega n}
\]  

(7.41)

\[
M_l(\omega)M_l(-\omega) = \frac{d_l}{\rho} p_l(\omega) + 1
\]  

(7.42)
FADING CHANNEL ESTIMATION FOR OFDM SYSTEMS

\[ H[n,1] \]

\[ H[n,2] \]

\[ H[n,K] \]

\[ F_1(w) \]

\[ F_K(w) \]

\[ U_H \]

\[ U \]

\[ \hat{n}[n,1] \]

\[ \hat{n}[n,2] \]

\[ \hat{n}[n,K] \]

Figure 7.5 Channel estimator for OFDM systems.

The DC component \( \gamma_{l}[0] \) in \( M_{l}(\omega) \) can be found by

\[
\gamma_{l}[0] = \exp \left\{ \frac{1}{2\pi} \int_{-\pi}^{\pi} \ln \left[ \frac{d_i}{\rho} p_i(\omega) + 1 \right] d\omega \right\} \quad \text{(7.43)}
\]

The \( d_i \) and \( u_i \) are the corresponding eigenvalues and eigenvectors of the frequency domain correlation matrix \( R_f \),

\[
R_f = \bar{U}^H \bar{D} \bar{U} \quad \text{or} \quad \bar{U} R_f \bar{U}^H = D \quad \text{(7.45)}
\]

and \( D = \text{diag}[d_1, \ldots, d_K] \) and \( \sum_k d_k = K \). The processing is illustrated in Figure 7.5.

The unitary linear inverse transform \( U^H \) and transform \( U \) in the figure perform the eigendecomposition of the frequency domain correlation. The estimator turns off the zero or small \( d_i \) to reduce the estimation noise. For those large \( d_i \), linear filters are used to take advantage of the time domain correlation.

One can show [1] that for Jakes’s model:

\[
\text{MMSE}_J(\omega_d) = \frac{\rho}{K} \sum_{l=1}^{K} \left( 1 - \left( \frac{\alpha_l}{\pi} \right)^{-(\omega_d)^2} \exp \left\{ -\frac{(\omega_d)^2 b(\alpha_l)}{\pi} \right\} \right) \quad \text{(7.46)}
\]

\[
\alpha_l = \frac{2d_i}{\omega_d \rho}
\]

\[
b(\alpha_l) = \begin{cases} 
\frac{\pi}{2} \alpha_l\sqrt{1-\alpha_l^2} \ln \frac{1+\sqrt{1-\alpha_l^2}}{\alpha_l}, & \text{if } \alpha_l < 1 \\
\frac{\pi}{2} \alpha_l\sqrt{\alpha_l^2-1} \left( \frac{\pi}{2} - \arcsin \frac{1}{\alpha_l} \right), & \text{if } \alpha_l \geq 1
\end{cases}
\]

### 7.2.4 FIR channel estimator

For a reader less familiar with eigenvalue decomposition, discussed in Chapter 5, we provide a simplified interpretation of the processing by using an approximation. Instead of proper eigenvectors
we use DFT matrix $W$ defined as

$$W \triangleq \frac{1}{\sqrt{K}} \begin{pmatrix}
1 & 1 & \cdots & 1 \\
1 & e^{j2\pi/K} & \cdots & e^{j2\pi(K-1)/K} \\
\vdots & \vdots & \ddots & \vdots \\
1 & e^{j2\pi(K-1)/K} & \cdots & e^{j2\pi(K-1)(K-1)/K}
\end{pmatrix}$$

(7.47)

and with notation

$$\bar{R}_l = 
\begin{pmatrix}
\bar{r}_l[0] & \bar{r}_l[1] & \cdots & \bar{r}_l[L-1] \\
\bar{r}_l[-1] & \bar{r}_l[0] & \cdots & \bar{r}_l[L-2] \\
\vdots & \vdots & \ddots & \vdots \\
\bar{r}_l[-L+1] & \bar{r}_l[-L+2] & \cdots & \bar{r}_l[0]
\end{pmatrix}$$

(7.48)

$$\bar{r}_l = (\bar{r}_l[0], \bar{r}_l[1], \cdots, \bar{r}_l[L-1])^T$$

$$\bar{r}_l[n] = \frac{\sin(n\omega_d)}{n\omega_d}$$

we have, for the coefficient matrix of the designed FIR channel estimator,

$$\bar{C}(\omega) = W^H \bar{\Phi}(\omega) W$$

(7.49)

with

$$\bar{\Phi}(\omega) = \text{diag}\{c(\omega), \ldots, c(\omega), 0, \ldots 0\}$$

$k_0$ elements

If the maximum delay spread is $t_d$, then for all $l \leq K_0$ ($K_0 = [Kt_d/T_s]$), $d_l \approx 0$, where $T_s$ is the symbol interval. In Equation (7.49), $c(\omega)$ is the FT of $c_n$ given by

$$\begin{pmatrix} c_0, c_1, \ldots, c_{L-1} \end{pmatrix}^T = \left( \begin{pmatrix} \bar{R}_l + \frac{K_0 \rho}{K} I \end{pmatrix} \right)^{-1} \bar{r}_l$$

and $L$ is the length of the FIR estimator. The estimation error

$$\text{MSE} = \frac{K_0 \rho}{K}$$

(7.50)

For the robust FIR channel estimator, the $U$ in Figure 7.5 is the DFT matrix $W$ and the $\Phi_k(\omega)s$ for $k = 1, \ldots, K$ are $c(\omega)$.

### 7.2.5 System performance

The set of assumptions used to generate the performance curves is the same as in [1]: a two-way Rayleigh fading with delay from 0 to 40 $\mu$s and Doppler frequency from 10 to 200 Hz; the channels corresponding to different receivers have the same statistics; two antennas are used for receiver diversity. For the OFDM signal, it is assumed that the entire channel bandwidth, 800 kHz, is divided into 128 subchannels. The four subchannels on each end are used as guard tones and the rest (120 tones) are used to transmit data. To make the tones orthogonal to each other, the symbol duration is 160 $\mu$s. An additional 40 $\mu$s guard interval is used to provide protection from ISI, the length $T_i = 200$ ms and a subchannel symbol rate $r_b = 5$ kBs.

To compare the performance of the OFDM system with and without the channel estimation, PSK modulation with coherent demodulation and differential PSK (DPSK) modulation with differential demodulation are used, respectively. The (40, 20) RS code, with each code symbol consisting of three quadrature PSK/differential quadrature PSK (QPSK/DQPSK) symbols grouped in frequency, is used in the system. Each OFDM block forms an RS code word. The RS decoder erases ten symbols, based on signal strength, and corrects five additional random errors. Hence, the simulated system can transmit data at 1.2 Mb/s before decoding or 600 kb/s after decoding, over an 800 kHz channel.
7.2.6 Reference generation

Four different ways to generate the reference signal are used:

1. Undecoded/decoded dual mode reference: If the RS decoder can successfully correct all errors in an OFDM block, the reference for the block can be generated by the decoded data; hence \( \tilde{a}[n, k] = \hat{a}[n, k] \). Otherwise, \( \tilde{a}[n, k] = \hat{a}[n, k] \).

2. Undecoded reference: \( \tilde{a}[n, k] = \hat{a}[n, k] \), no matter whether the RS decoder can successfully correct all errors in a block or not.

3. Decoded/CMA dual mode reference: The constant modulus algorithm (CMA) is used to generate a reference for the OFDM channel estimator. If the RS decoder can successfully correct all errors in a block, the reference for the block can be generated from the decoded data; hence \( \tilde{a}[n, k] = a[n, k] \). Otherwise, the reference can use the projection of \( y[n, k] \) on the unit circle, i.e. \( \tilde{a}[n, k] = y[n, k]/\mod(y[n, k]) \).

4. Error removal reference: If the RS decoder can successfully correct all errors in a block, the reference for the block can be generated by the decoded data. Otherwise, the \( \tilde{H}[n-1, k] \)’s are used instead of the \( \tilde{H}[n, k] \)’s for \( k = 1, \ldots, K \) respectively.

The results are shown in Figure 7.6.

In general a careful study of the results based on the previous discussion suggests the following conclusions:

- If a channel estimator is designed to match the channel with 40 Hz maximum Doppler frequency and 20 \( \mu \)s maximum delay spread, then for all channels with \( f_d \leq 40 \) Hz and \( t_d \leq 20 \) \( \mu \)s, the system performance is not worse than the channel with \( f_d = 40 \) Hz and \( t_d = 20 \) \( \mu \)s. For channels with \( f_d > 40 \) Hz or \( t_d > 20 \) \( \mu \)s, such as \( f_d = 80 \) Hz and \( t_d = 20 \) \( \mu \)s or \( f_d = 40 \) Hz and \( t_d = 40 \) \( \mu \)s, the system performance degrades dramatically.

- If the estimator is designed to match a Doppler frequency or delay spread larger than the actual ones, the system performance degrades only slightly compared with estimation that exactly matches the channel Doppler frequency and delay spread.

More details on channel estimation can be found in [2–11].

7.3 64 DAPSK AND 64 QAM MODULATED OFDM SIGNALS

In this section we discuss in more detail specific, high constellation, modulation schemes for OFDM signals. The transmission system and two options for the signal constellation are shown in Figures 7.7 and 7.8.

In this section we consider an OFDM system with \( N = 1024 \) subcarriers and parameters the same as in [12]. Each OFDM symbol transfers 6144 bits in total. The convolutional code has the memory length \( m = 6 \). The bit interleaver is a block interleaver with 83 rows and 74 columns, which means that only two bits of an OFDM symbol are not involved in the interleaving process. The performance of convolutional codes with code rates 1/2, 2/3 and 3/4 has been analyzed.

A (204, 188) RS code with \( p = 8 \) b/symbol has been chosen for the DTVB application. The objective of the concatenated code is to fulfil the requirement of a residual bit error rate (BER) of \( 10^{-11} \) at the output of the RS decoder. For this, the BER at the output of the Viterbi decoder is required to be lower than \( 2 \times 10^{-4} \) [13]. The above parameters are used in the system for digital terrestrial video broadcasting [14–17]. Such a system can transmit 34 Mbits/s over an 8 MHz radio channel.

The phase modulation is independent of the amplitude and identical to the well known 16 DPSK. The input bits \( b_{i,0}, b_{i,1}, b_{i,2}, b_{i,3} \) are used for this differential phase modulation in the 64 DAPSK scheme. The amplitude states \( |x_i| \) are chosen from the constellation diagram depending on the previous
amplitude state $|\tilde{x}_i|$ and the two information bits $b_{i,5}$ and $b_{i,4}$ according to Table 7.1. This means, e.g., for a subcarrier $i$ if the amplitude in the previous OFDM symbol was $|\tilde{x}_i|$ and the input information bits $b_{i,5}$ and $b_{i,4}$ were both zero, then the amplitude in the current OFDM symbol would be $|x_i| = |\tilde{x}_i|$ again. Table 7.1 shows the amplitude state $|x_i|$ for the subcarrier in the current OFDM symbol.

In each OFDM receiver after block synchronization, analog-to-digital (A/D) conversion, and removing of the guard interval, a fast Fourier transform (FFT) will produce the complex output states $y_i$ for each subcarrier $i$. If the coherent 64 QAM is used, after the FFT a channel equalization must be performed.

This means that the channel transfer factor $\alpha_i$ is assumed to be known exactly for each subcarrier $i$. With this information, the transmitted state $x_i$ of each subcarrier in the 64 QAM constellation diagram
Figure 7.7 Transmission system.

Figure 7.8 Constellation diagrams of 64 DAPSK and 64 QAM.
Table 7.1 Differential amplitude modulation for 64 DAPSK. Choice of the current amplitude state $|x_i|$ depending on the previous state $|\bar{x}_i|$ and the amplitude bits $b_{i,4}b_{i,5}$

| $|\bar{x}_i|$ | 00 | 01 | 11 | 10 |
|-------------|----|----|----|----|
| 1           | 1  | $a$ | $a^2$ | $a^3$ |
| $a$         | $a$ | $a^2$ | $a^3$ | 1   |
| $a^2$       | $a^2$ | $a^3$ | 1   | $a$ |
| $a^3$       | $a^3$ | 1   | $a$ | $a^2$ |

Table 7.2 Evaluation of the amplitude information bits $b_{i,5}$ and $b_{i,4}$ in the 64 DAPSK demodulation

| $|r_i| = |y_i/\bar{y}_i|$ |
|---------------------|
| $a^{-3}$ | $a^{-2}$ | $a^{-1}$ | 1 | $a^1$ | $a^2$ | $a^3$ |
| 01 | 11 | 10 | 00 | 01 | 11 | 10 |
| $a^{-2.5}$ | $a^{-1.5}$ | $a^{-0.5}$ | $a^{-0.5}$ | $a^{1.5}$ | $a^{-2.5}$ |
| $a^{-2.5}$ | $a^{-1.5}$ | $a^{-0.5}$ | $a^{0.5}$ | $a^{1.5}$ | $a^{2.5}$ | $a^{3.5}$ |

is evaluated by simple quotient

$$x_i \approx y_i/\alpha_i$$

in order to get the coded bit sequence for hard decision decoding or the metric increments for soft decision decoding.

For the non-coherent 64 DAPSK demodulation, first the quotient

$$r_i = y_i/\bar{y}_i \approx x_i/\bar{x}_i$$

of the currently received state $y_i$ and the preceding state $\bar{y}_i$ of the same subcarrier $i$ in the receiver (FFT output) are calculated. The resulting complex quotient $r_i$ is evaluated in order to get the phase and amplitude bits $b_{i,0}, b_{i,1}, \ldots, b_{i,5}$. This quotient $r_i$ is nearly independent of channel transfer factor $\alpha_i$ if the radio channel does not change the transmission behavior too quickly. Therefore, pilot symbols, channel estimation and equalization are not needed in the 64 DAPSK receiver, which reduces the computation complexity.

The four bits $b_{i,0}, b_{i,1}, b_{i,2}, b_{i,3}$ are determined depending on the phase difference between $y_i$ and $\bar{y}_i$ only. This phase demodulation process is the same as the demodulation of 16 DPSK.

For the amplitude demodulation, Table 7.2 shows how the amplitude bits $b_{i,4}$ and $b_{i,5}$ are obtained. For the evaluation of $|r_i|$, simple amplitude thresholds are used.

Performance results are given in Figure 7.9 and Tables 7.3–7.5. On average, 4 dB SNR degradation must be accepted in order to have much simpler-to-implement DAPSK. More details on coherent APSK and DAPSK can be found in [12–22].
Figure 7.9 (a) Performance of ideal 64 QAM and 64 DAPSK (ring ratio $a = 1.4$) with a convolutional code $m = 6, R = 3/4$ for hard and soft decision decoding in an AWGN channel; (b) performance of ideal 64 QAM (convolutional code $R = 3/4$) and 64 DAPSK ($R = 2/3$) over the Rayleigh and Rice-fading channel (fixed user data rate).

Table 7.3  SNR [dB] required for BER = $2 \times 10^{-4}$

<table>
<thead>
<tr>
<th>Code rate $R$</th>
<th>Metric</th>
<th>Ideal 64 QAM</th>
<th>64 DAPSK</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>AWGN channel</td>
<td>Rice channel</td>
</tr>
<tr>
<td>1/2</td>
<td>hard decision</td>
<td>17.3</td>
<td>19.6</td>
</tr>
<tr>
<td>2/3</td>
<td>hard decision</td>
<td>19.5</td>
<td>22.4</td>
</tr>
<tr>
<td>3/4</td>
<td>hard decision</td>
<td>20.9</td>
<td>23.9</td>
</tr>
<tr>
<td>1/2</td>
<td>soft decision</td>
<td>13.6</td>
<td>15.6</td>
</tr>
<tr>
<td>2/3</td>
<td>soft decision</td>
<td>16.8</td>
<td>19.1</td>
</tr>
<tr>
<td>3/4</td>
<td>soft decision</td>
<td>18.5</td>
<td>20.8</td>
</tr>
</tbody>
</table>
Table 7.4  Difference of required SNR [dB] for 64 QAM (ideal channel estimation) and 64 DAPSK modulation at BER $= 2 \times 10^{-4}$, fixed code rates

<table>
<thead>
<tr>
<th>Code rate $R$</th>
<th>Hard decision</th>
<th>Soft decision</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>AWGN channel</td>
<td>Rice channel</td>
</tr>
<tr>
<td>1/2</td>
<td>4.1</td>
<td>4.1</td>
</tr>
<tr>
<td>2/3</td>
<td>4.5</td>
<td>4.1</td>
</tr>
<tr>
<td>3/4</td>
<td>4.8</td>
<td>4.3</td>
</tr>
</tbody>
</table>

Table 7.5  Difference of required SNR [dB] for 64 QAM (ideal channel estimation) with $R = 3/4$ and 64 DAPSK modulation with $R = 2/3$ at BER $= 2 \times 10^{-4}$, fixed user data rate

<table>
<thead>
<tr>
<th>Code rate $R$</th>
<th>Hard decision</th>
<th>Soft decision</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>AWGN channel</td>
<td>Rice channel</td>
</tr>
<tr>
<td>QAM: 3/4</td>
<td>3.1</td>
<td>2.6</td>
</tr>
<tr>
<td>DAPSK: 2/3</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 7.10  Turbo convolutional (TC) or Reed–Solomon (RS) code used with space–time trellis (STT) or space–time block (STB) encoder.

7.4  SPACE–TIME CODING WITH OFDM SIGNALS

In this section we discuss space–time coding with an OFDM signal. The system block diagram is shown in Figure 7.10.

In the system, Alamouti’s $G_2$ space–time block code is used

$$G_2 = \begin{pmatrix} x_1 & x_2 \\ \bar{x}_2 & \bar{x}_1 \end{pmatrix}$$

For comparison the system parameters used in this section are the same as in [23] and are specified in Tables 7.6–7.8.
Table 7.6 Modulation parameters

<table>
<thead>
<tr>
<th>Modulation scheme</th>
<th>Bits per symbol BPS</th>
<th>Decoding algorithm</th>
<th>No. of states</th>
<th>No. of transmitters</th>
<th>No. of termination symbols</th>
</tr>
</thead>
<tbody>
<tr>
<td>4PSK</td>
<td>2</td>
<td>VA</td>
<td>4</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>8</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>16</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>32</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>8PSK</td>
<td>3</td>
<td>VA</td>
<td>16</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>32</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>

Table 7.7 The parameters associated with the turbo convolutional \((n, k, K)\) TC(2, 1, 3) code

<table>
<thead>
<tr>
<th>Code</th>
<th>Code rate (R)</th>
<th>Modulation mode</th>
<th>BPS</th>
<th>Random turbo interleaver depth</th>
<th>Random separation interleaver depth</th>
</tr>
</thead>
<tbody>
<tr>
<td>TC(2,1, 3)</td>
<td>0.50</td>
<td>QPSK</td>
<td>1</td>
<td>512 carriers</td>
<td>1024</td>
</tr>
<tr>
<td></td>
<td></td>
<td>16 QAM</td>
<td>2</td>
<td>256</td>
<td>768</td>
</tr>
<tr>
<td></td>
<td></td>
<td>64 QAM</td>
<td>3</td>
<td>384</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>16 QAM</td>
<td>2</td>
<td>1024</td>
<td>2048</td>
</tr>
</tbody>
</table>

Table 7.8 The coding parameters of the Reed–Solomon codes

<table>
<thead>
<tr>
<th>Code</th>
<th>Galois field</th>
<th>Rate</th>
<th>Correctable symbol errors</th>
</tr>
</thead>
<tbody>
<tr>
<td>RS(105, 51)</td>
<td>(2^{10})</td>
<td>0.49</td>
<td>27</td>
</tr>
<tr>
<td>RS(153, 102)</td>
<td>(2^{10})</td>
<td>0.67</td>
<td>25</td>
</tr>
</tbody>
</table>

7.4.1 Signal and channel parameters

A two-ray channel impulse response having equal amplitudes and differential delay of 5 \(\mu s\) is used. The average signal power received from each transmitter antenna is the same. All multipath components undergo independent Rayleigh fading; Jakes’s model. The receiver has a perfect knowledge of the CIR. A 128 subcarrier (160 \(\mu s\)) OFDM signal with a cyclic extension of 32 samples (40 \(\mu s\)) is used. The results are presented in Figure 7.11.

For increased delay spread and Doppler, the variation of the frequency domain fading envelope will eventually destroy the orthogonality of the space–time block code \(G_2\) (see Figure 7.13). For this reason the two transmission instants of the space–time block code \(G_2\) will have to be allocated to the same OFDM symbol. In the previous example they were allocated to the adjacent subcarriers. The transmission system for the time-varying channel is now modeled in Figure 7.12. The
Figure 7.11 FER performance comparison between various (a) 4PSK and (b, c) 8PSK space–time trellis codes and the space–time block code $G_2$ concatenated with the TC(2, 1, 3) code using (a, b) one or (c) two receivers and the 128 subcarrier OFDM modem over a channel having a CIR characterized by two equal-power rays separated by a delay spread of $5 \mu s$. The maximum Doppler frequency was 200 Hz. The effective throughput was (a) 2 BPS or (b, c) 3 BPS and the coding parameters are shown in Tables 7.6–7.7.
received signals are given by:

\[ \tilde{x}_1 = \bar{h}_{1,1}y_1 + h_{2,2}\bar{y}_2 \]
\[ = \bar{h}_{1,1}h_{1,1}x_1 + \bar{h}_{1,1}h_{2,1}x_2 + \bar{h}_{1,1}n_1 - h_{2,2}\bar{h}_{1,2}x_2 + h_{2,2}\bar{h}_{2,2}x_1 + h_{2,2}\bar{n}_2 \]
\[ = (|h_{1,1}|^2 + |h_{2,2}|^2)x_1 + (\bar{h}_{1,1}h_{2,1} - h_{2,2}\bar{h}_{1,2})x_2 + \bar{h}_{1,1}n_1 + h_{2,2}\bar{n}_2 \]
\[ \tilde{x}_2 = \bar{h}_{2,1}y_1 + h_{1,2}\bar{y}_2 \]
\[ = \bar{h}_{2,1}h_{1,1}x_1 + \bar{h}_{2,1}h_{2,1}x_2 + \bar{h}_{2,1}n_1 - h_{1,2}\bar{h}_{1,2}x_2 + h_{1,2}\bar{h}_{2,2}x_1 - h_{1,2}\bar{n}_2 \]
\[ = (|h_{2,1}|^2 + |h_{1,2}|^2)x_1 + (\bar{h}_{2,1}h_{1,1} - h_{1,2}\bar{h}_{1,2})x_1 + \bar{h}_{2,1}n_1 - h_{1,2}\bar{n}_2 \] (7.51)

The signal to interference ratio (SIR) for signal \( x_1 \) is:

\[ \text{SIR} = \frac{|h_{1,1}|^2 + |h_{2,2}|^2}{\bar{h}_{1,1}h_{2,1} - h_{2,2}\bar{h}_{1,2}} \] (7.52)

and for signal \( x_2 \) is:

\[ \text{SIR} = \frac{|h_{2,1}|^2 + |h_{1,2}|^2}{\bar{h}_{2,1}h_{1,1} - h_{1,2}\bar{h}_{2,2}} \] (7.53)

The two transmission instants are no longer assumed to be associated with the same complex transfer function values. Performance curves are given in Figures 7.13 and 7.14.

The fading amplitude variation versus time is slower than that versus the subcarrier index within the OFDM symbols. This implies that the SIR attained would be higher, if we were to allocate the two transmission instants of the space–time block code \( \mathbf{G}_2 \) to the same subcarrier of consecutive OFDM symbols. This increase in SIR is achieved by doubling the delay of the system, since in this scenario two consecutive OFDM symbols have to be decoded.
Figure 7.13 FER performance of the space–time block code $G_2$ concatenated with the TC(2, 1, 3) code using one receiver, the 128 subcarrier OFDM modem and 16 QAM. The CIR exhibits two equal-power rays separated by various delay spreads and a maximum Doppler frequency of 200 Hz. The coding parameters are shown in Tables 7.6–7.9.

Figure 7.14 FER performance comparison between adjacent subcarriers and adjacent OFDM symbols allocation for the space–time block code $G_2$ concatenated with the TC(2, 1, 3) code using one receiver, the 128 subcarrier OFDM modem and 16 QAM over a channel having a CIR characterized by two equal-power rays separated by a delay spread of 40 $\mu$s. The maximum Doppler frequency was 100 Hz. The coding parameters are shown in Tables 7.6–7.9.
7.4.2 The wireless asynchronous transfer mode system

The CIR used in this case has three paths and is referred to as the shortened WATM CIR, as shown in Figure 7.15. A 512 subcarrier (2.2756 µs) OFDM time domain signal with a cyclic extension of 64 samples (0.2844 µs) is used. The sampling rate is 225 Msamples/s and the carrier frequency is 60 GHz. The results are shown in Figure 7.16.

7.4.3 Space–time coded adaptive modulation for OFDM

All subcarriers in an adaptive OFDM (AOFDM) symbol are split into blocks of adjacent subcarriers, referred to as subbands. The same modulation scheme is employed for all subcarriers of the same subband. This substantially simplifies the task of signaling the modulation modes, since there are typically four modes and, for example, 32 subbands, requiring a total of 64 AOFDM mode signaling bits. The system is referred to as subband adaptive OFDM.

The system is presented in Figure 7.17. The choice of the modulation scheme to be used by the transmitter for its next OFDM symbol is determined by the channel quality estimate of the receiver, based on the current OFDM symbol. The following assumptions are used: the average signal power received from each transmitter antenna is the same; all multipath components undergo independent Rayleigh fading; the receiver has a perfect knowledge of the CIR and perfect signaling of the AOFDM modulation modes is available.

Optimized switching levels for adaptive modulation over a Rayleigh fading channel, shown in the instantaneous channel SNR (dB) are given in Table 7.9. The performance results are given in Figure 7.18.

7.4.4 Turbo and space–time coded adaptive OFDM

In this case the system parameters are as defined in Table 7.10. Performance results are shown in Figure 7.19.

For details on space–time coding with OFDMA signals and same set of parameters see [23–31].
Figure 7.16 FER performance comparison between the TC(2, 1, 3) coded space–time block code $G_2$ and the RS(102, 51) GF($2^{10}$) coded 16-state 4PSK space–time trellis code using one 512 subcarrier OFDM receiver over the shortened WATM channel at an effective throughput of (a) 1 BPS and (b) 2 BPS. The coding parameters are shown in Tables 7.6–7.9.
Table 7.9 Switching SNR

<table>
<thead>
<tr>
<th>System</th>
<th>NoTx</th>
<th>BPSK</th>
<th>QPSK</th>
<th>16 QAM</th>
<th>64 QAM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Speech</td>
<td>$-\infty$</td>
<td>3.31</td>
<td>6.48</td>
<td>11.61</td>
<td>17.64</td>
</tr>
<tr>
<td>Data</td>
<td>$-\infty$</td>
<td>7.98</td>
<td>10.42</td>
<td>16.76</td>
<td>26.33</td>
</tr>
</tbody>
</table>

Figure 7.17 System overview of the turbo coded and space–time coded adaptive OFDM.

Figure 7.18 BER and BPS performance of 16 subband AOFDM employing the space–time block code $G_2$ using multiple receivers for a target BER of $10^{-4}$ over the shortened WATM channel shown in Figure 7.15.

7.5 LAYERED SPACE–TIME CODING FOR MIMO OFDM

In this section we extend the discussion on space–time coded OFDM to the case with $n_t = n_r = 4$ assuming the Jakes fading model and a layered architecture. The channel modeling is discussed in Chapter 14, but in this section we assume the channel estimation procedures as in [29] and [32] and the TU channel model considered in [32]. The OFDM signals assume a channel bandwidth of 1.25 MHz, which is divided into 256 subchannels. Two subchannels at each end of the band are used
Table 7.10  Coding rates and switching levels (dB) for TC(2, 1, 3) and space–
time coded adaptive OFDM over the shortened W ATM channel of
Figure 7.15 for a target BER of $10^{-4}$

<table>
<thead>
<tr>
<th></th>
<th>NoTx</th>
<th>BPSK</th>
<th>QPSK</th>
<th>16 QAM</th>
<th>64 QAM</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Half rate TC(2, 1, 3)</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rate</td>
<td>–</td>
<td>0.50</td>
<td>0.50</td>
<td>0.50</td>
<td>0.50</td>
</tr>
<tr>
<td>Thresholds (dB)</td>
<td>$-\infty$</td>
<td>-4.0</td>
<td>-1.3</td>
<td>5.4</td>
<td>9.8</td>
</tr>
<tr>
<td><strong>Variable rate TC(2, 1, 3)</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rate</td>
<td>–</td>
<td>0.50</td>
<td>0.67</td>
<td>0.75</td>
<td>0.90</td>
</tr>
<tr>
<td>Thresholds (dB)</td>
<td>$-\infty$</td>
<td>-4.0</td>
<td>2.0</td>
<td>9.70</td>
<td>21.50</td>
</tr>
</tbody>
</table>

Figure 7.19  BER and BPS performance of 16 subband AOFDM employing the space–time
block code $G_2$ concatenated with both half rate and variable rate TC(2, 1, 3)
at a target BER of $10^{-4}$ over the shortened WATM channel.

as guard tones, with the other 252 tones used to transmit data. The symbol duration is taken to be
204.8 $\mu$s so that the tones are orthogonal. A 20.2 $\mu$s guard interval is used to provide protection from
inter-symbol interference, making the block duration $T_f = 225$ $\mu$s. The subchannel symbol rate is
$r_b = 4.44$ kbaud. Systems with the same signal parameters are discussed, for example, in [32, 33].

7.5.1  System model (two times two transmit antennas)

First we consider the $n_g = 2$ (groups of antennas) MIMO-OFDM implementation illustrated in
Figure 7.20. In this case, two antenna space–time codes are employed that use 16 states and QPSK
modulation. Data is grouped into blocks of 500 information bits, called words. Each word is coded
into 252 symbols to form an OFDM block. Since this system uses $n_g = 2$, it can transmit two of
these data blocks (1000 bits total) in parallel. Each time slot consists of ten OFDM blocks with the
first block used for training and the following nine blocks used for data transmission. This leads to a
system capable of transmitting 4 Mbit/s using 1.25 MHz of bandwidth, so the transmission efficiency
is 3.2 bit/s/Hz.
7.5.2 Interference cancellation

In general, all the interference cancellation schemes discussed in Chapter 5, modified for OFDM signals, can be used. One option has already been discussed in Chapter 4. An initial study of the system outlined in this section was provided in [34]. In [34] several interference cancellation approaches were described and performance was evaluated. Here, we focus on the successive interference cancellation approach based on signal quality (see Chapter 4). Basically, the strongest signal is detected first, subtracted from the input signal and then the procedure is repeated. We also assume that interleaving is employed. For the two-antenna, 16-state code given in Chapter 4 [3 (Figure 5)], the word error rate (WER) is given in Figure 7.21 for the case where the channel has a TU delay profile and for Doppler frequencies of 5, 40, 100, and 200 Hz. The other two curves in Figure 7.21 illustrate the performance improvement that can be obtained using the improved convolutional space–time codes given in Section 4.6. One of these codes was designed to be optimal for the quasi-static fading model from Section 4.2 (CC1). The other code was designed to be optimal for the rapid fading model in Section 4.2 (CC2). The improved codes from Section 4.6 are optimal codes based on the criterion given in Section 4.2. All these details are available in Chapter 4. In summary:

\[
\begin{bmatrix}
D + 2D^2 & 1 + 2D^2 \\
2D & 2
\end{bmatrix}
\]  
Figure 4.7 (TC)

\[
\begin{bmatrix}
D + D^2 & 2 + D \\
2 + D & 2 + 2D + 2D^2
\end{bmatrix}
\]  
quasi-static code Section 4.6 (CC1)

\[
\begin{bmatrix}
2D^2 & 2 + D + 2D^2 \\
2 + D & 2D + 2D^2
\end{bmatrix}
\]  
rapid fading code Section 4.6 (CC2)

7.5.3 Four transmit antennas

Next we investigate the approach that uses four-antenna space–time codes. We consider 16-state and 256-state codes, designed using an ad hoc approach. The codes are presented in Table 7.11 and the performance results are given in Figure 7.22.
Figure 7.21 Performance curves. (a) 5 Hz Doppler; (b) 40 Hz Doppler; (c) 100 Hz Doppler; (d) 200 Hz Doppler.

Table 7.11 Generator matrices for the four transmit antenna codes used in figure 7.22. These codes are in GF(4) with elements denoted by \{0, 1, a, 1 + a\}

<table>
<thead>
<tr>
<th>( (1 + a) + D )</th>
<th>( a + (1 + a)D )</th>
<th>( a + D )</th>
<th>( 1 + (1 + a)D )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( a + (1 + a)D )</td>
<td>( a + D )</td>
<td>( 1 + (1 + a)D )</td>
<td>( 1 + (1 + a)D )</td>
</tr>
<tr>
<td>( a + D )</td>
<td>( 1 + (1 + a)D )</td>
<td>( 1 + (1 + a)D )</td>
<td>( (1 + a) + aD )</td>
</tr>
<tr>
<td>( 1 + (1 + a)D )</td>
<td>( 1 + (1 + a)D )</td>
<td>( (1 + a) + aD )</td>
<td>( 1 + aD )</td>
</tr>
</tbody>
</table>

for the 16-state code

<table>
<thead>
<tr>
<th>( (1 + a) + (1 + a)D + aD^2 )</th>
<th>( (1 + a)D + aD^2 )</th>
<th>( 1 + D^2 )</th>
<th>( 1 + D^2 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( (1 + a)D + aD^2 )</td>
<td>( 1 + D^2 )</td>
<td>( 1 + (1 + a)D + (1 + a)D^2 )</td>
<td>( (1 + a) + aD + (1 + a)D^2 )</td>
</tr>
<tr>
<td>( 1 + D^2 )</td>
<td>( 1 + (1 + a)D + (1 + a)D^2 )</td>
<td>( (1 + a) + aD + (1 + a)D^2 )</td>
<td>( (1 + a) + aD )</td>
</tr>
<tr>
<td>( 1 + (1 + a)D + (1 + a)D^2 )</td>
<td>( (1 + a) + aD + (1 + a)D^2 )</td>
<td>( (1 + a) + aD )</td>
<td>( D + D^2 )</td>
</tr>
</tbody>
</table>

for the 256-state code
Figure 7.22 WER versus SNR of MIMO–OFDM systems with $n_t = n_r = 4$, TU channel with different Doppler frequencies. Here we compare the best code from the last figure with codes designed for four transmit antenna cases. See Table 7.11 and Figure 7.21 for details on the codes. (a) 5 Hz Doppler; (b) 40 Hz Doppler; (c) 100 Hz Doppler; (d) 200 Hz Doppler; (e) comparisons of WER for best MIMO–OFDM systems with perfect estimates and no Doppler.
7.6 SPACE–TIME CODED TDMA/OFDM RECONFIGURATION EFFICIENCY

In this section we compare two options for combating the fading in digital wireless communications, mainly the TDMA-based concept with equalization versus the OFDM concept. The structure of the code and the equivalent model of the encoder are shown in Figures 7.23 and 7.24 respectively. This structure is considered for the EDGE system. The presentation in this section is based mainly on [35–37].

7.6.1 Frequency selective channel model

The structure of this space–time trellis code (STTC) can be exploited to reduce the complexity of joint equalization and decoding in a frequency selective channel. This is achieved by embedding the space–time encoder in Figure 7.24 in the two channels $h_1$ and $h_2$, resulting in an equivalent single-input single-output (SISO) data-dependent channel impulse response (CIR) with memory $(v-1)$,
whose delay $D$-transform is given by

\[
h_{\text{STTC eqv}}^{\text{STTC}}(k, D) = h_1(0) + \sum_{m=1}^{v} (h_1(m) + p_k h_2(m - 1))D^m + p_k h_2(v)D^{v+1}
\]

where $D$ is the delay operator and $p_k = \pm 1$ is data dependent. Therefore, trellis-based joint space–time equalization and decoding with $8^{v+1}$ states can be performed on this equivalent channel. The traditional trellis equalization would require $8^v$ states, and STTC decoding requires eight states. The receiver block diagram is shown in Figure 7.25.

### 7.6.2 Front end prefilter

The objective of the prefilter is to shorten and shape the effective CIR seen by the equalizer to reduce its complexity (since the number of equalizer trellis states is exponential in the CIR memory).

### 7.6.3 Time-invariant channel

First, we describe the prefilter design problem for a time-invariant channel with memory $v$ and then extend it to the data-dependent time-varying channel case. Assume that the FIR prefilter has $N_f$ taps and denote its impulse response by the vector $w$. Then, the impulse response of the effective channel at the output of the prefilter is given by $h_{\text{eff}} = Hw$, where $H$ is the $(N_f + v) \times N_f$ Toeplitz convolution matrix. Let the vector $h_{\text{win}}$ contain the $(N_b + 1)$ taps (where $N_b < v$) of $h_{\text{eff}}$ to retain after shortening (whose energy is to be maximized), and let $h_{\text{wall}}$ contain the remaining taps (whose energy is to be minimized). Then

\[
h_{\text{win}} = J_{\text{win}} h_{\text{eff}} = J_{\text{win}} H w = H_{\text{win}} w
\]

where the $(N_b + 1) \times (N_f + v)$-dimensional matrix $J_{\text{win}}$ is constructed using columns of the identity matrix corresponding to tap positions of $h_{\text{win}}$ within $h_{\text{eff}}$. And

\[
h_{\text{wall}} = J_{\text{wall}} h_{\text{eff}} = J_{\text{wall}} H w = H_{\text{wall}} w
\]

where the $(N_f + v - N_b - 1) \times (N_f + v)$-dimensional matrix $J_{\text{wall}}$ is constructed from the columns of the identity matrix corresponding to tap positions of $h_{\text{wall}}$ within $h_{\text{eff}}$.

The prefilter design criterion maximizes the shortening signal to noise ratio (SSNR), the desired signal energy of the shortened channel contained in $h_{\text{win}}$, divided by the residual ISI energy in $h_{\text{wall}}$ plus the noise energy at the prefilter output.
7.6.4 Optimization problem

The problem reduces to the generalized eigenvector problem (for specific details see [36] and the references therein):

$$\max_w w^* B w \quad \text{subject to} \quad w^* A w = 1$$ (7.57)

where $(\cdot)^*$ denotes the complex conjugate transpose operation, $B = H_{\text{win}}^* H_{\text{win}}$, $A = H_{\text{wall}}^* H_{\text{wall}} + R_z$, and $H_z$ is the noise autocorrelation matrix at the prefilter input. The solution has the form

$$w_{\text{opt}} = (L_A)^{-1} u_{\text{max}} \quad (7.58)$$

here, $A = L_A L_A^*$ is the Cholesky factorization of the matrix $A$, and $u_{\text{max}}$ is the unitnorm eigenvector of matrix $(L_A)^{-1} B (L_A)^{-1}$ that corresponds to its largest eigenvalue $\lambda_{\text{max}}$. The resulting optimal SSNR is given by

$$\text{SSNR}_{\text{opt}} = \frac{w_{\text{opt}}^* B w_{\text{opt}}}{w_{\text{opt}}^* A w_{\text{opt}}} = \lambda_{\text{max}} \quad (7.59)$$

Equation (7.59) provides the optimal prefilter for a time-invariant channel.

7.6.5 Average channel

For the eight-state 8PSK STTC with two transmit antennas, we can design the prefilter for the average of the equivalent channel given in Equation (7.54). It can be shown [36] that the matrices $A$ and $B$ in this case are modified to

$$B = \left( H_{\text{win}}^1 \right)^* H_{\text{win}}^1 + \left( H_{\text{win}}^2 \right)^* H_{\text{win}}^2$$

$$A = \left( H_{\text{wall}}^1 \right)^* H_{\text{wall}}^1 + \left( H_{\text{wall}}^2 \right)^* H_{\text{wall}}^2 + R_z \quad (7.60)$$

where $H_{\text{win}}^i$ and $H_{\text{wall}}^i$ ($i = 1, 2$) are matrices corresponding to the constant channels $h_1$ and $h_2$ between the two transmit and the single receive antennas. The main attractive feature of the prefilter is that it is a single time-invariant (over a transmission block) FIR filter that shortens both channels $h_1$ and $h_2$ simultaneously without excessive noise enhancement.

7.6.6 Prefiltered M-BCJR equalizer

The algorithm as proposed in [38], is a reduced complexity version of the Bahl–Cocke–Jelinek–Raviv (BCJR) forward–backward algorithm presented in Appendix 2.1, also elaborated in [39], where at each trellis step, only the $M$ active states associated with the highest metrics are retained. An improved version of the $M$-BCJR algorithm was proposed in [40] based on a log domain implementation of the BCJR algorithm and operates as follows [37].

The forward and backward recursions independently select trees of active nodes without restricting one to be a subtree of the other. To form soft decisions at any time instant, we use all edges with at least one active node.

Let $L$ be the number of trellis steps; $Y_1 Y_2 \ldots Y_L$ the received outputs; $s_t$ the trellis state at time $t$; $S$ the number of trellis states and $u_t$ the input at time $t$. The quantity calculated by the algorithm is not $Pr (u_t = u | Y_1, \ldots, Y_L)$ as in BCJR, but an approximation, as detailed in what follows.

Using the channel observations and the channel description, calculate for each trellis step $t$ the quantities

$$\gamma_t (i, j) = Pr (s_t = j | Y_t, s_{t-1} = i) \quad (7.61)$$
The forward recursion (see Appendix 2.1)

1. For \( t = 0 \) (initialization) \( \alpha_0(0) = 1, \alpha_0(i) = 0 \) for \( i = 1 \ldots S \).

2. For \( t = 1, \ldots, L - 1 \)
   - \( \alpha_t(i) = \sum_j \alpha_{t-1}(j) \gamma_t(i, j) \)
   - Let \( A_t \) denote the \( M \) largest \( \gamma_t(i, j) \) at time \( t \). Any sorting algorithm can be used to construct \( A_t \).
     Set \( \alpha_t(i) = 0 \) if \( \alpha_t(i) \notin A_t \).

The backward recursion

1. For \( t = L \) (initialization) \( \beta_L(0) = 1, \beta_L(i) = 0 \) for \( i = 1 \ldots S \).

2. For \( t = L - 1, \ldots, 1 \)
   - \( \beta_t(i) = \sum_j \beta_{t+1}(j) \gamma_{t+1}(i, j) \)
   - Let \( B_t \) denote the \( M \) largest \( \gamma_t(i, j) \) at time \( t \). Any sorting algorithm can be used to construct \( B_t \).
     Set \( \beta_t(i) = 0 \) if \( \beta_t(i) \notin B_t \).

The probabilities of error \( E_{\alpha} \) and \( E_{\beta} \) (in the sense that the correct state is not included in the \( M \) selected states) can be calculated as follows:

\[
E_{\alpha} = Q\left(\sqrt{\frac{|h(0)|^2 d_{\min}^2}{2\sigma_z^2}}\right); \quad E_{\beta} = Q\left(\sqrt{\frac{|h(v)|^2 d_{\min}^2}{2\sigma_z^2}}\right) \tag{7.62}
\]

where \( Q(\cdot) \) is the standard \( Q \) function, \( d_{\min} \) the minimum Euclidean distance between any two constellation points and \( \sigma_z^2 \) the noise variance.

7.6.7 Decision

To make a decision at step \( 0 < t < L \) on the input \( u_t = u \), do the following:

1. Set \( P(u_t = u) = 0 \).

2. For all edges \((i, j)\) that have input \( u \)
   - if \( \beta_t(j) \neq 0 \) and \( \alpha_{t-1}(i) \neq 0 \), then \( P(u_t = u) = \alpha_{t-1}(i) \gamma_t(i, j) \beta_t(j) \);
   - if \( \beta_t(j) \neq 0 \) and \( \alpha_{t-1}(i) = 0 \), then \( P(u_t = u) = E_{\alpha} \gamma_t(i, j) \beta_t(j) \);
   - if \( \beta_t(j) = 0 \) and \( \alpha_{t-1}(i) \neq 0 \), then \( P(u_t = u) = \alpha_{t-1}(i) \gamma_t(i, j) E_{\beta} \).

The performance of the \( M \)-BCJR equalizer/decoder is further improved, especially for small \( M \), by using the prefilter of the previous subsection to concentrate the channel energy in a smaller number of taps. In fact, two different prefilters should be used for the forward and backward recursions since the forward recursion favors a close to minimum phase channel, whereas the backward recursion favors a close to maximum phase channel [40]. The value of \( M \) and the number of prefilter taps can be jointly optimized to achieve the best performance complexity tradeoffs.

7.6.8 Prefiltered MLSE/DDFSE equalizer complexity

To evaluate reconfiguration efficiency we need to estimate the complexity of the algorithm. For a size \( 2^b \) signal constellation, \( n_t \) transmit antennas and MIMO channel memory of \( v \), the MIMO MLSE equalizer has \( 2^{b n_t v} \) states in general. The number of equalizer states can be reduced by using the STTC trellis structure as shown in [41] or by a MIMO channel-shortening prefilter [42]. However, this complexity is still too high for large signal constellations, even for two transmit antennas and short-to-moderate MIMO channel memory. For example, for an 8PSK constellation and the EDGE TU channel (where \( V = 3 \)), the number of full MLSE equalizer states is equal to \( 8^6 = 262 144 \) states.
7.6.9 Delayed decision feedback sequence estimation (DDFSE)

In order to reduce complexity, DDFSE, as discussed in Chapter 6, was introduced in [43]. This is a hybrid scheme between MLSE and decision feedback equalization (DFE) for channels with long memory. Basically, the CIR is divided into a leading part and a tail. Then, an MLSE equalizer is constructed based on the leading part, and the interfering effect of the CIR tail is canceled by feedback using previous (hard) decisions (assumed correct).

At time \( k \), the branch metric \( \xi(k) \) of the DDFSE equalizer/decoder is given by

\[
\xi(k) = \left| y(k) - \sum_{i=0}^{n} h_{\text{STTC eqv}}^{\text{STTC}}(i)x(k-i) - \sum_{i=n+1}^{v+1} h_{\text{STTC eqv}}^{\text{STTC}}(i)\hat{x}(k-i) \right|^2
\]  

where

- \( y(k) \) is the \( k \)th received symbol;
- \( n \) is the design parameter (0 \( \leq n \leq v \)) that determines the number of DDFSE trellis states;
- \( h_{\text{STTC eqv}}^{\text{STTC}} \) is the impulse response vector of the equivalent channel;
- \( x(k) \) are all possible input symbols according to different transitions along the path history;
- \( \hat{x}(k) \) are the previous hard symbol decisions along the path history.

7.6.10 Equalization schemes for STBC

The focus is on the case of two transmit antennas where full-rate Alamouti-type space–time block codes can be constructed for any signal constellation.

7.6.10.1 Time-reversal space–time block coding (TR-STBC)

TR-STBC was introduced in [44] as an extension of the Alamouti STBC scheme to frequency selective channels by imposing the Alamouti orthogonal structure at a block, not symbol, level, as in the flat fading channel case. More specifically, the transmitted blocks from antennas one and two at time \((k+1)\) (which were denoted by \( x_1^{(k+1)} \) and \( x_2^{(k+1)} \), respectively) are generated by the encoding rule (for \( k = 0, 2, 4, \ldots \))

\[
x_1^{(k+1)} = -\bar{J}x_2^{(k)}; \quad x_2^{(k+1)} = Jx_1^{(k)}
\]  

where \( J \) is the time reversal matrix that consists of ones on the main antidiagonal and zeros everywhere else. To eliminate inter-block interference (IBI) between adjacent blocks due to channel memory, length-\( v \) all-zero guard sequences are inserted between information blocks.

The TR-STBC receiver in Figure 7.26 employs linear combining techniques (a spatio-temporal matched filter) to eliminate the mutual interference effects between the two transmit antennas while

![Figure 7.26 TR-STBC receiver block diagram. The operations (\( \bar{\cdot} \)) and (\( \tilde{\cdot} \)) denote complex conjugation and time reversal, respectively [37].](image-url)
still achieving the maximum diversity gain of $\|\mathbf{h}_1\|_2^2 + \|\mathbf{h}_2\|_2^2$ (where $\|\cdot\|$ denotes the norm of a vector).

TR-STBC uses a combination of complex conjugation, time reversal, and matched filtering operations, as described in detail in [29], to convert the two-input single-output channel to two single-input single-output (SISO) channels, each with an equivalent impulse response given by

$$h^{\text{TR- STBC}}_{\text{eqv}}(D) = h_1(D)\overline{h}_1(\overline{D}-1) + h_2(D)\overline{h}_2(\overline{D}-1)$$ (7.65)

to which standard SISO equalization schemes can be applied. In Equation (7.65), $h_i(D)$ is the $D$-transform of $h_i(k)$, and $\overline{h}_i(-k)$ for $i = 1, 2$. A whitened matched filter (WMF) front end can be used to convert $h^{\text{TR-STBC}}_{\text{eqv}}(D)$ to its minimum phase equivalent followed by trellis or feedback equalization, as will be further discussed later in the section. TR-STBC assumes that the two channels $h_1(D)$ and $h_2(D)$ are fixed over two consecutive transmission blocks and perfectly known at the receiver. In the next two subsections, we describe two alternative STBC joint equalization/decoding schemes that use frequency domain processing.

### 7.6.10.2 OFDM-STBC

In this case at the receiver end in Figure 7.27, received blocks are processed in pairs where their FFTs are computed and linearly combined. Finally, gain and phase adjustment is performed using minimum mean square error frequency domain equalization (MMSE-FDE) with a single complex tap for each subchannel, followed by a decision device. While the use of the Alamouti STBC modifies the channel frequency gain at subchannel $i$ from $|H(i)|^2$ to $|H_1(i)|^2 + |H_2(i)|^2$, which provides increased immunity against fading, decision errors occurring at any subchannel result in an irreducible error floor.

OFDM has two main drawbacks, namely, a high peak to average ratio (PAR), which results in larger backoff with non-linear amplifiers, and high sensitivity to frequency errors and phase noise [45]. An alternative equalization scheme that overcomes these two drawbacks of OFDM while retaining its reduced implementation complexity advantage (due to the use of FFT) is the single-carrier (SC) FDE [46], which has been extended to receive diversity systems in [47] and to Alamouti-type STBC transmit diversity systems in [48], and is described next.

These schemes can be extended to more than two transmit antennas using the theory of orthogonal designs presented in Chapter 4.

### 7.6.11 Single-carrier frequency domain equalized space–time block coding SC FDE STBC

The SC FDE, shown in Figure 7.28, is distinct from OFDM in that the IFFT block is moved to the receiver end and placed before the decision device. As noted in [46], this causes the effects of deep
nulls in the channel frequency response to be spread out, by the IFFT operation, over all symbols, thus reducing their effect and improving performance.

### 7.6.11.1 Encoder

Denote the $n$th symbol of the $k$th transmitted block (of length $N$) from antenna $i$ by $x_i^{(k)}(n)$. Then, the FDE-STBC encoding rule is given by [48]:

$$
\begin{align*}
x_1^{(k+1)}(n) &= -\bar{x}_1^{(k)}((-n)_N) \\
x_2^{(k+1)}(n) &= -\bar{x}_2^{(k)}((-n)_N)
\end{align*}
$$

for $n = 0, 1, \ldots, N-1$ and $k = 0, 2, 4, \ldots$ (7.66)

where $(-)_N$ denotes the modulo-$N$ operation that distinguishes this encoding scheme from TR-STBC, Equation (7.64). In addition, a cyclic prefix (CP) is added to each transmitted block to eliminate IBI and make the two channel matrices circulant. Taking the discrete Fourier 66 transform (DFT) of Equation (7.66), we see

$$
\begin{align*}
X_1^{(k+1)}(m) &= -\bar{X}_1^{(k)}; \\
X_2^{(k+1)}(m) &= -\bar{X}_2^{(k)}
\end{align*}
$$

for $m = 0, 1, \ldots, N-1$ and $k = 0, 2, 4, \ldots$ (7.67)

which reveals that this is also a *block-level* implementation of the symbol-level Alamouti encoding rule.

### 7.6.11.2 Receiver

After analog-to-digital (A/D) conversion, the CP part of each received block is discarded. Mathematically, we can express the input–output relationship over the $j$th received block as

$$
y^{(j)} = H_1^{(j)}x_1^{(j)} + H_2^{(j)}x_2^{(j)} + z^{(j)}
$$

(7.68)

where $H_1^{(j)}$ and $H_2^{(j)}$ are $N \times N$ circulant matrices whose first columns are equal to $h_1^{(j)}$ and $h_2^{(j)}$, respectively, appended by $(N - v - 1)$ zeros and $z^{(j)}$ is the noise vector. Since $H_1^{(j)}$ and $H_2^{(j)}$ are circulant matrices, they admit the eigendecompositions

$$
H_1^{(j)} = Q^*\Lambda_1^{(j)}Q; \quad H_2^{(j)} = Q^*\Lambda_2^{(j)}Q
$$

(7.69)

where $Q$ is the orthonormal FFT matrix and $\Lambda_1^{(j)}$ (respectively $\Lambda_2^{(j)}$) is a diagonal matrix whose $(n, n)$ entry is equal to the $n$th FFT coefficient of $h_1^{(j)}$ (resp. $h_2^{(j)}$). Therefore, applying the FFT to $y^{(j)}$, we
get (for \( j = k, k + 1 \))

\[
Y^{(j)} = Qy^{(j)} = \Lambda_1^{(j)}x_1^{(j)} + \Lambda_2^{(j)}x_2^{(j)} + z^{(j)} \quad (7.70)
\]

### 7.6.11.3 Processing

The length-\( N \) blocks at the FFT output are then processed in pairs, resulting in the two blocks (we drop the time index from the channel matrices since they are assumed fixed over the two blocks under consideration):

\[
\begin{bmatrix}
Y^{(k)} \\
\bar{Y}^{(k+1)}
\end{bmatrix} =
\begin{bmatrix}
\Lambda_1 & \Lambda_2 \\
\bar{\Lambda}_2 & -\bar{\Lambda}_1
\end{bmatrix}
\begin{bmatrix}
x_1^{(k)} \\
x_2^{(k)}
\end{bmatrix} +
\begin{bmatrix}
z^{(k)} \\
\bar{z}^{(k+1)}
\end{bmatrix} \quad (7.71)
\]

where \( x_1^{(k)} \) and \( x_2^{(k)} \) are the FFTs of the information blocks \( x_1^{(k)} \) and \( x_2^{(k)} \), respectively, and \( z \) is the noise vector. To eliminate inter-antenna interference, the linear combiner \( \Lambda \) is applied to \( Y \). Due to the orthogonal Alamouti-like structure of \( \Lambda \), a second-order diversity gain is achieved. By Alamouti-like we mean any \( 2 \times 2 \) complex orthogonal matrix of the form

\[
\begin{bmatrix}
c_1 & c_2 \\
\pm \bar{c}_2 & \mp \bar{c}_1
\end{bmatrix}
\]

Then, the two decoupled blocks at the output of the linear combiner are equalized separately, using the MMSE-FDE \([46]\), which consists of \( N \) complex taps per block that mitigate inter-symbol interference. Finally, the MMSE-FDE output is transformed back to the time domain using the inverse FFT where decisions are made. Note that the SC MMSE-FDE is equivalent to block MMSE linear equalization \([49]\); hence, its performance can be improved at the expense of increased complexity by adding a feedback section as discussed in \([50]\).

### 7.6.11.4 Channel estimator

Formulate the channel estimation problem for the two-transmit one-receive scenario as in Chapter 4. The analysis can be easily generalized to multiple transmit/receive antennas. Transmit two training sequences \( s_1 \) and \( s_2 \) from the first and second antennas simultaneously in synchronized data blocks, where each block consists of \( N \) information symbols and \( N_t \) training symbols. For two transmit antennas, the receiver uses the \( 2N_t \) known training symbols to estimate the \( 2(v + 1) \) unknown channel coefficients. The observed training sequence output, which does not have interference from information or preamble symbols, can be expressed as

\[
y = \begin{bmatrix} S_1 & S_2 \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \end{bmatrix} + z = Sh + z \quad (7.72)
\]

where the column vectors \( y \) and \( z \) are of size \((N_t - v)\), \( S_1 \), and \( S_2 \) are Toeplitz matrices of size \((N_t - v) \times (v + 1)\) that contain training symbols. The MMSE channel estimate assuming that \( S \) has full column rank, is given by \([51]\):

\[
\hat{h} = \begin{bmatrix} \hat{h}_1 \\ \hat{h}_2 \end{bmatrix} = (S^*S)^{-1}S^*y \quad (7.73)
\]

where \((\cdot)^{-1}\) denotes the inverse. The estimation error (mean square error) is given as

\[
MSE = E\left( (h - \hat{h})^* (h - \hat{h}) \right) = \sigma_z^2 \text{tr}((S^*S)^{-1}) \quad (7.74)
\]
where we assume that white noise with variance $\sigma_z^2$ and $\text{tr} (\cdot)$ denotes the trace of a matrix. The channel estimation MMSE is equal to

$$\text{MMSE} = \frac{\sigma_z^2(v + 1)}{(N_t - v)}$$

(7.75)

which is achieved if and only if

$$SS^* = \begin{bmatrix} S_1S_1 & S_1S_2 \\ S_2S_1 & S_2S_2 \end{bmatrix} = (N_t - v)I_{v+1}$$

(7.76)

where $I_{v+1}$ is the identity matrix of size $v + 1$. Two optimal training sequences that satisfy Equation (7.76) have an impulse-like autocorrelation sequence and zero cross-correlation. In this case, computing the channel estimates using Equation (7.73) reduces to a simple crosscorrelation (matrix–vector product).

### 7.6.11.5 Training sequences

For implementation purposes (to avoid non-linear amplifier distortion), it is desirable to use training sequences with constant amplitude. Optimal constant amplitude training sequences can be constructed from a $P$th root-of-unity alphabet $A_p = \{e^{i2\pi k/P} | k = 0, 1, \ldots, P - 1\}$ (where $i = \sqrt{-1}$) without constraining the alphabet size $P$. Such sequences are the perfect roots-of-unity sequences (PRUS), which are also known as polyphase sequences. Chu [52] showed that for any training sequence length $N_t$, there exists a PRUS with alphabet size $P = 2N_t$. In [53] and [54] the interested reader can find details on how to design PSK-type training sequences for dual-antenna transmissions with negligible performance loss from PRUS.

### 7.6.11.6 Performance results

For the performance results generation signal and channel parameters as in [37] are used. An 8PSK modulation with two transmit and one receive antennas on the TU EDGE channel is used. The overall CIR length is effectively four symbol periods, i.e. $v = 3$.

In EDGE, fading can be safely assumed to be quasi-static, i.e. the CIR can be assumed constant for the duration of a transmission block. This is due to the fact that the coherence time of the channel at around 1 GHz carrier frequency is much larger than the block duration of 577 $\mu$s, even for highway speeds. This eliminates the need for channel tracking at the receiver. In addition, assuming ideal frequency hopping, the fading process is independent from block to block. The noise samples are generated as independent samples of a zero mean complex Gaussian random variable with a variance of $1/$SNR per complex dimension. The reason for doubling the noise variance (compared with the single transmit antenna case) is that with two-antenna transmissions, we assume that the total transmitted power is the same as in the single antenna case and is divided equally between the two antennas. The average energy of the symbols transmitted from each antenna is normalized to one so that the signal to noise ratio is SNR. The results are shown in Figure 7.29.

For reconfiguration efficiency, relations from Chapter 4 are applicable where values for $D$ and $g_{12}$ can be derived from results presented in Tables 7.12 and 7.13. For two configurations, $g_{12}$ is obtained as a difference of the corresponding entries in column 2 and $D_r$ as a ratio of the corresponding complexity numbers from column 3.

In summary, for STTCs, the prefiltered $M$-BCJR equalizer/decoder outperforms the prefiltered DDFSE equalizer/decoder and has lower implementation complexity. For space–time block codes, TR-STBC achieves the best performance among the three investigated schemes. OFDM-STBC has the highest PAR and is the most sensitive to frequency errors but is also the most flexible among the three schemes in its support of multirate and multiQOS requirements. The three STBC schemes suffer the same amount of overhead (in the form of an all zero guard sequence for TR-STBC and a cyclic prefix guard sequence for FDE-STBC and OFDM-STBC).
Figure 7.29  (a) BER performance of two-transmit one-receive eight-state 8PSK STTC with prefilted $M$-BCJR equalizer as a function of $M$ (the number of active states). BER of a 4096-state full BCJR-MAP equalizer is shown as a benchmark; (b) BER performance of two-transmit one-receive eight-state 8PSK STTC with prefilted 64-state DDFSE, prefilted 16-state $M$-BCJR, and full BCJR-MAP equalizers; (c) BER performance of two-transmit one-receive eight-state 8PSK STTC with prefilted 16-state $M$-BCJR with perfect and estimated CSI. Full BCJR-MAP equalizer performance shown as BER lower bound; (d) BER performance of two-transmit one-receive OFDM-STBC, FDE-STBC, and TR-STBC. For OFDM and FDE-STBC, a size 64 FFT is assumed. For TR-STBC, an ideal whitened matched filter front end and a three-tap feedback filter are assumed; (e) effect of channel estimation on performance of SC FDE-STBC; (f) BER performance of two-transmit one-receive TR-STBC with 512-state full BCJR-MAP, eight-state $M$-BCJR, and SISO MMSE-DFE with $N_b = 3$ feedback taps. An ideal whitened matched filter front end is assumed.
Figure 7.29 (Cont.)

(b) EDGE TU Channel, 2TX 1RX, 8 State 8PSK STTC

(c) EDGE TU Channel, 2TX 1RX, 8 State 8PSK STTC
Figure 7.29 (Cont.)
Figure 7.29 (Cont.)

Table 7.12 Performance and complexity comparison summary between the equalization schemes for the eight-state 8PSK STTC over the TU EDGE channel

<table>
<thead>
<tr>
<th>Equalization scheme</th>
<th>SNR (dB) at BER = 10^{-3}</th>
<th>Receiver complexity (per block)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full BCJR-MAP</td>
<td>21.3</td>
<td>4096 states (each direction)</td>
</tr>
<tr>
<td>Prefiltered M-BCJR</td>
<td>23.1</td>
<td>16 states (each direction) 8-tap prefilter (each direction)</td>
</tr>
<tr>
<td>Prefiltered DDFSE</td>
<td>23.6</td>
<td>64 states and 32-tap prefilter</td>
</tr>
</tbody>
</table>

Table 7.13 Performance and complexity comparison summary between the Alamouti-type STBC equalization schemes over the TU EDGE channel, assuming 8PSK modulation and block size of 64

<table>
<thead>
<tr>
<th>Equalization scheme</th>
<th>SNR (dB) at BER = 10^{-3}</th>
<th>Receiver complexity (per block)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full BCJR-MAP</td>
<td>21.3</td>
<td>512 states (each direction) and 20-tap WMF</td>
</tr>
<tr>
<td>TR-STBC</td>
<td>22.2</td>
<td>20-tap WMF and 3 feedback taps</td>
</tr>
<tr>
<td>FDE-STBC</td>
<td>24.2</td>
<td>Size 64 FFT/IFFT and 64-tap FDE</td>
</tr>
<tr>
<td>OFDM-STBC</td>
<td>26.5</td>
<td>Size 64 FFT and 64-tap FDE</td>
</tr>
</tbody>
</table>

Ideal whitened matched filter assumed
7.7 MULTICARRIER CDMA SYSTEM

In Chapter 1 we discussed a variety of different structures for MC CDMA at the introductory level. In a number of the following sections we will provide more in-depth discussion on the performance of these systems. In the system shown in Figure 7.30 [55, 56] the MC-CDMA BPSK signal transmitted by the $k$th user is:

$$s_k(t) = \text{Re} \left\{ \sum_{n=-\infty}^{+\infty} u_k[n] h(t - nT_c - \tau_k) \sum_{l=0}^{L-1} e^{j(\omega_l t + \psi_{k,l})} \right\}$$  \hspace{1cm} (7.77)

where

$$u_k[n] = AC_{pk}[n] + BC_{dk}[n]d_k[n]$$  \hspace{1cm} (7.78)

$A$ and $B$ are the signal amplitudes of the pilot and the data channel respectively, $d_k[n]$ is binary data, $h(i)$ is the impulse response of the chip wave-shaping filter, $\omega_l$ and $\psi_{k,l}$ are the carrier frequency and carrier phase of the $l$th subcarrier, respectively, and $L$ is the number of subcarriers.

The signal is transmitted through a fading channel. The bandwidth of the subcarriers in this section are selected such that each subcarrier experiences independent, slowly varying, flat Rayleigh fading. Assuming perfect average power control, the received signal is given by

$$r(t) = \text{Re} \left\{ \sum_{k=0}^{K-1} \sum_{n=-\infty}^{+\infty} u_k[n] h(t - nT_c - \tau_k) \sum_{l=0}^{L-1} \alpha_{k,l} e^{j(\omega_l t + \theta_{k,l})} \right\} + n_w(t)$$  \hspace{1cm} (7.79)

where $K$ is the total number of users, $\tau_k$ is the relative time delay of user $k$, $\alpha_{k,l}$ and $\theta_{k,l}$ are the fading amplitude and phase, respectively, of the $l$th path for the $k$th user, and $n_w(t)$ is zero mean white Gaussian noise with two-sided spectral density $\eta_0/2$.

The channel estimator based on a pilot signal is shown in Figure 7.31. Assuming

$$X(f) \equiv |H(f)|^2 = \begin{cases} 
\frac{1}{W}, & -\frac{W}{2} < f < \frac{W}{2} \\
0, & \text{otherwise}
\end{cases}$$  \hspace{1cm} (7.80)

$$W = \frac{1}{T_c}$$

$$\mathcal{F}^{-1}|H(f)|^2 \equiv x(t)$$

and

$$\int_{-\infty}^{+\infty} |H(f)|^2 df \equiv 1$$

the $l$th complex channel estimate is given by

$$W_l = a_{0,l} e^{j\theta_{l,0}} AN_l + I_{p,l} + N_{p,l}$$  \hspace{1cm} (7.81)

![Figure 7.30 The complex transmitter block diagram [55].](image-url)
Figure 7.31 The complex channel estimator block diagram [55].

Figure 7.32 The complex data demodulation block diagram.

We have used the fact that the spreading sequences $C_{p[k]}$ and $C_{d[k]}$ are orthogonal in an estimation interval.

### 7.7.1 Data demodulation

Each subcarrier of the received signal is chip-matched filtered, demodulated, despread by the corresponding data spreading sequence, and then integrated over the bit interval of $N$ chips (see Figure 7.32). The $l$th demodulator output before combining is

$$Y_l = \alpha_0 e^{j\theta_0} B_d[l] + I_{p,l} + N_{d,l}$$

(7.82)

where the interference term is

$$I_{d,l} = \sum_{n=0}^{N-1} C_{d0}[n] R_n$$

(7.83)

and the noise term is

$$N_{d,l} = \sum_{n=0}^{N-1} \left[ n_w(t) * h^*(t) \right]_{t=nT_c} \cdot C_{d0}[n]$$

(7.84)

The combined signal is given by

$$Y = \sum_{l=0}^{L-1} W^*_l Y_l$$

(7.85)
and the final decision statistic is

\[ Z = \text{Re}\{Y\} = \sum_{i=0}^{L-1} \left[ \frac{1}{2} W_i^* Y_i + \frac{1}{2} W_i Y_i^* \right] \]  

(7.86)

Bit error rate analysis for such a system can be found in [55].

### 7.7.2 Performance examples

In this example the signal parameters are the same as in [55]. The bandwidth of each subcarrier is fixed to be the coherence bandwidth of the channel. The total bandwidth is proportional to the number of subcarriers. To make the comparison between different bandwidths, the total transmit power is kept constant, that is, decreasing the transmit power per subcarrier as the number of subcarriers increases. Traditionally, assuming perfect channel estimation, the probability of error improves monotonically with the number of subcarriers [56]. However, when there is estimation error, the situation is different.

The probability of error is plotted against the number of subcarriers \( L \) in Figure 7.33(a) with ten total users, the estimation interval \( N_i \) equaling 64 chips, and \( E_b/\eta_0 \) of 4, 7, 10 and 13 dB. A processing gain of 64 is used, and there is equal power in the pilot and the data channel. As the number of subcarriers increases, the bit error rate first improves and then degrades. The increasing \( L \) helps performance by introducing diversity gain. At the same time, as \( L \) goes up, the transmit energy-per-band goes down; this causes more estimation error, and in turn, results in performance degradation. Thus, an optimal value of \( L \) exists. When we increase the \( E_b/\eta_0 \), the optimal \( L \) becomes larger, because the higher signal to noise ratio reduces the degradation due to the estimation error. Some additional results are shown in Figures 7.33 (b) and (c).

### 7.8 MULTICARRIER DS-CDMA BROADCAST SYSTEMS

As pointed out in Chapter 1, multicarrier direct sequence code division multiple access (DS-CDMA) systems can be classified into two categories: those with overlapping bandwidths [57, 58] and those with disjoint bandwidths [56]. In this section, the non-overlapping bandwidth system is considered, employed in the forward link (base-to-mobile link) of a cellular system, wherein all user signals are synchronous. Using this type of multicarrier DS-CDMA to generate a wideband CDMA waveform, in particular by choosing the bandwidth of a subcarrier equal to that of a narrowband CDMA waveform, we can achieve some degree of compatibility between wideband (e.g. UMTS/WCDMA) and narrowband (e.g. UMTS/cdma2000) CDMA systems. The spectra of single-carrier and multicarrier CDMA are shown in Figures 7.34(a) and (b), respectively.

A base station transmitter using a multicarrier DS-CDMA system is shown in Figure 7.35. The transmitted signal is given by

\[ s(t) = \text{Re} \left( \sum_{m=1}^{M} S_m(t) \sqrt{2} e^{j(\omega_m t + \theta_m)} \right) \]  

(7.87)

\( S_m(t) \) is given by

\[ S_m(t) = \sum_{k=1}^{K} \sqrt{E_c} \sum_{n=-\infty}^{+\infty} d_v^{(k)} c_n^{(k)} e^{j\phi_k m} h(t - nT_c) \]  

(7.88)

where \( d_v^{(k)} \) and \( c_n^{(k)} \) are the data and the spreading sequences of the \( k \)th user, respectively, \( v = \lfloor n/N \rfloor \) where \( \lfloor x \rfloor \) is the largest integer less than or equal to \( x \) and \( N \) is the number of chips of the spreading sequence per data bit), \( h(t) \) is the impulse response of the chip wave-shaping filter, \( \phi_k m \) is a carrier phase randomly chosen by a base station for the \( m \)th subchannel of the \( k \)th user, and \( 1/T_c \) is the chip rate. Both \( \theta_m \) and \( \phi_k m \) are uniformly distributed over \([0, 2\pi]\). The energy per data bit is defined as \( E_b \equiv E_c NM \). Note that the need for \( \phi_k m \) to be random for different users and different
10 total users, $N_i = 4 \times 64, E_b/N_0 = 4, 7, 10, 13$ dB

(a) Number of subcarriers ($L$)/BW expansion

(b) $E_b/N_0 = 10$ dB, 5 users, $N = 64, N_i = N, 2N, 3N, 4N$

Figure 7.33 (a) Probability of error versus number of subcarriers for varying $E_b/N_0$; (b) probability of error versus number of subcarriers for different estimation intervals; (c) probability of error versus number of subcarriers for different numbers of users.
Figure 7.33 (Cont.).

Figure 7.34 Spectra of (a) single-carrier CDMA; (b) multicarrier CDMA; and (c) hybrid multicarrier CDMA/FDM.

Figure 7.35 Block diagram of a multicarrier CDMA base station transmitter.
subchannels is to obtain a proper processing gain in the multicarrier system. The reason for this is as follows. A multicarrier system is known to provide an effective processing gain of $MN$ [59] for an asynchronous communication link, where the carrier phase difference between different users’ signals can be assumed to be random, and the phases of any given user are uncorrelated in different subchannels. Therefore, coherent combining of the despread signals from $M$ correlators increases the processing gain by $M$ times that obtained by despreading the signals in each subchannel. However, for the forward link, without the $\phi_{k,m}$, the interference components of the $M$ correlator outputs are also combined coherently, and consequently, there is no increased processing gain achieved by the coherent combining of the $M$ correlator outputs, only a diversity gain. With the random $\phi_{k,m}$, it is possible for the receivers to demodulate the data coherently by using a common pilot signal from a base station if each receiver is provided the values of the $\phi_{k,m}$ by the base station when it establishes a communication link.

It is assumed that $c_n^{(k)} = a_n b_n^{(k)}$, where $a_n$ is a random sequence commonly used by all users and $b_n^{(k)}$ is a member of either an orthogonal or a quasi-orthogonal code set assigned to the $k$th user. We also have

$$E\{c_n^{(k)} c_{n+1}^{(k')}\} = 0 \quad \text{for} \quad i \neq 0 \quad (7.89)$$

for all $k$ and $k'$. It is also assumed that the $b_n^{(k)}$ satisfy the following relationship for all $k$:

$$\frac{1}{K-1} \sum_{k'=1}^{K} \left( \sum_{n=0}^{N-1} b_n^{(k)} b_n^{(k')} \right)^2 = (1-q)N \quad (7.90)$$

where $q$ is a measure of the orthogonality of the set of quasi-orthogonal codes. The quasi-orthogonal codes with these characteristics are standardized [60].

Further, we assume that the channel is a slowly varying frequency selective Rayleigh fading channel and is modeled as a finite length tapped delay line. The complex low-pass equivalent response for the $m$th subchannel is given by

$$c_m(t) = \sum_{i=0}^{L-1} \alpha_{m,i} e^{j\beta_{m,i}} \delta(t - iT_c) \quad (7.91)$$

where $L$ is the number of resolvable paths, $\alpha_{m,i}$ are independent but not necessarily identically distributed Rayleigh random variables, and the $\beta_{m,i}$ are independently, identically distributed (i.i.d.), uniform random variables over $[0, 2\pi)$. A unit energy constraint is assumed, i.e. $\sum_{i=0}^{L-1} E[\alpha_{m,i}^2] = 1$. Then, for a constant multipath intensity profile (MIP), the second moment of each path of a subchannel is given by $E[\alpha_{m,i}^2] = 1/L$. For an exponential MIP, the second moments are assumed to be related to the second moment of the initial path strength by $E[\alpha_{m,i}^2] = E[\alpha_{m,0}^2] \exp(-ri)$, where $r$ is the MIP decay factor. For comparison of the performance of single-carrier and multicarrier systems, we use the facts that $r = Mr_1$ and $L = L_1/M$, where $r_1$ and $L_1$ are the MIP decay factor and the number of resolvable paths, respectively, for the single-carrier system [61].

Since all signals that arrive from the base station at a given mobile unit propagate over the same path, they all fade in unison. The received signal at the desired mobile unit is then given by

$$r(t) = \text{Re}[R_c(t)] + n(t) \quad (7.92)$$

where $R_c(t)$ is the complex representation of the received signal given by:

$$R_c(t) = \sum_{m=1}^{M} \sum_{i=0}^{L-1} \alpha_{m,i} S_m(t - iT_c) \sqrt{2} e^{j(\omega_m t + \theta_{m,i})} \quad (7.93)$$

$n(t)$ is additive white Gaussian noise (AWGN) with a two-sided power spectral density of $\eta_0/2$, and $\theta_{m,i} = \theta_m + \beta_{m,i}$. The corresponding expression for the single-carrier system is obtained by setting $M = 1$ and $L$ equal to $L_1$.

The receiver of the desired user ($k = 1$) is shown in Figure 7.36, where a RAKE receiver in each subchannel, and perfect phase recovery of each carrier from the pilot signal detector is assumed [59].
The chip wave-shaping filter given in [59] is assumed, where \( X(f) \equiv |H(f)|^2 \) is a raised cosine filter. The DS waveforms do not overlap and therefore, adjacent channel interference may be ignored.

Each data stream modulates \( H = M/R \) disjoint carriers, and the number of chips of the spreading sequence per data bit is \( N_1/H \). To make a fair comparison of the performance, given a fixed information rate and total bandwidth, the relationship \( HRL = ML = L_1 \) must be satisfied. Bit error rate analysis for such a system can be found in [62] and some results are shown in Figure 7.37.

### 7.9 FRAME BY FRAME ADAPTIVE RATE CODED MULTICARRIER DS-CDMA SYSTEM

In this section we discuss an adaptive rate convolutionally coded multicarrier direct sequence code division multiple access (DS-CDMA) system. In order to accommodate a number of coding rates easily and make the encoder and decoder structure simple, the rate compatible punctured convolutional (RCPC) code discussed in Chapter 2 is used. We choose the coding rate that has the highest data throughput in the signal to interference and noise ratio (SINR) sense. To achieve maximum data throughput, a rate adaptive system is used, based on the channel state information (the signal to interference to noise ratio, SINR, estimate). The SINR estimate is obtained by the soft decision Viterbi decoding metric. It will be demonstrated that the rate adaptive convolutionally coded multicarrier DS-CDMA system can enhance spectral efficiency and provide frequency diversity.
Figure 7.37 (a) Performance of a single-carrier and a multicarrier CDMA system in multipath fading channel when both systems employ an orthogonal code set; (b) performance comparison of a single-carrier CDMA, a multicarrier CDMA and a hybrid multicarrier CDMA/FDM system for $K = 120$ in a multipath fading channel with a constant MIP; (c) probability of bit error versus $K$ for $E_b/\eta_0 = 15$ (decibels).
Figure 7.37 (Cont.).

Figure 7.38 A typical CC-OM power spectral density.

7.9.1 Transmitter

The power spectral density of a convolutionally coded orthogonal multicarrier (CC-OM) signal and the transmitter for the rate adaptive CC-OM DS-CDMA system considered in this section are shown in Figure 7.38 and Figure 7.39 respectively. For user $k$, the (information) bits $\{b_i^k\}$, each with duration $T_b$, are encoded by the RCPC encoder of rate $r$. The relationship between $T_b$ and the duration $T_s$ of a coded binary symbol can be written as

$$T_s = r M T_b \quad (7.94)$$

where $M$ is the number of subchannels. The $M$ coded binary symbols are allocated to $M$ subchannels to get frequency diversity. They are interleaved to get time diversity as well as frequency diversity, and are spread by each user’s pseudonoise (PN) signature waveform $c_k(t)$ with chip duration $T_c = T_s / N$, where $N$ is the processing gain of the DS narrowband waveforms modulated by subcarriers. For CC-OM systems, we have $M = (2B_T / B_S) - 1$, where $B_T$ and $B_S$ are the total and subchannel bandwidths, respectively. Since we fix the subchannel bandwidth $B_S$ (or equivalently, the symbol duration $T_s$) in this section, $T_b$ varies according to Equation (7.94) when the code rate $r$ changes.
Figure 7.39 The transmitter model for user \( k \) in the adaptive rate CC-OM DS-CDMA system.

The transmitted signal \( s_k(t) \) of user \( k \) can be written as

\[
s_k(t) = \sqrt{2P} \sum_{j=-\infty}^{\infty} \sum_{m=1}^{M} x_{k,m}^j c_k(t - jT_s) \cos(\omega_m t + \varphi_{k,m})
\]  

(7.95)

The channel is assumed to be frequency selective Rayleigh fading and not to vary during one symbol duration. However, the subchannels are assumed to be non-selective by choosing the number of subcarriers appropriately as [56]:

\[
MT_c \geq T
\]  

(7.96)

where \( T \) is the maximum delay spread of the channel. Then the complex low-pass impulse response of the subchannels of user \( k \) can be modeled as

\[
h_{k,m}(t) = \alpha_{k,m} e^{j\beta_{k,m}} \delta(t)
\]  

(7.97)

where \( \alpha_{k,m} \) is the fading amplitude and \( \beta_{k,m} \) is the random phase of the \( m \)th subchannel, \( m = 1, 2, \ldots, M \). The phases \( \{\beta_{k,m}\} \) are i.i.d. uniform random variables on \([0, 2\pi)\). In general, the fading amplitudes \( \{\alpha_{k,m}\} \) are correlated, but we can assume that they are i.i.d. Rayleigh random variables once the coded symbols are properly interleaved in the time domain.

Frame by frame transmission is assumed. Such a frame by frame transmission is typical of many cellular systems. The frame discussed in this section is shown in Figure 7.40. Each frame of duration \( T_f \) consists of a header of duration \( T_p \) and data symbols of duration \( T_d \). We have \( T_d = N_s T_s \) and \( T_p = N_p T_s \), where \( N_s \) is the number of data symbols and \( N_p \) is normally \( 6–10 \). The header contains pilot symbols and information on the rate and channel state. The function of the MUX in Figure 7.39 is to combine the header and data symbols to make frames as shown in Figure 7.40.

### 7.9.2 Receiver

The receiver for the adaptive rate CC-OM DS-CDMA system in this section is shown in Figure 7.41. Let us assume that there are \( K \) users in a cell and power control is employed. Then the received signal
at the base station can be written as

\[ r(t) = \sqrt{\frac{2}{P}} \sum_{j=-\infty}^{\infty} \sum_{k=1}^{K} \sum_{m=1}^{M} \alpha_{k,m} x_{k,m}^j c_k(t - \tau_k - jT_s) \cos(\omega_m t + \phi_{k,m}) + n(t) \]  

(7.98)

### 7.9.3 Rate-compatible punctured convolutional (RCPC) codes

RCPC codes are discussed in Chapter 2. For this section, some additional details are specified. Let the code rate and constraint length of the parent code be \( R = \frac{1}{n} \) and \( L_c \), respectively. The parent code is completely specified by the \( n \) generator polynomials \( G^j(D) = g_0^j + g_1^j D + \cdots + g_{L_c-1}^j D^{L_c-1}, \) \( j = 1, 2, \ldots, n \), where \( g_i^j \in \{0, 1\} \). The puncturing is done according to the rate compatibility criterion, which requires that lower rate codes use the same coded bits as the higher rate codes plus one or more additional bit(s). The bits to be punctured are described by an \( n \times p \) puncturing matrix \( P \) consisting of zeros and ones, where \( p \) is called the puncturing period. At time instant \( t \), the output from each generator \( G^j(D) \) is transmitted if \( P(j, t \mod p) = 1 \) and punctured otherwise. Here, \( P(a, b) \) denotes the element on row \( a \) and column \( b \) in the matrix \( P \). The number \( p \) of columns determines the number of code rates and the rate resolution that can be obtained. Generally, from a parent code of rate \( 1/n \),
we can obtain a family of \((n−l)p\) different codes with rates

\[
r = \frac{p}{np}, \frac{p}{np-1}, \ldots, \frac{p}{p+1}
\]

(7.99)

The code rate of RCPC codes can be changed during even one information bit transmission and, thus, unequal error protection can be obtained [8]. In this section, however, the code rate of RCPC codes is changed frame by frame, not bit by bit, because we assumed frame by frame transmission. An example of the RCPC encoder is shown in Figure 7.42.

### 7.9.4 Rate adaptation

A threshold-based adaptation scheme is used which adaptively changes the coding rate depending upon the SINR estimated. Let \(\theta_0 = -\infty, \theta_1, \theta_2, \ldots, \theta_Q = \infty\) be the SINR threshold values, which are chosen such that between \(\theta_{j−1}\) and \(\theta_j\) the channel coding rate \(r_j\) has the highest throughput. Here, \(Q\) is the number of possible code rates. Then, as discussed in Chapter 3, the transmitter mode (rate) adaptation scheme can be defined as follows:

\[
\text{Choose } r_j \text{ if } \theta_{j−1} \leq \text{SINR} < \theta_j, \quad j = 1, \ldots, Q
\]

(7.100)

In this method, bit by bit adaptation is not assumed due to the feedback delay. Instead, we choose the adaptation interval \(T_a\) in such a way that \(T_a\) is long enough to allow the transmission of at least one frame and short enough to react quickly to the possible change of the SINR. The transmitter can then adapt its data rate every \([T_a/T_i]T_i\). This allows efficient error recovery through ARQ mechanisms even with dynamic rate adaptation. The rate at which the transmitter reacts to the changes in the SINR depends on the SINR estimate and feedback delay in the system [64].

#### 7.9.4.1 Example

RCPC codes with rate 1/4 convolutional codes of constraint length \(L_c = 5\) and 9 are used as the parent codes [64, 65], as the error control capability varies when the constraint length changes, we use two values of \(L_c\). The number, \(M\), of subcarriers is four and nine. The processing gain \(N\) is 192
and 96, for \( M = 4 \) and \( M = 9 \), respectively, when the total bandwidth
\[
B_T = N(M + 1)/T_s
\]
is fixed. We assume each frame contains 144 symbols with \( T_f = 10 \) ms and \( T_p = 0 \) (that is, we assume perfect feedback to simplify the simulations).

The adaptive rate CC-OM DS-CDMA system was implemented based on the above discussion in [66]. When \( L_c = 5 \), fixing the coding rate to \( 1/2 \) allows us to get the highest throughput and the result is Figure 7.43(a). When \( L_c = 9 \), using the thresholds \( \theta_1 = 2.5 \) dB and \( \theta_2 = 5.5 \) dB, we get

![Throughput curves](image)

Figure 7.43 (a) The adaptive throughput curves and the conventional throughput curves when \( L_c = 5 \); (b) the adaptive throughput curves and the conventional throughput curves when \( L_c = 9 \).
Figure 7.43(b). In these figures, the throughput of the conventional system (fixed rate with $1/M$) is also shown. It is clear that we can get much higher throughput with the adaptive system.

7.10 INTERMODULATION INTERFERENCE SUPPRESSION IN MULTICARRIER CDMA SYSTEMS

In this section, a coded multicarrier direct sequence code division multiple access (DS-CDMA) system is presented that, by the use of a minimum mean squared error receiver, achieves frequency diversity (instead of path diversity as in a conventional single carrier (SC) RAKE DS-CDMA). It also has the ability to suppress the intermodulation distortion and partially compensate for the signal distortion introduced by a non-linear amplifier at the transmitter. A frequency selective Rayleigh fading channel is decomposed into $M$ frequency non-selective channels, based on the channel coherence bandwidth. A rate $1/M$ convolutional code, after being interleaved, is used to modulate $M$ different DS-CDMA waveforms. The system is shown to effectively combat intermodulation distortion in the presence of multiple access interference.

7.10.1 Transmitter

In this section, the transmitter shown in Figure 7.44 is considered. The input signal to the power amplifier for the $k$th user $s_k(t)$ is given by

$$s_k(t) = A_k \sum_{i=-\infty}^{\infty} \sum_{m=1}^{M} d_{k,m}^{(i)} p_{k,m}(t - iY - \tau_k) \cos(\omega_m t + \theta_{k,m}) \quad (7.102)$$

and $p_k(t)$ is a spreading (or signature) waveform given by

$$p_{k,m}(t) = \sum_{n=0}^{N-1} c_{k,m}^{(n)} h(t - nMT_c) \quad (7.103)$$

Figure 7.44 Transmitter block diagram for user $k$. 
In Equation (7.103), \( c_{k,m} \in \{ \pm 1 \} \) is the \( n \)th chip of the spreading sequence, \( N \) is the processing gain, which is taken to be equal to the period of the spreading sequence, \( h(t) \) is the impulse response of the chip wave-shaping filter, \( 1/MT_c \) is the chip rate of the band-limited MC DS-CDMA system, and \( 1/T_c \) is the chip rate of a band-limited single carrier (SC) DS-CDMA system that occupies the same spread bandwidth as does the MC system. That is, \( T = NMT_c \).

### 7.10.2 Non-linear power amplifier model

For an input signal formed by the sum of \( M \) subcarrier signals, i.e.

\[
x(t) = \text{Re} \left\{ \sum_{m=1}^{M} a_m(t) \exp[j\omega_0 t + j\psi_m(t)] \right\}
\]

\[
= \text{Re} \left\{ A(t) \exp[j\omega_0 t + j\Psi(t)] \right\}
\]

(7.104)

the output signal of a non-linear power amplifier can be represented by

\[
\hat{x}(t) = \text{Re} \left\{ g[A(t)] \exp[j\omega_0 t + j\Psi(t) + jf[A(t)]] \right\}
\]

(7.105)

where \( g(A) \) and \( f(A) \) represent the AM/AM (see Figure 7.45) and AM/PM conversion characteristics of the non-linear power amplifier.

### 7.10.3 MMSE receiver

As shown in Figure 7.46, \( M \) MMSE filters are combined with a soft-decision Viterbi decoder so that the soft outputs from the \( M \) MMSE filters are parallel-to-serial converted, deinterleaved, and decoded. The received DS-CDMA signal after the LPF for each subcarrier is despread (either partially or fully) over consecutive \( F \) chips, which is characterized by a parameter \( N_t = \lceil N/F \rceil \). The decision symbols needed by the \( M \) MMSE filters can be obtained via an interleaver with a serial-to-parallel converter and an encoder which is identical to that used in the transmitter. The tap weight vector of
the $m$th MMSE filter $\omega_m$ is chosen so as to minimize the conditional mean square error, conditioned on all parameters of the desired user (user 1) and certain parameters of the MAI and IM, i.e. with $q = \{\alpha_1, \theta_1, \rho_{k,I(m,h)}\}$, we have

$$\text{MSE} = E\{(\omega'_m z_m - d_{1,m})^2 | q\} \quad (7.106)$$

For the notation see Figure 7.46. Note that we omit the superscript $i$, which denotes the estimated bit, for notational simplicity. From Chapter 5, the optimum tap weight vector for Equation (7.106) is given by:

$$(\omega_m)_{opt} = R_m^{-1}a_m \quad (7.107)$$

where

$$R_m = E\{z_m z'_m | q\} \quad (7.108)$$

and

$$a_m = E\{d_{1,m} z_m | q\} \quad (7.109)$$

### 7.10.3.1 Example

For the numerical example, we use the same parameters as in [67], the PA model given in the previous section and $M = 4$, $N = 32$, $R = 1/4$ and constraint length $K = 3$ coded MC-CDMA system and an SC coded RAKE system with the same coding scheme and the same bandwidth as the coded MC-CDMA system. Note that there are not any IM terms in the SC system, but the non-linear distortion introduced by the PA is taken into account. The same PA output power is assumed for both the SC input signal and the MC input signal. The system performance is obtained by simulation. In Figures 4.47–48, the maximal number of resolvable paths for the SC is denoted $L_p$, and $L_t$ is the actual number of RAKE taps.
Figure 7.47 Probability of bit error in the presence of IMD (different PN (DPN) code for each subcarrier).

Figure 7.48 Comparison between the systems with the same (SPN) and the different (DPN) PN codes for each subcarrier.
7.11 SUCCESSIVE INTERFERENCE CANCELLATION IN MULTICARRIER DS-CDMA SYSTEMS

This section presents a successive interference cancellation (SIC) scheme for a multicarrier (MC) asynchronous DS-CDMA system, wherein the output of a convolutional encoder modulates band-limited spreading waveforms at different subcarrier frequencies. In every subband, the SIC receiver successively detects the interferers signals and subtracts them from that of the user of interest. The SIC receiver employs maximal ratio combining (SIC-MRC) for detection of the desired user, and feeds a soft decision Viterbi decoder.

7.11.1 System and channel model

A $K$-user asynchronous communication system is presented. We assume knowledge of the time delays and the spreading sequences of all the users, but no knowledge of the channel gains of the interferers. The user data symbols are input to a rate $1/M$ convolutional encoder. The output code symbols are interleaved and serial-to-parallel (S/P) converted such that $M$ parallel code symbols may be transmitted simultaneously. Then, each of the $M$ code symbols is replicated by a rate $1/R$ repetition code, and transmitted over $MR$ subcarriers.

For example, if $M = 4$ and $R = 2$, then the $M = 4$ code symbols are mapped to a total of eight subcarriers in the following way: the first code symbol is transmitted on the first and the fifth subcarriers, the second code symbol is transmitted on the second and the sixth subcarriers, the third code symbol is transmitted on the third and the seventh subcarriers, and the fourth code symbol is transmitted on the fourth and the eighth subcarriers. Therefore, the minimum subcarrier distance for the same code symbol is maximized. The mapped code symbols are then multiplied by the spreading sequence assigned to the given user. The transmitted signal for the $k$th user is:

$$S_k(t) = \sqrt{2E_{ck}} \left\{ \sum_{j=-\infty}^{+\infty} a_k(t - jT_k + \tau_k) \times \sum_{m=1}^{MR} b_{k,[m]}^j M \cos(\omega_m t + \theta_{k,m}) \right\}$$  \hspace{1cm} (7.110)

where

$$a_k(t) = \sum_{n=0}^{N-1} c_k^{(n)} h(t - nT_c)$$

We assume that the channel in each subband is a slow-varying frequency non-selective Rayleigh channel with transfer function

$$\zeta_{k,m} = \alpha_{k,m} \exp(j\beta_{k,m})$$

The received signal is given by

$$r(t) = \sum_{k=1}^{K} \sqrt{2E_{ck}} \sum_{j=-\infty}^{+\infty} a_k(t - jT_k + \tau_k) \times \sum_{m=1}^{MR} b_{k,[m]}^j M \alpha_{k,m} \cos(\omega_m t + \theta_{k,m}) + n_w(t)$$ \hspace{1cm} (7.111)

The receiver structure for the desired user, user 1, is shown in Figures 7.49–50. Interferers 2,3, . . . , $K$ are renumbered as $1_m, 2_m, \ldots, (K-1)_m$, which defines the cancellation order such that in the $m$-th subband, interferer $1_m$ is the strongest, $2_m$ is the second strongest, and so on. The cancellation order in each subband is different, i.e. for every subband there is a distinct successive interference cancellation order.
7.11.1.1 Example

In the following numerical example the signal parameters are the same as in [68]. Gold sequences are used with \( N = 31 \) when convolutional coding is employed, and the constraint length of the convolutional code is three. Ideal power control is assumed, i.e. \( E_{ck} = E_c \) for every user \( k \). Also, \( X(f) \) is a raised cosine function with a rolloff factor \( \alpha = 0.5 \), and all of the comparisons are based on the same transmitted data rate. The analytical results presented in [68] are averaged over 1000 realizations, and the cancellation order in the simulation of SIC on the \( m \)th subband is based upon the decreasing order of \( |Z_{km,m}(i)| \), i.e., \( |\hat{Z}_{1,m,m}(i)| \geq \cdots \geq |\hat{Z}_{(K-1),m,m}(i)| \). To simulate the time-correlated Rayleigh fading channel, the Jakes model (see Chapter 14) is used with a bit rate of 20000 bits/s and a maximum Doppler frequency of 100 Hz, while a block interleaver/deinterleaver is employed to separate adjacent data bits into a block size of \( 30 \times 30 \), which results in a delay of 45 ms. Some results are given in Figure 7.51.

7.12 MMSE DETECTION OF MULTICARRIER CDMA

The results from the previous section have already demonstrated that an MMSE detector performs better than a receiver with SIC. For this reason, in this section the minimum mean squared error (MMSE) detection of multicarrier code division multiple access (CDMA) signals is presented in more detail. The performance of two different design strategies for MMSE detection is compared. In one case, the MMSE filters are designed separately for each carrier, while in the other case the optimization of the filters is done jointly. Naturally, the joint optimization produces a better receiver, but the difference in performance is shown to be substantial. The multicarrier CDMA performance is then compared to that of a single-carrier CDMA system on a frequency selective fading channel. A mechanism to track the channel fading parameters for all the users’ signals is presented which enables joint optimization of the receiver filters in a time-varying channel. Simulation results show
that the performance of this receiver is close to the ideal theoretical results for moderate vehicle speeds. Performance begins to degrade when the normalized Doppler rate is higher than about 1 %.

The received signal on the system's reverse link on the $m$th carrier is given by

$$r_m(t) = \Re \left\{ \sum_{i=-\infty}^{+\infty} \sum_{k=1}^{K} \frac{\sqrt{P_k}}{\sqrt{M}} \gamma_{k,m}(i) d_k(i) \times c_{k,m}(t - iT_b - \tau_k) \exp(j\omega_m t) \right\} + n_m(t)$$

(7.112)

The received signal is processed with a chip-matched filter, which consists of an integrator with duration $MT_c$. The samples are stored for one bit interval, giving a column vector of length $N/M$:

$$r_m(i) = \sum_{k=1}^{K} \sqrt{\frac{P_k}{P_1}} \gamma_{k,m}(i) [d_k(i) c_{k,m} + d_k(i - 1) g_{k,m}] + n_m(i)$$

(7.113)

where $f_{k,m}$ and $g_{k,m}$ depend on the left- and right-cyclic shifts of $c_{k,m}$, the spreading code of the $k$th user on the $m$th carrier.

A block diagram of a general linear receiver is shown in Figure 7.52. Each of the $M$ received vectors is processed with a receiver filter $w_m(i)$ to form a statistic $Z_m(i) = w_m^H(i)r_m(i)$, for $m = 1, 2, \ldots, M$. Note the time dependence of the filters in the time-varying fading channel. The individual statistics are summed to form an overall decision statistic $Z(i) = \sum_{m=1}^{M} Z_m(i)$. Equivalently, we can define an overall receiver filter as

$$w(i) = \begin{bmatrix} w_1^T(i) & w_2^T(i) & \cdots & w_M^T(i) \end{bmatrix}^T$$

(7.114)

and an overall received vector as

$$r(i) = \begin{bmatrix} r_1^T(i) & r_2^T(i) & \cdots & r_M^T(i) \end{bmatrix}^T$$

(7.115)

which gives $Z(i) = w^H(i)r(i)$. 

---

**Figure 7.51** Comparisons of the analytical bounds with simulation results for the MF and SIC receivers in convolutionally coded MC CDMA, where perfect CSI is assumed.
Figure 7.52 General receiver for multicarrier CDMA.

We next consider two different design strategies for performing MMSE detection. The best performance is obtained when the filters \( w_1(i), w_2(i), \ldots, w_M(i) \) are designed jointly so as to minimize the composite mean squared error

\[
J = E[|d_1(i) - w_1^H(i)r_1(i)|^2] \tag{7.116}
\]

This gives the well-known Wiener solution \( w(i) = R^{-1}(i)p(i) \), with

\[
R(i) = E[r(i)r^H(i)] \quad \text{and} \quad p(i) = E[d_1^*(i)r(i)] \tag{7.117}
\]

representing the correlation matrix and steering vector, respectively. These can be further decomposed as

\[
R(i) = \begin{bmatrix}
R_{1,1}(i) & R_{1,2}(i) & \cdots & R_{1,M}(i) \\
R_{2,1}(i) & R_{2,2}(i) & \cdots & R_{2,M}(i) \\
\vdots & \vdots & \ddots & \vdots \\
R_{M,1}(i) & R_{M,2}(i) & \cdots & R_{M,M}(i)
\end{bmatrix} \tag{7.118}
\]

where the individual submatrices are defined as

\[
R_{m,n}(i) = E[r_m(i)r_n^H(i)] \tag{7.119}
\]

and

\[
p(i) = \begin{bmatrix} p_1^T(i), p_2^T(i), \ldots, p_M^T(i) \end{bmatrix}^T \tag{7.120}
\]

with

\[
p_m(i) = E[d_1^*(i)r_m(i)] \tag{7.121}
\]

An alternative suboptimal approach is to design the \( M \) filters separately by choosing each of the filters \( w_1(i), w_2(i), \ldots, w_M(i) \) to minimize the individual mean squared error quantities \( J = E[|d_1(i) - w_m^H(i)r_m(i)|^2] \), which leads to \( w_m(i) = R_m^{-1}(i)p_m(i) \). Together with the previous notation, an overall
Figure 7.53 Probability of error versus number of users for the multicarrier CDMA system, with $E_b/N_0 = 17$ dB, composite processing gain of 32 chips/bit and $M$ carriers. The solid lines are the Wiener solutions when all users’ fading processes are tracked, the dashed lines are the Wiener solutions when only the desired user’s fading processes are tracked.

A filter using this design strategy can then be written as $w(i) = R_p^{-1}(i)p(i)$ where

$$R_p(i) = \begin{bmatrix} R_{1,1}(i) & 0 & \cdots & 0 \\ 0 & R_{2,2}(i) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & R_{M,M}(i) \end{bmatrix}$$

An example of performance results is given in Figure 7.53. One can see that the joint detector demonstrates better performance.

Before proceeding to present a tracking algorithm, it is worthwhile to compare the performance of a multicarrier CDMA system to that of a single-carrier system which realizes diversity inherently by operating on a frequency selective fading channel. In order to get a fair comparison to the multicarrier case, the system will be assumed to employ spreading waveforms of length $N$ chips/bit, where $N$ in the multicarrier case is the composite processing gain. The received signal will consist of $M$ resolvable components, delayed with respect to one another by a sufficient number of chips. Results for this case are shown for comparison to the multicarrier case in Figure 7.54.

### 7.12.1 Tracking the fading processes

The presentation in this section is based on [69–71]. The received vector on the $m$th carrier, from Equation (7.113), may be written as

$$r_m(i) = (F_mD(i) + G_mD(i - 1))PT_m(i) + n_m(i)$$

(7.123)
Figure 7.54 Probability of error versus number of users for the multicarrier CDMA system (solid curves) and the single-carrier system on a frequency selective fading channel (dashed curves). For each $E_b/N_0 = 17$ dB, the composite processing gain is 32 chips/bit, and $M$ is the number of carriers used (for multicarrier) or the number of resolvable paths present (for frequency selective). The Wiener solution is formed, with all users’ fading processes tracked perfectly.

where the matrices in this expression are defined as

$$\begin{align*}
F_m &= \begin{bmatrix} f_{1,m}, f_{2,m}, \ldots, f_{K,m} \end{bmatrix} \\
G_m &= \begin{bmatrix} g_{1,m}, g_{2,m}, \ldots, g_{K,m} \end{bmatrix} \\
D(i) &= \begin{bmatrix} d_1(i) \\ d_2(i) \\ \vdots \\ d_K(i) \end{bmatrix} \\
P(i) &= \begin{bmatrix} 1 \\ \sqrt{P_2/P_1} \\ \ddots \\ \sqrt{P_K/P_1} \end{bmatrix}
\end{align*}$$

(7.124)

and the column vector of fading coefficients on the $m$th carrier, which is to be tracked, is

$$\Gamma_m(i) = [\gamma_{1,m}(i), \gamma_{2,m}(i), \ldots, \gamma_{K,m}(i)]^T$$

(7.125)

Also, in Equation (7.123), recall that $\mathbf{n}_m(i)$ is a column vector of independent, complex Gaussian noise samples, with the real and imaginary parts independent from each other and each with variance
\[ \sigma^2 = N/(2E_b/N_0). \] Then, Equation (7.123) can be rewritten as
\[
r_m(i) = \Lambda_m(i) \Gamma_m(i) + n_m(i)
\]
(7.126)
with
\[
\Lambda_m(i) = (F_mD(i) + G_mD(i-1))P
\]
(7.127)

For estimation purposes, we next assume that the fading processes are essentially constant over a window of \( L \) bit intervals, which gives
\[
r_{m(L)}^T(i) = \Lambda_{m(L)}^T(i) \Gamma_m(i) + n_{m(L)}^T(i)
\]
(7.128)
where matrices from \( L \) bit intervals have been concatenated to form
\[
\Lambda_{m(L)}^T(i) = \begin{bmatrix} \Lambda_{m(L)}^T(i - (L - 1)), & \cdots, & \Lambda_{m(L)}^T(i) \end{bmatrix}
\]
(7.129)
\[
n_{m(L)}^T(i) = \begin{bmatrix} n_{m(L)}^T(i - (L - 1)), & \cdots, & n_{m(L)}^T(i) \end{bmatrix}
\]

We now assume that the receiver has knowledge of the data bits of all of the users. This would be reasonable either when the receiver is in training mode, or when decision feedback is used. In this case, the maximum likelihood estimate of the vector \( \Gamma_m(i) \) minimizes the quadratic cost function
\[
C(\Gamma_m(i)) = \| r_{m(L)}^T(i) - \Lambda_{m(L)}^T(i) \Gamma_m(i) \|^2
\]

This form of the matrices suggests recursive estimates using exponentially weighted windows
\[
Q_m(i) = \lambda Q_m(i-1) + \Lambda_{m(L)}^T(i) \Lambda_m(i)
\]
(7.132)
\[
S_m(i) = \lambda S_m(i-1) + \Lambda_{m(L)}^T(i) n_m(i)
\]
where \( 0 < \lambda < 1 \) is the forgetting factor. Once the fading has been estimated according to this procedure, an estimate of the true Wiener solution of the tap weights may be formed.

To obtain some insight into the operation of this channel estimator, consider a single bit estimator, that is, let \( \lambda = 0 \). We have
\[
\Gamma_{m,ML}(i) = \left[ \Lambda_{m(L)}^T(i) \Lambda_m(i) \right]^{-1} \left[ \Lambda_{m(L)}^T(i) r_{m(L)}^T(i) \right]
\]
(7.133)

Thus, the estimate of \( \Gamma_m(i) \) is equal to the true value plus a term due only to the thermal noise, and independent of the multiaccess interference.
Figure 7.55 Block diagram of the estimator for fading processes using a single bit observation window.

Furthermore, if the system were synchronous, then the matrix $G_m$ would be a zero matrix, and the estimate of $\Gamma_m(i)$ could be written as

$$\Gamma_{m,ML}(i) = D^{-1}(i)[(F_m P)^T (F_m P)]^{-1}[(F_m P)^T r_m(i)] (7.134)$$

The estimator could then be visualized as in Figure 7.55. The received signal is processed first with a matched filter bank. The MAI is then removed with a decorrelator. Finally, the data are removed, leaving an unbiased estimate of $\Gamma_m(i)$.

One should keep in mind that the matrix $(F_m P)^T (F_m P)$ will be invertible only if the columns of $F_m$ are linearly independent. This condition will be violated as the number of users surpasses $N/M$ and the decorrelator will not exist. However, as the observation window was increased, which would obviously be done in order to get good estimates of the fading processes, the existence of the decorrelator would be almost certain. A similar interpretation of the estimator would still apply, that is, matched filter/decorrelator/data removal.

The performance of this algorithm will be illustrated first for a multicarrier CDMA system with the same parameters as in [69]. A user with a data rate of 10 kHz, a carrier frequency of 900 MHz, a processing gain of 32 chips/bit, and an $E_b/N_0$ of 17 dB was simulated. With 15 users present, results for the average probability of bit error are shown in Figure 7.56(a) as a function of the vehicle speed, which was varied from 20 mph up to 200 mph. Single-carrier, two-carrier and four-carrier systems were considered. Results for the ideal Wiener solution are shown for comparison. It is seen that the performance is very close to ideal for vehicle speeds below about 80 mph, or a normalized Doppler frequency under 1%.

In Figure 7.56(b), the identical system was simulated, this time with a fixed vehicle speed of 80 mph, and with the number of users varying between 1 and 30. Again, the results are seen to be close to ideal at this speed.

As mentioned previously, it is straightforward to extend this tracking algorithm to the frequency selective case. Without going into such details, in Figure 7.56(c), an identical system to that used to generate the results shown in Figure 7.56(a) for the multicarrier case was applied to a single-carrier system operating on a frequency selective channel with $M$ resolvable paths. The average probability of bit error is shown as a function of the vehicle speed for 15 users. Again, the tracking algorithm gives results which are very close to ideal for vehicle speeds below about 80 mph, or a normalized Doppler frequency under 1%. In Figure 7.56(d), the vehicle speed was fixed at 80 mph, and the number of users was varied between 1 and 30, the results are again close to ideal at this speed.

Additional discussions on MMSE detectors for MC CDMA systems can be found in [69–81].

### 7.13 APPROXIMATION OF OPTIMUM MULTIUSER RECEIVER FOR SPACE–TIME CODED MULTICARRIER CDMA SYSTEMS

In this section we extend the discussion on multiuser detection for MC CDMA to the case when space–time coding is included. The presentation is based on [82–84]. The structure of the MC CDMA modulator of the $k$th user is illustrated in Figure 7.57. BPSK symbols of the $k$th user are first serial-to-parallel converted, by grouping every $P$ symbols into a vector

$$s^k = [s^k[1], \ldots, s^k[P]]^T (7.135)$$
Figure 7.56 (a) Probability of error versus vehicle speed for multicarrier CDMA system with $E_b/N_0 = 17$ dB, composite processing gain of 32 chips/bit, 15 asynchronous users, bit rate of 10 000 bits/s, and carrier frequency of 900 MHz. $M$ is the number of carriers used. Curves show the performance of the tracking algorithm, and solid straight lines show the Wiener solutions for 15 users. (b) Probability of error versus number of users for multicarrier CDMA system with $E_b/N_0 = 17$ dB, composite processing gain of 32 chips/bit, vehicle speed of 80 mph, bit rate of 10 000 bits/s, and carrier frequency of 900 MHz. $M$ is the number of carriers used. Dashed lines show the performance of the tracking algorithm, and solid lines show the Wiener solutions. (c) Probability of error versus vehicle speed for single-carrier CDMA system with $M$ resolvable paths. The system has 15 users, $E_b/N_0 = 17$ dB, composite processing gain of 32 chips/bit, bit rate of 10 000 bits/s, and carrier frequency of 900 MHz. Curves show the performance of the tracking algorithm, and solid straight lines show the Wiener solutions for 15 users. (d) Probability of error versus number of users for single-carrier system with $M$ resolvable paths. System has $E_b/N_0 = 17$ dB, composite processing gain of 32 chips/bit, vehicle speed of 80 mph, bit rate of 10 000 bits/s, and carrier frequency of 900 MHz. Dashed lines show the performance of the tracking algorithm, and solid lines show the Wiener solutions.
Figure 7.56 (cont.)
In the next step, symbol \( s^k[p] \) in \( s^k \) is spread by a spreading sequence \( c^k[p], \ p = 1, \ldots, P \). The sequences corresponding to all \( P \) binary phase shift keying (BPSK) symbols are represented by an \( N \)-vector as

\[
\mathbf{c}^k = [c^k[1], \ldots, c^k[P]]^T
\]  

(7.136)

and the spread signals are represented by a componentwise product of the above vectors

\[
\mathbf{s}^k \mathbf{c}^k = [s^k[1]c^k[1]^T, \ldots, s^k[P]c^k[P]^T]^T
\]  

(7.137)

In order to avoid the strong correlation among the \( G \) subcarriers occupied by a particular symbol, all \( N \) spread chips in Equation (7.137) are interleaved before they are transmitted from the \( N \) subcarriers. The interleaving function is denoted by \( \mathbf{T} \) in Figure 7.57; for simplicity, the interleaver is assumed to be the same for all \( K \) users.

The Alamouti space–time block codes (STBC), discussed in Chapter 4, are employed to further increase the system capacity. In Chapter 4, the STBC was used to transmit scalar symbols. In this section, we extend it to the vector form and apply it in MC CDMA systems. In vector form, the simplest \( (2 \times 2) \) STBC discussed in Chapter 4, as illustrated in Figure 7.58, takes two time slots to
transmit two symbol vectors $s_{1k}^k$, $s_{2k}^k$ [cf. Equation (7.135)]. At the first time slot, signals $T(s_{1k}^k c_{1k})$ are transmitted from the first antenna on $N$ subcarriers and $T(s_{2k}^k c_{2k})$ is transmitted from the second antenna. At the second time slot, $T(-s_{2k}^k c_{1k})$ is transmitted from the first antenna and $T(s_{1k}^k c_{2k})$ is transmitted from the second antenna. From Figure 7.58, we can see that different spreading sequences $c_{1k}$ and $c_{2k}$ are assigned to different transmitter antennas; this structure is shown to be an efficient way to resolve the so-called ‘antenna-ambiguity’ in blind algorithms, as will be discussed. The structure of the transmitter for the $k$th user is given in Figure 7.59.

### 7.13.1 Frequency selective fading channels

The system with one receiver antenna is considered and the extension to multiple receiver antennas is straightforward. The time domain channel impulse response of the $k$th user between the $j$th transmitter antenna and the receiver antenna will be modeled as

$$h_j^k(\tau) = \sum_{l=0}^{L-1} \alpha_j^k(l) \delta \left( \tau - \frac{l}{\Delta_f} \right)$$

(7.138)

where $\delta(\cdot)$ is the Kronecker delta function; $L = \lceil \tau_m \Delta_f + 1 \rceil$, with $\tau_m$ being the maximum multipath spread of all $K$ users (note that we have assumed the synchronous transmission of all $k$ users) and $\Delta_f$ being the whole bandwidth of multicarrier systems; $\alpha_j^k(l)$ is the complex amplitude of the $l$th tap associated with the $j$th transmitter antenna of the $k$th user, whose relative delay is $l/\Delta_f$.

For MC CDMA systems with proper cyclic extensions and sample timing, with tolerable leakage, the channel frequency response of the $k$th user at its $j$th transmitter antenna and at the $n$th subcarrier
can be expressed as

\[ H_{jk}^i[n] \triangleq H_{jk}^i(n \Delta_f) \]

\[ = \sum_{l=0}^{L-1} \alpha_{jk}^i(l) \exp \left( -j \frac{2\pi nl}{N} \right) \]

\[ = w_{jk}^H(n) h_{jk}^i \]

where

\[ h_{jk}^i \triangleq \left[ \alpha_{jk}^i(0), \alpha_{jk}^i(1), \ldots, \alpha_{jk}^i(L - 1) \right]^T \]

contains the time response of all \( L \) taps; and

\[ w_f(n) \triangleq \left[ 1, \exp \left( -j \frac{2\pi n}{N} \right), \ldots, \exp \left( -j \frac{2\pi (N - 1)}{N} \right) \right]^H \]

contains the corresponding discrete Fourier transform (DFT) coefficients.

### 7.13.2 Receiver signal model of STBC MC CDMA systems

The transmitted signals of all \( K \) users propagate through their respective frequency selective fading channels and finally reach the receiver antenna. We assume that the fading processes associated with different transmitter–receiver antenna pairs are uncorrelated. At the receiver, after matched filtering, the discrete Fourier transform (DFT) is applied to the received chip rate sampled discrete time signals. Consider all the DFT-ed signals in one signal frame, which spans \( M \) STBC slots, or equivalently, \( 2M \) time slots [recall that each STBC slot consists of two neighboring time slots (see Figure 7.58).] Using Equation (7.139) and assuming that the fading processes are time invariant within one signal frame, the received signal model can be represented as

\[ y_i = \sum_{k=1}^{K} \left[ \begin{array}{cc} S_{k,1}^i & C_{1}^i \\ -S_{k,2}^i & C_{2}^i \end{array} \right] \left[ \begin{array}{c} T W_f \\ 0 \end{array} \right] \left[ \begin{array}{c} h_{k,1}^i \\ h_{k,2}^i \end{array} \right] + v_i \]

\[ = \sum_{i=0}^{M-1} X_i^t W h^i + v_i, \quad i = 0, \ldots, M - 1 \]

(7.142)

with

\[ C_j^i \triangleq \text{diag} \left[ C_j^i \right]_{N \times N} \]

\[ C_j^i \triangleq \left[ c_j^i[1], \ldots, c_j^i[P] \right]^T, \quad j = 1, 2 \]

\[ S_{k,j}^i \triangleq \text{diag} \left[ s_{k,j}^i \otimes I_G \right]_{N \times N}, \quad j = 1, 2 \]

\[ W_t \triangleq [w_f(0), w_f(1), \ldots, w_f(N - 1)]_{N \times L}^H \]

where \( \otimes \) is the Kronecker matrix product; \( s_{k,j}^i, j = 1, 2 \) are the symbol vectors input to the STBC encoder at the \( i \)th STBC slot; \( c_j^i \) is the spreading sequence assigned to the \( j \)th transmitter antenna of the \( k \)th user; \( T \) is an \((N \times N)\) permutation matrix, which acts as an interleaver mapping the \( N \) chips to their assigned subcarriers; \( y_i \) is the received signal during the \( i \)th STBC slot; \( v_i \) is circularly symmetric complex Gaussian ambient noise, with covariance matrix \( \sigma^2 I_{2N} \).

Note that Equation (7.142) can be used to describe both the slow-fading (when \( M \) is large) and the fast-fading (when \( M \) is small, e.g., \( M = 1 \)) cases. In this section, the fading channels are assumed...
to be static during two neighboring time slots (i.e. one STBC slot), this is the only limitation of the signal model in Equation (7.142). Due to the orthogonality property of the STBC, i.e. \( X_k^H X_k = 2I_{2N} \), the orthogonality property of the OFDM multicarrier modulation, i.e. \( W_f^H W_f = N \cdot I_L \), the fact that the permutation matrix satisfies \( T^T T = I_N \) and the definitions given in Equation (7.142), we have

\[
W^H X^H f X^f W = \begin{bmatrix} W^H f^T & 0 \\ 0 & W^H f^T \end{bmatrix} \begin{bmatrix} 2I_N & 0 \\ 0 & 2I_N \end{bmatrix} \times \begin{bmatrix} T W_f & 0 \\ 0 & T W_f \end{bmatrix} = 2N \cdot I_{2L} \tag{7.143}
\]

As we know from orthogonal design, discussed in Chapter 4, the structure of Equation (7.143) can be exploited to reduce the computational complexity of the optimal receiver for the STBC MC CDMA system. Note that the interleaver function \( T \) is the same for all \( K \) users; hence, the signal model Equation (7.142) can be written in an alternative form, which decouples the signal components corresponding to different symbols

\[
y_i[p] = \sum_{k=1}^{K} X^i_k[p] W[p] h^i + v_i[p], \quad i = 0, \ldots, M - 1
\]

\[
p = 1, \ldots, P \tag{7.144}
\]

where \( X^i_k[p] \) is a \((2G \times 2G)\) submatrix decimated from \( X^i_k \), which contains all the rows and columns related to the symbol \( s^i_{1,k}[p] \) and \( s^i_{2,k}[p] \); \( y_i[p] \), \( W[p] \) and \( v_i[p] \) are then the corresponding decimations from \( y_i \), \( W \) and \( v_i \). Equation (7.144) will be used later in deriving the conditional posterior distributions of the unknown symbols [see Equation (7.158)].

### 7.13.3 Blind approach

Since the channel state information is unknown to the receiver, there are two types of ambiguity inherent in the design of blind receivers for the STBC MC CDMA system: phase ambiguity and antenna ambiguity.

To resolve the phase ambiguity, differential encoding is employed before the STBC encoding. For each signal frame, a block of BPSK bits \( b^k = \{b^k[1], b^k[2], \ldots, b^k[2MP-1]\} \) is input to the differential encoder, and the output \( d^k = \{d^k[1], \ldots, d^k[2MP]\} \) is given by

\[
\begin{aligned}
    d^k[1] &= 1 \\
    d^k[n] &= d^k[n-1]b^k[n-1], \quad n = 2, \ldots, 2MP
\end{aligned} \tag{7.145}
\]

These differentially encoded bits \( \{d^k[n]\} \) are the same set of bits \( \{s^i_{1,j}[p]\}_{i,i',j} \) [defined by Equation (7.142)] input to the STBC encoder, where they are related as \( s^i_{1,j}[p] = d^k[n] = s^i_{1,j}[p]_{i,n=2(i+j)p+p} \). Henceforth, the index \( n \) of \( d^k[n] \) is understood as an implicit function of the index \((i,j,p)\) of \( s^i_{1,j}[p] \).

One possible approach to resolving the antenna ambiguity is to employ the differential space–time modulation as discussed in Chapter 4; however, in that case, the signal constellation will be changed. Fortunately, in the STBC MC CDMA system, the antenna ambiguity can be resolved by using different spreading sequences on different transmitter antennas. Note that the usage of orthogonal sequences will result in a maximum 50 % system loading (defined as \( K/G \)) in the \((2 \times 2)\) STBC MC CDMA system. The problem with this is that the channel frequency selectivity destroys the orthogonality of the sequences. For this reason, in this section, random sequences are assumed and a sufficient number of these sequences, such that they impose no constraint on the system loading. When the system employs the outer channel code, it is possible to use the same spreading sequence at different antennas of the same user. In this case, the antenna ambiguity can be resolved by exploiting the coding structure. For example, for the system considered here, due to the antenna ambiguity, we have two possible code bit sequences at the output of the multiuser detector. We can send both of them to the channel decoder and count the number of bit corrections. The correct code bit sequence will have only a few bit errors, whereas the incorrect one will have many errors.
7.13.4 Bayesian optimal blind receiver

From here on the following matrix notation will be used

\[ Y \triangleq \{y_i\}_{i=0}^{M-1}, \quad H \triangleq \{h^k\}_{k=1}^K, \quad D \triangleq \{d^k\}_{k=1}^K; \quad B \triangleq \{b^k\}_{k=1}^K \]

The optimal blind receiver estimates the a posteriori probabilities (APP) of the multiuser data bits

\[ P[b^k[n] = +1|Y], \quad n = 1, 2, \ldots, MP - 1, \quad k = 1, \ldots, K \quad (7.146) \]

based on the received signals \( Y \), the signal structure Equation (7.142), the spreading sequences of all users, and the prior information of \( B \), without knowing the channel response \( H \) and the noise variance \( \sigma^2 \).

So, the unknown parameters have to be averaged out from the a posteriori function. The Bayesian solution to Equation (7.146) is given by

\[ P[b^k[n] = +1|Y] = \sum_{B:b^k[n]=+1} \int p[Y|H, \sigma^2, B] p[H|\sigma^2] p[B] \, dH \, d\sigma^2 \quad (7.147) \]

where \( P[Y|H, \sigma^2, B] \) is a Gaussian density function [see Equation (7.142)]; \( p[H], p[\sigma^2] \) and \( p[B] \) are prior distributions of the independent and unknown quantities \( H, \sigma^2 \) and \( B \) respectively. Clearly the computation in Equation (7.147) involves a very high-dimensional integral which is certainly infeasible for any practical implementations. Thus, the Gibbs sampler, a Monte Carlo method, is used to calculate the a posteriori probabilities of the unknown symbols.

7.13.5 Blind Bayesian Monte Carlo multiuser receiver approximation

In this section, we consider the problem of computing the a posteriori bit probabilities in Equation (7.146). The problem is solved under a Bayesian framework, by treating the unknown quantities as realizations of random variables with some prior distributions. The Gibbs sampler [85–89] is then employed to compute the Bayesian estimates.

7.13.6 Gibbs sampler

The Gibbs sampler [85–89] is a Markov chain Monte Carlo (MCMC) procedure for numerical Bayesian computation. Let \( \theta = [\theta_1, \ldots, \theta_d]^T \) be a vector of unknown parameters. Let \( Y \) be the observed data. To generate random samples from the joint posterior distribution \( p[\theta|Y] \), given the samples at the \((j-1)\)th iteration, \( \theta^{(j-1)} = [\theta_1^{(j-1)}, \ldots, \theta_d^{(j-1)}]^T \) at the \( j \)th iteration, the Gibbs algorithm iterates as follows to obtain samples \( \theta^{(j)} = [\theta_1^{(j)}, \ldots, \theta_d^{(j)}]^T \).

For \( i = 1, \ldots, d \), draw \( \theta_i^{(j)} \) from the conditional distribution

\[ p[\theta_i|\theta_1^{(j)}, \ldots, \theta_{i-1}^{(j)}, \theta_{i+1}^{(j)}, \ldots, \theta_d^{(j)}, Y] \]

It is known that under regularity conditions [86–89]:

1. The distribution of \( \theta_i^{(j)} \) converges geometrically to \( p[\theta|Y] \), as \( j \to \infty \);

2. \( (1/J) \sum_{j=1}^J f(\theta^{(j)}) \xrightarrow{a.s.} \int f(\theta)p[\theta|Y] \, d\theta \), as \( J \to \infty \), for any integrable function \( f \).

Hence, the marginal a posteriori distribution of any parameter \( \theta_i \) can be computed easily from the samples drawn by the Gibbs sampler.
7.13.7 Prior distributions

For simplicity, we choose the sampling space, the set of unknown parameters sampled by the Gibbs sampler, to be \{D, H, \sigma^2\}, which are assumed to be independent of each other. Next, we specify their prior distributions \(p[H], p[\sigma^2]\) and \(p[D]\).

1. For the unknown channel \(h^k\), a complex Gaussian prior distribution is assumed
   \[
p[h^k] \sim N_c(h_{k,0}, \Sigma_{k0})
   \]
   Note that large value of \(\Sigma_{k0}\) corresponds to less informative prior distributions.

2. For the noise variance \(\sigma^2\), an inverse chi squared prior distribution is assumed
   \[
p[\sigma^2] \sim \chi^{-2}(2\nu_0, \lambda_0)
   \]
   a small value of \(2\nu_0\) corresponds to the less informative prior distributions.

3. The data bit sequence \(d^k\) is a Markov chain, encoded from \(b^k\). Its prior distribution can be expressed as
   \[
p[d^k] = p(d^{k}[1])p(d^{k}[2]|d^{k}[1]) \cdots p(d^{k}[2PM]|d^{k}[2PM−1])
   \]
   \[
   = p(d^{k}[1])p(b^{k}[1] = d^{k}[2]|d^{k}[1])
   \cdots p(b^{k}[2PM−1] = d^{k}[2PM]|d^{k}[2PM−1])
   \]
   \[
   = \frac{1}{2} \prod_{n=2}^{PM} \exp(\rho^k[n−1]|d^k[n−1]|d^k[n]) \over 1 + \exp(\rho^k[n−1]|d^k[n−1]|d^k[n])
   \]
   where \(\rho^k[n]\) denotes the \(a\ priori\) log likelihood ratio (LLR) of \(b^k[n]\),
   \[
   \rho^k[n] \Delta \log \frac{P(b^k[n] = +1)}{P(b^k[n] = −1)}
   \]
   Note that in Equation (7.150), we set \(p(d^{k}[1]) = 1/2\) to account for the phase ambiguity in \(d^{k}[1]\).

7.13.8 Conditional posterior distributions

The following conditional posterior distributions are required by the Bayesian multiuser detector. The derivations can be found in Appendix 7.1 [83].

1. The conditional distribution of the \(k\)th user’s channel response \(h^k\) given \(\sigma^2, D, H^k, \text{ and Y}\) is
   \[
p[h^k|D, \sigma^2, H^k, Y] \sim N_c(h_{k,\Delta}, \Sigma_{k\Delta})
   \]
   where \(H^k \triangleq H \setminus h^k\) with
   \[
   \Sigma_{k\Delta} \triangleq \Sigma_{k0} + \frac{1}{\sigma^2} \sum_{i=0}^{M-1} W_i h_i^H X_i W
   \]
   \[
   = \Sigma_{k0} + \frac{2MN}{\sigma^2} I_{2L}
   \]
   and
   \[
   h_{k,\Delta} \triangleq \Sigma_{k\Delta}^{-1} \Sigma_{k0}^{-1} h_{k,0} + \frac{1}{\sigma^2} \sum_{i=0}^{M-1} W_i h_i^H X_i \left( y_i - \sum_{l \neq k} X_l W_l^H h^l \right)
   \]
   Equation (7.153) follows from the orthogonality property in Equation (7.143).
2. The conditional distribution of the noise variance $\sigma^2$ given $H$, $D$, and $Y$ is given by

$$p[\sigma^2 | H, D, Y] \sim \chi^{-2}\left(2[\nu_0 + MN], \frac{\nu_0 \lambda_0 + s^2}{\nu_0 + 2MN}\right)$$ (7.155)

with

$$s^2 \triangleq \sum_{i=0}^{M-1} \left| y_i - \sum_{k=1}^{K} X_k^i W^i h^i \right|^2$$ (7.156)

3. The conditional distribution of the data bit $d^k[n]$, given $H$, $\sigma^2$, $D^k$ and $Y$ can be obtained from

$$\frac{P\left[d^k[n] = +1 \mid H, \sigma^2, D^k, Y\right]}{P\left[d^k[n] = -1 \mid H, \sigma^2, D^k, Y\right]} = \exp\left[d^k[n + 1] \rho^k[n] + d^k[n - 1] \rho^k[n - 1] - \frac{\Delta s^2}{\sigma^2}\right]$$ (7.157)

where $D^k \triangleq D \setminus d^k[n]$ and

$$\Delta s^2 \triangleq \left| y[p] - \sum_i X_i^p W[p] h_i^p \right|^2_{i_i^p[p]=+1} - \left| y[p] - \sum_i X_i^p W[p] h_i^p \right|^2_{i_i^p[p]=-1}$$ (7.158)

On the right-hand side of Equation (7.157), the first two items correspond to the prior information of the differentially encoded bits $D$, through which the prior information of data bits $B$ is incorporated.

### 7.13.9 Gibbs multiuser detection

Given the initial values of the unknown quantities $(H^{(0)}, \sigma^{2(0)}, D^{(0)})$ drawn from their prior distributions (Equations 7.148–7.150), at the $j$th iteration, the Gibbs multiuser detector operates as follows.

1. For $k = 1, \ldots, K$, draw $h^{k(j)}$ from $p[h^k | H^{(j-1)}, \sigma^{2(j-1)}, D^{(j-1)}, Y]$, given by Equation (7.152), with $H^{(j-1)} \triangleq (h^{1(j)}, \ldots, h^{k-1(j)}, h^{k+1(j)}, \ldots, h^{K(j-1)});$.
2. Draw $\sigma^{2(j)}$ from $p[\sigma^2 | H^{(j)}, D^{(j-1)}, Y]$ given by Equation (7.155);
3. For $n = 1, \ldots, 2PM$, and for $k = 1, \ldots, K$, draw $d^{k(j)[n]}$ from $P[d^k[n] | H^{(j)}, \sigma^{2(j)}, D^{(j-1)}, Y]$ given by Equation (7.157), where $D^{(j-1)} \triangleq [d^{1(j)[1]}, \ldots, d^{K(j)[2PM]}, \ldots, d^{k(j)[n - 1]}, d^{k(j)[n + 1]}, \ldots, d^{K(j)[2PM]}].$

To ensure convergence, the Gibbs iteration is usually carried out for $(J_0 + J)$ iterations. The initial $J_0$ iterations represent the burn-in period and only the samples from the last $J$ iterations are used to calculate the Bayesian interference. In particular, the posterior distribution of the multiuser data bits $b^k[n]$ can be obtained by

$$P[b^k[n] = +1 \mid Y] \propto \frac{1}{J} \sum_{j=J_0+1}^{J_0+J} \delta_{kn}^{(j)}$$ (7.159)

where $\delta_{kn}^{(j)}$ is an indicator such that

$$\delta_{kn}^{(j)} = \begin{cases} 1, & \text{if } d^{k(j)[n]} d^{k(j)[n - 1]} = +1 \\ 0, & \text{if } d^{k(j)[n]} d^{k(j)[n - 1]} = -1 \end{cases}$$ (7.160)
7.13.10 Sampling space of data

As we are interested in computing the posterior probabilities of the multiuser data bits, direct sampling can be done on the data bits $B$. Note that the conditional posterior distribution of $b^k[n]$, given $H$, $\sigma^2$, $B^k$, and $Y$, involves a large number of received signals, i.e. $\{y, \ldots, y_{u-1}\}$. This long memory in the receiver signal processing will increase the computational complexity and decrease the convergence speed of the Gibbs procedure. To avoid these disadvantages, the Gibbs procedure described in this section samples the differentially encoded bits $D$, and outputs a sampling sequence $\{d^{(j)}[n]\}$. It is shown in Equation (7.159) that the marginal posterior probability of $b^k[n]$ can be computed easily from the output samples $\{d^{(j)}[n]\}$.

7.13.11 The orthogonality property

The dominant computations involved in the Gibbs sampler are Equations (7.153–7.154) By exploiting the orthogonality property, Equation (7.143), in STBC MC CDMA systems, the matrix $\Sigma^{-1}_{k^*}$ in Equation (7.153) is simply a constant matrix. In addition to this, with no matrix inversion involved in computing $h^k$, the numerical stability is also improved.

7.13.12 Blind turbo multiuser receiver

In this section, we consider employing iterative multiuser detection and decoding to improve the performance of the Bayesian multiuser receiver in a coded STBC MC CDMA system. Because it utilizes the a priori bit probabilities, and it produces the a posteriori bit probabilities, the Bayesian multiuser detector is well suited for iterative processing, which allows the multiuser detector to refine its processing based on the information from the decoding stage and vice versa. The $k$th user’s transmitter structure is shown in Figure 7.60, with block of information bits $\{a^k[l]\}$ encoded using some channel code (e.g. block code, convolutional code or turbo code). A code bit interleaver is used to reduce the influence of error bursts at the input of the channel decoder. The interleaved code bits are then mapped to BPSK symbols $\{b^k[n]\}$. These BPSK symbols are differentially encoded to yield the symbol stream $\{d^k[n]\}$, which is then serial-to-parallel converted and reorganized in to a vector form as $\{s^k_{i,j}\}$ to feed into the STBC encoder followed by the MC CDMA modulator, and finally transmitted from two antennas.

An iterative (turbo) receiver structure is shown in Figure 7.61. It consists of two stages: the Bayesian multiuser detector developed in the previous sections, followed by a soft-input soft-output channel decoder. The two stages are separated by a deinterleaver and an interleaver. Assume that $\{b^k[n]\}$ is mapped into $\{b^k[\pi(n)]\}$ after deinterleaving.

In the first stage, the blind Bayesian multiuser detector incorporates the a priori information $\{\lambda_2^k(b^k[n])\}$, which is computed by the channel decoder in the previous iteration. At the first iteration, it is assumed that all code bits are equally likely. At the output of the blind Bayesian multiuser detector, the a posteriori LLR is given by

$$A_1(b^k[n]) \triangleq \log \frac{P[b^k[n] = +1 | Y]}{P[b^k[n] = -1 | Y]} \quad (7.161)$$

Figure 7.60 Transmitter structure of the $k$th user in an STBC MC CDMA system employing outer channel code, where $\Pi$ represents an interleaver.
According to the ‘turbo principle’ described in Chapter 2, the *a priori* information \( \lambda_2^p(b^k[n]) \) should be subtracted from the *a posteriori* LLR \( A_1(b^k[n]) \) to obtain the extrinsic information to deliver to the channel decoders. However, the posterior distribution delivered by the blind Bayesian multiuser detector is a quantized value instead of the true value, due to the finite number of samples. Therefore, to ensure numerical stability, the posterior LLR \( A_1(b^k[n]) \) is regarded as the approximated extrinsic information, deinterleaved and fed back to the channel decoder of the \( k \)th user.

The soft-input soft-output channel decoder, using the MAP decoding algorithm (Appendix 7.1), computes the *a posteriori* LLR of each code bit of the \( k \)th user

\[
A_2(b^k[\pi(n)]) = \log \frac{P[b^k[n] = +1 | \{ A_1^p(b^k[\pi(i)]) \}_t^{PM-1}]}{P[b^k[n] = -1 | \{ A_1^p(b^k[\pi(i)]) \}_t^{PM-1}}
\]

\[
= \lambda_2^p(b^k[\pi(n)]) + A_1^p(b^k[\pi(n)])
\]

(7.162)

It is seen from Equation (7.162) that the output of the MAP decoder is the sum of the prior information \( A_1^p(b^k[\pi(n)]) \), and the extrinsic information \( \lambda_2^p(b^k[\pi(n)]) \) delivered by the channel decoder. After interleaving, the extrinsic information delivered by the channel decoder \( \{ \lambda_2^p(b^k[n]) \}_{t=1}^{PM-1} \) is then fed back to the blind Bayesian multiuser detector as the refined prior information \( p^k[n] \) [see Equation (7.151)] for the next iteration.

### 7.13.13 Decoder-assisted convergence assessment

Although it is desirable to have the Gibbs sampler reach convergence within the burn-in period (\( J_0 \) iterations), this may not always be the case. Hence, we need some mechanism to detect the convergence. In the coded system considered here, the blind multiuser detector is followed by a bank of channel decoders; we can assess convergence by monitoring the number of bit corrections made by the channel decoders [90]. The number of corrections is determined by comparing the signs of the code bit LLR at the input and output of the MAP channel decoder. If this number exceeds some predetermined threshold, then we decide convergence is not achieved, in which case, the Gibbs multiuser detector will be applied again to the same data block.

### 7.13.14 Performance example

In this section, we present some computer simulation results to illustrate the performance of the Bayesian multiuser receivers in the STBC MC CDMA system, where there are two transmitter
antennas, one receiver antenna, \( N \) subcarriers and \( K \) users. The signal parameters are the same as in [83]. All spreading sequences used in simulations are randomly and independently generated for each transmitter antenna of each user. The frequency selective fading channels are assumed to be uncorrelated and have the same statistics for different transmitter–receiver antenna pairs. For simplicity, all \( L \) taps of a particular fading channel are assumed to be of equal power and normalized such that \( \sum_{j=1}^{2} ||h_j||^2 = 1 \), and have delays \( \tau_l = (l/\Delta f), l = 0, 1, \ldots, L - 1 \); and all \( K \) users in the system are assumed to have equal transmission power. Such a system setup is also the worst case scenario from the interference mitigation point of view (Chapter 5). For STBC MC CDMA systems employing outer channel codes, a four-state, rate-1/2 convolutional code with generator (5,7) in octal notation is chosen for all users. For each block of received signals, \( J_0 + J = 100 \) samples are drawn by the Gibbs sampler, with the first \( J_0 = 50 \) samples discarded. As discussed in the previous section, at the end of the 100 Gibbs iterations, the convergence of the Gibbs sampler is tested. In very few cases, when the Gibbs sampler is not convergent, it is restarted for another round of 100 Gibbs iterations.

The performance is demonstrated in two forms: one (Figure 7.62) is in terms of the bit error rate (BER) and OFDM word error rate (WER) versus the number of users at a particular signal to noise ratio (SNR), where \( \text{SNR} = (1/\sigma^2) \) [cf. Equation (7.142)]; the other (Figure 7.63) is in terms of the BER/WBR versus SNR for the system with a particular number of users. Figure 7.62, demonstrates the expected effects of a multiuser detector where BER does not change significantly with the increase in the number of users. These curves are rather close to the single user (SU) case which is also expected.

7.14 PARALLEL INTERFERENCE CANCELLATION IN OFDM SYSTEMS IN TIME-VARYING MULTIPATH FADEING CHANNELS

In order to increase the system capacity the 4G systems will try to bring as much as possible of WLAN technology into cellular wireless networks with high mobility. Time varying multipath channel will generate inter-channel interference (ICI) and how to deal with this problem in a systematic way will be the topic of the next few sections.
Figure 7.63 BER and OFDM WER of an STBC MC CDMA system employing outer convolutional channel code in five-tap frequency selective fading channels, where $N = 256$, $G = 16$, $L = 5$, $K = 12$.

In this section we further specify the notation introduced in Section 7.1. The sequence $S_n(m)$ (the frequency-domain symbol) is fed to an IDFT, producing the OFDM signal $s_n(k)$ (the time-domain symbol) with:

$$s_n(k) = \text{IDFT}\{S_n(m)\} = \frac{1}{N} \sum_{m=0}^{N-1} S_n(m)e^{j2\pi km/N}, \quad k = 0, \ldots, N-1$$

(7.163)

The IDFT output sequence with the addition of the guard interval is:

$$s^g_n(k) = s_n(k + N - G)\mod N, \quad 0 \leq k \leq N + G - 1$$

(7.164)

where $G$ is the length of the guard interval and $(k)_N$ denotes the residue of $k$ modulo $N$. As this waveform is transmitted over the multipath channel, the received sampling data $y^g_n(k)$ at the $k$th interval of the $n$th OFDM symbol can be expressed as:

$$y^g_n(k) = s^g_n(k) * h^g_n(k, l) + v^g_n(k) = \sum_{l=0}^{k} s^g_n(k - l)h^g_n(k, l) + \sum_{l=k+1}^{L} s^g_{n-1}(k - l + N + G)h^g_n(k, l) + v^g_n(k)$$

(7.165)

where $*$ denotes the convolution operation, $L = \lfloor T_m/T_s \rfloor$ represents the maximum delay spread, $v^g_n(l)$ the ambient channel noise, and $h^g_n(k, l)$ the equivalent discrete channel at position $l$ and instant $k$.

After removal of the guard interval, we get:

$$y_n(k) = y^g_n(k + G) = \sum_{l=0}^{L} s_n(k - l)h_n(k, l) + v_n(k); \quad 0 \leq k \leq N - 1$$

(7.166)
where $h_n(k, l) = h_n^k(k + G, l)$ and $v_n(k) = v_n^k(k + G)$. The DFT, at the receiver gives:

$$Y_n(m) = DFT\{y_n(k)\} = \sum_{k=0}^{N-1} y_n(k)e^{-\frac{j2\pi km}{N}} = \frac{1}{N} \sum_{d=0}^{L} \sum_{d=0}^{N-1} S_n(d)H_n'(m - d)Ne^{-\frac{j2\pi km}{N}}$$

$$+ V_n(m) = \alpha_n(m, m)S_n(m) + \sum_{d=0, d \neq m}^{N-1} \alpha_n(m, d)S_n(d) + V_n(m),$$

for $m = 0, \ldots, N - 1$  \hspace{1cm} (7.167)

where $H_n'(m - d)N$ is the frequency response of a time-varying multipath channel with:

$$H_n'(m - d)N = \sum_{k=0}^{N-1} h_n(k, l)e^{-\frac{j2\pi (m - d)k}{N}}$$

$$\alpha_n(m, d) = \frac{1}{N} \sum_{d=0}^{L} H_n'(m - d)Ne^{-\frac{j2\pi ld}{N}}$$

$$V_n(m) = DFT\{v_n(l)\}. \hspace{1cm} (7.168)$$

Numerical analysis of these coefficients shows that the frequency responses of a time-varying multipath channel, $E[H_n'(0)] \ldots E[H_n'(N - 1)]$, and the complex weighting coefficients, $E[\alpha_n(0, 0)] \ldots E[\alpha_n(0, N - 1)]$, for different Doppler spread cases, are in the lower frequency bands, and the weighting coefficients decay smoothly. That is, the ICI comes from only a few neighboring subcarriers. In the conventional receivers the ICI is modeled as a white Gaussian process and (7.167) can be written as:

$$Y_n(m) = \alpha_n(m, m)S_n(m) + J_n(m)$$

$$J_n(m) = \sum_{d=0, d \neq m}^{N-1} \alpha_n(m, d)S_n(d) + V_n(m) \hspace{1cm} (7.169)$$

A conventional equalization method uses a one-tap equalizer to equalize the distorted samples and, the symbol samples are equalized by $w_n(m)$ as:

$$z_n(m) = w_n(m)Y_n(m), \hspace{1cm} m = 1, \ldots, N - 1 \hspace{1cm} (7.170)$$

and, the decision $dec^*(\cdot)$ output is $\hat{S}_n(m) = dec(z_n(m))$.

The zero-forcing (ZF) and the MMSE solutions can be obtained as:

$$w_n(m) = \frac{1}{\alpha_n(m, n)} \hspace{1cm} (7.171a)$$

and

$$w_n(m) = \frac{\alpha_n^*(m, m)}{|\alpha_n(m, m)|^2 + \frac{\sigma_j^2}{E_S}} \hspace{1cm} (7.171b)$$

respectively, where $\sigma_j^2$ is the variance of noise $J_n(m)$ and $E_S$ is the variance of the transmitted data symbols. Note that the MMSE solution reduces to the ZF solution for $\sigma_j^2 = 0$. In the case of a rapidly time-varying channel, the adaptation of the equalizer’s tap coefficients has to be carried out via a channel estimator. Pilot-based correction is an efficient method for this, and several types of pilot arrangement for time-frequency domains have been studied. If the equalizer considers the ICI as an additive Gaussian random process, the performance of equalizer degrades significantly for larger channel variation. As will be demonstrated latter, a parallel interference cancellation (PIC) equalizer consisting of a set of prefilters in the first stage and a set of ICI cancellation filters in the second stage, will perform better. In order to subtract the ICI term from the symbol $Y_n(m)$, we need to know all the symbols $S_n(j)$ ($j \neq m$) of the ICI term. However, we cannot get prior knowledge of the symbols of the ICI term. In order to obtain the accurate symbols of the ICI term, the received signal vector $Y_n(m)$ is fed to the prefilter to obtain an initial decision in the first stage. Then, the initial decision is fed to the ICI cancellation filter to cancel the ICI term in the second stage, as shown in Figure 7.64.
For the $p$th subcarrier, in the first stage of PIC equalization the sampled signal vector, $(Y_n(0), \ldots, Y_n(N-1))$, is fed to the prefilter $w_p(m)$. The prefilter output $z_n(p)$ at the $n$th symbol interval is:

$$z_n(p) = \sum_{m=0}^{N-1} Y_n(m)w_p(m) = \sum_{m=0}^{N-1} \alpha_n(m, p)S_n(p)w_p(m) + \sum_{m=0}^{N-1} \alpha_n(m, d)S_n(d)w_p(m) + \sum_{m=0}^{N-1} V_n(m)w_p(m), \quad 0 \leq p \leq N-1$$

(7.172)

The first term is the desired signal, the second the ICI, and the third the noise. If the prefilters compensate for the multiplicative distortion completely, the output signal $z_n(p)$ of the $p$th subcarrier contains only the desired signal, ICI, and noise. Then, an initial decision is made as $\hat{S}_n'(p)$, given by:

$$\hat{S}_n'(p) = \text{dec}(z_n(p)), \quad p = 0, \ldots, N-1$$

(7.173)

where $\text{dec()}$ denotes the decision function.

In the second stage, the ICI cancellation filters utilize the initial decisions $\hat{S}_n'(p)$; $p = 1, \ldots, K$, to cancel the ICI. For the $p$th desired subcarrier, the initial decisions $\hat{S}_n'(d)$, for $d = 1, \ldots, p-1, p+1, \ldots, N$, are passed through the ICI cancellation filters $w_p'(d)$, where the subscript $p$ denotes the $p$th desired subcarrier and $d = 1, \ldots, N, d \neq p$ denotes the input of the $p$th ICI cancellation filter from the $d$th subcarrier’s initial decision. The final decision is made in the second stage of PICs as:

$$\hat{S}_n(p) = z_n(p) + \sum_{d=0, d \neq p}^{N-1} \hat{S}_n'(d)w_p'(d) = \sum_{m=0}^{N-1} \alpha_n(m, p)S_n(p)w_p(m) + \sum_{m=0}^{N-1} \alpha_n(m, d)S_n(d)w_p(m) + \hat{S}_n'(d)w_p'(d) + \sum_{m=0}^{N-1} V_n(m)w_p(m)$$

and

$$\hat{S}_n(p) = \text{dec}\left(\{\hat{S}_n(p)\}\right)$$

(7.174)
The PIC equalizer is more complex to implement than is a one-tap equalizer. Since most of the energy of the time-varying channel is concentrated in the neighborhood of the dc component in the frequency domain, and the ICI term mainly comes from only a few neighboring subcarriers, the ICI terms that do not significantly affect \( Y_n(m) \) in (7.167) can be ignored as:

\[
\alpha_n(m, d) \approx 0, \quad \text{for } |m - d| > q \tag{7.175}
\]

Therefore, we can simplify the PIC equalizer by using 2\( q \) + 1-taps’ prefilter and 2\( q \)-taps’ ICI cancellation filter in each subcarrier. The outputs in the first stage and the second stage become:

\[
z_n(p) = \sum_{m=p-q}^{p+q} Y_n(m) w_p(m) = \sum_{m=p-q}^{p+q} \alpha_n(m, p) S_n(p) w_p(m)
+ \sum_{m=p-q}^{p+q} \sum_{d=0, d \neq p}^{N-1} \alpha_n(m, d) S_n(d) w_p(m) + \sum_{m=p-q}^{p+q} V_n(m) w_p(m)
\tag{7.176}
\]

\[
\hat{S}_n(p) = z_n(p) + \sum_{d=p-q, d \neq p}^{p+q} \hat{S}_n(d) w_p'(d) = \\
= \sum_{m=p-q}^{p+q} \alpha_n(m, p) S_n(p) w_p(m) + \sum_{d=p-q, d \neq p}^{p+q} \left[ \sum_{m=p-q}^{p+q} \alpha_n(m, d) S_n(d) w_p(m) + \sum_{d=0, d \neq p}^{N-1} \alpha_n(m, d) S_n(d) w_p(m) \right] + \sum_{m=p-q}^{p+q} V_n(m) w_p(m)
\tag{7.177}
\]

In order to improve convergence, a cost function developed from the error signal is used. If error signals are defined as

\[
e'_p = S_n(p) - z_n(p) = S_n(p) - y_{p,n}^T W_p
\]

\[
e_p = S_n(p) - \hat{S}_n(p) = S_n(p) - \left( y_{p,n}^T W_p + \hat{S}_n(p) \right)
\]

where

\[
y_{p,n} = [Y_n(p-q), \ldots, Y_n(p+q)]^T
\]

\[
\hat{S}_{p,n} = [\hat{S}_n(p-q), \ldots, \hat{S}_n(p-1), \hat{S}_n(p+1), \ldots, \hat{S}_n(p+q)]^T
\]

\[
W_p = [w(p-q), \ldots, w(p+q)]^T
\]

\[
W'_p = [w'(p-q), \ldots, w'(p-1), w'(p+1), \ldots, w'(p+q)]^T
\tag{7.178}
\]

then, under the assumption of perfect decision, the MSE cost function has the following form

\[
U_{p,n} = \gamma E \left[ |e'_p|^2 \right] + (1 - \gamma) E \left[ |e_p|^2 \right]
= \gamma [E \hat{S} I - R_{sy} W_p - W_p^H R_{sy}^H + W^H_p R_{sy} W_p] + (1 + \gamma) [E \hat{S} I - R_{sy} W_p - W_p^H R_{sy}^H + W_p^H R_{sy} W_p + W_p^H R_{sy} W_p + W_p^H R_{sy} W_p + \ldots + W_p^H R_{sy} W_p]
\tag{7.179}
\]
In order to compute $\partial \text{MMSE}$ solution is derived from $\text{fdmax NTs}$ variance of transmitted data symbols, $R_{yy} = E[y_n^*y_n] = E_S \sum_{d=0}^{N-1} R(d) + \sigma^2 I$ and:

$$
R(d) = \begin{bmatrix}
|\alpha_n(p-q,d)|^2 & \ldots & \alpha_n^*(p-q,d)\alpha_n(p+q,d) \\
\alpha_n^*(p,d)\alpha_n(p-q,d) & \ldots & \alpha_n^*(p,d)\alpha_n(p+q,d) \\
\alpha_n^*(p+q,d)\alpha_n(p-q,d) & \ldots & |\alpha_n(p+q,d)|^2
\end{bmatrix}
$$

$$
R_{yy} = E\left[ y_n^*y_n \right] = E_S \left[ \alpha_n(p-q,p) \ldots \alpha_n(p,p) \ldots \alpha_n(p+q,p) \right]
$$

MMSE solution is derived from $\partial U_{p,n}/\partial W_p = 0$ and $\partial U_{p,n}/\partial W_p = 0$ resulting in

$$
\frac{\partial U_{p,n}}{\partial W_p} = \gamma \left(-2R_{yx}^H + 2R_{yx} W_p\right) + (1 + \gamma) \left(-2R_{yx}^H + 2R_{yx} W_p + 2R_{yy} W_p^*\right) = 0
$$

$$
\frac{\partial U_{p,n}}{\partial W_p} = (1 - \gamma) \left(2R_{yx}^H W_p + 2E_S W_p\right) = 0
$$

So, the MMSE optimum tap coefficients estimates are

$$
W_{p_{opt}} = \left[ R_{yy} + (\gamma - 1) \frac{R_{yx}^H R_{yx}^H}{E_S} \right] R_{yy}^H ; \ W_{p_{opt}}^r = -\frac{R_{yx}^H}{E_S} W_{p_{opt}}.
$$

In order to compute $W_{p_{opt}}$ and $W_{p_{opt}}^r$ accurately, we need to know the impulse response of the channel.

Performance curves are shown for the following simulation scenario. The data modulation scheme is 16-QAM and the total number of subcarriers is $N = 64$. A frequency-selective Rayleigh channel is generated by using Jake’s model for two-path with a multipath spread of 2 $\mu$s. The carrier frequency is 5 GHz, the symbol rate of 16-QAM is 0.47 Msymbols/s (bit rate is 1.88 Mb/s). The OFDM block is composed of 68 samples, among them one for cyclic prefix and three for pilot signals ($N_p = 1$). The relative velocity between the transmitter and receiver is up to 216 km/h, resulting in $f_{d_{max}} NT_s$ equal to 0.032 for 54 km/h, 0.064 for 108 km/h, and 0.128 for 216 km/h.

Figure 7.65(a) shows the ratio of MSE$_1$ over MSE$_2$ versus $\gamma$, where MSE$_1$, is at the output of the prefilters, and MSE$_2$ is at the output of the ICI cancellation filters. MSE$_1$ represents the performance

![Figure 7.65(a)](image1)

![Figure 7.65(b)](image2)

Figure 7.65 Simulation results of SER and MSE$_1$/MSE$_2$ versus $\gamma$ for $f_{d_{max}} NT_s = 0.032$ and $q = 1$. 
Figure 7.66 Simulation results of SER and MSE1/MSE2 versus $\gamma$ for $f_{d_{\text{max}}}NT_S = 0.128$ and $q = 2$ performance.

of the initial decision of the equalizer and MSE2 represents the performance of the overall equalizer. Figure 7.65(b) presents the symbol error rate for the two cases. Figure 7.66 shows the same parameters for higher Doppler.

### 7.15 ZERO FORCING OFDM EQUALIZER IN TIME-VARYING MULTIPATH FADING CHANNELS

Samples of an OFDM signal, implemented by an inverse fast Fourier transform (IFFT), can be expressed as follows:

$$x_n = \sum_{m=0}^{N-1} X_m e^{j2\pi nm/N} \quad 0 \leq n \leq N$$

(7.183)

where $x_n$ represents the $n$th sample of the output of the IFFT. For the multipath fading channel consisting of $L$ discrete paths, the received signal can be written as:

$$y_n = \sum_{l=0}^{L-1} h_{n,l} x_{n-l} + w_n = h_{n,0} x_n + h_{n,1} x_{n-1} + \ldots + h_{n,L-1} x_{n-L+1} + w_n$$

which after FFT becomes:

$$Y_m = \sum_{l=0}^{L-1} \sum_{k=0}^{N-1} X_k H_l^{(m-k)} e^{-j2\pi lk/N} + W_m =$$

$$= \left[ \sum_{l=0}^{L-1} H_l^{0} e^{-j2\pi lm/N} \right] X_m + \sum_{k \neq m}^{N-1} \sum_{l=0}^{L-1} X_k H_l^{(m-k)} e^{-j2\pi lk/N} + W_m =$$

$$= \alpha_m X_m + \beta_m + W_m, \quad 0 \leq m \leq N - 1$$

(7.184)

where $W_m$ denotes the FFT of $w_n$, and $H_l^{(m-k)}$ represents the FFT of a time-variant multipath channel $h_{n,l}$ given as:

$$H_l^{(m-k)} = \frac{1}{N} \sum_{n=0}^{N-1} h_{n,l} e^{-j2\pi (m-k)n/N}$$

(7.185)

Here, $\alpha_m$ and $\beta_m$ represent the multiplicative distortion at the desired subchannel and the ICI, respectively. If the channel is assumed to be time invariant during a block period, $H_l^{(m-k)}$ in Equation (7.185) vanishes, implying that there exists no ICI for time-invariant channels. In this case, $Y_m$ in Equation (7.184) contains only the multiplicative distortion, which can be easily compensated for by a one-tap frequency-domain equalizer.
In the general case where the multipath channel cannot be regarded as time invariant during a block period, Equation (7.184) can be expressed in vector form as:

$$Y = HX + W \quad (7.186)$$

where $Y = [Y_0, \ldots, Y_{N-1}]^T$, $X = [X_0, \ldots, X_{N-1}]^T$, $W = [W_0, \ldots, W_{N-1}]^T$, and:

$$H = \begin{bmatrix}
a_{0,0} & a_{0,1} & \cdots & a_{0,N-1} \\
a_{1,0} & a_{1,1} & \cdots & a_{1,N-1} \\
\vdots & \vdots & \ddots & \vdots \\
a_{N-1,0} & a_{N-1,1} & \cdots & a_{N-1,N-1}
\end{bmatrix} \quad (7.187)$$

Here, $a_{m,k}$ in Equation (7.187) is defined as:

$$a_{m,k} = H_0^{(m-k)} + H_1^{(m-k)} e^{-j\pi k/N} + \cdots + H_{L-1}^{(m-k)} e^{-j\pi k(L-1)/N} \quad 0 \leq (m, k) \leq N - 1 \quad (7.188)$$

In order to solve for $X$ in (7.186), we need to estimate the channel matrix $H$ and calculate its matrix inverse. Since $H$ can have a large size, it is difficult to process in real time. For that reason, similar approximations like those indicated in the previous section in (7.175) will be used. Here we provide additional arguments.

If the multipath fading channel is slowly time varying (e.g., $\Delta f_D = T_b \cdot f_D < 0.1$), the time variations of the CIR, $h_{n,j}$, for all L paths, can be approximated by straight lines with low slopes during a block period $T_b$. For the channel with $\Delta f_D > 0.1$, the assumption that the CIR varies in a linear fashion during a block period no longer holds and gives rise to an error floor. When the multipath fading channel is slowly time varying, the matrix equation in (7.186) can be greatly simplified. Since most energy of the straight line with a low slope is concentrated in the neighborhood of the dc component in the frequency domain, the ICI terms which do not significantly affect $Y_m$ in (7.187) can be ignored, i.e.:

$$a_{m,k} = 0 \quad \text{for} \quad |m - k| > q/2 \quad (7.189)$$

where $q$ denotes the number of dominant ICI terms. Figure 7.67(a) shows the time variation of the CIR in a block and Figure 7.67(b) the corresponding magnitude response (absolute value of the Fourier transform) for three different Doppler frequencies. From this figure, one can see that the time variation of the CIR can be approximated as a straight line, and most of the energy is concentrated in

Figure 7.67 The characteristic of a CIR in a slowly time-varying environment. (a) Time variation of the CIR for different Doppler frequencies within a block period, and (b) corresponding magnitude responses for different Doppler frequencies.
the neighborhood of the dc component. By using the approximation in Equation (7.189), we have:

\[
H' = \begin{bmatrix}
    a_{0,0} & a_{0,1} & \ldots & a_{0,n/2} & 0 & \cdots & 0 \\
a_{1,0} & a_{1,1} & \cdots & \cdots & \cdots & \cdots & \cdots \\
    \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \ddots \\
a_{n/2,0} & \cdots & \cdots & 0 \\
    0 & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
    \vdots & 0 & \cdots & \cdots & \cdots & \cdots & \cdots \\
0 & \cdots & 0 & a_{N-1,n/2} & a_{N-1,n/2} & a_{N-1,N-1} & a_{N-1,N-1} \\
\end{bmatrix}
\]

(7.190)

The matrix in Equation (7.190) has nonzero element around diagonal \( h_{ij}' \neq 0, i - q/2 < j < i + q/2 \). Since the matrix becomes a sparse matrix for \( q \ll N \), it is not efficient to calculate the matrix inverse to estimate the transmitted sequence. By transforming the matrix \( H' \) of order \( N \times N \) to a block-diagonal matrix \( H \) of order \((N - q)(q + l) \times (N - q)(q + l)\) as shown in Figure 7.68, we obtain:

\[
H = \begin{bmatrix}
    A_0 & \cdots & 0 \\
    \cdot & A_1 & \cdots \\
    \cdot & \cdots & \cdots \\
    0 & \cdots & A_{N-1-q} \\
\end{bmatrix}
\]

(7.191)

where \( A_n \) is

\[
A_n = \begin{bmatrix}
    a_{n,n} & a_{n,n+1} & \ldots & a_{n,n+\frac{q}{2}} & 0 & \ldots & 0 \\
    a_{n+1,n} & a_{n+1,n+1} & \cdots & \cdots & \cdots & \cdots & \cdots \\
    \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \ddots \\
    a_{n+\frac{q}{2},n} & \cdots & \cdots & 0 \\
    0 & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & a_{n+\frac{q}{2},n+\frac{q}{2}} \\
    \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \\
0 & \cdots & 0 & a_{n+q-n,n+q} & \ldots & a_{n+q-n,n+q} & a_{n+q-1,n+q} & a_{n+q-1,n+q} & a_{n+q, n+q} \\
\end{bmatrix}
\]

(7.192)

and input–output relationship of the multipath channel becomes:

\[
Y = HX + W
\]

(7.193)

where

\[
X = [X_0 X_1 \ldots X_{N-1-q}]^T, \quad X_n = [X_n X_{n+1} \ldots X_{n+q}]^T \\
Y = [Y_0 Y_1 \ldots Y_{N-1-q}]^T, \quad Y_n = [Y_n Y_{n+1} \ldots Y_{n+q}]^T \\
W = [W_0 W_1 \ldots W_{N-1-q}]^T \text{ and } W_n = [W_n W_{n+1} \ldots W_{n+q}]^T
\]
Multiplying (7.193) by the inverse of $H$, using is estimated value of $H$, gives:

$$\hat{X} = \hat{H}^{-1}Y \rightarrow \hat{X}_n = \tilde{A}_q^{-1}Y_n, \quad 0 \leq n \leq N - 1 - q$$  \hspace{1cm} (7.194)

where

$$\hat{H}^{-1} = \begin{bmatrix} \tilde{A}_0^{-1} & 0 \\ \tilde{A}_1^{-1} & \cdots & \tilde{A}_{N-1-q}^{-1} \\ 0 & \cdots & \tilde{A}_{N-1-q}^{-1} \end{bmatrix}$$

Finally, the transmitted symbols $\hat{X}_{\frac{q}{2}+1}, \ldots, \hat{X}_{N-\frac{q}{2}}$ are estimated by selecting the elements in the middle of $\hat{X}_{n,1} \leq n \leq N - 2 - q (N \gg q)$. The remaining symbols $\hat{X}_0, \ldots, \hat{X}_\frac{q}{2}$ $(\hat{X}_{N-1+\frac{q}{2}}, \ldots, \hat{X}_{N-1})$ are estimated by taking the first (last) $q/2$ elements of $\hat{X}_0(\hat{X}_{N-1-q})$.

The size of the matrix inverse is reduced to $(q + 1) \times (q + 1)$, implying that the transmitted sequence can be obtained with a moderate amount of computational complexity for a small value of $q$. Thus, the linear approximation assumption discussed in this paper has transformed a large-size $(N)$ matrix inverse problem into $(N - q)$ small-size $(q)$ matrix inverse problems. The solution $(q = 2)$ for the transmitted sequence in a multipath fading channel is listed in Table 7.12. The required multiplications and additions for $q = 2$ are $6N + 2$ and $3N$ respectively.

In order to construct the matrix equation in Equation (7.193), it is necessary to estimate the channel matrix $H$. To demonstrate the effectiveness of the above equalizer for time-variant multipath channels, the simulations were performed for the scenario similar to the one used to generate Figures 7.65 and

Table 7.12 Frequency-domain equalization $(q = 3)$ for an OFDM system in a multipath fading channel

<table>
<thead>
<tr>
<th>$\hat{X}$</th>
<th>$\tilde{b}$</th>
<th>$\tilde{a}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\hat{X}<em>0 = (b</em>{0,0}Y_0 + b_{0,1}Y_1 + b_{0,2}Y_2)/\Delta_0$</td>
<td>$b_{0,0}$</td>
<td>$\tilde{a}<em>{1,1}\tilde{a}</em>{2,2} - \tilde{a}<em>{1,2}\tilde{a}</em>{2,1}$</td>
</tr>
<tr>
<td>$\hat{X}<em>{n+1} = (b</em>{n+1,n}Y_n + b_{n+1,n+1}Y_{n+1} + b_{n+1,n+2}Y_{n+2})/\Delta_n$</td>
<td>$b_{n+1,n}$</td>
<td>$-\tilde{a}<em>{n+1,n}\tilde{a}</em>{n+2,n+2}$</td>
</tr>
<tr>
<td>$\hat{X}<em>{N-1} = (b</em>{N-1,N-3}Y_{N-3} + b_{N-1,N-2}Y_{N-2} + b_{N-1,N-1}Y_{N-1})/\Delta_{N-2}$</td>
<td>$b_{N-1,N-3}$</td>
<td>$\tilde{a}<em>{N-2,N-3}\tilde{a}</em>{N-1,N-2}$</td>
</tr>
<tr>
<td>$\Delta_n = \tilde{a}<em>{n,n}\tilde{a}</em>{n+1,n+1}\tilde{a}<em>{n+2,n+2} - \tilde{a}</em>{n,n}\tilde{a}<em>{n+2,n+2}\tilde{a}</em>{n+1,n+1} - \tilde{a}<em>{n+2,n+2}\tilde{a}</em>{n+1,n}\tilde{a}_{n,n+1}$</td>
<td>$\tilde{a}_{n,n}$</td>
<td>$\tilde{a}_{n,n}$</td>
</tr>
</tbody>
</table>
Figure 7.69 BER for the two types of equalizers in time-variant channels with three different Doppler frequencies: (a) $f_D = 20$ Hz, (b) $f_D = 100$ Hz, (c) $f_D = 200$ Hz.

7.66. A two-path fading channel with a multipath spread of 2 $\mu$s, bandwidth 500 kHz, carrier at 1 GHz, 64 subbands, the size of FFT became 64, OFDM block 68 samples, one for a cyclic prefix and three for pilot signals ($N_p = 1$), modulation 16-QAM, the data rate 1.88 Mb/s (64 subcarriers $\times$ 4 bits per symbol/136 $\mu$s), Doppler up to 200 Hz $\rightarrow \Delta f_D = 0.272$ % for 20 Hz, 1.36 % for 100 Hz, and 2.72 % for 200 Hz. Perfect carrier and symbol synchronizations were assumed.

The results for conventional frequency-domain equalizer, with one tap in an OFDM system which compensates for the frequency-selectivity of a multipath fading channel, assuming that the channel is stationary over the period of an FFT block, are denoted in Figure 7.69 by C.

The results for zero forcing equalizer, described in this section are denoted in Figure 7.69 by P.

7.16 CHANNEL ESTIMATION FOR OFDM SYSTEMS USING MULTIPLE RECEIVE ANTENNAS

Figure 7.70 represents a block diagram of a space division multiple access (SDMA) uplink scenario, as observed on an OFDM subcarrier basis. Each of the $L$ users is equipped with a single transmit antenna and the BS’s receiver has $P$-element antenna. For simplicity, we have omitted the subcarrier index $k$ ($K$ subcarriers, $k = 0, \ldots, K - 1$). A decision-directed channel estimator (DDCE) aided OFDM receiver is shown in Figure 7.71 and parallel interference canceller (PIC)-assisted channel transfer function estimator in Figure 7.72.

The complex output signal $x_p[n, k]$ of the $p$th receiver-antenna element in the $k$th subcarrier of the $n$th OFDM symbol is given by:

$$x_p[n, k] = \sum_{i=1}^{L} H_p^{(i)}[n, k]s^{(i)}[n, k] + n_p[n, k]$$  \hspace{1cm} (7.195)

In vector notation, we have:

$$x_p[n] = \sum_{i=1}^{L} S^{(i)}[n]H_p^{(i)}[n] + n_p[n]$$  \hspace{1cm} (7.196)
From Equation (7.196) an \textit{a posteriori} (apt) estimate of the channel is:

$$ \hat{\mathbf{H}}_{\text{apt}}^{(i)}[n] = \mathbf{S}[n] \left( \mathbf{x}[n] - \sum_{i=1, i \neq j}^{L} \mathbf{S}[n] \hat{\mathbf{H}}_{\text{apt}}^{(i)}[n] \right) $$ (7.197)

and its prediction (\textit{a priori} channel transfer-factor estimation) as:

$$ \mathbf{H}_{\text{ap}r}^{(i)}[n] = f \left( \hat{\mathbf{H}}_{\text{apt}}^{(i)}[n-1], \ldots, \hat{\mathbf{H}}_{\text{apt}}^{(i)}[n-N_{\text{ap}r}^{(i)}] \right). $$ (7.198)
of $\tilde{U}$ respect to the unitary transform matrix $\tilde{U}$ are associated with a significant value of $\gamma$ and $I_{\gamma}$.

The Jakes model is used, having an OFDM-symbol-normalized Doppler frequency of $F_{D}$ assumed to have the same Doppler power spectrum. For the channel’s space–time correlation function were assumed. The channels between the different transmit antennas and each receiver antenna were assumed. The MSE transfer-factor estimation errors. Matrix $R_{\text{appr}}$ has been kept constant during each OFDM symbol’s transmission period.

Furthermore, we considered ‘frame-invariant’ fading, where the fading envelope of each CIR-related tap has been kept constant during each OFDM symbol’s transmission period.

which corresponds to a vehicular speed of 50 km/h, or equivalently, 31.25 miles/h in the context of the indoor wireless asynchronous transfer mode (WATM) system’s parameters (see Section 7.4).

The simulation results with the same parameters versus Doppler frequency are shown in Figure 7.74.

**Figure 7.72** PIC-assisted channel transfer-function estimation.

with optimum predictor coefficients obtained as:

$$
\hat{z}_{\text{pre}}^{(j)}|_{\text{opt}} = \left[ R^{(j)} + \frac{K_0}{\text{Trace}(\gamma^{(j)} I_{K_0}^{(j)})} \alpha_j \sigma_j^2 \right]^{-1} \cdot r^{(j)}
$$

(7.199)

$$
R^{(j)} = E \left[ \hat{H}^{(j)} \hat{H}^{(j)\dagger} \right],
$$

$$
\text{MSE}_{\text{appr}}^{(j)}[n] = \frac{1}{K} \text{Trace} \left( R_{\Delta H_{\text{appr}}}^{(j)}[n] \right)
$$

$$
\Delta H_{\text{appr}}^{(j)}[n] = H_j[n] - \hat{z}_{\text{appr}}^{(j)}[n].
$$

(7.200)

where $R_{\Delta H_{\text{appr}}}^{(j)}[n] \in C^{K \times K}$ denotes the autocorrelation matrix of the vector $\Delta H_{\text{appr}}^{(j)}$ of $a \text{ priori}$ channel transfer-factor estimation errors. Matrix $R^{(j)}$ is the decomposition of the $j$th user’s channel’s space-frequency correlation matrix $R^{(j)}$ with respect to the unitary transform matrix $\tilde{U}^{(j)}$, which is formulated as $\gamma^{(j)} = \tilde{U}^{(j)} R^{(j)} \tilde{U}^{(j)\dagger}$, and $I_{K_0}^{(j)}$ is a sparse identity matrix having unity entries only at those $K_0$ number of positions, which is associated with a significant value of $\gamma^{(j)}$. Hence, we note that the evaluation of $\text{Trace}(\gamma^{(j)} I_{K_0}^{(j)})$ requires the knowledge of $R^{(j)}$ which is not directly available in practice.

For the simulation results shown in Figure 7.73, four simultaneous equal-power OFDM users with one transmit antenna and the same modulation scheme and a BS with four receiving antennas were assumed. The channels between the different transmit antennas and each receiver antenna were assumed to have the same Doppler power spectrum. For the channel’s space–time correlation function the Jakes model is used, having an OFDM-symbol-normalized Doppler frequency of $F_{D} = 0.007$, which corresponds to a vehicular speed of 50 km/h, or equivalently, 31.25 miles/h in the context of the indoor wireless asynchronous transfer mode (WATM) system’s parameters (see Section 7.4).

Furthermore, we considered ‘frame-invariant’ fading, where the fading envelope of each CIR-related tap has been kept constant during each OFDM symbol’s transmission period.

The sparse identity matrix $I_{K_0}^{(j)}$ could be designed for retaining the first $K_0$ CIR-related coefficients of $\gamma^{(j)}$—rather than the $K_0$ largest one—or alternatively, for retaining the first $K_0^{(j)}$ and the last $K_0^{\dagger}$ CIR-related coefficients of $\gamma^{(j)}$, where $K_0 = K_0^{(j)} + K_0^{\dagger}$.

The simulation results with the same parameters versus Doppler frequency are shown in Figure 7.74.
Figure 7.73 A priori channel-estimation MSE versus SNR performance PIC-assisted DDCE, using the optimum recursive predictor coefficients evaluated with the aid of the iterative approach. Fr.-Inv. Fad. SWATM from Section 7.4. L Rec.-Antennas, L Users, MPSK.

Figure 7.74 A priori channel-estimation MSE versus OFDM-symbol-normalized Doppler-frequency performance exhibited by the PIC-assisted DDCB, using optimum recursive predictor coefficients. The predictor coefficients were optimized for $\tilde{F}_D = 0.05$, using the iterative approach.

7.17 TURBO PROCESSING FOR AN OFDM-BASED MIMO SYSTEM

In this section we will discuss performance of a receiver using constrained list sphere decoder (CLSD)-based soft-detector and the space-time bit-interleaved coded modulation (STBICM) approach.

Let $H = [h_{n,m}] \in C^{N \times M}$ denote the MIMO channel matrix for the $k$th subcarrier, where $h_{n,m}$ denotes the channel gain from the $m$th Tx antenna to the $n$th Rx antenna for subcarrier $k$. For
simplicity index \( k \) is dropped. A received data vector \( y \) for this subcarrier, can be written as:

\[
y = Hx + w \tag{7.201}
\]

where \( x = [x_1, x_2]^T \) is the \( 2 \times 1 \) symbol vector transmitted on this subcarrier and \( w \sim N(0, \sigma^2 I_N) \) is the additive white circularly symmetric complex Gaussian noise with variance \( \sigma^2 \).

We consider the general case of MIMO soft-detection where \( H \) is:

\[
N \times M. \text{Let } \beta = BM \text{ and } b = [b_1, b_2, \ldots, b_\beta]^T, \text{ with } b_i \in \{-1, +1\}, \text{ } i = 1, 2, \ldots, \beta, \text{ be the bit vector that maps to } x. \text{ Then } x \text{ can be expressed as } x(b) \text{ to stress its dependence on } b.
\]

The bit metric for the \( i \)th bit, \( i = 1, 2, \ldots, \beta, \) is defined as:

\[
l_D(i) = \ln \frac{P(b_i = +1 \mid y, H)}{P(b_i = -1 \mid y, H)} \tag{7.202}
\]

which, by using the Bayes’ theorem, can be written as

\[
l_D(i) = \ln \frac{P(b_i = +1)}{P(b_i = -1)} + \ln \frac{P(b_i + 1, H)}{P(b_i - 1, H)} \triangleq l_A(i) + l_E(i). \tag{7.203}
\]

Here, \( l_A(i) \) and \( l_E(i) \) are referred to as the \textit{a priori} and extrinsic information, respectively. The \textit{a priori} information can be obtained from the BCJR algorithm presented in Appendix 2.1, a soft-in/soft-out decoder, and the extrinsic information can be calculated by the STBICM-based soft-detector:

\[
l_E(i) \approx \frac{1}{2} \max_{b \in B_{i, +1}} \left\{ -\frac{1}{\sigma^2} \| y - Hx(b) \|^2 + b^T I_A - l_A(i) \right\} - \frac{1}{2} \max_{b \in B_{i, -1}} \left\{ -\frac{1}{\sigma^2} \| y - Hx(b) \|^2 + b^T I_A + l_A(i) \right\} \tag{7.204}
\]

where \( B_{i, +1} \) and \( B_{i, -1} \) are the set of \( 2^\beta - 1 \) bit vectors with \( b_i \) being +1 and -1, respectively, and \( I_A = [l_A(1) \ l_A(2) \ldots l_A(\beta)]^T \).

In the iterative decoding process, as shown in Figure 7.75, the \textit{a priori} information \( l_A(i) \) for the MIMO detector i.e., the inner decoder, is obtained from the soft output of the BCJR decoder, i.e., the outer decoder. Specifically, it is from the incremental soft information (denoted as \( L_E \) in the figure) due to the BCJR decoder. Similarly, the input for the BCJR decoder (denoted as \( L_A \) in the figure) is the deinterleaved extrinsic information \( l_E \), which is the incremental soft information from the MIMO detector. The iterative decoding starts from the MIMO detection with the \textit{a priori} information being 0.

The optimal but extremely inefficient STBICM-based soft detector of Equation (7.204) can be simplified by using list sphere decoder (LSD). The LSD-based soft-detector keeps the STBICM framework while improving its efficiency by searching in much smaller subsets \( B_{i, +1} \subset B_{i, +1} \) and \( B_{i, -1} \subset B_{i, -1} \) with \( |B_{i, +1}| \ll 2^{\beta - 1} \) and \( |B_{i, -1}| \ll 2^{\beta - 1} \).

![Figure 7.75 Turbo processing for the OFDM-based MIMO system.](image-url)
The LSD-based soft detector is implemented in the following two steps.

(i) Obtain a set $\tilde{B}$ of vectors $\mathbf{b}$, referred to as the candidate pool, which satisfies

$$\begin{align*}
\| \mathbf{y} - \mathbf{Hx}(\mathbf{b}) \|^2 &\leq d_i \quad \forall \mathbf{b} \in \tilde{B} \\
\| \mathbf{y} - \mathbf{Hx}(\mathbf{b}) \|^2 &> d_i \quad \forall \mathbf{b} \notin \tilde{B}
\end{align*}$$

(7.205)

by using the LSD algorithm with a fixed sphere radius $d_i$ determined by the antenna numbers and noise variance.

(ii) Calculate $\tilde{B}_{i+1} = B_{i+1} \cap \tilde{B}$ and $\tilde{B}_{i-1} = B_{i-1} \cap \tilde{B}$ for each $i = 1, 2, \ldots, \beta$ and obtain the bit metric using Equation (7.204) with $B_{i+1}$ and $B_{i-1}$ being replaced by $\tilde{B}_{i+1}$ and $\tilde{B}_{i-1}$, respectively.

The LSD-based soft detector is orders of magnitude more efficient than the STBICM-based soft detector. However, this efficiency is obtained at the cost of some performance degradation due to the limited candidate pool used by LSD. Due to the limited candidate pool, the a priori information from the outer decoder can be quite poor. When the poor a priori information is used blindly without being censored, the performance of the iterative processing can degrade with the increase in iteration number.

The detection/decoding can be improved by constraining the a priori information from the outer decoder. We follow the simple rules below to constrain the a priori information:

(R1) The larger the sphere radius, the larger the maximum allowable a priori information, denoted $l_{max}$.

(R2) The larger the value of $\beta$, the smaller the maximum allowable a priori information $l_{max}$.

(R3) The higher the SNR, the larger the maximum allowable a priori information $l_{max}$.

R1 is chosen because for a given SNR and a given number of candidates in a sphere, the larger the radius, the better the channel. We give more freedom or larger maximum allowable a priori information $l_{max}$ to a better channel. R2 is chosen because the more complicated the constellation and the larger the number of transmitted symbols, the larger the $\mathbf{b}^T \mathbf{l}_A$. We put more constraint on the maximum allowable value of $\mathbf{l}_A$ to prevent the value of $\mathbf{b}^T \mathbf{l}_A$ from growing too large. Finally, R3 is chosen because the higher the SNR or the smaller the noise variance $\sigma^2$, the more trust we can put on the larger a priori information from the outer decoder.

Based on the aforementioned rules, the value of the a priori information for our MIMO system is constrained as:

$$l_{max} = \frac{d}{\min[1, k_C \sigma^2]}$$

(7.206)

where $d$ is the radius of the sphere and $k_C$ is a constant related to $\beta$. For example, for 64-QAM, we choose $k_C = 4 \times 42$ for above MIMO system, where 42 is the average power of the 64-QAM constellation. We choose $d$ so that the sphere contains five candidates.

Once we have determined the maximum allowable a priori information $l_{max}$, we can constrain the a priori information from the outer decoder

$$l_A^{(c)} = \begin{cases} 
\mathbf{l}_A, & \max\{\mathbf{l}_A\} \leq l_{max} \\
\max\{\mathbf{l}_A\}, & \max\{\mathbf{l}_A\} > l_{max}
\end{cases}$$

(7.207)

This constraining scheme in Equation (7.207) is by no means the best; however, it is very simple and can lead to excellent results in WLAN applications. We also clip the soft outputs of the CLSD-based soft detector (by ±50) in addition to constraining the a priori information.

Some performance illustrations are given in Figure 7.76.

### 7.18 PAPR REDUCTION OF OFDM SIGNALS

It has been already pointed out in Section 7.10 that nonlinear effects can cause clipping of OFDM signal due to a large peak-to-power ratio (PAPR). In fading channels, this problem becomes even
more serious. A possible technique for PAPR reduction in orthogonal frequency division multiplexing (OFDM) systems is utilization of signal scrambling. Golay sequences (with dual capabilities of error correction and peak reduction) [91] and partial transmit sequences (PTS) [92, 93] have been suggested for these purposes. Computation of optimal PTS weight factors via exhaustive search requires exponential complexity in the number of subblocks; consequently, many suboptimal strategies have been developed.

Suboptimal PTS strategies include the following. The iterative flipping algorithm (FA) [94] has complexity linearly proportional to the number of subblocks, and each phase factor is individually optimized regardless of the optimal value of other phases. A neighborhood search is proposed in [95] using gradient descent search. In [96] dual layered phase sequencing is used to reduce complexity, at the price of PAPR performance degradation. In [93] a suboptimal strategy is developed by modifying the problem into an equivalent problem of minimizing the sum of phase-rotated vectors. An initial set of phase vectors is computed by reducing the peak amplitude of each sample and the best phase vector of the set is chosen as the final solution. Finally, [97] gives an orthogonal projection-based approach for computing PTS phase factors.

In this section we present an efficient algorithm for computing the optimal PTS weights that has lower complexity than does exhaustive search.

For the system model, let \( X = [X_1, \ldots, X_N]^T \) be a block of \( N \) symbols being transmitted, where each symbol is modulated to one of the carrier frequencies \( \{ f_n, n = 1, \ldots, N \} \) where \( f_n = n \Delta f, \Delta f = 1 / NT \) and \( T \) is the signal period. The complex envelope of the transmitted signal is:

\[
 x(t) = \frac{1}{\sqrt{N}} \sum_{n=1}^{N} X_n e^{j2\pi f_n t} \quad 0 \leq t < NT \tag{7.208}
\]

where \( j = \sqrt{-1} \). The PAPR of the OFDM signal \( x(t) \) is defined as:

\[
PAPR = \max \frac{|x(t)|^2}{E[|x(t)|^2]} \tag{7.209}
\]

\( E[|x(t)|^2] = 1 \) for unitary signal constellations. The signal in Equation (7.208) can be oversampled by generating \( LN \) samples, where \( L > 1 \) is the oversampling factor. These samples can be computed by using appropriate zero-padding and using an inverse fast Fourier transform (IFFT).
In the PTS approach, $X$ is divided into $M$ disjoint sub-blocks $X_m$ ($1 \leq m < M$) of length $U$ where $N = MU$ for some integers $M$ and $U$. For $m = 1, \ldots, M$, let $[R_{1,m}, \ldots, R_{LN,m}]^T$ be the zero-padded IFFT of $X_m$ (all zeros except for the block $m$). PTS combines phase-rotated versions of these sub-block IFFTs in order to minimize the PAPR. The signal samples at the PTS output can be written as:

$$
X' = \left[ \begin{array}{c}
R_{1,1} & R_{1,2} & \cdots & R_{1,M} \\
R_{2,1} & R_{2,2} & & \\
\vdots & & & \\
R_{LN,1} & & \cdots & R_{LN,M}
\end{array} \right] \left[ \begin{array}{c}
b_1 \\
b_2 \\
\vdots \\
b_M
\end{array} \right]
$$

(7.210)

where $x' = [x'_1, \ldots, x'_{LN}]$ is the block of optimized signal samples. The optimization problem is to find optimum phases $b_m$ according to:

$$
\{b_1^*, \ldots, b_M^*\} = \arg \min_{\{b_1, \ldots, b_M\}} \left( \max_{1 \leq k < LN} \left| \sum_{m=1}^M b_m R_{k,m} \right| \right)
$$

(7.211)

where $b_m \in P = \{e^{j2\pi k/N}, k = 0, \ldots, q - 1\}$. The last phase factor can be fixed ($b_M = 1$) without loss of generality. Therefore, $q^{M-1}$ distinct possible vectors $b$ should be tested to solve Equation (7.211). Accordingly, in an exhaustive search approach, the computational complexity increases exponentially with the number of sub-blocks.

The optimization algorithm that solves Equation (7.211) with lower complexity, is motivated by the shortest vector problem (SVP) in a lattice [100]. An $M$-dimensional lattice is the set of vectors (lattice points) $[Ab, b \in \mathbb{Z}]$, where $b = (b_1, \ldots, b_M)$ and the columns of matrix $A \in \mathbb{R}^{N \times M}$ are called the basis for the lattice. The SVP requires finding the shortest non-zero vector in the lattice, where the length can be measured in any $l_p (p \geq 1)$ norm. The $l_p$ norm of a vector $x = (x_1, x_2, \ldots, x_N)$ is defined as $\|x\|_p = (\sum |x_i|^p)^{1/p}$ and $\|x\|_\infty = \max_i |x_i|$. Fincke and Phost [98] have developed an efficient algorithm for SVP in $l_2$ (i.e., Euclidean distance) by enumerating all the lattice points inside a sphere centered at the origin. This is one example of sphere decoding that has wide application in communication problems (see [99] for a detailed survey). The signal vectors (7.210) can be interpreted as lattice points generated by $R$. However, (7.211) is equivalent to the SVP in $l_\infty$ norm. As such, the original Fincke–Phost sphere decoder (FPSD) cannot be directly applied to problem at hand. Nevertheless, the basic premise of FPSD – to generate only lattice points $x$ for which $\|x\|_2 \leq \mu$ can be adapted; consequently, only lattice points for which $\|x\|_\infty \leq \mu$ are generated, and this is equivalent to $|x_i| \leq \mu \forall k$. We refer to this as a sphere decoder based algorithm (SDBA) (see Appendix 4.2 or [100]).

Let $x' = [x'_1, \ldots, x'_{LN}]$ be defined as in Equation (7.210) and let $R_k$ represent the $k$th row of the matrix $R$. Then each element of $x'$ can be expressed as $x'_k = R_k \cdot b$. To find the PAPR of the OFDM signal, the amplitude of $x'_k$ is computed according to

$$
|x'_k|^2 = x'^H_k \cdot x_k = b^H \cdot R_k^H \cdot R_k \cdot b
= b^H \cdot \left[ R_k^H \cdot R_k + \alpha^2 I \right] b - \alpha^2 b^H \cdot b = b^H \cdot A_k \cdot b - \alpha^2 M
$$

(7.212)

where $\alpha$ is an arbitrary non-zero real number and $(\cdot)^H$ denotes conjugate transpose. The resulting $M \times M$ matrix $A_k$ is positive-definite due to the addition of $\alpha^2 I$, and therefore can be Cholesky factorized as $A_k = Q_k^H \cdot Q_k$ where $A_k$ is an upper-triangular matrix. Substituting $A_k$ into Equation (7.212) gives:

$$
|x'_k|^2 = b^H \cdot Q_k^H \cdot Q_k \cdot b - \alpha^2 M = \|Q_k \cdot b\|^2 - \alpha^2 M
$$

(7.213)

where the signal sample is now a function of the phase vector $b$. 


If we wish to limit the PAPR Equation (7.209) to
\[ \mu^2 E \left[ |x(t)|^2 \right] \] for some positive number \( \mu \) the candidate phase vectors can be generated from Equation (7.213) subject to the following constraint:
\[
\left| \begin{bmatrix} Q_{1,1}^k & \cdots & Q_{1,M}^k \\ 0 & Q_{2,2}^k & \cdots \\ \vdots & \ddots & \vdots \\ 0 & 0 & Q_{M,M}^k \end{bmatrix} \right| \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_M \end{bmatrix} \right| ^2 < \mu^2 + \alpha^2 M \tag{7.214}
\]
for \( 1 \leq k \leq LN \). Sphere decoding only searches among those candidates that lie inside the sphere of radius \( \mu^2 + \alpha^2 M \) and, therefore, reduces the complexity of the search. If we rewrite Equation (7.214) as
\[
\sum_{\nu=1}^{M} \left| \sum_{\mu=0}^{M} Q_{\nu,\mu}^k b_\mu \right|^2 < \mu^2 + \alpha^2 M, \ 1 \leq k \leq LN \tag{7.215}
\]
then in order to satisfy Equation (7.215), the following set of inequalities must be satisfied for \( 1 \leq k \leq LN \):
\[
\left| Q_{M,M}^k b_M \right|^2 < \mu^2 + \alpha^2 M, \\
\sum_{\nu=M-1}^{M} \left| \sum_{\mu=0}^{M} Q_{\nu,\mu}^k b_\mu \right|^2 < \mu^2 + \alpha^2 M \\
\sum_{\nu=M-2}^{M} \left| \sum_{\mu=0}^{M} Q_{\nu,\mu}^k b_\mu \right|^2 < \mu^2 + \alpha^2 M \\
\vdots \\
\sum_{\nu=1}^{M} \left| \sum_{\mu=0}^{M} Q_{\nu,\mu}^k b_\mu \right|^2 < \mu^2 + \alpha^2 M \tag{7.216}
\]
Note that the first equation contains \( b_M \) only, the second \( b_{M-1} \) and \( b_M \) only, and so on. We fix \( b_M = 1 \) without loss of generality. However, the first term in Equation (7.216) constrains the parameter \( \mu \) (which specifies the achievable PAPR reduction). We use the second line of Equation (7.216) and \( b_M \) to generate candidates for \( b_{M-1} \). These candidates and (7.218) again are used to generate candidates for \( b_{M-2} \). This process is repeated until the candidates for the whole phase vector \( b \) are generated. The resulting number of candidates is substantially smaller than \( q^{M-1} \). Therefore, the search space is reduced, compared with exhaustively searching all \( q^{M-1} \) phase vectors, which reduces complexity.

For simulation an OFDM signals with 512 8-PSK subcarriers, \( L = 4 \) and \( \alpha = \sqrt{1/M} \) is used. Figure 7.77 compares the complementary cumulative density functions (CCDF’s) of the PAPR where the number of sub-blocks in PTS is \( M = 8 \) and 4. Also, the PTS phase factors are chosen from \( P = \{ +1, -1 \} \). Note that both the SDBA algorithm presented in this section and exhaustive search perform identically, verifying that the SDBA algorithm is optimal, resulting in approximately 1-dB additional reduction compared with the FA.

Instead of scrambling, care can be taken over low peak-to-average power ratios in the stage of channel coding [101–120].
Figure 7.77 CCDF of the PAPR for exhaustive search (same as SDBA) and flipping algorithm, (a) \( M = 8 \), (b) \( M = 4 \).

**APPENDIX 7.1**

**Derivation of Equation (7.152)**

As explained earlier, \( d^i[n] \) is also denoted as \( s^i_j[p] \), and among all the received signals, only \( y_i[p] \) is related to \( s^i_j[p] \). Hence, \( \Delta s^2 \) can be further simplified to

\[
\Delta s^2 = \begin{bmatrix}
\left| y_i[p] + \sum_{l} X^l_i[p] W[p] h^l \right|^2 & - \left| y_i[p] + \sum_{l} X^l_i[p] W[p] h^l \right|^2
\end{bmatrix}_s \rightarrow +1
\]

\[
\left| y_i[p] + \sum_{l} X^l_i[p] W[p] h^l \right|^2
\]

\[
p[h^k | H^k, \sigma^2, D, Y] \propto p[Y|H, \sigma^2, D]^p[h^k]
\]

\[
\propto \exp \left\{ -\frac{1}{\sigma^2} \sum_{i=0}^{M-1} \left| y_i + \sum_{l=1}^{K} X^l_i W h^l \right|^2 \right\} \exp \left\{ -(h^k - h_{k0})^H \Sigma_{k0}^{-1} (h^k - h_{k0}) \right\}
\]

\[
\propto \exp \left\{ -h^H \left( \Sigma_{k0}^{-1} + \frac{1}{\sigma^2} \sum_{i=0}^{M-1} W^H X^l_i X^l_i W \right) h^k \right\}
\]

\[
\propto \exp \left\{ +2R \left( h^H \left( \Sigma_{k0}^{-1} h_{k0} + \frac{1}{\sigma^2} \sum_{i=0}^{M-1} W^H X^l_i \left( y_i - \sum_{l \neq k} X^l_i W \right) \right) \right) \right\}
\]

\[
\propto \exp \left\{ -(h^k - h_{k0})^H \Sigma_{k0}^{-1} (h^k - h_{k0}) \right\} \sim N_c(H_{k*}, \Sigma_{k*})
\]  

(A.7.1)
Derivation of Equation (7.155)

\[
p[\sigma^2 | H, D, Y] \propto p[Y|H, \sigma^2 | D]p[\sigma^2]
\]

\[
\propto \left( \frac{1}{\sigma^2} \right)^{2 MN} \exp \left( -\frac{1}{\sigma^2} \sum_{i=0}^{M-1} \left[ y_i + \sum_{k=1}^{K} X_i^k \text{Wh}^k \right]^2 \right) \times \left( \frac{1}{\sigma^2} \right)^{v_0+1} \exp \left( -\frac{v_0 \lambda_0}{\sigma^2} \right)
\]

\[
= \left( \frac{1}{\sigma^2} \right)^{v_0+2MN+1} \exp \left( -\frac{v_0 \lambda_0 + s^2}{\sigma^2} \right)
\]

\[
\sim \chi^2 \left( 2[v_0 + 2MN], \frac{v_0 \lambda_0 + s^2}{v_0 + 2MN} \right)
\]

(A.7.2)

Derivation of Equation (7.157)

\[
p[d^k[n] = +1 | H, \sigma^2, D^k_n, Y] = \frac{p[d^k[n] = +1 | H, \sigma^2, D^k_n | Y]}{p[H, \sigma^2, D^k_n | Y]}
\]

\[
= p[Y | d^k[n] = +1, D^k_n, H, \sigma^2] \frac{P[d^k[n] = +1 | D^k_n]P[H]P[\sigma^2]}{P[H, \sigma^2, D^k_n | Y]}
\]

\[
\Rightarrow \frac{p[d^k[n] = +1 | H, \sigma^2, D^k_n, Y]}{p[d^k[n] = -1 | H, \sigma^2, D^k_n, Y]} = \frac{P[d^k[n] = +1 | D^k_n]P[Y | d^k[n] = +1, D^k_n, H, \sigma^2]}{P[d^k[n] = -1 | D^k_n]P[Y | d^k[n] = -1, D^k_n, H, \sigma^2]}
\]

\[
= \frac{P[d^k[n+1] | d^k[n] = +1]}{P[d^k[n+1] | d^k[n] = -1]} \times \frac{P[d^k[n] = +1 | d^k[n-1]]}{P[d^k[n] = -1 | d^k[n-1]]}
\]

\[
\times \exp \left( -\frac{1}{\sigma^2} \sum_{i=0}^{M-1} \left[ y_i + \sum_{i} X_i^k \text{Wh}^k \right]^2 \right) \left[ y_i + \sum_{i} X_i^k (d^k_i) \text{Wh}^k \right]^2 \right)
\]

\[
= \exp \left[ 2 \rho^k[n] d^k[n+1] + 2 \rho^k[n-1] d^k[n-1] - \frac{\Delta s^2}{\sigma^2} \right].
\]

(A.7.3)

REFERENCES


REFERENCES


REFERENCES


Ultra Wide Band Radio

In this chapter we discuss technology which is based on the spread spectrum concept, such as CDMA, as described in Chapter 5. The difference is that the pulse (called chip in Chapter 5) period used in this field is below 1 ns, resulting in a bandwidth of over 1 GHz, hence the name Ultra Wide Band (UWB) Radio. The second important characteristic is that the signal can be transmitted with no carrier. This is why very often the system is also referred to as Impulse Radio (IR). The above characteristics of the signal will require the modification of the signal format and detection concepts. In addition to these issues the chapter will also cover the basic characteristics of the UWB channel.

8.1 UWB MULTIPLE ACCESS IN A GAUSSIAN CHANNEL

A typical time-hopping format used in this case can be represented as [1–31]:

\[ s_{tr}^{(k)}(t^{(k)}) = \sum_{j=-\infty}^{\infty} \omega_{tr}(t^{(k)} - jT_f - c^{(k)}_j T_c - \delta d_{i,j/N_s}) \]  

(8.1)

where \( t^{(k)} \) is the \( k \)th transmitter’s clock time and \( T_f \) is the pulse repetition time. The transmitted pulse waveform \( \omega_{tr} \) is referred to as a monocycle. To eliminate collisions due to multiple access, each user (indexed by \( k \)) is assigned a distinctive time shift pattern \( \{c^{(k)}_j\} \) called a time-hopping sequence. This provides an additional time shift of \( c^{(k)}_j T_c \) seconds to the \( j \)th monocycle in the pulse train, where \( T_c \) is the duration of addressable time delay bins. For a fixed \( T_f \), the symbol rate \( R_s \) determines the number \( N_s \) of monocytes that are modulated by a given binary symbol as \( R_s = (1/N_sT_f) s^{-1} \). The modulation index \( \delta \) is chosen to optimize performance. For performance prediction purposes, most of the time the data sequence \( \{d^{(k)}_j\}_{j=-\infty}^{\infty} \) is modeled as a wide-sense stationary random process composed of equally likely symbols. For data, a pulse position modulation is used.

8.1.1 The multiple access channel

When \( K \) users are active in the multiple access system, the composite received signal at the output of the receiver’s antenna is modeled as

\[ r(t) = \sum_{k=1}^{K} A_k s_{rec}^{(k)}(t - \tau_k) + n(t) \]  

(8.2)
Figure 8.1 A typical ideal received monocycle $\omega_{\text{rec}}(t)$ at the output of the antenna subsystem as a function of time in nanoseconds.

The antenna/propagation system modifies the shape of the transmitted monocycle $\omega_{\text{tr}}(t)$ to $\omega_{\text{rec}}(t)$ at its output. An idealized received monocycle shape $\omega_{\text{rec}}(t)$ for a free-space channel model with no fading is shown in Figure 8.1.

8.1.2 Receiver

The optimum receiver for a single bit of a binary modulated impulse radio signal in additive white Gaussian noise (AWGN) is a correlation receiver defined as

\[
\text{decide } d_0^{(1)} = 0 \quad \text{if} \quad \frac{\text{pulse correlator output}}{\text{test statistic}} = \frac{\sum_{j=0}^{N_s-1} \int_{\tau_1+jT_f}^{\tau_1+(j+1)T_f} r(u,t) \nu(t - \tau_1 - jT_f - c_j^{(1)}T_f) dt}{\alpha(u)} > 0 \tag{8.3}
\]

where $\nu(t) \triangleq \omega_{\text{rec}}(t) - \omega_{\text{rec}}(t - \delta)$.

The optimal detection in a multiuser environment, with knowledge of all time-hopping sequences, leads to complex parallel receiver designs [2]. However, if the number of users is large and no such multiuser detector is feasible, then it is reasonable to approximate the combined effect of the other users dehopped interfering signals as a Gaussian random process [2]. Hence, the single-link reception algorithm (8.3) as shown in Figure 8.2 can be used for practical implementations. The test statistic in Algorithm (8.3) consists of summing the $N_s$ correlations $\alpha_j$ of the correlators template signal $\nu(t)$ at various time shifts with the received signal $r(t)$.

For the monocycle waveform of Figure 8.1, the optimum choice of $\delta$ is 0.156 ns. By choosing $\delta = 0.156$ ns and $T_f = 100$ ns, we achieve the results shown in Figure 8.3. For the evaluation of the system efficiency, formulas from Chapter 4 are applicable with

\[
D_r = K, \quad \left( \frac{k_{02}}{k_{01}} \right) = K \quad \text{and} \quad -g_{12} = \text{Additional Required Power (ARP)}
\]

Both $K = \text{total number of users}$ and ARP are available in Figure 8.3.
Figure 8.2 Receiver block diagram for the reception of the first user’s signal. Clock pulses are denoted by Dirac delta functions $\delta_D(\cdot)$ [31] © 2001, IEEE.

Figure 8.3 Total number of users versus additional required power (decibels) for the impulse radio example. Ideal power control is assumed at the receiver. Three different BER performance levels with the data rate set at 19.2 kb/s are considered.
8.2 THE UWB CHANNEL

8.2.1 Energy capture

In Section 8.1 a Gaussian channel was assumed. For a UWB signal, a high resolution of multipath channel is expected. In this section we discuss some characteristics of a UWB signal in such a channel.

8.2.2 The received signal model

In general, the received signal can be presented as

\[ r(u, t) = s(u_s, t) + n(u_n, t) \]  \hspace{1cm} (8.4)

where \( u \) characterizes the set of parameters defining the environment (position of the receiver in the room). The RAKE correlator structure is modeled as

\[ \sum_{i=1}^{L_\text{P}} c_i \omega(t - \tau_i) \]  \hspace{1cm} (8.5)

For experimental purposes, the pulse in Figure 8.1 that can be represented as \( \omega_{\text{rec}}(t + 1.0) = \lfloor 1 - 4\pi(t/\tau_m)^2 \rfloor \exp[ -2\pi(t/\tau_m)^2] \) with \( \tau_m = 0.78125 \) is used. The ML estimates of the amplitude vector \( \hat{c}(\tilde{u}) \) and delay vector \( \hat{\tau}(\tilde{u}) \) based on a specific observation \( r(\tilde{u}, t) \) are the values \( c \) and \( \tau \) which minimize the following mean squared error:

\[ E(\tilde{u}, L_\text{P}) = \int_0^T \left| r(\tilde{u}, t) - \sum_{i=1}^{L_\text{P}} c_i \omega(t - \tau_i) \right|^2 \, dt \]  \hspace{1cm} (8.6)

The minimum value of the above mean squared error is denoted by \( E_{\text{min}}(u, L_\text{P}) \). The energy capture, a function of \( L_\text{P} \) for each observation \( r(\tilde{u}, t) \), is defined mathematically as

\[ \text{EC} (\tilde{u}, L_\text{P}) = 1 - \frac{E_{\text{min}}(\tilde{u}, L_\text{P})}{E_{\text{tot}}(\tilde{u})} \]  \hspace{1cm} (8.7)

8.2.3 The UWB signal propagation experiment 1

A UWB signal propagation experiment performed in a typical modern office building [4] is described. The bandwidth of the signal used in this experiment is in excess of 1 GHz, resulting in a differential path delay resolution of less than a nanosecond.

The transmitter is kept stationary in the central location of the building. Multipath profiles are measured using a digital sampling oscilloscope on one floor at various locations in 14 different rooms and hallways. In each office, multipath measurements are made at 49 different locations. They are arranged spatially in a level \( 7 \times 7 \) square grid with six inch (15 cm) spacing, covering \( 3' \times 3' \) (90 \times 90 cm).

Measurements from three different offices are used in the following discussions as typical examples of propagation environments. In these offices, the receiving antennas are located 6, 10 and 17 m away from the transmitter, representing typical UWB signal transmissions characterized as a ‘high signal to noise ratio (SNR)’ environment, ‘low SNR’ environment, and ‘extremely low SNR’ environment, respectively. The transmitter and receiving antenna are located in different rooms in these examples. Detailed results of this UWB signal propagation experiment can be found in [4, 13]. The results are shown in Figure 8.4 in the form of upper and lower bound curves.
Figure 8.4 The required diversity level, $L_p$, in a UWB RAKE receiver as a function of percentage energy capture for each of the 49 received waveforms in an office representing a ‘high SNR’ environment (courtesy Moe Win, Massachusetts Institute of Technology) [4] © 1997, IEEE.

Figure 8.5 Transmitted pulse shape captured at 1 m separation from the transmit antenna.

8.2.4 UWB propagation experiment 2

The propagation experiment described in this section uses two vertically polarized diamond-dipole antennas [6], each 1.65 m above the floor and 1.05 m below the ceiling in an office/laboratory environment [7]. The equivalent received pulse at 1 m in free space can be estimated as the ‘direct path’ signal in an experiment in which there is no multipath signal, as shown in Figure 8.5.

A collection of results of recovered signal locations (delay and azimuth) is shown in Figure 8.6.
Clustering models for the indoor multipath propagation channel

A number of models for the indoor multipath propagation channel [8–12] have reported a clustering of multipath components, in both time and angle. In the model presented in [11], the received signal amplitude $\beta_{kl}$ is a Rayleigh distributed random variable with a mean square value that obeys a double exponential decay law, according to

$$\beta_{kl}^2 = \beta_{k0}^2 e^{-T_l/\Gamma} e^{-\tau_{kl}/\gamma}$$ (8.8)

where $\beta_{k0}$ is the average power of the first arrival of the first cluster, $T_l$ represents the arrival time of the $l$th cluster, and $\tau_{kl}$ is the arrival time of the $k$th arrival within the $l$th cluster, relative to $T_l$. The parameters $\Gamma$ and $\gamma$ determine the inter-cluster signal level rate of decay and the intra-cluster rate of decay, respectively. The parameter $\Gamma$ is generally determined by the architecture of the building, while $\gamma$ is determined by objects close to the receiving antenna, such as furniture. The results presented in [11] make the assumption that the channel impulse response as a function of time and azimuth angle is a separable function, or

$$h(t, \theta) = h(t) h(\theta)$$ (8.9)

from which independent descriptions of the multipath time-of-arrival and angle-of-arrival are developed. This is justified by observing that the angular deviation of the signal arrivals within a cluster from the cluster mean does not increase as a function of time.
The cluster decay rate $\Gamma$ and the ray decay rate $\gamma$ can be interpreted for the environment in which the measurements were made. For the results, presented later in this section, at least one wall separates the transmitter and the receiver. Each cluster can be viewed as a path that exists between the transmitter and the receiver, along which signals propagate. This cluster path is generally a function of the architecture of the building itself. The component arrivals within a cluster vary because of secondary effects, e.g. reflections from the furniture or other objects. The primary source of degradation in the propagation through the features of the building is captured in the decay exponent $\Gamma$. Relative effects between paths in the same cluster do not always involve the penetration of additional obstructions or additional reflections, and therefore tend to contribute less to the decay of the component signals. Results for $p(\theta)$ generated from the data in Figure 8.6 are shown in Figure 8.7.

\[
Laplacian distribution: p(\theta) = \frac{1}{\sqrt{2\sigma}} e^{-\sqrt{2\theta}/\sigma}
\]

Figure 8.7 (a) Ray arrival angles at 1° of resolution and a best fit Laplacian density with $\sigma = 38^\circ$; (b) distribution of the cluster azimuth angle of arrival, relative to the reference cluster [5] © 2002, IEEE.
Table 8.1  Channel parameters

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Gamma$</td>
<td>27.9 ns</td>
<td>33.6 ns</td>
<td>78.0 ns</td>
<td>60 ns</td>
</tr>
<tr>
<td>$\gamma$</td>
<td>84.1 ns</td>
<td>28.6 ns</td>
<td>82.2 ns</td>
<td>20 ns</td>
</tr>
<tr>
<td>$1/\Lambda$</td>
<td>45.5 ns</td>
<td>16.8 ns</td>
<td>17.3 ns</td>
<td>300 ns</td>
</tr>
<tr>
<td>$1/\lambda$</td>
<td>2.3 ns</td>
<td>5.1 ns</td>
<td>6.6 ns</td>
<td>5 ns</td>
</tr>
<tr>
<td>$\sigma$</td>
<td>37°</td>
<td>25.5°</td>
<td>21.5°</td>
<td>. .</td>
</tr>
</tbody>
</table>

*Interarrival times* are hypothesized [11] to follow exponential rate laws, given by

$$p(T_l | T_{l-1}) = \Lambda e^{-\Lambda(T_l - T_{l-1})}$$

$$p(T_{kl} | T_{k-1,l}) = \lambda e^{-\lambda(T_l - T_{l-1})}$$

where $\Lambda$ is the cluster arrival rate and $\lambda$ is the ray arrival rate. Channel parameters are summarized in Table 8.1.

### 8.2.6 Path loss modeling

In this segment we are interested in a transceiver operating at approximately 2 GHz center frequency with a bandwidth in excess of 1.5 GHz, which translates to sub-nanosecond time resolution in the CIRs.

#### 8.2.6.1 Measurement procedure

The measurement campaign is described in [14] and is conducted in a single-floor, hard-partition office building (fully furnished). The walls are constructed of drywall with vertical metal studs; there is a suspended ceiling ten feet (three metres) in height with carpeted concrete floor. Measurements are conducted with a stationary receiver and mobile transmitter, both transmit and receive antennas are five feet (1.5 metres) above the floor. For each measurement, a 300 ns time domain scan is recorded and the LOS distance from transmitter to receiver is recorded. A total of 906 profiles are included in the dataset with seven different receiver locations recorded over the course of several days. Except for a reference measurement made for each receiver location, all successive measurements are NLOS links, chosen randomly throughout the office layout, that penetrate anywhere from one to five walls. The remainder of the datapoints are taken in a similar fashion.

#### 8.2.6.2 Path loss modeling

The average path loss for an arbitrary T-R separation is expressed using the power law as a function of distance. The indoor environment measurements show that at any given $d$, shadowing leads to signals with a path loss that is lognormally distributed about the mean [15, 16]. That is:

$$PL(d) = PL_0(d_0) + 10N \log \left( \frac{d}{d_0} \right) + X_\sigma \quad (8.10)$$

where $N$ is the path loss exponent, $X_\sigma$ is a zero mean lognormally distributed random variable with standard deviation $\sigma$ (dB) and $PL_0$ is the free space path loss at reference distance, $d_0$. Some results are shown in Figure 8.8.

Assuming a simple RAKE with four correlators where each component is weighted equally, we can calculate the path loss versus distance using the peak CIR power plus RAKE gain, $PL_{\text{peak}+\text{RAKE}}$, 

for each CIR, as shown in Figure 8.8(c). The exponent $N$ obtained from performing a least squares fit is 2.5, with a standard deviation of 4.04 dB. The results for delays are shown in Figure 8.9.

### 8.2.6.3 In-home channel

For the in-home channel, Equation (8.10) can also be used to model path loss. Some results are shown in Table 8.2 [17–20].

Table 8.3 presents the results for delay spread in the in-home channel [17].

![Figure 8.8](image-url)  
(a) Peak PL vs. distance; (b) total PL vs. distance; (c) peak PL + RAKE gain vs. distance [14] © 2002, IEEE.
8.3 UWB SYSTEM WITH \(M\)-ARY MODULATION

8.3.1 Performance in a Gaussian channel

We will first assume that the transmitted pulse and the received signal are \(p_{\text{TX}}(t) = \int_{-\infty}^{t} p(\xi) d\xi\) and \(p(t) + n(t)\) respectively (we ignore effects of propagation). The effect of the antenna system in the transmitted pulse is modeled as a differentiation operation. The noise \(n(t)\) is AWGN with two-sided power density \(N_0/2\). The UWB pulse \(p(t)\) has duration \(T_p\) and energy \(E_p = \int_{-\infty}^{\infty} [p(t)]^2 dt\). The normalized signal correlation function of \(p(t)\) is

\[
\gamma_p(\tau) = \frac{1}{E_p} \int_{-\infty}^{\infty} p(t) p(t-\tau) dt > -1 \quad \forall \tau
\]

Parameter \(\gamma_{\text{min}} = \gamma_p(\tau_{\text{min}})\) is defined as the minimum value of \(\gamma_p(\tau)\), \(\tau \in (0, T_p]\). The transmitted signal is a PPM signal, and each is composed of \(N_s\) time-shifted pulses

\[
\Psi_{\text{TX}}^{(j)}(t) = \sum_{k=0}^{N_s-1} p_{\text{TX}} \left( t - kT_f - \alpha_k^j \tau_{\text{min}} \right)
\]

\(j = 1, 2, \ldots, M\). In the absence of noise, the received signals are composed of \(N_s\) time-shifted UWB pulses

\[
\Psi_j(t) = \sum_{k=0}^{N_s-1} p \left( t - kT_f - \alpha_k^j \tau_{\text{min}} \right)
\]

Each \(\Psi_j(t)\) represents the \(j\)th signal in an ensemble of \(M\) signals, each signal identified by the sequence of time shifts \(\alpha_k^j, \tau_{\text{min}} \in \{0, \tau_{\text{min}}\}\) (this choice of time shifts allows us to produce \(M\)-ary PPM signals, which are equally correlated). The \(\alpha_k^j\) is a 0, 1 pattern representing the \(j\)th cyclic shift of an
Figure 8.9 (a) RMS delay spread vs. distance; (b) RMS delay spread vs. path loss [14]
© 2002, IEEE.

$m$-sequence of length $N_s$. Since there are at most $N_s$ cyclic shifts in an $m$-sequence, we require that $2 \leq M < N_s$.

The pulse duration satisfies $T_p + \tau_{\text{min}} < T_f$, where $T_f$ is the time shift value corresponding to the frame period. Each signal $\Psi_j(t)$ has duration $\Delta = N_s T_f$ and energy $E_{\psi} = N_s E_p$. The signals in Equation (8.133) have normalized correlation values

$$
\beta_{ij} = \frac{\int_{-\infty}^{\infty} \Psi_i(t) \Psi_j(t) \, dt}{E_{\psi}} = \beta = \frac{1 + \gamma_{\text{min}}}{2}
$$

for all $i \neq j$, i.e. they are equally correlated.
Table 8.2  Statistical values of the path loss parameters

<table>
<thead>
<tr>
<th>LOS</th>
<th>NLOS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean</td>
<td>Standard deviation</td>
</tr>
<tr>
<td>PL₀ (dB)</td>
<td>47</td>
</tr>
<tr>
<td>N</td>
<td>1.7</td>
</tr>
<tr>
<td>σ (dB)</td>
<td>1.6</td>
</tr>
</tbody>
</table>

Table 8.3  Percentage of power contained in profile, number of paths, mean excess delay and RMS delay spread for 5, 10, 15, 20 and 30 dB threshold level.

<table>
<thead>
<tr>
<th>Threshold</th>
<th>50% NLOS</th>
<th>90% NLOS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>% Power</td>
<td>τₘ (ns)</td>
</tr>
<tr>
<td>5 dB</td>
<td>46.8</td>
<td>7</td>
</tr>
<tr>
<td>10 dB</td>
<td>89.2</td>
<td>27</td>
</tr>
<tr>
<td>15 dB</td>
<td>97.3</td>
<td>39</td>
</tr>
<tr>
<td>20 dB</td>
<td>99.4</td>
<td>48</td>
</tr>
<tr>
<td>30 dB</td>
<td>99.97</td>
<td>60</td>
</tr>
</tbody>
</table>

The optimum receiver is a bank of filters matched to the M signals \( \Psi_j(t) \), \( j = 1, 2, \ldots, M \). The receiver is assumed to be perfectly synchronized with the transmitter.

The union bound on the bit error probability using these equally correlated signals can be written [22]:

\[
\text{UBPb} = \frac{1}{M} \sum_{i=1}^{M} \sum_{j=1}^{M} Q \left( \sqrt{\frac{E_{\Psi}}{N_0}}(1 - \beta) \right) = \frac{M}{2} \int_{\log_2(M)SNR_b}^{\infty} \frac{\exp(-\xi^2/2)}{\sqrt{2\pi}} d\xi \quad (8.15)
\]

where

\[
\text{SNR}_b = \frac{1}{\log_2(M)} \frac{E_{\Psi}}{N_0}(1 - \beta) \quad (8.16)
\]

is the received bit SNR and \( Q(\xi) \) is the Gaussian tail function.

As an example for \( p(t) \), we consider a UWB pulse that can be modeled by properly scaling the second derivative of a Gaussian function \( \exp\left(-2\pi [t/t_n]^2\right) \). In this case, we have

\[
p_{\text{RX}}(t) = t \exp\left(-2\pi \left[ \frac{t}{t_n} \right]^2\right) \quad (8.17)
\]

\[
p(t) = \left[ 1 - 4\pi \left[ \frac{t}{t_n} \right]^2 \right] \exp\left(-2\pi \left[ \frac{t}{t_n} \right]^2\right) \quad (8.18)
\]

where the value \( t_n = 0.7531 \) ns was used to fit the model \( p(t) \) to a measured waveform \( p_{m}(t) \) from a particular experimental radio link [23]. This resulted in \( T_p \approx 2.0 \) ns.
The normalized signal correlation function corresponding to \( p(t) \) is calculated using Equation (8.11) to give:

\[
\gamma_p (t) = \left[ 1 - 4\pi \left( \frac{\tau}{T_a} \right)^2 + \frac{4\pi^2}{3} \left( \frac{\tau}{T_a} \right)^4 \right] \exp \left( -\pi \left( \frac{\tau}{T_a} \right)^2 \right)
\] (8.19)

For this \( \gamma_p (t) \), we have \( \tau_{\text{min}} = 0.4073 \text{ ns} \) and \( \gamma_{\text{min}} = -0.6183 \), so \( \beta = 0.1909 \) in Equation (8.14). Both \( p(t-T_p/2) \) and \( \gamma_p (\tau) \) are depicted in Figure 8.10. Figure 8.11 shows the spectrum of the impulse \( p(t) \). The 3 dB bandwidth of the pulse is close to 1 GHz. The center frequency is around 1.1 GHz.

Figure 8.10 (a) The pulse \( p(t-T_p/2) \) as a function of time \( 0 \leq t \leq 4 \text{ ns} \); (b) the signal autocorrelation \( \gamma_p (\tau) \) as a function of time shift \( -2 \leq \tau \leq 2 \text{ ns} \) [21] © 2001, IEEE.

Figure 8.11 The magnitude of the spectrum of the pulse \( p(t) \).
The specific values of $N_s$ and $T_f$ do not affect SNRb, as long as $M < N_s$ and $T_p + \tau_{\min} < T_f$. Hence, we set arbitrarily $T_f = 500$ ns and $N_s > 1000$ [21]. The BER in AWGN can now be calculated using UBPb from Equation (8.15). Results for different values of $M$ are shown in Figure 8.12. Values as large as $M = 128$ are easily obtained with the PPM signal design in Equation (8.13), allowing us to exploit the benefits of $M$-ary modulation without an excessive increase in the complexity of the receiver [24].

### 8.3.2 Performance in a dense multipath channel

In this section we discuss performance of $M$-ary UWB signals in a dense multipath channel with AWGN. The channel can be, for example, an indoor radio channel as discussed in Section 8.2. In the analysis, we will assume that the transmitter is placed at a certain fixed location, and the receiver is placed at a variable location denoted $u_0$. The transmitted pulse is the same pulse $p_{\text{TX}}(t)$ as in the AWGN case, and the received UWB signal is $\sqrt{E_a} \hat{p}(u_0, t) + n(t)$. The pulse $\sqrt{E_a} \hat{p}(u_0, t)$ is a multipath spread version of $p(t)$ received at position $u_0$ with average duration $T_a \gg T_p$. The pulse has ‘random’ energy $\bar{E}(u_0) \equiv E_a \bar{\alpha}^2(u_0)$, where $E_a$ is the average energy and

\[
\bar{\alpha}(u_0) \equiv \int_{-\infty}^{\infty} [\hat{p}(u_0, t)]^2 \, dt
\]

(8.20)
is the normalized energy. The pulse has normalized signal correlation

\[
\bar{\gamma}(u_0, \tau) \equiv \frac{\int_{-\infty}^{\infty} \hat{p}(u_0, t) \hat{p}(u_0, t - \tau) \, dt}{\int_{-\infty}^{\infty} [\hat{p}(u_0, t)]^2 \, dt}
\]

(8.21)
The transmitted signals are the same $\Psi_{\text{TX}}^{(j)}(t)$ as given in Equation (8.12). In the absence of noise, the received signals are composed of $N_s$ time-shifted UWB pulses

\[
\tilde{\Psi}_j(u_0, t) = \sum_{k=0}^{N_s-1} \sqrt{E_a} \hat{p} \left( u_0, t - kT_f - \alpha_j^k \tau_{\min} \right)
\]

(8.22)
for \( j = 1, 2, \ldots, M \). The UWB PPM signal \( \tilde{\Psi}_j(u_0, t) \) is a multipath spread version of \( \Psi_j(t) \) received at position \( u_0 \). Assume that \( \tilde{\Psi}_j(u_0, t) \) has fixed duration \( T_s \approx N_s T_f \), provided that \( T_a + \tau_{\min} < T_f \). The signals in Equation (8.22) have ‘random’ energy

\[
\tilde{E}_\Psi(u_0) = \int_{-\infty}^{\infty} [\tilde{\Psi}_j(u_0, \xi)]^2 d\xi = \tilde{E}_\psi \tilde{\alpha} u_0
\]

for \( j = 1, 2, \ldots, M \), where \( \tilde{E}_\psi = N_s E_a \) is the average energy. The signals in Equation (8.22) have normalized correlation values

\[
\tilde{\beta}_{ij}(u_0) \triangleq \frac{\int_{-\infty}^{\infty} \tilde{\Psi}_i(u_0, \xi) \tilde{\Psi}_j(u_0, \xi) d\xi}{\tilde{E}_\Psi(u_0)} = \tilde{\beta}(u_0) = \frac{1 + \tilde{\gamma}(u_0, \tau_{\min})}{2}
\]

for all \( i \neq j \), i.e. they are equally correlated. The multipath effects change with the particular position \( u_0 \), and therefore the \( M \)-ary set of received signals \( \{\tilde{\Psi}_j(u_0, t)\}_{j=1}^M \) also changes with the particular position \( u_0 \).

### 8.3.3 Receiver and BER performance

Conditioned on a particular physical location \( u_0 \), the optimum receiver (matched filter) is a kind of perfect Rake receiver that is able to construct a reference signal \( \tilde{\Psi}_j(u_0, T_s - t) \) that is perfectly matched to the signal received \( \tilde{\Psi}_j(u_0, t) \) over the multipath conditions at that location \( u_0 \). We will assume that the receiver is perfectly synchronized with the transmitter. Performance analysis for the perfect Rake receiver can be calculated using standard techniques, conditioned on a particular physical location \( u_0 \), the union bound on the bit error probability using these equally correlated signals can be written

\[
\text{UBP}_b(u_0) = \frac{M}{2} \int_{-\infty}^{\infty} \frac{\exp(-\xi^2/2)}{\sqrt{2\pi}} d\xi
\]

where

\[
\text{SNR}_b(u_0) = \frac{1}{\log_2(M) E_\Psi(u_0)} \frac{E_\Psi(u_0)}{N_0} (1 - \tilde{\beta}(u_0)) = \frac{1}{\log_2(M) E_\Psi \tilde{\alpha}^2(u_0)} \frac{E_\Psi \tilde{\alpha}^2(u_0)}{N_0} (1 - \tilde{\beta}(u_0))
\]

is the received bit SNR [25].

### 8.3.4 Time variations

The \( \tilde{\beta}(u_0) \) value accounts for changes in the correlation properties of the received signals. These changes in \( \tilde{\beta}(u_0) \) translate into changes in the Euclidean distance between signals. Therefore, the \( (1 - \tilde{\beta}(u_0)) \) value accounts for energy variations at the output of the perfect matched filter due to distortions in the shape of the signal correlation function caused by multipath. The \( \tilde{\alpha}^2(u_0) \) value accounts for variations in the received signal energy due to fading caused by multipath. The average performance can be obtained by taking the expected value \( E_u(\cdot) \) over all values of \( u_0 \)

\[
\text{UBP}_b \left( \frac{E_\Psi}{N_0} \right) = E_u \{\text{UBP}_b(u)\}
\]

where

\[
\left( \frac{E_\Psi}{N_0} \right) \triangleq E_u \{\text{SNR}_b(u)\}
\]

is the average received bit SNR. This BER analysis provides a theoretical matched filter bound for the best performance attainable when the multipath channel is perfectly estimated.
Instead of using a channel model, the calculations based on the received waveforms can be used. This is possible since the expected value in Equation (8.27) is taken with respect to the quantities $\hat{\alpha}^2(u_0)$ and $\beta(u_0)$. Histograms of these quantities can be calculated for a particular indoor channel environment and a first approximation can be obtained using the sample mean value

$$\text{UBPb} \left( \frac{\bar{E}_b}{N_0} \right) \approx \frac{1}{u_*} \sum_{u_0=1}^{u_*} \text{UBPb} (u_0) \quad (8.27a)$$

This calculation represents a rough approximation to the performance of UWB signals in the presence of dense multipath in a particular indoor radio environment. The histograms for $\hat{\alpha}^2(u_0)$ and $\beta(u_0)$ can be derived from their definitions in Equations (8.20) and (8.24) using the ensemble of pulse responses

$${\bar{\rho}}(u_0,t), \ u_0 = 1, 2, \ldots, u_*$$

taken in a measurement experiment in the multipath channel of interest.

### 8.3.5 Performance example

The channel responses $\bar{\rho}(u_0,t)$ were measured in eight different rooms and hallways in a typical office building described in Section 8.2 (see also [21]). In every room and hallway, 49 different locations are arranged spatially in a $7 \times 7$ square grid with six inch spacing. At every location $u_0$, the $T_a = 300$ ns-long pulses $\bar{\rho}(u_0,t)$ are recorded, keeping the transmitter, the receiver and the environment stationary.

The UWB transmitter is placed at a fixed location inside the building. It consists of a step recovery diode-based pulser connected to a UWB omnidirectional antenna. The pulser produces a train of UWB ‘Gaussian monocycles’ $p_{\text{TX}}(t)$. The train of $p_{\text{TX}}(t)$ is transmitted as an excitement signal to the propagation channel. The train has a repetition rate of 500 ns with a tightly controlled average monocycle-to-monocycle interval. The clock driving the pulser has resolution in the order of picoseconds.

The $\bar{\rho}(u_0,t)$ represents the convolution of $p_{\text{TX}}(t)$ with the channel impulse response at location $u_0$. The 500 ns repetition rate is long enough to make sure that pulse responses $\bar{\rho}(u_0,t)$ corresponding to adjacent impulses $p_{\text{TX}}(t)$ do not overlap. The receiver consists of a UWB antenna and a low-noise amplifier. The output of this amplifier is captured using a high-speed digital sampling scope. The scope takes samples in windows of 50 ns at a sampling rate of 20.5 GHz. Noise in the measured $\bar{\rho}(u_0,t)$ is reduced by averaging 32 consecutive received pulses measured at exactly the same location $u_0$. These samples are sent to a data storage and processing unit.

A total of $u_* = 392$ channel pulse responses $\bar{\rho}(u_0,t)$ were measured. An equal number of normalized energy values $\hat{\alpha}^2(u_0)$ and normalized correlation functions $\bar{\gamma}(u_0,\tau)$ were calculated using Equations (8.20) and (8.21), respectively. Figure 8.13 shows histograms of $\hat{\alpha}^2(u_0)$, $\beta(u_0)$ and the product $\hat{\alpha}^2(u_0) \times (1 - \beta^2(u_0))$. The measured $\bar{\rho}(u_0,t)$ have $T_a \approx 300$ ns. The rest of the parameter values are the same as those used in the AWGN case, i.e. $\tau_{\text{min}} = 0.4073$ ns, $T_t = 500$ ns and $N_t > 1000$. With these values, the conditions $M < N_t$ and $T_a + \tau_{\text{min}} < T_t$ will be satisfied.

For the results from Figure 8.13, the average BER curves, Equation (8.27), are shown in Figure 8.14.

### 8.4 M-ARY PPM UWB MULTIPLE ACCESS

The signal format introduced in Section 8.3 is now used for the multiple access system [26]. The TH PPM signal conveying information exclusively in the time shifts is now represented as

$$x^{(v)}(t) = \sum_{k=0}^{N_t-1} \omega(t - kT_t - c^{(v)}_k T_e - \delta_k u^{(v)}_{(k/N_t)}) \quad (8.29)$$
Figure 8.13 (a) Normalized correlation function $\tilde{\gamma}(u_0, t)$ of the pulse $\tilde{p}(u_0, t)$ in Figure 8.5; (b) a closer view of the correlation in (a). The spreading caused by multipath is notorious. The long tails in the correlation function are the effect of the pulse spreading [21]. The histogram of the normalized values of (c) the received energy $\tilde{\alpha}^2(u_0)$; (d) the correlation value $\tilde{\beta}(u_0)$; and (e) the product $\tilde{\alpha}^2(u_0) \times (1 - \tilde{\beta}^2(u_0))$. The ordinate represents appearance frequency, and the abscissa represents the value of the parameter. The size of the sample is $u_0 = 392$ [21] © 2001, IEEE.
Figure 8.13 (Cont.).

Figure 8.14 $\text{UBPb}(E_{\psi}/N_0)$ in Equation (8.27a); $M = 2, 4, 8, 16, 32, 64$ and $132$ signals.

The superscript $(\nu)$, $1 \leq \nu \leq N_u$, indicates user-dependent quantities, where $N_u$ is the number of simultaneous active users. The index $k$ is the number of time hops that the signal $x^{(\nu)}(t)$ has experienced, and also the number of pulses that have been transmitted. $T_1$ is the frame (pulse repetition) time and equals the average time between pulse transmissions. The notation $[q]$ stands for the integer part of $q$. The $\{c_k^{(\nu)}\}$ is the pseudorandom time-hopping sequence assigned to user $\nu$. It is periodic with period $N_p$ (i.e. $c_k^{(\nu)} = c_l^{(\nu)}$, for all $k, l$ integers) and each sequence element is an integer in the range $0 \leq c_k^{(\nu)} \leq N_h$. For a given time shift parameter $T_c$, the time-hopping code provides an additional time shift to the pulse in every frame, each time shift being a discrete time value $c_k^{(\nu)}T_c$, with $0 \leq c_k^{(\nu)}T_c \leq N_hT_c$. The time shift corresponding to the data modulation is

$$\delta_k^{(\nu)} \in \{\tau_1 = 0 < \tau_2 < \cdots < \tau_\eta\}$$  \hspace{1cm} (8.30)
with $\eta \geq 2$ an integer. The data sequence $\{d_m^{(v)}\}$ of user $v$ is an $M$-ary symbol stream, $1 \leq d_m^{(v)} \leq M$, that conveys information in some form. The system under study uses fast time-hopping, which means that there are $N_s \geq 1$ pulses transmitted per symbol. The data symbol changes only every $N_s$ hops. Assuming that a new data symbol begins with pulse index $k = 0$, the data symbol index is $[k/N_s]$.

We will use the following notation

$$H_m^{(v)}(t) \doteq \sum_{k=mN_s}^{(m+1)N_s-1} T_c e^{(v)}_k p(t - kT_t)$$

and

$$S_i(t) \doteq \sum_{k=0}^{N_s-1} \omega(t - kT_t - \delta_i^k)$$

for $i = 1, 2, \ldots, M$, then Equation (8.29) can be written

$$x^{(v)}(t) = \sum_{m=0}^{\infty} S_{m,i}^{(v)}(t - mN_s T_t - H_m^{(v)}(t)) \doteq \sum_{m=0}^{\infty} X_{m,i}^{(v)}(t)$$

The signal $S_i(t)$ in Equation (8.32) is the received signal corresponding to the transmitted signal $\int_{-\infty}^{\infty} S_i(\xi) d\xi = \sum_{k=0}^{N_s-1} \omega_{TX}(t - kT_t - \delta_i^k)$. With this notation, the signal correlation function is defined as

$$R_{ij} \doteq \int_{-\infty}^{\infty} X_{m,i}^{(v)}(\xi) X_{m,j}^{(v)}(\xi) d\xi$$

$$= \int_{-\infty}^{\infty} S_i(\xi) S_j d\xi = E_s \sum_{k=0}^{N_s-1} \gamma_s (\delta_i^k - \delta_j^k)$$

since the pulses are non-overlapping. The energy in the $i$th signal $X_{m,i}^{(v)}$ is $E_S = R_{ii} = N_s E_s$, and the normalized correlation value is

$$\alpha_{i,j} \doteq \frac{R_{ij}}{E_S} = \frac{1}{N_s} \sum_{k=0}^{N_s-1} \gamma_s (\delta_i^k - \delta_j^k) \geq \gamma_{\text{min}}$$

### 8.4.1 $M$-ary PPM signal sets

The PPM signal $S_i(t)$ in Equation (8.32) represents the $i$th signal in an ensemble of $M$ information signals, each signal completely identified by the pulse shape $\omega(t)$ and the sequence of time shifts $\{\delta_i^k\}, k = 0, 1, 2, \ldots, N_s - 1$. The most interesting $M$-ary PPM sets from the practical point of view are orthogonal (OR), equally correlated (EC), and $N$-orthogonal (NO) signal sets.

In general, these signal designs have the property that the structure of the $M$-ary autocorrelation matrix is preserved for different $\omega(t)$. This is important because $\omega(t)$ is, in general, a non-standard pulse, and these signal designs reduce the dependence of the MA performance on the shape of $\omega(t)$. The time shift patterns defining each $M$-ary PPM signal set and their respective correlation properties are studied in detail in [27–29] and are summarized in Table 8.4. In the EC case, the $a_k^1$ is a 0, 1 pattern representing the $i$th cyclic shift, $i = 1, 2, \ldots, M$, of an $m$-sequence [31] of length $N_s = 2^m - 1, m \geq 1$, and $N_s \geq M$.

The $M$-ary correlation receiver, shown in Figure 8.15, consists of $M$ filters matched to the signals $X_{m,j}^{(1)}(t - \tau^{(1)})$, $j = 1, 2, \ldots, M$, $t \in T_m$, followed by samplers and a decision circuit that selects the maximum among the decision variables

$$\int_{\tau} \{ r(t) X_{m,j}^{(1)}(t - \tau^{(1)}) \} dr, \quad j = 1, 2, \ldots, M$$

(8.35)
Table 8.4 Time-shift patterns and normalized correlation values of the $M$-ary PPM signals under study. Orthogonal (OR), equally correlated (EC), $N$-orthogonal design 1 (NO1) and $N$-orthogonal design 2 (NO2) [26] © 2001, IEEE

<table>
<thead>
<tr>
<th>Type of signal</th>
<th>Time shift pattern ${\delta^k_i}$</th>
<th>Normalized correlation coefficients $\alpha_{ij}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>OR</td>
<td>$\delta^k_i = [(k + i - 1) \mod M] T_{OR}$</td>
<td>$\delta^k_i = a^k_i \tau_2$</td>
</tr>
<tr>
<td></td>
<td>$T_{OR} = 2T_{\omega}$</td>
<td>$\alpha_{ij}^{(OR)} = \begin{cases} 1, &amp; i = j \ 0, &amp; i \neq j \end{cases}$</td>
</tr>
<tr>
<td>EC</td>
<td>$\delta^k_i = \tau_2 \in (0, T_{\omega})$</td>
<td>$\alpha_{ij}^{(EC)} = \begin{cases} 1, &amp; i = j \ \lambda, &amp; i \neq j \end{cases}$</td>
</tr>
<tr>
<td>NO1</td>
<td>$</td>
<td>\lambda</td>
</tr>
<tr>
<td></td>
<td>$I = i - \left[\frac{i - 1}{2}\right] 2$</td>
<td>$J = j - \left[\frac{i - 1}{2}\right] 2$</td>
</tr>
<tr>
<td></td>
<td>$\hat{I} \equiv \frac{M}{2}$</td>
<td>$\hat{L} \equiv \frac{L}{2}$</td>
</tr>
<tr>
<td></td>
<td>$T_{NO1} \equiv \tau_2 + T_{OR}$</td>
<td>$\alpha_{ij}^{(NO1)} = \begin{cases} 1, &amp; i = j \ 0, &amp; \left[\frac{i - 1}{2}\right] \neq \left[\frac{j - 1}{2}\right] \end{cases}$</td>
</tr>
<tr>
<td></td>
<td>$0 = \tau_1 &lt; \tau_2 &lt; T_{\omega}$</td>
<td>$\beta_{ij} = \gamma_\omega (\tau_J - \tau_I)$</td>
</tr>
<tr>
<td>NO2</td>
<td>$\delta^k_i = a^k_i \tau_2 + [(k + 2\hat{I}) \mod L] T_{NO2}$</td>
<td>$\alpha_{ij}^{(NO2)} = \begin{cases} 1, &amp; i = j \ 0, &amp; \left[\frac{i - 1}{2}\right] \neq \left[\frac{j - 1}{2}\right] \end{cases}$</td>
</tr>
<tr>
<td></td>
<td>$T_{NO2} \equiv \tau_2 + T_{OR}$</td>
<td>$\lambda, \left[\frac{i - 1}{2}\right] = \left[\frac{j - 1}{2}\right]$</td>
</tr>
<tr>
<td></td>
<td>$0 = \tau_1 &lt; \tau_2 &lt; T_{\omega}$</td>
<td></td>
</tr>
</tbody>
</table>

Figure 8.15 $M$-ary correlator receiver for the TH PPM signals.
8.4.2 Performance results

In this subsection, we illustrate the theoretical MA performance of this system for a specific \( \omega(t) \) under perfect power control (i.e. \( A^{(v)} = A^{(1)} \) for \( v = 2, 3, \ldots, N_u \)). The system parameters are the same as in [26]. The \( \omega(t) \) considered here is the second derivative of a Gaussian function

\[
\omega(t) = \left[ 1 - 4\pi \left( \frac{t}{t_n} \right)^2 \right] \exp \left( -2\pi \left( \frac{t}{t_n} \right)^2 \right)
\]

(8.36)

where the value \( t_n \) is used to fit the model \( \omega(t) \) to a measured waveform from a particular experimental radio link. The normalized signal correlation function corresponding to \( \omega(t) \) in Equation (8.36) is

\[
\gamma_{\omega}(\tau) = \left[ 1 - 4\pi \left( \frac{\tau}{t_n} \right)^2 + \frac{4\pi^2}{3} \left( \frac{\tau}{t_n} \right)^4 \right] \exp \left( -\frac{\pi}{t_n} \left( \frac{\tau}{t_n} \right)^2 \right)
\]

(8.37)

In this case, \( T_\omega \) and \( \tau_{\min} \) depend on \( t_n \), and \( \gamma_{\min} = -0.6183 \) for any \( t_n \). Using \( t_n = 0.4472 \) ns, we get \( T_\omega \approx 1.2 \) ns and \( \tau_{\min} = 0.2149 \) ns. Figure 8.16 depicts \( \omega(t - T_\omega/2), \gamma_{\omega}(\tau) \) and the spectrum of \( \omega(t) \). The 3 dB bandwidth of \( \omega(t) \) is in excess of 1 GHz.

Given \( \omega(t) \), the signal design is complete when we specify \( N_s, T_1 \) and \( \delta_f(t) \) in Table 8.4. For OR signals \( T_{OR} = 2T_\omega \), for EC signals \( \tau_1 = 0 \) and \( \tau_2 = \tau_{\min} \), for NO1 signals \( \tau_1 = 0, \tau_2 = \tau_{\min} \) and \( T_{NO1} = \tau_{\min} + 2T_\omega \), and for NO2 signals \( \tau_1 = 0, \tau_2 = \tau_{\min} \) and \( T_{NO2} = \tau_{\min} + 2T_\omega \).

To choose a value for \( T_1 \), we need \( 0 < NBT_c < (T_1/2) - 2(T_\omega + \tau_0) \). Also notice that \( \tau_0 = 16T_{OR} \) for \( M = 16 \) (the maximum value of \( M \), in this example). By choosing \( T_1 = 100 \) ns, we have that \( 0 < NBT_c < 16 \) ns.

To choose a value for \( N_s \), notice that the \( M \)-ary PPM signal designs considered in [27] require that \( N_s = 1/(R_bT_1) = \log_2(M)/R_bT_1 = \log_2(M)N_u^{(2)} \). Hence, for a fixed \( T_1 \), the value of \( N_s \) is determined by \( R_b \). However, in particular, the EC PPM signal design additionally requires \( N_s \geq M \).

Combining these two requirements on \( N_s \) we have that, in the EC PPM case, both \( R_b \) and \( N_s \) satisfy the relation \( \log_2(N_s)/N_u \geq R_bT_1 \). In this example, we use \( R_b = 100 \) kb/s, \( N_u^{(2)} = 100, 2 \leq M \leq 16, \) and \( N_s = \log_2(M)100 \); hence, \( \log_2(N_s)/N_u \geq R_bT_1 \) holds, and both relations \( N_s \geq M \) and \( N_s = \log_2(M)/R_bT_1 \) are satisfied.

For a single-link communications bit, \( E_b/N_0 = 14.30 \) dB, \( \text{SNR}_{\text{OR}}(1) = 14.30 \) dB [27], and \( \text{SNR}_{\text{TSK}}(1) = 16.40 \) dB. The BER results are given in Figure 8.17. If we define SNR(1) as the required signal to noise ratio for a certain BER with only one user in the network and \( \text{SNR}(N_u) \) as the required initial signal to noise ratio with one user which, after adding \( N_u - 1 \) additional users, will still give SNR(1), then we define the degradation factor \( DF \) as \( DF = \text{SNR}(N_u)/\text{SNR}(1) \). In the next example we demonstrate some results for \( N_u(DF) \) when using the pulse \( \omega(t) \) in Equation (8.36) with \( t_n = 0.4472 \) ns, \( T_\omega = 1.2 \) ns, \( \tau_{\min} = 0.2149 \) ns, \( \tau_1 = 0, \tau_2 = \tau_{\min} \), and \( T_1 = 100 \) ns. Figure 8.18 shows \( N_u(DF) \) for different values of \( DF \) using \( R_b = 100 \) kb/s, \( N_u^{(2)} = 100, 2 \leq M \leq 256, \) and \( N_s = \log_2(M)100 \). Notice that \( \log_2(N_u)/N_u \geq R_bT_1 \) still holds for \( M = 256 \).

From \( N_u(DF) \) we can also find \( R_b(DF) \) for a particular value of \( N_u \). Figure 8.19 shows \( R_b(DF) \) for different values of \( DF \) using \( N_u = 1000 \) active users.

The values of the upper bound on maximum capacity \( C_{\text{sup}} \) are given in Table 8.5. The upper bound on maximum data rate is shown in Figure 8.20.

8.5 CODED UWB SCHEMES

We can consider the simple time-hopping spread spectrum from Section 8.4 as a coded system in which a simple repetition block code with rate \( 1/N_s \) is used. As outlined in Chapter 2 the efficiency of the repetition code is very low. Thus, by applying a near optimal code instead of the above simple repetition code, we should expect the system performance to improve significantly. In [32, 33], a class of low-rate superorthogonal convolutional codes that have near optimal performance is introduced. In a superorthogonal code with constraint length \( K \), the rate is equal to \( 1/2^{K-2} \). Since in the TH CDMA (UWB) system, \( N_s \) pulses are sent for each data bit, we must set \( 2^{K-2} = N_s \) or \( K = 2 + \log_2 N_s \). The
Figure 8.16 (a) Pulse $\omega(t - T_\omega/2)$ as a function of time $t$; (b) signal autocorrelation function $\gamma_\omega(\tau)$ as a function of time shift $\tau$; (c) magnitude of the spectrum of the pulse $\omega(t)$ [26] © 2001, IEEE.

location of each pulse in each frame is determined by the user-dedicated pseudorandom sequence, along with the code symbol corresponding to that frame.

Decoding is performed using the Viterbi algorithm. The state diagram of this decoder consists of $2^{K-1}$ states. Two branches, corresponding to bit zero and bit one, exit from each state in the trellis diagram. To update the state metrics, it is first necessary to calculate the branch metrics,
Figure 8.17 Base 10 logarithm of the probability of bit error as a function of $N_u$ for different values of $M$, using $R_b = 100$ kb/s. (a) OR PPM signals, $\text{SNR}_{\text{OR}}(1) = 14.30 \text{ dB}$; (b) EC PPM signals, $\text{SNR}_{\text{EC}}(1) = 13.39 \text{ dB}$; (c) NO PPM signals, design 1, $\text{SNR}_{\text{OR}}(1) = 14.30 \text{ dB}$ and $\text{SNR}_{\text{TSK}}(1) = 16.40 \text{ dB}$; (d) NO PPM signals, design 2, $\text{SNR}_{\text{OR}}(1) = 14 : 30 \text{ dB}$.

Figure 8.18 Number of simultaneous active links (users) $N_u(\text{DF})$ for EC PPM signals for $2 \leq M \leq 256$ with $P_e(1) = \text{UBP}_{b}^{(\text{EC})}(1) \approx 10^{-6}$ and $R_b = 100$ kb/s.
ULTRA WIDE BAND RADIO

Figure 8.19 Data transmission rate per user $R_b(\text{DF})$ for EC PPM signals for $2 \leq M \leq 256$ with $P_e(1) = \text{UBF}_b^{(\text{EC})}(1) \approx 10^{-6}$ and $N_u = 1000$ active users.

Table 8.5 Values of $C_{\text{sup}}$ in (Gb/s) calculated using three different pulse widths. Also included are the values of $N_{\text{sup}}$ for $R_b = 100$ kb/s [26]

<table>
<thead>
<tr>
<th>Set I of parameters</th>
<th>Set II of parameters</th>
<th>Set III of parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t_n = 0.2877$ ns</td>
<td>$t_n = 0.4472$ ns</td>
<td>$t_n = 0.7531$ ns</td>
</tr>
<tr>
<td>$T_\omega = 0.75$ ns</td>
<td>$T_\omega = 1.2$ ns</td>
<td>$T_\omega = 2.0$ ns</td>
</tr>
<tr>
<td>$\tau_{\text{min}} = 0.1556$ ns</td>
<td>$\tau_{\text{min}} = 0.2419$ ns</td>
<td>$\tau_{\text{min}} = 0.4073$ ns</td>
</tr>
<tr>
<td>$C_{\text{sup}}^{(I)} = 3.6394$ (Gbps)</td>
<td>$C_{\text{sup}}^{(II)} = 2.3412$ (Gbps)</td>
<td>$C_{\text{sup}}^{(III)} = 1.3903$ (Gbps)</td>
</tr>
<tr>
<td>$N_{\text{sup}}^{(I)} = 36394$ (users)</td>
<td>$N_{\text{sup}}^{(II)} = 23412$ (users)</td>
<td>$N_{\text{sup}}^{(III)} = 13903$ (users)</td>
</tr>
</tbody>
</table>

using the received signal $r(t)$. For this purpose, in each frame $j$, the quantity:

$$\alpha_j \doteq \int_{t_1+jT_f}^{t_1+(j+1)T_f} r(t) \nu(t - \tau_1 - jT_f - c_j^{(1)}T_f) \, dt$$

(the pulse correlator output) is obtained. Because of the special form of the Hadamard–Walsh sequence that is used in the structure of superorthogonal codes, the branch metrics can be simply evaluated based on the outputs of pulse correlators $\alpha_j$ [33]. This is also elaborated in Chapter 4 for space–time codes from orthogonal designs. The processing complexity of this decoder grows only linearly with $K$ (or logarithmically with $N_s$); the required memory, however, grows exponentially with $K$ (or equally linearly with $N_s$) [33]. Since in time-hopping, spread spectrum applications, the value of $K$ is relatively low (the typical value is in the range 3–12), the system can be considered to be practical.
Figure 8.20 Upper bound on the bit transmission rate per user \( R_{sup}(N_u) \) in b/s, calculated using the sets I, II and III of parameters in Table 8.5.

### 8.5.1 Performance

The path generating function of the code for a superorthogonal code is computed as \([33, 34]\):

\[
T_{SO}(\gamma, \beta) = \frac{\beta W^{K+2}(1 - W)}{1 - W[1 + \beta(1 + W^{K-3} - 2W^{K-2})]} \tag{8.38}
\]

in which \( W = \gamma^{2K-3} \) and \( K \) is the constraint length of the code. Expanding the above expression, we get a polynomial in \( \gamma \) and \( \beta \). The coefficient and the powers of \( \gamma \) and \( \beta \) in each term of the polynomial indicate the number of paths and output–input path weights respectively. The free distance of this code is obtained from the first term of the expansion as \( d_f = 2^{K-3}(K + 2) = N_s(\log_2 N_s + 4)/2 \).

### 8.5.2 The uncoded system as a coded system with repetition

In this case, the distance will be \( N_s \). Comparing the free distances of these two schemes, it is clear that the coded scheme outperforms the uncoded scheme significantly.

An upper bound on the probability of error per bit for a memoryless channel is obtained using the union bound as follows:

\[
P_b < \left. \frac{dT_{SO}(W, \beta)}{d\beta} \right|_{\beta=1} = \frac{W^{K+2}}{(1 - 2W)^2} \left( \frac{1 - W}{1 - W^{K-2}} \right)^2 \tag{8.39}
\]

where \( W = Z^{2K-3} \). The parameter \( Z \) is calculated from the Bhattacharyya bound as

\[
Z = \int_{-\infty}^{\infty} \sqrt{p_0(y) p_1(y)} \, dy \tag{8.40}
\]

where \( p_0(y) \) and \( p_1(y) \) are pdfs of the pulse correlator output conditioned on the input symbol being zero and one, respectively.
Figure 8.21 Probability of bit error as a function of number of users for synchronous uncoded and coded (upper bound) schemes in exact and Gaussian cases (a) at $R_s = 5 \text{ Mb/s}$ ($N_s = 2$); (b) at $R_s = 1.25 \text{ Mb/s}$ ($N_s = 8$); (c) at $R_s = 325 \text{ kb/s}$ ($N_s = 32$); (d) at $R_s = 78.1 \text{ Mb/s}$ ($N_s = 128$).

A lower bound on the probability of error per bit is obtained by considering only the first term of the path generating function, Equation (8.38), The result is

$$P_b \geq P_{df}$$

where $P_{df}$ is the probability of pairwise error in favor of an incorrect path that differs in $d_f$ symbols from the correct path over the unmerged span in the trellis diagram.

For $\omega_{nc}(t + T_w/2) = [1 - 4\pi(t/\tau_m)^2] \exp(-2\pi(t/\tau_m)^2)$, $\tau_m = 0.2877$ and $\delta$ and $T_f$ set to 0.156 and 100 ns respectively, the performance curves are given in Figures 8.21 and 8.22 for different $R_s$.

### 8.6 MULTIUSER DETECTION IN UWB RADIO

In this section we consider a system based on using orthogonal codes in a synchronous or quasi-synchronous context, coupled with TDD for transmissions through frequency selective channels. This is achieved by assigning to each user different orthogonal time-hopping sequences and designating two time slots for transmission: one for the uplink and one for the downlink. Binary PPM modulation from Section 8.5 is used with eight users ($N'_c = 8$) and two symbols per burst ($K = 2$). As per [35], frame duration $T_f = 100 \text{ ns}$ and there is a maximum delay spread equal to 100 ns.

Each symbol is repeated $N_f$ frames. A time guard of duration $T_c$ is used, hence, $N_g = 1$. The chip duration $T_c = T_f/(N'_c + N'_g) = 11.11 \text{ ns}$. The sampling rate is equal to $t_c$ which leads to a
The channel model from Section 8.2 is used. This model is based on clusters of rays. The received signal is composed of attenuated and delayed versions of the transmitted signal arriving in clusters. The times of arrival are modeled as a Poisson process and the amplitudes as Gaussian. For the simulations, the receivers will be assumed to be synchronized on the strongest path. In the PPM-IRMA system, the received signal is the second derivative of the Gaussian function \( \sqrt{\tau/3}(2/\pi)^{1/4}\exp(-t^2/\tau^2) \) (normalized to have \( r_\omega(0) = 1 \)); hence, we have \( r_\omega(t) = \exp(-t^2/(2\tau^2))(1 - 2(t/\tau)^2 + (t/\tau)^4/3) \) where \( r_\omega(t) \) is the correlation function of \( \omega(t) \) and the parameter \( \tau = 0.1225 \) ns is adjusted to yield a pulse width equal to 0.7 ns.

![Graph showing probability of bit error as a function of number of users for asynchronous uncoded and coded (upper bound) schemes in exact and Gaussian cases.](image)

Figure 8.22 Probability of bit error as a function of number of users for asynchronous uncoded and coded (upper bound) schemes in exact and Gaussian cases (a) at \( R_s = 5 \) Mb/s \( (N_s = 2) \); (b) at \( R_s = 2.5 \) Mb/s \( (N_s = 4) \); (c) at \( R_s = 1.25 \) Mb/s \( (N_s = 8) \).
Figure 8.22 (Cont.).

For multiuser detection schemes (MF, zero forcing (ZF)-decorrelator and MMSE), based on the same logic as in Chapter 5, the results are shown in Figures 8.23 and 8.24. Details of the derivation of the detector transfer function are given in [36].

8.7 UWB WITH SPACE–TIME PROCESSING

8.7.1 Signal model

In this segment, the Generalized Gaussian Pulse (GGP) $\Omega(t)/E_0$ shown in Figure 8.25 will be used. The self-steering array beamforming system for UWB impulse waveforms is shown in Figure 8.26.

The response of the $i$th sensor to the received wavefront can be expressed in terms of the voltage signal as

$$v_i(t, \phi) = \Omega(t - \tau_i(\phi)) = \frac{E_0}{(1 - \alpha)} \left(\exp \left\{-4\pi \left[\frac{(t - \tau_i(\phi))}{\Delta T}\right]^2\right\} - \alpha \exp \left\{-4\pi \left[\alpha (t - \tau_i(\phi)) / \Delta T\right]^2\right\}\right)$$

The relative time delay $\tau_i(\phi)/\Delta T$ is a function of the angle of incidence $\phi$ and the distance $d$ between adjacent sensors:

$$\tau_i(\phi) / \Delta T = \frac{id}{c\Delta T} \sin \phi = \frac{i}{2m} \rho \sin \phi$$

$$\rho = 2md/c\Delta T = L/c\Delta T = L\Delta f / c$$

In order to achieve electronic beamsteering for enhancing the quality of signal reception from a desired look direction, e.g. $\phi = \phi_0$ the variable delay circuit VDC$_i$ applies to the incoming signal $Y_i(t)$ a time delay $\tau_i = (id/c) \sin \phi_0$. In analogy to Equation (8.44), the relative time delay $\tau_i / \Delta T$
Figure 8.23 Comparing the different receivers (MF, ZF, MMSE) for the binary PPM-IRMA scheme with $K = 2$, $T_g = 1$ and eight users. (a) For $N_f = 1$; (b) for $N_f = 5$, $N_f = 15$ and $N_f = 25$.

can also be expressed in terms of $\rho$:

$$\bar{\xi}/\Delta T = (id/c\Delta T)\sin \phi_0 = (i/2m)\rho \sin \phi_0$$

Finally, the delayed signals $\gamma(t + \bar{\xi} - \tau_\phi)$, $i = 0, \pm 1, \pm 2, \ldots, \pm m$ from the VDC's are summed by SUM1, SUM2 and SUM3 to produce the beamformer’s response function

$$\tilde{\gamma}(t, \phi) = \sum_{i=-m}^{m} \gamma(t + \bar{\xi} - \tau_\phi)$$

Some results are shown in Figures 8.27 to 8.31.
Figure 8.24 Comparing the different receivers (MF, ZF, MMSE) for the binary PPM-IRMA scheme for 100 Monte Carlo channel trials with the same parameters as in Figure 8.23(a).

Figure 8.25 (a) Normalized time variation of the generalized Gaussian pulse $\Omega(t)/E_0$; (b) autocorrelation function $Y(t)/Y(0)$; (c) energy density spectrum $\Psi(f) = |\Lambda(f)|^2$ for values of the scaling parameter $\alpha = 0$ (dotted line), $\alpha = 0.75$ (dashed line), $\alpha = 1.5$ (dashed-dotted line) and $\alpha = 3$ (solid line).
Figure 8.25 (Cont.).

(b) \[ \gamma(t) = \int_{-\infty}^{\infty} \Omega(\lambda) \Omega(\lambda + t) d\lambda \]

Energy density spectrum \[ \psi(f) / (E_0/2\Delta f)^2 \]

(c) \[ A(f) = \int_{-\infty}^{\infty} \Omega(t) \exp(-j2\pi ft) dt \]
Figure 8.26 A self-steering array beamforming system for UWB impulse waveforms. The beamformer includes $M = 2m + 1$ sensors, sliding correlator ($SC_i$), variable delay circuit ($VDC_i$), delay adjustment computer (DAC) and summer circuits (SUM) [37] © 2002, IEEE.

8.7.2 The monopulse tracking system

The monopulse signal delivered to DAC by SUM4 is:

$$
\Upsilon(t, \phi) = \sum_{i=0}^{+m} \Upsilon(t + \tilde{\tau}_i - \tau_i(\phi)) - \sum_{i=-m}^{0} \Upsilon(t + \tilde{\tau}_i - \tau_i(\phi))
$$

(8.47)

Angle-of-arrival estimation based on slope processing is illustrated in Figure 8.32. Basically, for different signs of $\phi$, the control signal $\Upsilon(t, \phi)$ has different slopes [see Figure 8.32(a)]. For larger $\phi$, the control signal $\Upsilon(t, \phi)$ is smaller. Based on this, the control mechanism is defined so that the delays of the $VDC_i$s are adjusted by DAC according to the relationship

$$
\tilde{\tau}_i > \tilde{\tau}_{i+1}, \quad \text{for} \quad -\pi/2 \leq \phi < \phi_0
$$

$$
\tilde{\tau}_i < \tilde{\tau}_{i+1}, \quad \text{for} \quad \phi_0 < \phi \leq \pi/2
$$

(8.48)

More details can be found in [38].
Figure 8.27 Space–time resolution function $\Theta(t, \phi)$ for the value of spacial frequency bandwidth $\rho = 10$. (a) Scaling parameter $\alpha = 3$ and the angular range $-5^0 \leq \phi \leq +5^0$; (b) $0^0 \leq \phi \leq +3^0$. 
Figure 8.28 Peak amplitude pattern $\tilde{A}(\phi)$ for: (a) $\phi_0 = 0^0$, $\alpha = 3$, $d = c\Delta T/2$, and $M = 5$ (dotted line), 9 (dashed line), 13 (dashed-dotted line) and 17 (solid line); and (b) $M = 9$ and $d = c\Delta T/2$ (dotted line), $c\Delta T$ (dashed line), $3c\Delta T/2$ (dashed-dotted line) and $2c\Delta T$ (solid line).
Figure 8.29 Peak power pattern $P(\phi)$ for: (a) $\phi_0 = 0^0$ and $\rho = 10$, $\alpha = 0.5$ (dotted line), 0.75 (dashed line), 1.5 (dashed-dotted line) and 3 (solid line); and (b) $\alpha = 3$ and $\rho = 4$ (dotted line), 6 (dashed line), 8 (dashed-dotted line) and 10 (solid line).

8.8 BEAMFORMING FOR UWB RADIO

8.8.1 Circular array

The received signal is modeled as a Gaussian pulse [38–40]

$$\Omega(t) = \frac{E}{\Delta T} \exp[-\pi(t/\Delta T)^2]$$

For the geometry shown in Figure 8.33 we have:

$$R_n = (r^2 + a^2 - 2ar \cos \Psi_n)^{1/2}$$
Figure 8.30 Energy pattern $\tilde{W}(\phi)$ for: (a) $\phi_0 = 0^0$ and $\rho = 10, \alpha = 0.5$ (dotted line), 0.75 (dashed line), 1.5 (dashed-dotted line) and 3 (solid line); and (b) $\alpha = 3$ and $\rho = 4$ (dotted line), 6 (dashed line), 8 (dashed-dotted line) and 10 (solid line).

\[ \tilde{U}(\phi) = \int_{-\infty}^{\infty} |\tilde{\Upsilon}(\tau \phi)|^2 \, d\tau \]

\[ \tilde{W}(\phi) = \frac{\tilde{U}(\phi)}{\tilde{U}(\phi_0)} \]
Figure 8.31 Peak power pattern $P(\phi)$ and energy pattern $\tilde{W}(\phi)$ for the case of an ideal Gaussian pulse with $\alpha = 0$. The plots are calculated for $\rho = 4$ (dotted line), 6 (dashed line), 8 (dashed-dotted line) and 10 (solid line).

which, for $r \gg a$, can be approximated as

$$R_n = r - a \cos \Psi_n = r - a \sin \theta \cos (\phi - \phi_n)$$

with $\phi_n = 2\pi \left(\frac{n}{N}\right)$, $n = 1, 2, \ldots, N$.

We assume that the signal source is located in the far field at the point $P(r, \theta, \phi)$; $r \gg a$. So at the $n$th array element we have:

$$V_n(t) = \Omega(t + \tau_n) = (E/\Delta T) \exp[-\pi((t + \tau_n)/\Delta T)^2]$$

(8.50)
Figure 8.32 Normalized monopulse signal $\Upsilon/(m \Delta T E_0^2)$ plotted as a function of relative time $t/\Delta T$ for (a) $\alpha = 3, \rho = 10$, and the values of the angle of incidence $\phi = +20^0$ (solid line), and $\phi = -20^0$ (dotted line) and repeated for (b) $\phi = 20^0$ (solid line), $\phi = 30^0$ (dashed line), $\phi = 40^0$ (dotted line).

with $\tau_n = (a/c) \sin \theta \cos(\phi - \phi_n)$. To steer the peak of the main beam of the array in the $(\theta_0, \phi_0)$ direction, a delay

$$\alpha_n = -\tau_n (\theta_0, \phi_0) = -(a/c) \sin \theta_0 \cos(\phi_0 - \phi_n)$$

must be applied to the voltage signal $V_n(t)$. So, the overall received signal is

$$V_T (t, \theta, \phi) = \sum_{n=1}^{N} \Omega (t + \tau_n + \alpha_n)$$

We will use notation

$$\rho_c = (a/c \Delta T) = 2\Delta f a/c, \quad \Delta f = 1/2\Delta T$$

$$\eta_s \equiv \sin \theta \sin \phi - \sin \theta_0 \sin \phi_0$$

$$\eta_c \equiv \sin \theta \cos \phi - \sin \theta_0 \cos \phi_0$$
Equations (8.52) and (8.54) together result in:

\[(\alpha_n + \tau_n) / \Delta T = \rho_c \eta_0 \cos (\xi - \phi_n)\]  \hspace{1cm} (8.54)

and we have

\[V_T(t, \theta, \phi) = (E / \Delta T) \sum_{n=1}^{N} \exp\{-\pi [(t / \Delta T) + \rho_c \eta_0 \cos (\xi - \phi_n)]^2\}\]  \hspace{1cm} (8.55)

For a main beam in the vertical direction along the \(z\)-axis \(\theta_0 = 0\) which yields \(\alpha_n = \alpha_{nv} = 0, \ \eta_0 = \eta_{0v} = \sin \theta, \ \xi = \xi_v = \phi\) and we have:

\[V_v(t, \theta, \phi) = (E / \Delta T) \sum_{n=1}^{N} \exp\{-\pi [(t / \Delta T) + \rho_c \sin \theta \cos (\phi - \phi_n)]^2\}\]  \hspace{1cm} (8.56)

A main beam can be formed in the principal horizontal plane \((z = 0)\) by setting \(\theta_0 = \pi / 2\) which gives

\[\alpha_n = \alpha_{nh} = - (a / c) \cos (\phi_0 - \phi_n)\]

\[\eta_0 = \eta_{0h} = 2 \sin \left(\frac{\phi_0 - \phi_n}{2}\right)\]  \hspace{1cm} (8.57)

\[\xi = \xi_h = \frac{\pi + \phi_0 + \phi_n}{2}\]
Figure 8.34 The time variation of the normalized voltage signal $V_v(t, \theta, \phi)$ for $\rho_c \sin \theta = 0$, 0.2, 0.4, 0.6, 0.8, 1, and 1.2.

and we have

$$V_h(t, \phi) = \frac{(E/\Delta T) \times \sum_{n=1}^{N} \exp \left\{ -\pi \left[ (t/\Delta T) + 2\rho_c \sin \left( \frac{\phi - \phi_0}{2} \right) \right] \cos \left( \frac{\pi + \phi_0 + \phi - 2\phi_n}{2} \right)^2 \right\}}{N \sum_{n=1}^{N} \exp \left\{ -\pi \left[ \rho_c \eta_0 \cos \left( \xi - \phi_n \right) \right]^2 \right\}}$$

(8.58)

A graph of $V_v(t, \theta, \phi)/N(E/\Delta T)$ is shown in Figure 8.34.

By using Figure 8.34, we define the peak amplitude pattern

$$A(\theta, \phi) = \frac{V_T(0, \theta, \phi)/N(E/\Delta T)}{1 - \frac{N}{\sum_{n=1}^{N} \exp \{-\pi [\rho_c \eta_0 \cos (\xi - \phi_n)]^2\}}}$$

(8.59)

Similarly, we have, for the vertical and the horizontal peak amplitude patterns, the following expressions

$$A_v(\theta, \phi) = \frac{V_v(0, \theta, \phi)/N(E/\Delta T)}{1 - \frac{N}{\sum_{n=1}^{N} \exp \{-\pi [\rho_c \sin \theta \cos (\phi - \phi_n)]^2\}}}$$

$$A_h(\phi) = \frac{1}{N} \sum_{n=1}^{N} \exp \left\{ -\pi \left[ 2\rho_c \sin \left( \frac{\phi - \phi_0}{2} \right) \cos \left( \frac{\pi + \phi_0 + \phi - 2\phi_n}{2} \right)^2 \right] \right\}$$

(8.60)

In a similar way, one can show that for sinusoidal waves with the time variation $\exp(j\omega t)$, the Array factor, $AF$, of an $N$-element circular array is given by [41, 42]:

$$AF(\theta, \phi) = \sum_{n=1}^{N} I_n \exp \left\{ j\rho_c \eta_0 \cos (\xi - \phi_n) \right\}$$

(8.61)

where

$$\rho_c = \frac{2\pi a}{\lambda}, \quad \lambda = \text{wavelength},$$

(8.62)
In is the amplitude excitation of the array element \( n \). For constant \( In = I \) we have:

\[
AF(\theta, \phi) = NI \sum_{n=1}^{N} J_{mN} (\rho, \eta_0) \exp \left[ j m N \left( \frac{\pi}{2} - \xi \right) \right]
\] (8.63)

where \( J_x(\cdot) \) is the Bessel function of the first kind. The vertical and the horizontal amplitude patterns are now given as:

\[
AF_v(\theta, \phi) = NI \sum_{n=1}^{N} J_{mN} (\rho, \sin \theta) \exp \left[ j m N \left( \frac{\pi}{2} - \phi \right) \right]
\] (8.64)

\[
AF_h(\phi) = NI \sum_{n=1}^{N} J_{mN} (2\rho, \sin \frac{\phi}{2}) \exp \left[ - j m N \frac{\phi}{2} \right]
\] (8.65)

For periodic sinusoidal signals, the square of the amplitude pattern, \( AF(\theta, \phi) \), represents the average power pattern \( P_{av}(\theta, \phi) \). We also define the vertical and horizontal power patterns as

\[
P_{av,v}(\theta, \phi) = [AF_v(\theta, \phi)]^2
\] (8.66)

\[
P_{av,h}(\theta, \phi) = [AF_h(\theta, \phi)]^2
\] (8.67)

The peak power pattern is defined as

\[
P(\theta, \phi) = [A(\theta, \phi)]^2 = \frac{1}{N^2} \left[ \sum_{n=1}^{N} \exp \left\{ -\pi \left[ (t/\Delta T) + \rho, \eta_0 \cos (\xi - \phi_n) \right]^2 \right\} \right]^2
\] (8.68)

We also define the vertical and horizontal peak power patterns as

\[
P_v(\theta, \phi) = [A_v(\theta, \phi)]^2
\] (8.69)

\[
P_h(\phi) = [A_h(\phi)]^2
\] (8.70)

The average power or energy pattern is defined as

\[
W(\theta, \phi) = \frac{\int_{-\infty}^{\infty} [VT(t, \theta, \phi)]^2 dt}{\int_{-\infty}^{\infty} [VT(t, \theta_0, \phi_0)]^2 dt}
\]

\[
= \frac{\int_{-\infty}^{\infty} \left[ \sum_{n=1}^{N} \exp \left\{ -\pi \left[ (t/\Delta T) + \rho, \eta_0 \cos (\xi - \phi_n) \right]^2 \right\} \right]^2 dt}{N^2 \int_{-\infty}^{\infty} \exp \left\{ -2\pi [(t/\Delta T)]^2 \right\} dt}
\] (8.66)

As before, for the vertical and horizontal components, we have

\[
W_v(\theta, \phi) = \frac{\int_{-\infty}^{\infty} \left[ \sum_{n=1}^{N} \exp \left\{ -\pi \left[ (t/\Delta T) + \rho, \sin \theta \cos (\phi - \phi_n) \right]^2 \right\} \right]^2 dt}{N^2 \int_{-\infty}^{\infty} \exp \left\{ -2\pi [(t/\Delta T)]^2 \right\} dt}
\] (8.69)

\[
W_h(\theta, \phi) = \frac{\int_{-\infty}^{\infty} \left[ \sum_{n=1}^{N} \exp \left\{ -\pi \left[ (t/\Delta T) + 2\rho, \sin \left( \frac{\phi - \phi_0}{2} \right) \cos \left( \frac{\phi + \phi_0 + \pi - 2\phi_n}{2} \right) \right]^2 \right\} \right]^2 dt}{N^2 \int_{-\infty}^{\infty} \exp \left\{ -2\pi [(t/\Delta T)]^2 \right\} dt}
\] (8.70)

Some numerical results are shown in Figures 8.35–8.44.
Figure 8.35 Normalized amplitude pattern \( AF_v(\theta, \phi) \) given in Equation (8.65) for an infinitely extended periodic sinusoidal wave received by the circular array in Figure 8.33 with \( N = 16 \) elements and (a) \( \rho_s = 1 \); (b) \( \rho_s = 3 \); (c) \( \rho_s = 6 \); (d) \( \rho_s = 12 \).
Figure 8.35 (Cont.).
Figure 8.36 Normalized average power pattern $P_{av,v}(\theta, \phi)$ given in Equation (8.66) for an infinitely extended periodic sinusoidal wave received by the circular array in Figure 8.33 with $N = 16$ elements and (a) $\rho_s = 1$; (b) $\rho_s = 3$; (c) $\rho_s = 6$; (d) $\rho_s = 12$. 
Figure 8.36 (Cont.).
Figure 8.37 Normalized amplitude pattern $AF_v(\theta, \phi)$ given in Equation (8.65) for an infinitely extended periodic sinusoidal wave received by the circular array in Figure 8.33 with $\rho_s = 12$ and (a) $N = 10$ elements; (b) $N = 32$ elements.
Figure 8.38 Normalized average power pattern $P_{av,v}(\theta, \phi)$ given in Equation (8.66) for an infinitely, extended periodic sinusoidal wave received by the circular array in Figure 8.33 with $\rho_s = 12$ and (a) $N = 10$ elements; (b) $N = 32$ elements.
Figure 8.39 Peak amplitude pattern $A_v(\theta, \phi)$ given in Equation (8.61) for non-sinusoidal Gaussian pulses received by the circular array in Figure 8.33 with $N = 16$ elements and (a) $\rho_s = 1$; (b) $\rho_s = 3$; (c) $\rho_s = 6$; (d) $\rho_s = 12$. 
A slope pattern $S(\theta, \phi)$, a vertical slope pattern $S_v(\theta, \phi)$ and a horizontal slope pattern $S_h(\phi)$ can be derived by using a linear regression algorithm to calculate the slope of the ramp that best fits the rising section of the Gaussian pulse $V_T(t, \theta, \phi)$, $V_v(t, \theta, \phi)$ and $V_h(t, \phi)$. A plot of the ramp slope versus angle results in a slope pattern.

More details on practical aspects of UWB antenna and receiver design can be found in [43–88].
Figure 8.40 Peak power pattern $P_v(\theta, \phi)$ given in Equation (8.68) for non-sinusoidal Gaussian pulses received by the circular array in Figure 8.33 with $N = 16$ elements and (a) $\rho_s = 1$; (b) $\rho_s = 3$; (c) $\rho_s = 6$; (d) $\rho_s = 12$. 
Figure 8.40 (Cont.).
Figure 8.41 Energy pattern \( W_v(\theta, \phi) \) given in Equation (8.70) for non-sinusoidal Gaussian pulses received by the circular array in Figure 8.33 with \( N = 16 \) elements and
(a) \( \rho_s = 1 \); (b) \( \rho_s = 3 \); (c) \( \rho_s = 6 \); (d) \( \rho_s = 12 \).
Figure 8.41 (Cont.).
Figure 8.42 Peak amplitude pattern $A_v(\theta, \phi)$ for non-sinusoidal Gaussian pulses received by the circular array with a large number ($N \to \infty$) of elements and (a) $\rho_s = 1$; (b) $\rho_s = 3$; (c) $\rho_s = 6$; (d) $\rho_s = 10$. 
Figure 8.42 (Cont.).
Figure 8.43 Peak power pattern $P_v(\theta, \phi)$ for non-sinusoidal Gaussian pulses received by the circular array with a large number ($N \rightarrow \infty$) of elements and (a) $\rho_s = 1$; (b) $\rho_s = 3$; (c) $\rho_s = 6$; (d) $\rho_s = 10$. 
Figure 8.43 (Cont.).
Figure 8.44 Energy pattern $W_r(\theta, \phi)$ for non-sinusoidal Gaussian pulses received by the circular array with a large number ($N \to \infty$) of elements and (a) $\rho_s = 1$; (b) $\rho_s = 3$; (c) $\rho_s = 6$; (d) $\rho_s = 10$. 

(a) 

(b)
Figure 8.44 (Cont.).
REFERENCES


REFERENCES


9

Linear Precoding for MIMO Channels

9.1 SPACE–TIME PRECODERS AND EQUALIZERS FOR MIMO CHANNELS

In this section the problem of equalization, which was initiated in Chapter 6 for single-input–single-output (SISO) systems, is extended to multiple-input–multiple-output (MIMO) channels.

9.1.1 ISI modelling in MIMO channels

Let us assume a MIMO channel with \( p \) transmitters and \( q \) receivers. Let \( s_j(k) \) denote the sequence from transmitter \( j (j = 1, \ldots, p) \), and let \( h_{ij}(k) \) be the channel response from input \( j \) to output \( i (i = 1, \ldots, q) \). Consequently, the output sequence at the receiver \( i \) is:

\[
x_i(k) = \sum_{j=1}^{p} \sum_{l=0}^{d} h_{ij}(l)s_j(k-l)
\]

(9.1)

where \( d \) denotes the maximal ISI degree. The above convolution can be equivalently expressed in a \( z \)-transform notation:

\[H(D)s(D) = x(D)\]

(9.2)

where \( s(D) = [s_1(D) s_2(D) \ldots s_p(D)]^T \), \( x(D) = [x_1(D) \ldots x_q(D)]^T \), and \( H(D) = \{h_{ij}(D)\} \) are the \( z \)-transform vectors (matrix) of the corresponding sequences or impulse responses where we use a delay operator \( D = z^{-1} \). A \( p \)-input–\( q \)-output MIMO system can then be represented by the \( q \times p \) matrix \( H(D) \), called the transfer function of the MIMO system.

A transfer function \( H(D) \) is perfectly recoverable (PR) if and only if there exists a polynomial matrix \( G(D) \) such that:

\[G(D)H(D) = \text{Diag}[D^{k_j}]\]

(9.3)

where \( k_j \) denotes the necessary delays incurred on the recovered \( j \)th source signal. The equality in Equation (9.3) is referred to as a (generalized) Bezout identity and \( G(D) \) is referred to as a Bezout inverse of \( H(D) \), which can perfectly recover the original signals in the absence of noise.
MIMO system in Equation (9.1), the \( j \)th source signal is said to be perfectly recoverable (PR) with a \( \rho_j \)-tap equalizer (i.e., degree = \( \rho_j - 1 \)) if there exists a (row) polynomial vector \( g(D) \) with degree lower than \( \rho_j \) such that:

\[
g(D) x(D) = s_j(D) D^{k_j}
\]  

(9.4)

A polynomial matrix \( C(D) \) is said to be a (right) common factor of the columns in \( H(D) \) if \( H(D) = H(D) C(D) \) and \( H(D) \) is itself a polynomial matrix. Moreover, a polynomial matrix is unimodular if and only if its determinant is a constant. A polynomial matrix is said to be (right) coprime if and only if there exists no nonunimodular right common factor. The right coprimeness of \( H(D) \) requires it to have more rows than columns, i.e., \( q > p \), except when it is a unimodular matrix.

In the sequel we will use some results from matrix theory [1]:

(R1) A MIMO system with transfer function \( H(D) \) is perfectly recoverable (PR) if and only if \( H(D) \) is (right) coprime, except for a common factor with determinant \( D^k \). (This property will be called delay-permissive coprimeness.)

(R2) A MIMO system with transfer function \( H(D) \) is perfectly recoverable if and only if \( H(\lambda) \) has full column rank for any (complex value), \( \lambda \neq 0 \).

For any \( p \)-input--\( q \)-output MIMO transfer function \( H(D) \), there exists a unimodular polynomial matrix \( U(D) \) [2] such that \( H(D) U(D) = \hat{H}(D) \) where \( \hat{H}(D) \) is a minimal basis (MB). The \( q \times p \) polynomial matrix \( \hat{H}(D) \) is called a minimal basis if and only if \( \hat{H}(D) \) is column reduced, i.e. its (column-wise) highest-degree coefficient matrix has full column rank. In the sequel, we will denote the \( k \)th column degree of the MB \( \hat{H}(D) \) by \( \mu_k \).

The MacMillan degree (which is denoted as \( \eta \)) of a \( q \times p \) \((p < q)\) polynomial matrix \( H(D) \) is defined as the highest degree of the determinants of all the \( p \times p \) minors in \( H(D) \). It can be shown [2] that \( \eta = \sum_{k=1}^{p} \mu_k \).

Given a \( q \times p \) \((p < q)\) polynomial matrix \( H(D) \) with full column rank (on the polynomial ring), a \((q - p) \times q\) polynomial matrix \( N(D) \) is called a null-space minimal basis (NMB) of \( H(D) \) if and only if:

1. \( N(D) \) is in the null space of \( H(D) \), i.e., \( N(D) H(D) = 0 \);
2. \( N(D) \) is left-coprime;
3. \( N(D) \) is row reduced, i.e., its (row-wise) highest-degree coefficient matrix has full row rank.

We denote the row degrees of \( N(D) \) by \(\{ v_i \}_{i=1}^{\eta-p} \). The degree of the minimal basis of the null space of \( H(D) \) is defined as:

\[
v = \max_i \{ v_i \}
\]

Any polynomial vector in the null subspace of \( H(D) \) can be expressed as a (polynomial-wise) combination of the row vectors in \( N(D) \). For more properties of minimal basis, see [2, Chapter 6] and [3]

*The MIMO equivalent matrix* \( \Gamma^\rho [H] \) is a block Toeplitz matrix (with \( \rho \) block-rows, i.e. \( \rho \times q \) rows):

\[
\Gamma^\rho [H] = \begin{bmatrix}
H_d & H_{d-1} & \cdots & H_0 & 0 & \cdots & 0 \\
0 & H_d & H_{d-1} & \cdots & H_0 & \cdots & 0 \\
\vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\
0 & \cdots & 0 & H_d & H_{d-1} & \cdots & H_0
\end{bmatrix}
\]  

(9.5)

where \( H_i \) is the \( i \)th degree coefficient matrix of \( H(D) \), i.e. \( H(D) = H_0 + D H_1 + \cdots + D^\rho H_d \).

The *equivalent* matrix can provide an effective tool for testing the coprimeness and, hence, the recoverability (PR) of the MIMO system.
Reference [4] defines *equivalent matrix test for MIMO recoverability* as follows:

(a) A MIMO system $H(D)$ is coprime if and only if there exists an integer $\rho$ such that

\[
\text{rank} \{\Gamma^\rho[H]\} = \eta + p \times \rho
\]  
(9.6)

where $\eta$ is the McMillan degree of $H(D)$, (The smallest integer $\rho$ to meet the above equality is equal to the NMB degree of $H(D)$, i.e., $v$).

(b) A MIMO system $H(D)$ is PR if and only if there exists an integer $\rho$ such that

\[
\text{rank} \{\Gamma^\rho[H]\} = \eta' + p \times \rho
\]  
(9.7)

where $\eta'$ is the reduced McMillan degree of $H(D)$, which can be obtained by simply subtracting the degree contributed by those pure delay factors from the full McMillan degree. More precisely, $\eta' = \eta - \text{degree associated with pure delay factors}$.

Again, the smallest integer to meet the above equality is equal to the NMB degree $v$ [4].

### 9.1.2 MIMO system precoding and equalization

The above system theory serves as a theoretical foundation for flexible transceiver design of MIMO systems. The analysis can be logically divided into the following categories:

(a) $p < q$, and the channel is known to receiver;
(b) $p > q$, and the channel is fed back to the transmitter;
(c) $p \geq q$, and the channel is known to the receiver;
(d) When $p < q$, the channel is known to the transmitter and the receiver.

Case (d) is usually handled without using transmitter channel knowledge, and hence, it can be treated as (a). As outlined in the flowchart in Figure 9.1, case (c) will be treated in Section 9.1.3, while cases (a) and (b) are treated in Sections 9.1.2.1 and 9.1.2.2 respectively.

Figure 9.1 Design strategy is outlined in the flow chart. The pure Bezout solutions are outlined in the upper dashed box. The joint Bezout-STBC solutions will be treated in the next section as shown in the lower dashed box.
According to (R1) and (R2) of the previous section, when \( p < q \) and the channel is known to receiver, there exists a PR Bezout equalizer if and only if the \( q \times p \) transfer function \( H(D) \) is delay-permissive right coprime. More exactly, \( G(D) \) is a PR Bezout equalizer if and only if \( G(D) H(D) = \text{Diag}[D^{k_j}] \)

Similarly, when \( p > q \) and the channel is known to the transmitter, there exists a PR Bezout precoder if and only if the \( q \times p \) transfer function \( H(D) \) is delay-permissive left coprime. More exactly, \( F(D) \) is a PR Bezout precoder if and only if \( F(D) H(D) = \text{Diag}[D^{k_j}] \)

With such a Bezout precoder, the symbols received by the receiver will not only be ISI-free but also ICI-free (containing only the desired stream’s information).

### 9.1.2.1 MIMO Bezout equalizer

To analyze the system performance under noisy conditions, we need to investigate the SNR in Bezout inverse system.

The post-processing SNR is defined as the SNR at the output of the equalizer. Assuming BPSK constellation and that the bit is decided by zero thresholding, the BER can be determined as

\[
\text{BER} = Q\left(\sqrt{\text{SNR}}\right).
\]

The Bezout equalization (when \( p < q \) and channel is known to receiver) is illustrated in Figure 9.2. For a Bezout equalizer system \( G(D) H(D) = \text{Diag}[D^{k_j}] \), let us denote one row of \( G(D) \) as \( g(D) = g_0 + D g_1 + \cdots + D^{\rho - 1} g_{\rho - 1} \), and let \( \vec{g} \) denote the expanded row vector:

\[
\vec{g} \equiv [g_{\rho - 1} \ldots g_1 g_0]
\]

Obviously, the equalizer system can be designed separately for each individual stream. For the design of an individual equalizer \( g(D) \), note that the i.i.d. additive-white-Gaussian noise (AWGN) will be filtered by \( g(D) \) before its effect appears at the output of the equalizer. This leads to the post-processing noise power:

\[
\sigma_n^2 \|\vec{g}\|^2 = \frac{N_0}{2} \|\vec{g}\|^2,
\]

where \( N_0 \) is the noise spectral density. Therefore, the SNR pertaining to a Bezout equalizer can be derived as:

\[
\text{equalizer SNR} = \frac{2E_b}{N_0 \|\vec{g}\|^2}.
\]

![Figure 9.2 Bezout equalizer](image-url)
where $E_b$ is the transmit energy per bit. This immediately suggests a design criterion of minimizing the 2-norm of the equalizer.

From Figure 9.2, we note that the problem of designing an optimal Bezout equalizer for all the source inputs can be decoupled into the task of separately designing many individual equalizers, one for each input. The strategy of designing an optimal (individual) Bezout equalizer (for the $j$th input) is to find $g(D)$ with minimal two-norm $\|g\|$ such that:

$$g(D)H(D) = [0 \; \ldots \; D^{k_j} \; \ldots \; 0], \quad j = 1, \ldots, p$$

Equivalently, in a resultant matrix notation:

$$\bar{g}\Gamma^\rho [H] = e_i$$

where $e_i$ is a row vector with all elements being zero except an entry 1 at $j + p(d + \rho - 1 - k_j)$, $k_j = 0, \ldots, d + \rho - 1$. The derivation of Bezout inverse depends not only on the equalizer order but also on the system delay $k_j$. The solution of an individual Bezout inverse is given in Equation (9.10).

Take singular value decomposition (SVD) on $\Gamma^\rho[H]: \Gamma^\rho[H] = U\Sigma V$, where $\Sigma$ is a square diagonal matrix of positive singular values. Then, a Bezout inverse for Equation (9.10) exists if (and only if) there is a solution $b$ for $bV = e_i$, and thereafter, the Bezout solution $\bar{g} = b\Sigma^{-1}U = e_iV\Sigma^{-1}U^H$ yields a minimum-norm solution with the optimal 2-norm

$$\|\bar{g}\|^2 = e_iV\Sigma^{-2}Ve_i^H$$

Note that the system delay $k_j$ provides an extra (and important) degree of freedom for the Bezout inverse. Selection of $k_j$ is just the same as selecting $i$ since $i = j + p(d + \rho - 1 - k_j)$. The best integer $i$ minimizing the 2-norm in Equation (9.11) is:

$$i^* = \arg \min_i \{(V\Sigma^{-2})_{ii} \mid i = j \pmod p, \quad e_i \in \text{Row Span} \{V\}\}$$

### 9.1.2.2 Bezout MIMO precoder

When $p > q$ and the channel is known to transmitter a Bezout precoder system exists as illustrated in Figure 9.3. Mathematically

$$H(D)F(D) = \text{Diag}[D^{k_j}]$$

Let us denote one column of $F(D)$ as $f(D) = f_0 + Df_1 + \cdots + D^{p-1}f_{p-1}$, and its expanded column vector $\tilde{f} = [f_{p-1}^T \; \ldots \; f_1^T \; f_0^T]^T$. Let $\sigma_i^2$ be the input variance for one stream. Each precoding

![Figure 9.3 Bezout precoder: $H(D)F(D) = \text{Diag}[D^{k_j}]$ with $p < q$.](image-url)
column vector \( \mathbf{f}(D) \) amplifies the transmitter’s power to \( \sigma_s^2 \|\mathbf{f}\|^2 \). In order to have an ISI-free communication, we must have \( \mathbf{H}(D)\mathbf{f}(D) = [0 \ldots D^{k_j} \ldots 0]^T \), which produces a signal power of \( \sigma_s^2 \) at each receiver. Moreover, we must set \( \sigma_n^2 = \frac{E_b}{\|\mathbf{f}\|^2} \) in order to normalize the transmit energy per bit to \( E_b \). In this case, the SNR for the Bezout precoder becomes

\[
\text{precoder SNR} = \frac{2E_b}{N_0\|\mathbf{f}\|^2} \tag{9.14}
\]

Note that the noise power at each receiver is \( N_0/2 \).

Equation (9.14) establishes a useful duality between the optimal designs of Bezout equalizer and precoder systems. The optimal precoder design for a MIMO system \( \mathbf{H}(D) \) is equivalent to the optimal equalizer design for a MIMO with transfer function \( \mathbf{H}(D)^H \). Similarly to the previous equalizer design, design of an optimal Bezout precoder can be decoupled into the task of separately designing many individual precoders: one for each output.

An optimal (individual) Bezout precoder can be obtained by finding \( \mathbf{f}(D) \), with minimal two-norm such that \( \mathbf{H}(D)\mathbf{f}(D) = [0 \ldots D^{k_j} \ldots 0]^H \). Equivalently, in a resultant matrix notation

\[
\mathbf{V}^H \mathbf{V} = \mathbf{I}
\]

\[
\mathbf{f} = \mathbf{V}^H \mathbf{e}
\]

where \( \mathbf{e} \) is now a column vector with all zeros, except an entry 1 at \( i = j + q(d+ \rho - 1 - k_j) \), \( k_j = 0, \ldots, d + \rho - 1 \). Take SVD on \( \mathbf{f} = \mathbf{U} \Sigma \mathbf{V} \) as

\[
\mathbf{f} = \mathbf{U} \Sigma \mathbf{V} \]

For a given index \( i \), a Bezout precoder \( \tilde{\mathbf{f}} \) for Equation (9.15) exists iff a solution \( \mathbf{b} \) for \( \mathbf{U} \mathbf{b} = \mathbf{e} \) exists. Similarly, the optimal Bezout precoder is \( \tilde{\mathbf{f}} = \mathbf{V}^H \Sigma^{-1} \mathbf{b} = \mathbf{V}^H \Sigma^{-1} \mathbf{U}^H \mathbf{e} \), with 2-norm:

\[
\|\tilde{\mathbf{f}}\|^2 = (\mathbf{U} \Sigma^{-2} \mathbf{U}^H)_{ii} \tag{9.17}
\]

The optimal integer \( i^* \) corresponding to the optimal delay \( k_j \) is:

\[
i^* = \arg \min_{i} \{(\mathbf{U} \Sigma^{-2} \mathbf{U}^H)_{ii} | i = j (\mod q) \ \mathbf{e} \in \text{Column Span} \{\mathbf{U}\}\} \tag{9.18}
\]

### 9.1.3 Precoder and equalizer design for STBC systems

In many down-link communication scenarios, we have fewer receivers than transmitters (i.e., \( p \geq q \)). In this case it may be effective to adopt a design combining the Bezout and STBC techniques.

STBC, introduced in Chapter 4, can be regarded as a combination of channel coding techniques and equalization techniques. The performance can be enhanced by incorporating some structured redundancy at the transmitter side, e.g. with zero-padding or cyclic-prefixing. The basic STBC system used in this section is illustrated in Figure 9.4. Given a \( p \times q \) transfer function of a \( (p-\text{input} - q-\text{output}) \) physical channel model, the STBC system can generate an expanded virtual \( q' \times p' \) transfer function with \( p' = pW \) and \( q' = qN \). Namely, it results in a \( qN \times pW \) virtual transfer function. Here, \( N \) denotes the block size, and \( W \) is the number of symbols sent per transmitter per block. For convenience, this will be referred to as a \( N,W \)-STBC MIMO system.

If the original channel is PR, there is no need to incorporate redundancy; then, we can set \( W = N \). Such a \( (N,N) \)-STBC system is obtained by adding interleaving and deinterleaving operators to the transmission channel inputs and outputs, respectively.

The virtual transfer function corresponding to a physical SISO channel has been well studied [5, 8, 9]. Given a transfer function \( \mathbf{H}(D) \) of a \( p \)-input–\( q \)-output MIMO physical channel \( \mathbf{H}(D) = \sum_{n=0}^{d} \mathbf{H}_n D^n \) and \( \mathbf{H}_{j/N}(D) \) is its \( j \)th forward polyphase component out of \( N \):

\[
\mathbf{H}_{j/N}(D) = \sum_{n=0}^{[d/N]} \mathbf{H}_{Nn+j} D^n \tag{9.19}
\]
where $0 \leq j \leq N - 1$. Following [5], we can obtain a $qN \times pN$ virtual transfer function denoted as $\overline{H}(D)$:

$$
\overline{H}(D) = \begin{bmatrix}
H_{0/N}(D) & DH_{N-1/N}(D) & \cdots & DH_{1/N}(D) \\
H_{1/N}(D) & H_{0/N}(D) & \cdots & DH_{2/N}(D) \\
& \ddots & \ddots & \ddots \\
H_{N-2/N}(D) & H_{N-3/N}(D) & \cdots & DH_{N-1/N}(D) \\
H_{N-1/N}(D) & H_{N-2/N}(D) & \cdots & H_{0/N}(D)
\end{bmatrix}
$$ (9.20)

If $H(D)$ is a $q \times p$ (assuming $p \leq q$) transfer function of a physical MIMO channel and $\overline{H}(D)$ the virtual transfer function of the $(N,N)$-STBC system, which is given in Equation (9.20), then there is a one-to-one correspondence between the common zeros of $H(D)$ (denoted by $\{\lambda_i\}$) and that of $\overline{H}(D)$ (denoted by $\{\eta_i\}$). More exactly [4]:

$$
\eta_i = \lambda_i^N.
$$ (9.21)

and the $(N,N)$-STBC preserves the PR property of the original transfer function [6].

Problems arise for $(N,N)$-STBC systems when the original $H(D)$ is not PR or when it is PR but not robustly recoverable. An effective solution is via a redundant precoding matrix $F$:

$$
\tilde{H}(D) = H(D) F
$$ (9.22)

The simplest $(N,W)$-STBC precoding is represented by a precoding matrix:

$$
F = \begin{bmatrix}
I_{pW \times pW} & 0_{pW \times p(N-W)}
\end{bmatrix}^T
$$ (9.23)
This will lead to a $qN \times pW$ virtual transfer function, which is denoted as $\tilde{H}(D)$

$$\tilde{H}(D) = \begin{bmatrix}
H_{0/N}(D) & DH_{N-1/N}(D) & \cdots & DH_{N-W+1/N}(D) \\
H_{1/N}(D) & H_{0/N}(D) & \cdots & DH_{N-W+2/N}(D) \\
\vdots & \vdots & \ddots & \vdots \\
H_{N-2/N}(D) & H_{N-3/N}(D) & \cdots & H_{N-W-1/N}(D) \\
H_{N-1/N}(D) & H_{N-2/N}(D) & \cdots & H_{N-W-1/N}(D)
\end{bmatrix} \tag{9.24}$$

Since $W$ is the number of symbols sent per transmitter per block, the number of symbols being transmitted during one block period is $pW$; thus, the data transmission rate is $pW/N$. By using fewer columns than rows in the precoding matrix $F$, some redundancy can be incorporated for the purpose of enhancing the robust recoverability of the new transfer function $\tilde{H}(D)$.

If $H(D)$ is PR, then $\tilde{H}(D)$ in Equation (9.24) is PR if and only if the precoding matrix $F$ has full column rank. In particular, for the $(N,N)$-STBC system, the $qN \times pW$ virtual transfer function $\tilde{H}(D)$ is PR, provided that $H(D)$ is PR [4].

## 9.2 LINEAR PRECODING BASED ON CONVEX OPTIMIZATION THEORY

In this section, we discuss the design of a centralized precoder given fixed linear MIMO transmitter, channel, and receiver as shown in Figure 9.5. We define a precoder as a linear transformation on the transmitted symbols. If the precoded symbols are sent as is to the channel, then the precoder is the transmitter itself. However, in general, the precoded symbols may be transformed again before the channel. We refer to this transformation as the transmitter, and we assume that it is a fixed design parameter. The output of the transmitter is then sent over a fixed MIMO channel (or channels) and is received using a fixed linear receiver (or receivers).

A precoder that applied a linear transformation on the transmitted symbols prior to the spreading was derived in [11]. This precoder inverted the channel at the transmitter side and is usually referred to as the transmit zero-forcing (ZF) precoder. The main drawback of the transmit ZF precoder is noise enhancement. Better solutions are transmit MF precoders and transmit rakes, which perform better in low SNR and transmit minimum mean-squared error (MMSE) precoders that compensate for the performance in the different SNR regions [12–16].

Variants of the previous precoders are discussed in [17–22]. Linear precoders based on an approximate maximum-likelihood approach and maximum asymptotic multiuser efficiency with different power constraints are derived in [23]. A linear precoding technique based on a decomposition approach is proposed in [24], and a linear precoder design for nonlinear maximum-likelihood
(ML) receivers is discussed in [25]. Among the nonlinear precoders are the Tomlinson–Harashima precoder (THP) [26] and the ‘Dirty Paper’ precoder [27]. The nonlinear precoder that optimizes the transmitted symbols vector itself is derived in [28].

The problem of precoder design is highly related to other problems in the literature. In this section, we consider the design of linear precoders for fixed linear receivers. A related problem is the problem of jointly optimizing the precoder/transmitter and the receiver, which has been treated, e.g., in [29–37]. Another related problem is the joint design of rank-one transmit beamforming design and optimal power control [38–40].

In this section we discuss precoder design based on the framework of convex optimization theory [41], which allows for efficient numerical solutions using standard optimization packages [42]. It was shown in [10] that the power optimization problem can be formulated as a second-order cone program (SOCP) [43] or a semidefinite program (SDP) [44], known as a linear matrix inequalities (LMI) program. The SINR optimization can also be formulated as a standard conic program known as the generalized eigenvalue problem (GEVP) [45].

9.2.1 Generalized MIMO systems

We assume that at each time instant, a block of symbols is modulated and transmitted over the channels. The possibly distorted output is then processed at the receivers in a linear fashion, as depicted in Figure 9.5. Denoting by \( y_i \) the length \( L \) output of the \( m \)th receiver, for \( m = 1, \ldots, M \), we have

\[
\begin{bmatrix}
  y_1 \\
  \vdots \\
  y_M \\
  y
\end{bmatrix} =
\begin{bmatrix}
  H_{Rx,1} \& H_{Ch,1} \\
  \vdots \\
  H_{Rx,M} \& H_{Ch,M}
\end{bmatrix} H_{Tx} b +
\begin{bmatrix}
  H_{Rx,1} w_1 \\
  \vdots \\
  H_{Rx,M} w_M
\end{bmatrix} w
\]

(9.25)

where the matrices \( H_{Rx,m} \) and \( H_{Ch,m} \) denote the receiver and channel associated with the \( m \)th user, the matrix \( H_{Tx} \) is the centralized transmitter, \( b \) is the length \( K = M \cdot L \) vector of independent, and unit variance transmitted symbols, and \( w \) are the noise vectors. The noise vectors may be correlated, and the channels are completely arbitrary. The only restriction is that the transmitter is centralized and has access to all of the \( K \) transmit components.

We assume a single stream per MIMO dimension, i.e., the length of \( b \) is equal to the length of \( y \). However, these streams can be dedicated to a single user or to multiple users. A single user, point-to-point communication system using \( L \) multiple receive and \( L \) transmit antennas is a special case of Equation (9.25) with \( M = 1 \). The downlink channel of a CDMA system with \( L \) users is a special case of Equation (9.25) with \( L = 1 \), where \( H_{Tx} \) is a signature matrix whose columns are the signatures of each of the users, \( H_{Ch,i} = I \) and \( H_{Rx,j} \) are row vectors representing the linear receive filters of each of the users. A multiuser system with transmit beamforming in which \( L \) transmit antennas signal to \( L \) users each using a single receive antenna is a special case of Equation (9.25) with \( L = 1 \). Here, \( H_{Tx} \) is a beamforming matrix whose columns are the antenna weights of each of the \( L \) users, \( H_{Ch,i} \) are row vectors that represent the paths from the transmit antennas to the \( i \)th receive antenna, and \( H_{Rx,i} \) are arbitrary scalars.

In the sequel, we will assume that the transmitter \( H_{Tx} \), the channels \( H_{Ch,i} \) and the receivers \( H_{Rx,j} \) are fixed and cannot be altered due to budget restrictions, standardization, or physical problems. Given this fixed structure, we will try to improve the performance by introducing a linear precoder. The precoder, denoted by \( T \), linearly transforms the original symbol vector prior to the transmission.

As suggested by Equation (9.25), for ease of representation, we will use the following notation:

\[
y = HTb + w
\]

(9.26)

with \( H = H_{RxCh}H_{Tx} \) and the rest of the variables are defined in Equation (9.25).
The system performance (or quality of service) QoS is related to the output SINRs, and in particular to the worst SINR. The output SINR of the $i$th subchannel is defined as

$$\text{SINR}_i = \frac{\left| (\mathbf{H}^H)_{i,i} \right|^2}{\sum_{j \neq i} \left| (\mathbf{H}^H)_{i,j} \right|^2 + \sigma_i^2}$$

(9.27)

for $i = 1, \ldots, K$, where $\sigma_i^2 = E \left| w_i \right|^2 > 0$. Other criteria deal with the use of system resources, e.g., peak-to-average ratio, or maximal transmitted power. The most common resource measure is average transmitted power, which is defined as:

$$P = E \left\{ \| \mathbf{H}_T \mathbf{T} \|^2 \right\} = \text{Tr} \left\{ \mathbf{T}^H \mathbf{H}_T^H \mathbf{H}_T \mathbf{T} \right\}$$

(9.28)

The SINR metric and average power metric conflict, and one cannot maximize the SINRs while also minimizing the power, and vice versa. Depending on the application, the designer must decide which criterion is stricter. We therefore consider one of the following two complementary strategies. The first optimization strategy seeks to minimize the average transmitted power subject to QoS constraints. Given the required QoS, the system tries to satisfy it with minimum transmitted power [34, 39] (see also Chapter 16) as follows:

$$P(\gamma_o) = \begin{cases} \min_{\mathbf{T}} & \text{Tr} \left\{ \mathbf{T}^H \mathbf{H}_T^H \mathbf{H}_T \mathbf{T} \right\} \\ \text{s.t.} & \frac{\left| (\mathbf{H}^H)_{i,i} \right|^2}{\sum_{j \neq i} \left| (\mathbf{H}^H)_{i,j} \right|^2 + \sigma_i^2} \geq \gamma_o \end{cases}$$

(9.29)

where $\gamma_o > 0$ is the given worst SINR constraint.

The second strategy is maximizing the minimal SINR subject to a power constraint [33, 46] (see also Chapter 16). In this case, the problem can be formulated as:

$$S(P_o) = \begin{cases} \max_{\mathbf{T}} & \min_i \frac{\left| (\mathbf{H}^H)_{i,i} \right|^2}{\sum_{j \neq i} \left| (\mathbf{H}^H)_{i,j} \right|^2 + \sigma_i^2} \\ \text{s.t.} & \text{Tr} \left\{ \mathbf{T}^H \mathbf{H}_T^H \mathbf{H}_T \mathbf{T} \right\} \leq P_o \end{cases}$$

(9.30)

where $P_o > 0$ is the given power constraint.

### 9.2.2 Convex optimization

In the sequel the following notation is used. Boldface capital letters denote matrices, boldface lower-case letters denote column vectors, and standard lower-case letters denote scalars. The superscripts($\cdot^T$, $(\cdot)^*$, $(\cdot)^H$, $(\cdot)^{-1}$, and $(\cdot)^\dagger$) denote the transpose, the complex conjugate, the Hermitian, the matrix inverse operators, and the Moore–Penrose pseudoinverse, respectively. $[\mathbf{X}]_{i,j}$ denotes the $(i\text{th}, j\text{th})$ element of the matrix $\mathbf{X}$. By diag$\{x_i\}$, we denote a diagonal matrix with $x_i$ being the $(i\text{th}, i\text{th})$ element; by vec$\{\mathbf{X}\}$, we denote stacking the elements of $\mathbf{X}$ in one long column vector; by $e_i$, we denote a zeros vector with a one at the $i\text{th}$ element; by $I$, we denote an all ones vector; and by $\mathbf{I}$, we denote the identity matrix of appropriate size. Tr{$\cdot$}, $\Re\{$$\cdot$$\}$, $| \cdot |$, $\| \cdot \|_2$, and $\| \cdot \|_\infty$ denote the trace operator, the real part, the absolute value, the standard Euclidean norm, and the induced row sum matrix norm, respectively. Finally, $\mathbf{X} \succ 0$ denotes that the matrix $\mathbf{X}$ is a Hermitian positive semidefinite matrix, and $\mathbf{N}\{\cdot\}$ denotes the Null space operator.

The most widely used field in optimization is convex optimization. A convex program is a program with a convex objective function and convex constraints. It is well known that in such programs a local minimum is also a global minimum. The most common convex program is the linear program (LP) [41], i.e. an optimization with a linear objection function and linear (affine) constraints. Recently, conic programs, i.e. LPs with generalized inequalities are also used. The two standard conic programs
are SOCP and SDP optimization. The standard form of an SOCP is [43]:

\[
\text{SOCP : } \begin{cases}
\min_x & \Re \{ f^H x \} \\
\text{s.t.} & \begin{bmatrix} c_i^H x + d_i \\ A_i^H x + b_i \end{bmatrix} \succ_K 0, \quad i = 1, \ldots, N
\end{cases}
\] (9.31)

where the optimization variable is the vector \( x \) of length \( n \) and \( f, A_i, b_i, c_i, \) and \( d_i \) for \( i = 1, \ldots, N \) are the data parameters of appropriate sizes. The notation \( \succ_K \) denotes the following generalized inequality:

\[
\begin{bmatrix} z \\ z \end{bmatrix} \succ_K 0 \iff \| z \| \leq z
\] (9.31a)

The standard form of an SDP is [44]:

\[
\text{SDP : } \begin{cases}
\min_x & \Re \{ f^H x \} \\
\text{s.t.} & A(x) > 0
\end{cases}
\] (9.32)

where \( A(x) = A_0 + \sum_{i=1}^{n} x_i A_i \) is a Hermitian matrix that depends affinely on \( x \). The data parameters are the Hermitian matrices \( A_i \) for \( i = 0, \ldots, n \). The notation \( > \) denotes the positive semidefinite generalized inequality. A simple case of an SDP is an SOCP. For example, each of SOC constraints in Equation (9.31) can be written as an LMI [21], as follows:

\[
\begin{bmatrix} c_i^H x + d_i \\ x^H A_i + b_i^H \\ A_i^H x + b_i \\ (c_i^H x + d_i) \mathbf{1} \end{bmatrix} > 0
\] (9.31c)

A common optimization package designed to solve SOCP and SDP is SEDUMI [42].

Among nonconvex problems is the GEVP [45], which can still be efficiently solved. Its standard form is:

\[
\text{GEVP : } \begin{cases}
\min_{\beta, x} & \beta \\
\text{s.t.} & \beta B(x) - A(x) > 0 \\
& B(x) > 0 \\
& C(x) > 0
\end{cases}
\] (9.33)

where \( \beta \) is a real-valued optimization variable and \( A(x) = A_0 + \sum_{i=1}^{n} x_i A_i \), \( B(x) = B_0 + \sum_{i=1}^{n} x_i B_i \), and \( C(x) = C_0 + \sum_{i=1}^{n} x_i C_i \) are Hermitian matrices that depend affinely on \( x \). The data parameters are the Hermitian matrices \( A_i, B_i, \) and \( C_i \) for \( i = 0, \ldots, n \). The name of the GEVP arises from its resemblance to the well-known problem of minimizing the maximal generalized eigenvalue of the pencil \( [A, B] \), i.e. minimizing the largest \( \beta \) such that \( Av = \beta Bv \). It is easy to show that this problem can be expressed as

\[
\begin{cases}
\min_{\beta} & \beta \\
\text{s.t.} & \beta B - A > 0
\end{cases}
\] (9.34)

which is a simple SDP. The GEVP generalizes this program to the case where \( A \) and \( B \) also depend on the optimization variables.

### 9.2.3 Precoding for power optimization

The first important property of any optimization problem is its feasibility (admissibility), i.e. whether a solution exists. In other words, we need to verify whether for a given \( \gamma_o \) there exists a \( T \) such that the constraint in Equation (9.29) is satisfied:

\[
\min T \frac{|[HT]_{i,i}|^2}{\sum_{j \neq i} |[HT]_{i,j}|^2 + \sigma_i^2} \geq \gamma_o
\] (9.29a)
Since the noise variances are positive, the SINRs are strictly lower than the signal-to-interference ratios (SIRs), i.e.

\[
\sum_{j \neq i} \frac{|[HT]_{i,j}|^2}{|[HT]_{i,i}|^2 + \sigma_i^2} < \frac{|[HT]_{i,i}|^2}{\sum_{j \neq i} |[HT]_{i,j}|^2}
\]

for \(i = 1, \ldots, K\). By scaling \(T\) to \(aT\) for large enough \(a > 0\), the difference between the SIRs and the SINRs can be made insignificant.

Therefore, for the sake of examining the feasibility, the interesting metrics are the SIRs.

It was shown in [10] that there exists a \(T\) such that:

\[
\min_i \frac{|[HT]_{i,i}|^2}{\sum_{j \neq i} |[HT]_{i,j}|^2} \geq \gamma_o
\]

only if

\[
\gamma_o \leq \frac{1}{K - 1}
\]

If the effective channel \(H\) is full rank, then the condition results in \(\gamma_o \leq \infty\), i.e. any SIR is feasible. This is easily verified as the condition in Equation (9.29) can be satisfied by choosing \(T = aH^{-1}\) for large enough \(a > 0\). This choice of precoder inverts the channel and eliminates all interference. When the effective channel is rank deficient, the interference cannot be eliminated, and there is an upper bound on the maximal SIRs.

Experimenting with arbitrary channels shows that in almost all practical channels the bound can be achieved even for a fixed suboptimal receiver. For example, consider a rank 1 channel \(H\) with the normalized null vector \(u \in N \{HH^H\}\). Except for the case in which \(u_i = 0\) for some \(i = 1, \ldots, K\), the bound can always be attained by choosing

\[
T = H^\dagger \text{diag} \left\{ 1/\|u_i\| \right\} Q
\]

where \(Q\) is a matrix with unit diagonal elements and \([Q]_{i,j} = -1/(K - 1)\) for the nondiagonal \(i \neq j\) elements. This can be shown by considering the following chain:

\[
HT = HH^\dagger \text{diag} \left\{ 1/\|u_i\| \right\} Q = \text{diag} \left\{ 1/\|u_i\| \right\} Q
\]

where we have used \(HH^\dagger = I - uu^H\) and the fact that \(I \in N \{Q\}\). Substituting the above \(HT\) into the SIRs yields the maximal SIRs in rank \(K - 1\) channels, as follows:

\[
\frac{|[HT]_{i,i}|^2}{\sum_{j \neq i} |[HT]_{i,j}|^2} = \frac{1}{K - 1}, \quad i = 1, \ldots, K
\]

### 9.2.3.1 Conic optimization solution

We now show that the P problem of Equation (9.29) can be represented as a standard conic optimization program. Thus, using off-the-shelf optimization packages, we can numerically verify its feasibility and find its optimal solution. In order to use the standard forms of the conic programs, we must cast our problem constraints using the standard notations described in Section 9.2.2.

Using a real-valued slack variable \(p_o\), the program can be rewritten as:

\[
P (\gamma_o) : \begin{cases} \min_{T, p_o} & P_o \\ \text{s.t.} & \sum_{j \neq i} \frac{|[HT]_{i,j}|^2}{\sum_{j \neq i} |[HT]_{i,j}|^2 + \sigma_i^2} \geq \gamma_o, \quad i = 1, \ldots, K \\ & \text{Tr} \left\{ T^H H^H T \right\} \leq P_o \end{cases}
\]

The argument \(T\) of the \(P\) program is defined up to a diagonal phase scaling on the right, i.e. if \(T\) is optimal, then \(T \text{diag} \{ e^{i\phi_i} \}\), where \(\phi_i\) for \(i = 1, \ldots, K\) are arbitrary phases, is also optimal. This is easy to verify, as the phases do not change the objective nor the constraints. Therefore, we can restrict
ourselves to precoders in which $[HT]_{i,i} \geq 0$ for $i = 1, \ldots, K$, i.e. each has a nonnegative real part and a zero imaginary part. Taking this into account, we now recast the SINR constraints in standard form. Rearranging the constraints and using matrix notations, the constraints yield
\begin{equation}
\left(1 + \frac{1}{\gamma_0}\right) \| [HT]_{i,i} \|_2^2 \geq \left\| \frac{T^H H^H e_i}{\sigma_i} \right\|^2, \quad i = 1, \ldots, K \tag{9.42}
\end{equation}
Since $[HT]_{i,i} \geq 0$ for $i = 1, \ldots, K$, we can take the square root of $\| [HT]_{i,i} \|_2$, resulting in
\begin{equation}
\sqrt{1 + \frac{1}{\gamma_0}} [HT]_{i,i} \geq \left\| \frac{T^H H^H e_i}{\sigma_i} \right\|, \quad i = 1, \ldots, K \tag{9.43}
\end{equation}
which can be written as the SOCs
\begin{equation}
\begin{bmatrix}
\sqrt{1 + \frac{1}{\gamma_0}} [HT]_{i,i} \\
\frac{T^H H^H e_i}{\sigma_i}
\end{bmatrix} \succ_k 0, \quad i = 1, \ldots, K \tag{9.44}
\end{equation}
Similarly, the power constraint in Equation (9.41) can be reformulated using the vec($\cdot$) operator as
\begin{equation}
\| \text{vec} \left( H^T x^T \right) \| \leq \sqrt{P_o}, \quad \text{which is equivalent to the SOC:}
\begin{bmatrix}
\sqrt{P_o} \\
\text{vec} \left( H^T x^T \right)
\end{bmatrix} \succ_k 0 \tag{9.45}
\end{equation}
Using $p = \sqrt{P_o}$, the program in Equation (9.27) can now be cast in the standard SOCP form [43] as follows:
\begin{align*}
P (\gamma_0) : \\
\begin{cases}
\min_{\mathbf{T}, \mathbf{p}} & p \\
\text{s.t.} & \begin{bmatrix}
\sqrt{1 + \frac{1}{\gamma_0}} [HT]_{i,i} \\
\frac{T^H H^H e_i}{\sigma_i}
\end{bmatrix} \succ_k 0, \quad i = 1, \ldots, K \\
& \begin{bmatrix}
p \text{vec} \left( H^T x^T \right)
\end{bmatrix} \succ_k 0
\end{cases}
\end{align*} \tag{9.46}
and it can be efficiently solved using any standard SOCP package [42]. Such a solver can also numerically determine the feasibility of the optimization problem. A similar approach was taken in [40] in the context of transmit beamforming.
As explained in Section 9.2.2, each SOC constraint can be replaced with an SDP constraint using Equation (9.32). Thus, the problem can also be expressed as a standard SDP, as follows:
\begin{align*}
P (\gamma_0) : \\
\begin{cases}
\min_{\mathbf{T}, \mathbf{p}} & p \\
\text{s.t.} & A_i (\mathbf{T}) \succ 0, \quad i = 1, \ldots, K \\
& C (\mathbf{T}) \succ 0
\end{cases}
\end{align*} \tag{9.47}
where
\begin{equation}
A_i (\mathbf{T}) = \begin{bmatrix}
\sqrt{1 + \frac{1}{\gamma_0}} [HT]_{i,i} & \left[ e^H H^H \sigma_i \right] \\
\frac{T^H H^H e_i}{\sigma_i} & \sqrt{1 + \frac{1}{\gamma_0}} [HT]_{i,i} \mathbf{I}
\end{bmatrix} \tag{9.48}
\end{equation}
for $i = 1, \ldots, K$, and
\begin{equation}
C (\mathbf{T}) = \begin{bmatrix}
p \text{vec} \left( H^T x^T \right) \\
\text{vec}^H \left( H^T x^T \right)
\end{bmatrix} \tag{9.49}
\end{equation}
Solving SOCPs via SDP is not very efficient. Interior point methods that solve SOCP directly have a much better worst case complexity than do their SDP counterparts [43].
Define now the dual variables $\lambda_i > 0$ for $i = 1 \ldots K$ and denote $\Lambda = \text{diag} \{ \lambda_i \}$ and $G(\lambda_i) = H^H \Lambda H + H_i^H H_{Rx}$. If there exist $\lambda_i > 0$ such that

$$
\gamma_o = \frac{1}{\Lambda^{\frac{1}{2}} H G(\lambda_i) H^H \Lambda^\frac{1}{2}} - 1, \quad i = 1, \ldots, K \tag{9.50}
$$

holds, then the program is strictly feasible [10]. Also, if the condition in Equation (9.50) holds, then the optimal $T$ is of the form

$$
T = G^\dagger (\lambda_i) H^H \Lambda^\frac{1}{2} \text{diag} \{ \delta_i \} \tag{9.51}
$$

where $\delta_i$ are the positive weights that allocate the power between the users, as follows:

$$
\delta_i = \frac{1}{\sum_j \left[ \left( \frac{\gamma_o}{1 + \gamma_o} \right)^{-1} \right]} \lambda_j \sigma_j^2 \tag{9.52}
$$

$$
[F]_{i,j} = \Lambda^{\frac{1}{2}} H G(\lambda_i) H^H \Lambda^\frac{1}{2} \tag{9.53}
$$

for $i, j = 1, \ldots, K$. This structure of $T$ is unique within the range of $H_i^H H_{Rx}$. At this optimal solution, all the constraints are active, i.e., there are equal SINRs for all the subchannels. The optimal objective value is

$$
P_o = \sum_i \lambda_i \sigma_i^2 \tag{9.54}
$$

Equations (9.50)–(9.54) provide a simple strategy for designing the precoder.

Given a feasible $\gamma_o$, all one has to do is find $\lambda_i > 0$ which satisfy Equation (9.50). Once these are found, $T$ can be derived through Equations (9.51)–(9.54). In some special cases, these variables can be derived in closed form. Otherwise, [10] provides two alternative methods for finding these variables. The structure of (9.50) motivates a fixed-point iteration for finding $\lambda_i$. By rearranging Equation (9.50), we arrive at the following simple iteration:

$$
\lambda_i^{(n+1)} = \frac{\gamma_o}{1 + \gamma_o} \left[ HG^\dagger(\lambda_i^{(n)}) H^H \right]_{i,i}, \quad i = 1, \ldots, K \tag{9.55}
$$

Clearly, the optimal $\lambda_i$ satisfy this fixed point. If $P(\gamma_o)$ is feasible, then the above iteration will converge from any $\lambda_i^{(0)}$ to a set $\lambda_i^{(n)} > 0$ that satisfies Equation (9.50) [10].

### 9.2.4 Precoder for SINR optimization

The power optimization problem of Equation (9.29) and the SINR optimization problem of Equation (9.30) are inverse problems [10]:

$$
\gamma_o = S(\gamma_o) \quad P_o = P(S(\gamma_o)) \tag{9.56}
$$

In addition, the optimal objective value of each program is continuous and strictly monotonic increasing in its input argument

$$
\gamma_o > \gamma_o^* \Rightarrow P(\gamma_o) > P(\gamma_o^*) \quad P_o > P^*_o \Rightarrow S(\gamma_o) > S(P^*_o) \tag{9.57}
$$

Using Equations (9.56)–(9.57), we can solve $S(P_o)$ for a given $P_o$ by iteratively solving $P(\gamma_o)$ for different $\gamma_o$s. Due to the inversion property, if $P(\gamma)$, then its solution will be optimal also for $S(P_o)$. The strict monotonicity and continuity guarantees that a simple one-dimensional bisection search will efficiently find the required $\gamma_o$. This procedure is summarized in the following algorithm (see also [47, 10]).
\begin{equation}
S(P_o) \begin{align*}
1. & \gamma_{\text{max}} \leftarrow \text{MaxSINR} \\
2. & \gamma_{\text{min}} \leftarrow \text{MinSINR} \\
3. & \text{repeat} \\
4. & \gamma_o \leftarrow (\gamma_{\text{min}} + \gamma_{\text{max}})/2 \\
5. & \hat{P}_o \leftarrow P(\gamma_o) \\
6. & \text{if } \hat{P}_o \leq P_o \\
7. & \text{then } \gamma_{\text{min}} \leftarrow \gamma_o \\
8. & \text{else } \gamma_{\text{max}} \leftarrow \gamma_o \\
9. & \text{until } \hat{P}_o = P_o \\
10. & \text{return } \gamma_o
\end{align*}
\end{equation}

where MinSINR and MaxSINR define a range of relevant SINRs for a specific application, and where we have used the convention that \( \infty = P(\gamma_o) \) if it is infeasible. Theoretically, this means that the SINR optimization problem can be solved through the previous results concerning the power optimization.

The SINR optimization can be also cast as a standard GEVP program. Using a real-valued slack variable \( \gamma_o \), the problem can be rewritten as:

\begin{equation}
S(P_o) : \begin{cases}
\max_{T, \gamma_o} & \gamma_o \\
\text{s.t.} & \frac{|[H]_{i,i}|^2}{\sum_{j \neq i} |[H]_{i,j}|^2 + \sigma_i^2} \geq \gamma_o, \quad i = 1, \ldots, K \\
& \text{Tr} \left\{ T^H [H]^H \sigma_i \right\} \leq P_o
\end{cases}
\end{equation}

Although Equation (9.58) seems similar to Equation (9.41), it turns out to be more complicated. This is because the SINR matrix inequalities in Equation (9.48) are linear in \( \beta = \sqrt{1 + 1/\gamma_o} \) or in \( T \), but not in both simultaneously. Thus, when \( \beta \) is an optimization variable and not a parameter, these constraints are no longer LMIs. In fact, the sets which they define are not convex. Even so, we can still express them using generalized matrix inequalities as in Equations (9.47) and (9.48). If we rewrite the \( A_i(T) \)'s in Equation (9.48) and separate out the terms which are linear, we have:

\begin{equation}
A_i(T) = \beta A^1_i(T) - A^2_i(T)
\end{equation}

where \( A^1_i(T) \) and \( A^2_i(T) \) are matrices that depend affinely on \( T \), as follows:

\begin{align*}
A^1_i(T) &= \begin{bmatrix} \frac{[H]_{i,i}^2}{\sigma_i} & 0 \\
0 & \frac{[H]_{i,i}^2}{\sigma_i} \end{bmatrix} \\
A^2_i(T) &= \begin{bmatrix} 0 & -[\sigma_i^H \sigma_i] \\
-\sigma_i^H [H]_{i,i} & 0 \end{bmatrix}
\end{align*}

Using Equation (9.59), we can express \( S \) in the standard GEVP form

\begin{equation}
S(P_o) : \begin{cases}
\min_{T, \beta} & \beta \\
\text{s.t.} & \beta A^1_i(T) > A^2_i(T), \quad i = 1, \ldots, K \\
& A^1_i(T) > 0, \quad i = 1, \ldots, K \\
& C(T) > 0
\end{cases}
\end{equation}

which can be solved using appropriate software [45].
The SINR optimization problem can also be solved using the conditions in (9.50) by fixed-point iteration for finding $\lambda_i$. As explained in Equations (9.56)–(9.57), $S$ and $P$ are inverse problems. Thus, the optimal solution of the SINR optimization is also optimal for an inverse power optimization problem, and therefore must satisfy its optimality conditions as well. Thus, to optimize the SINRs, we need to find $\lambda_i > 0$ that satisfy Equations (9.50) and (9.54). Unfortunately, in this case, $\gamma_o$ is an optimization variable and not a parameter and has to be found as well. This can be overcome by adjusting the fixed-point iteration in Equation (9.55), as follows:

$$\tilde{\lambda}_i = \frac{1}{\left[ H G^\dagger \left( \lambda_i^{(n)} \right) H^H \right]_{i,i}}, \quad i = 1, \ldots, K$$

(9.62)

and then normalizing the result so that it will satisfy Equation (9.54):

$$\lambda_i^{(n+1)} = \frac{P_o \tilde{\lambda}_i}{\sum_j \sigma_j^2 \tilde{\lambda}_j}, \quad i = 1, \ldots, K$$

(9.63)

If this iteration converges to a fixed point $\lambda_i^{(n)} > 0$, then it will satisfy Equations (9.50) and (9.54). Numerous numerical simulations with arbitrary initial points and parameters show a rapid convergence rate.

### 9.2.5 Performance example

Consider a multiuser precoded downlink system. At each symbol’s period, the base station transmits using an $N \times K$ nonorthogonal signatures matrix $H_{Tx} = S$. The maximal average transmitted power is $P_o = K$, and the cross correlations between the signatures are denoted by $\rho_{i,j} = [S^H S]_{i,j}$ with $\rho_{i,i} = 1$ for all $i$. For simplicity, we assume ideal channels, i.e., $H_{Ch,i} = I$, and equal noise variances, i.e., $\sigma_i^2 = \sigma^2$. Denoting by $y$ the output vector of the multiple user receiver, we have

$$y = H_{Rx} S T b + H_{Rx} w$$

(9.64)

where from Chapter 5, $H_{Rx}$ is one of the standard filters, as follows:

- MF receiver: $H_{Rx} = S^H$;
- ZF receiver: $H_{Rx} = (S^H S)^{-1} S^H$;
- MMSE receiver: $H_{Rx} = S^H \left( S S^H + \sigma^2 I \right)^{-1}$.

An interesting result of the precoder is its performance in an equal power and equal cross correlations multiuser system, i.e., $\rho_{i,j} = \rho$ for all $j \neq i$ and $\sigma_i^2 = \sigma^2$.

When the matrices $H$ and $H_{Tx}$ have equal diagonal elements and equal off-diagonal elements, and the variances $\sigma_i^2 = \sigma^2$ are equal, due to the symmetry, it is clear that choosing $\lambda_i = P_o / (K \sigma_i^2)$ will satisfy the conditions in Equation (9.59). Therefore, the solution for the SINR optimization problem is:

$$\gamma_o = \frac{1}{1 - \left[ H \left( H^H + \frac{K \sigma^2}{P_o} H_{Tx}^H H_{Tx} \right)^{-1} H^H \right]_{i,i}}$$

(9.65)

$$T = c \left[ H^H H + \frac{K \sigma^2}{P_o} H_{Tx}^H H_{Tx} \right]^\dagger H^H$$

(9.66)

where $c$ is a constant that scales the matrix to satisfy the power constraint.

In Figure 9.6, we plot the output SINRs given by Equation (9.65) for the three linear receivers. For comparison, we also plot the output SINRs that result from similar systems without a precoder [48]. As expected, using the precoder always improves the output SINR.
9.3 CONVEX OPTIMIZATION-THEORY-BASED BEAMFORMING

In this section we discuss the joint design of transmit and receive beamforming for multicarrier multiple-input–multiple-output (MIMO) channels under a variety of design criteria. As already indicated in Section 9.2, this linear processing is commonly termed linear precoding at the transmitter and equalization at the receiver. Instead of considering each design criterion in a separate way, we generalize the existing results by developing a unified framework based on considering two families of objective functions that embrace most reasonable criteria to design a communication system: Schur-concave and Schur-convex functions. Once the optimal structure of the transmit–receive processing is known, as in Section 9.2, the design problem simplifies and can be formulated within the framework of convex optimization theory, in which a number of interesting design criteria can be easily accommodated and efficiently solved, even though closed-form expressions may not exist.

A general convex optimization problem (convex program), introduced in Section 9.2, will be further specified in this section as follows [65]:

\[
\begin{align*}
\min_{x} & \quad f_{0}(x) \\
\text{s.t.} & \quad f_{i}(x) \leq 0, \quad 1 \leq i \leq m \\
& \quad h_{i}(x) = 0, \quad 1 \leq i \leq p
\end{align*}
\]

where \( x \in \mathbb{R}^{n} \) is the optimization variable, \( f_{0}(x), \ldots, f_{m}(x) \) are convex functions, and \( h_{1}(x), \ldots, h_{p}(x) \) are linear functions (affine functions). The function \( f_{0} \) is the objective function or cost function. The inequalities \( f_{i}(x) \leq 0 \) are called inequality constraints, and the equations \( h_{i}(x) = 0 \) are called equality constraints. When the functions \( f_{i} \) and \( h_{i} \) are linear (affine), the problem is called a linear program (LP) and is much simpler to solve.

In general, as already indicated in Section 9.2, some manipulations are required to convert the problem into a convex one (unfortunately, this is not always possible). The interest of expressing a problem in convex form is that although an analytical solution may not exist and the problem may be difficult to solve (it may have hundreds of variables and a nonlinear nondifferentiable objective function), it can still be solved (numerically) very efficiently [64]. Another interesting feature of expressing a problem in convex form is that additional constraints can be straightforwardly added, as long as they are convex.

In some cases, convex optimization problems can be analytically solved using the Karush–Kuhn–Tucker (KKT) optimality conditions, and closed-form expressions can be obtained. In general,
however, one must resort to iterative methods [64, 67]. Interior-point methods can be used to iteratively solve convex problems [72]. In addition, the difference between the objective value at each iteration and the optimum value can be upper bounded using duality theory [64, 67]. This allows the utilization of nonheuristic stopping criteria such as stopping when some prespecified resolution has been reached. Another interesting family of iterative methods is cutting-plane methods [64].

In the sequel we will also use some results from majorization theory. For these purposes we need the following definitions (D) [63]:

(D1) For any $x \in R^n$, let $x_{[i]} \geq \cdots \geq x_{[n]}$ denote the components of $x$ in decreasing order (also termed order statistics of $x$).

(D2) Let, $x, y \in R^n$. Vector $x$ is majorized by vector $y$ (or $y$ majorizes $x$) if

$$
\sum_{i=1}^{k} x_{[i]} \leq \sum_{i=1}^{k} y_{[i]}, \quad i \leq k \leq n - 1, \quad \sum_{i=1}^{n} x_{[i]} = \sum_{i=1}^{n} y_{[i]}
$$

and represent it by $x \prec y$.

(D3) A real-valued function $\phi$ defined on a set $A \subseteq R^n$ is said to be Schur-convex on $A$ if $x \prec y$ on $A \Rightarrow \phi(x) \leq \phi(y)$. Similarly, $\phi$ is said to be Schur-concave on $A$ if $x \prec y$ on $A \Rightarrow \phi(x) \geq \phi(y)$. As a consequence, if $\phi$ is Schur-convex on $A$, then $-\phi$ is Schur-concave on $A$ and vice versa. It is important to remark that the sets of Schur-convex and Schur-concave functions do not form a partition of the set of all functions. In fact, neither are the two sets disjoint (the intersection is not empty), nor do they cover the entire set of all functions. By using these definitions we additionally introduce the following results [63]:

(R1) Let $R$ be an $n \times n$ Hermitian matrix with diagonal elements denoted by the vector $d$ and eigenvalues denoted by the vector $\lambda$. Then $d \prec \lambda$.

(R2) Let $x \in R^n$ and $1 \in R^n$ denote the constant vector with $1_i = \frac{\lambda}{n}$, $\sum_{i=1}^{n} x_i / n$. Then $1 \prec x$.

(R3) For any $x \in R^n$, there exists a real symmetric (and therefore Hermitian) matrix with equal diagonal elements and eigenvalues given by $x$.

### 9.3.1 Multicarrier MIMO signal model

We analyse a communication system with $n_T$ transmit and $n_R$ receive antennas. To cope with the frequency-selectivity of the channel, we use a multicarrier signal so that we have

$$
y_k = H_k s_k + n_k, \quad 1 \leq k \leq N
$$

(9.67)

where $k$ denotes the carrier index, $N$ is the number of carriers, $s_k \in R^{n_T \times 1}$ is the transmitted vector, $H_k \in R^{n_R \times n_T}$ is the channel matrix, $y_k \in R^{n_R \times 1}$ is the received signal vector, and $n_k \in R^{n_R \times 1}$ is $n_k \sim CN(0, R_n)$ a zero-mean circularly symmetric complex Gaussian noise vector with arbitrary covariance matrix $R_n$. The channel is assumed fixed during the transmission of a block and known at both sides of the communication link as well as the noise covariance matrix.

At each carrier $k$, the matrix channel has $K \leq \min(n_T, n_R)$ channel eigenmodes or spatial subchannels (i.e., nonvanishing singular values of the channel matrix) [50] (see also Section 4.14). We can use them as a means of spatial multiplexing [51] to transmit simultaneously $L_k$ symbols by having $L_k$ established substreams. Notice that established substreams and spatial subchannels (or channel eigenmodes) are different concepts that may or may not coincide, depending on whether the channel is diagonalized or not. This will be further elaborated in the next section. In a practical system, we will typically have to have $L_k \leq K$ an acceptable performance. In the sequel boldface capital letters denote matrices, boldface lower-case letters denote column vectors, and italics denote scalars. $[X]_{i,j}$ (also $[X]_{i,j}$) and $[X]_{i,j}$ denote the $(i$th, $j$th) element and $j$th column of matrix $X$, respectively. By $A \succeq B$, we mean that $A - B$ is positive semidefinite. The trace, determinant, and Frobenius norm of a matrix are denoted by $\text{Tr}(\cdot)$, $| \cdot |$, and $|| \cdot ||_F$, respectively. By diag ($\{X_l\}$), we denote a block-diagonal matrix with diagonal blocks given by the set $\{X_l\}$. The gradient of a function with respect to $x$ is written as $\nabla_x f (x)$. We define $(x)^{\Delta} \overset{\Delta}{=} \max (0, x)$. The transmitted vector at the
Figure 9.7 Matrix processing and multiple beamforming interpretations of the communication system. (We assume for the clarity of the figure that $L_k = L \forall k$.) (a) Matrix processing interpretation at carrier $k$. (b) Multiple beamforming interpretation at carrier $k$.

$k$th carrier after linear precoding can be now represented as (see Figure 9.7(a)).

$$s_k = B_k x_k = \sum_{i=1}^{L_k} b_{k,i} x_{k,i}$$  (9.68)

where $x_k \in \mathbb{R}^{L_k \times 1}$ represents the $L_k$ transmitted symbols. We assume zero-mean unit-energy uncorrelated (white) symbols, i.e., $\mathbb{E}[x_k x_k^H] = I_{L_k}$. $B_k \in \mathbb{R}^{n_T \times L_k}$ is the precoding matrix, $b_{k,i} \triangleq [B_k]_{i,:}$, and $x_{k,i} \triangleq [x_k]$. Each column of $B_k$ can be considered as a different beamvector corresponding to each transmitted symbol, resulting into multiple beamforming architecture shown in Figure 9.7(b).

Note that if only one symbol is transmitted per carrier ($L_k = 1 \forall k$), then Equation (9.68) reduces to a classical beamforming structure with a single beamvector: $s_k = b_k x_k$. The transmitter is constrained in its average total transmit power:

$$\sum_{k=1}^{N} \mathbb{E}[\|B_k x_k\|^2] = \sum_{k=1}^{N} \|B_k\|^2_F \leq P_T$$  (9.69)

where $P_T$ is the power in units of energy per block-transmission (or, equivalently, per OFDM symbol).

The received vector at the $k$th carrier after the equalizer is

$$\hat{x}_k = A_k^H y_k$$  (9.70)

where $A_k^H \in \mathbb{R}^{L_k \times n_R}$ is the equalizer matrix, and $\hat{x}_k \in \mathbb{R}^{L_k \times 1}$ is the estimation of $x_k$. Again, each column of $A_k$ can be interpreted as a beamvector adapted to each spatial channel substream at carrier $k$, i.e., $a_{k,i} = A_k^H y_k$ [see Figure 9.7(b)].

In the sequel, only independent processing at each carrier has been considered, called the *carrier-noncooperative approach* [49] [see Figure 9.8(a)]. This scheme, however, can be further generalized by allowing cooperation among carriers, called *carrier-cooperative approach* [49] [see Figure 9.8(b)]. The signal model is obtained by stacking the vectors corresponding to all carriers (e.g., $x^T = [x_1^T, \ldots, x_N^T]$), by considering global transmit and receive matrices $B \in \mathbb{R}^{n_T \times N L_T}$ (the transmit...
power constraint reduces to $\|B\|_F^2 \leq P_T$ and $A^H \in R^{L_T \times (n_R \cdot N)}$, where $L_T = \sum_{k=1}^{N} L_k$ is the total number of transmitted symbols, and by defining the global channel as $H = \text{diag} ([H_k]) \in R^{(n_R \cdot N) \times (n_T \cdot N)}$. This model can cope with intermodulation terms, unlike the noncooperative model, that implicitly assumes orthogonal carriers.

9.3.2 Channel diagonalization

For some specific design criteria, the system optimization problem is greatly simplified because the channel turns out to be diagonalized by the precoder-equalizer processing, which allows a scalarization of the problem (all matrix equations are substituted with scalar ones). Examples are the minimization of the (weighted) sum of the MSEs of all channel spatial substreams [52–54], the minimization of the determinant of the MSE matrix [55], and the maximization of the mutual information [50, 56, 57].

In [49], these results are generalized by developing a unified framework. Instead of analyzing each design criterion in a separate way, the design is based on the minimization of some arbitrary objective function of the MSEs of all channel substreams $f_0 (\{\text{MSE}_k,i\})$, where $\text{MSE}_k,i$ is the MSE of the $i$th spatial substream at the $k$th carrier (objective functions of the SINRs and of the BERs are also readily incorporated).
9.3.2.1 Optimum equalizer

To design the system, we first easily derive the optimum equalizer $A_k$’s, assuming the precoders $B_i$’s fixed, and then deal with, the derivation of the optimal precoders $B_i$’s. This two-step derivation has been independently used in [49] and [58]. The MSE matrix at the $k$th carrier is defined as the covariance matrix of the error vector (given by $e_k \triangleq \hat{x}_k - x_k$):

$$E_k(B_k, A_k) \triangleq E[(\hat{x}_k - x_k)(\hat{x}_k - x_k)^H] = A_k^H R_{y_k} A_k + I - A_k^H H_k B_k - B_k^H H_k^H A_k$$  \hspace{1cm} (9.71)

where $R_{y_k} \triangleq E[y_k y_k^H] = H_k B_k B_k^H H_k^H + R_{n_k}$. The MSE of the $(k, i)$th substream is the $i$th diagonal element of $E_k$, i.e.,

$$\text{MSE}_{k,i}(B_k, a_{k,i}) = [E_k]_{ii} = a_{k,i}^H R_{y_k} a_{k,i} + 1 - a_{k,i}^H H_k b_{k,i} - b_{k,i}^H H_k^H a_{k,i}$$  \hspace{1cm} (9.72)

where $a_{k,i}$ (or $b_{k,i}$) is the $i$th column of $A_k$ (or $B_k$). Expression (9.72) is nonconvex in $(B_k, a_{k,i})$ whereas, for a given $B_k$, MSE$_{k,i}$ is convex in $a_{k,i}$ and independent of the other columns of $A_k$ and of the other carriers. This means that each $a_{k,i}$ can be independently optimized. To obtain the optimal equalizer$A_k^{\text{opt}}$ in a more direct way, it suffices to find $A_k$ such that the diagonal elements of $E_k$ are minimized. This can be done regardless of the specific choice of the objective function $f_0$ since we know it is increasing in each argument. Alternatively, we can obtain $A_k^{\text{opt}}$ so that $E_k(B_k, A_k^{\text{opt}}) \leq E_k(B_k, A_k)$, which in particular implies that the diagonal elements are minimized (in fact, both criteria are equivalent as shown in [59]). In other words, we want to solve $\min_{A_k} c^H E_k(B_k, A_k) c$, $\forall c$.

Setting the gradient of $c^H E_k c = \text{Tr}(E_k c c^H)$ to zero $\nabla A_k \text{Tr}(E_k c c^H) = R_{y_k} A_k c c^H - H_k B_k c c^H = 0$, $\forall c$ and particularizing for all the vectors of the canonical base, it follows that:

$$A_k^{\text{opt}} = (H_k B_k B_k^H H_k^H + R_{n_k})^{-1} H_k B_k$$

Expression (9.73) is the linear minimum MSE (LMMSE) receiver or Wiener filter [see Section 5]. Using the optimal equalizer $A_k^{\text{opt}}$, we obtain the following concentrated error matrix:

$$E_k(B_k) \triangleq E_k(B_k, A_k^{\text{opt}}) = I - B_k^H H_k^H (H_k B_k B_k^H H_k^H + R_{n_k})^{-1} H_k B_k$$

where we have used the matrix inversion lemma, $(A + BCD)^{-1} = A^{-1} - A^{-1} B(DA^{-1} B + C^{-1})^{-1} DA^{-1}$ and we have defined $R_{H_k} \triangleq H_k^H R_{n_k} H_k$ (note that the eigenvectors and eigenvalues of $R_{H_k}$ are the right singular vectors and the squared singular values, respectively, of the whitened channel $R_{n_k}^{-1/2} H_k$). However, many objective functions are naturally expressed as functions of the SINR of each substream. The SINR at the $k$th carrier and the $i$th spatial substream is:

$$\text{SINR}_{k,i} \triangleq \frac{\|a_{k,i}^H H_k b_{k,i}\|^2}{\|a_{k,i}^H R_{H_k} a_{k,i}\|} \leq \frac{b_{k,i}^H H_k^H R_{n_k}^{-1} H_k b_{k,i}}{\|a_{k,i}^H R_{H_k} a_{k,i}\|}$$

(9.75)

where $R_{H_k} \triangleq H_k B_k B_k^H H_k^H + R_{n_k} - H_k b_{k,i} b_{k,i}^H H_k^H$ is the interference-plus-noise covariance matrix seen by the $(k, i)$th substream, the inequality comes from Cauchy–Schwarz’s inequality [60] [with vectors $(R_{H_k}^{-1/2} H_k b_{k,i})$ and $(R_{H_k}^{-1/2} a_{k,i})$], and the upper bound is achieved by $a_{k,i} \propto R_{H_k}^{-1} H_k b_{k,i} \propto R_{H_k}^{+1} H_k b_{k,i}$, i.e. the Wiener filter again. Noting that the MSE can be expressed as:

$$\text{MSE}_{k,i} = \left[ I + B_k^H H_k^H R_{n_k}^{-1} H_k B_k \right]^{-1}_{ii} = \frac{1}{1 + b_{k,i}^H H_k^H R_{n_k}^{-1} H_k b_{k,i}}$$

(9.76)

the SINR can be easily related to the MSE as:

$$\text{SINR}_{k,i} = \frac{1}{\text{MSE}_{k,i}} - 1$$  \hspace{1cm} (9.77)

So, maximizing the SINR is equivalent to minimizing the MSE. The performance of a digital communications system is given by the bit error rate (BER). Under the Gaussian assumption, the symbol
error probability $P_e$ will be analytically expressed as in the previous section as a function of the SINR $P_e(\text{SINR}) = \alpha Q(\sqrt{\beta}\text{SINR})$ where $\alpha$ and $\beta$ are constants that depend on the signal constellation, and $Q$ is the Q-function defined as $Q(x) = (1/\sqrt{2\pi}) \int_x^{\infty} e^{-\lambda^2/2} d\lambda$. It is sometimes convenient to use the Chernoff upper bound of the tail of the Gaussian distribution function $Q(x) \leq (1/2) e^{-x^2/2}$ to approximate the symbol error probability as $P_e \approx (1/2) \alpha e^{-\beta/2\text{SINR}}$ (which becomes a good approximation for high values of the SINR). The BER can be approximately obtained from the symbol error probability (assuming that a Gray encoding is used to map the bits into the constellation points) as $\text{BER} \approx P_e/k$ where $k = \log_2 M$ is the number of bits per symbol, and $M$ is the constellation size.

It can be seen from Figure 9.9 that BER is a convex function.

### 9.3.2.2 Optimum precoder

To obtain the set of precoder matrices $\{B_k\}$, we now consider the minimization of an arbitrary objective function of the diagonal elements of Equation (9.74). We start with optimization problem defined by:

$$
\min_B f_0(\mathbf{d}(\mathbf{E}(B)))
$$

s.t. \quad \text{Tr}(\mathbf{BB}^H) \leq P_T \quad (9.78)

where matrix $\mathbf{B} \in \mathbb{R}^{n_T \times L}$ is the optimization variable, $\mathbf{d}(\mathbf{E}(\mathbf{B}))$ is the vector of diagonal elements of the MSE matrix $\mathbf{E}(\mathbf{B}) = (\mathbf{I} + \mathbf{B}^H \mathbf{R}_H \mathbf{B})$ [the diagonal elements of $\mathbf{E}(\mathbf{B})$ are assumed in decreasing order w.l.o.g., $\mathbf{R}_H \in \mathbb{R}^{n_T \times n_T}$ is a positive semidefinite Hermitian matrix, and $f_0 : \mathbb{R}^L \rightarrow \mathbb{R}$ is an arbitrary objective function (increasing in each variable). It then follows that there is an optimal solution $\mathbf{B}$ of at most rank $\tilde{L} \leq \min(L, \text{rank} (\mathbf{R}_H))$ rank with the following structure [49]:

- If $f_0$ is Schur-concave, then

$$
\mathbf{B} = \mathbf{U}_{H,1} \sum_{\mathbf{B},1}
$$

where $\mathbf{U}_{H,1} \in \mathbb{R}^{n_T \times \tilde{L}}$ has as columns the eigenvectors of $\mathbf{R}_H$ corresponding to the $\tilde{L}$ largest eigenvalues in increasing order, and $\sum_{\mathbf{B},1} = [0 \text{ diag}([\sigma_{\mathbf{B},1}])] \in \mathbb{R}^{L \times L}$ has zero elements, except along the rightmost main diagonal (which can be assumed real).
If \( f_0 \) is Schur-convex, then
\[
B = U_{H,1} \sum_{B,1} V_{B}^H
\]  
(9.80)

where \( U_{H,1} \) and \( \sum_{B,1} \) are defined as before, and \( V_{B} \in R^{L \times L} \) is a unitary matrix such that \((I + B^H R_B B)^{-1}\) has identical diagonal elements. This rotation can be computed using the algorithm given in [62] or with any rotation matrix \( Q \) that satisfies \(|Q_{ii}| = |Q_{jj}|, \forall i, k, l, \) such as the discrete Fourier transform (DFT) matrix or the Hadamard matrix (when the dimensions are appropriate such as a power of two [61]).

For the simple case in which only one symbol per carrier is transmitted at each transmission, i.e. a single spatial eigenmode \( L = 1 \) is utilized, Equations (9.78)–(9.80) simplify, and the diagonal structure simply means that the spatial subchannel (eigenmode) with highest gain is used.

For Schur-concave objective functions, the global communication process including pre- and post-processing \( A^H \text{HB} \) is fully diagonalized [see Figure 9.10(b)] as well as the MSE matrix \( E \).

Among the \( L \) established substreams, only \( \bar{L} \) are associated with nonzero channel eigenvalues, whereas the remainder \( L_0 = L - \bar{L} \) are associated with zero eigenvalues. Having in mind that \( A = (HBB^H + R_B)^{-1} \text{HB} = R_B^{-1} \text{HB}(I + B^H R_B^{-1} \text{HB})^{-1} \) the global communication process is \( \hat{x} = (I + \sum_{B,1} H_{B} \sum_{B,1})^{-1} \sum_{B,1} H_{B} \sum_{B,1} x + w \) or,

\[
\hat{x}_i = \begin{cases} 0, & 1 \leq i \leq L_0 \\ \frac{\sigma_{B,i}^2 - \lambda_{H,i} \hat{x}_i}{1 + \sigma_{B,i}^2 - \lambda_{H,i}} \hat{x}_i + \frac{\sigma_{B,i}^2}{1 + \sigma_{B,i}^2 - \lambda_{H,i}} u_i, & L_0 < i \leq L 
\end{cases}
\]  
(9.81)

where \( D_{H,1} = \text{diag}(\{\lambda_{H,i}\}_{i=1}^{\bar{L}}) \), the \( \lambda_{H,i} \)s are the \( \bar{L} \) largest eigenvalues of \( R_H \) in increasing order, the \( \sigma_{B,i}^2 \)s represent the allocated power, and \( w \) is a normalized equivalent white noise.

The MSE matrix is \( E = (I + \sum_{B,1} H_{B} \sum_{B,1})^{-1} \), and the corresponding MSEs are given by:

\[
\text{MSE}_i = \begin{cases} 1, & 1 \leq i \leq L_0 \\ \frac{1}{1 + \sigma_{B,i}^2 - \lambda_{H,i}}, & L_0 < i \leq L 
\end{cases}
\]  
(9.82)

Similarly, using Equation (9.77) we have

\[
\text{SINR}_i = \begin{cases} 0, & 1 \leq i \leq L_0 \\ \frac{1}{\sigma_{B,i}^2 - \lambda_{H,i}}, & L_0 < i \leq L 
\end{cases}
\]  
(9.83)

For Schur-convex objective functions, the global communication process including pre- and post-processing \( A^H \text{HB} \) is diagonalized only up to a very specific rotation of the data symbols [see Figure 9.10(c)], and the MSE matrix \( E \) is nondiagonal with equal diagonal elements (equal MSEs). In particular, assuming a pre-rotation of the data symbols at the transmitter \( \hat{x} = V_{B}^H x \) and a post-rotation of the estimates at the receiver \( \hat{x} = V_{B}^H \hat{x} \), the same diagonalizing results of Schur-concave functions apply [see Figure 9.10(c)]. Since the diagonal elements of the MSE matrix \( E = (I + B^H R_B B)^{-1} \) are equal whenever the appropriate rotation is included, the MSEs are identical and given by:

\[
\text{MSE}_i = \frac{1}{L} \text{Tr}(E) = \frac{1}{L} \left( L_0 + \sum_{j=1}^{\bar{L}} \frac{1}{1 + \sigma_{B,j}^2 - \lambda_{H,j}} \right), \quad 1 \leq i \leq L 
\]  
(9.84)
Similarly by using Equation (9.77) we have:

$$\text{SINR}_i = \frac{L}{L_0 + \sum_{j=1}^{i} \frac{1}{\sigma_{B_{ki}}^2 \lambda_{H_{ij}}}} - 1, \quad 1 \leq i \leq L \tag{9.85}$$

The result defined by Equations (9.78)–(9.80) is easily extended to the multicarrier case as follows. For any carrier $k$, consider the matrices corresponding to the rest of the carriers $\{B_l\}_{l \neq k}$ fixed, and Equations (9.78)–(9.80) can be directly invoked to show the optimal structure for $B_k$.

### 9.3.3 Convex optimization-based beamforming

In the sequel we use notation $z_{k,j} \triangleq \sigma_{B_{kj}}^2$ and $\lambda_{k,j} \triangleq \lambda_{H_{kj}}$. Note also that for Schur-concave functions with $L_k > \text{rank}(R_k)$, the $L_k - \tilde{L}_k$ substreams associated with zero eigenvalues are simply ignored in the optimization process.
9.3.3.1 Minimization of the ARITH-MSE

For the minimization of the (weighted) arithmetic mean of the MSEs (ARITH-MSE) the objective function is

\[
    f_0 \left( \{\text{MSE}_{k,i}\} \right) = \sum_{k,i} \left( w_{k,i} \text{MSE}_{k,i} \right) \quad (9.86)
\]

The function \( f_0 (\{x_i\}) = \sum_i (w_i x_i) \) (assuming \( x_i \geq x_{i+1} \)) is minimized when the weights are in increasing order \( w_i \leq w_{i+1} \), and it is then a Schur-concave function [49]. So, the objective function Equation (9.86) is Schur-concave on each carrier \( k \), and by Equations (9.78)–(9.80), the diagonal structure is optimal, and the MSEs are given by Equation (9.82). The problem in convex form (the objective is convex and the constraints linear) is:

\[
    \min_{\{z_{k,i}\}} \quad \sum_{k,i} \frac{1}{1 + \lambda_{k,i}, z_{k,i}} \quad \text{s.t.} \quad \sum_{k,i} z_{k,i} \leq P_T \quad (9.87)
\]

\[ z_{k,i} \geq 0, \quad 1 \leq k \leq N, \quad 1 \leq i \leq \bar{L}_k \]

This particular problem can be solved very efficiently because the solution has a water-filling interpretation (from the KKT optimality conditions):

\[
    z_{k,i} = \left( \mu^{-1/2} w_{k,i}^{1/2} \lambda_{k,i}^{-1/2} - \lambda_{k,i}^{-1} \right)^+ \quad (9.88)
\]

where \( \mu^{-1/2} \) is the water-level chosen to satisfy the power constraint with equality.

9.3.3.2 Minimization of the GEOM-MSE

The objective function corresponding to the minimization of the weighted geometric mean of the MSEs (GEOM-MSE) is

\[
    f_0 \left( \{\text{MSE}_{k,i}\} \right) = \prod_{k,i} (\text{MSE}_{k,i})^{w_{k,i}} \quad (9.89)
\]

The function \( f_0 (\{x_i\}) = \prod_i x_i^{w_i} \) (assuming \( x_i \geq x_{i+1} > 0 \)) is minimized when the weights are in increasing order \( w_i \leq w_{i+1} \), and it is then a Schur-concave function [49]. So, the objective function, Equation (9.90), is Schur-concave on each carrier \( k \) and by Equations (9.78)–(9.80), the diagonal structure is optimal, and the MSEs are given by Equation (9.82). The problem in convex form (since the objective is log-convex, it is also convex [64]) is:

\[
    \min_{\{z_{k,i}\}} \quad \prod_{k,i} \left( \frac{1}{1 + \lambda_{k,i}, z_{k,i}} \right)^{w_{k,i}} \quad \text{s.t.} \quad \sum_{k,i} z_{k,i} \leq P_T \quad (9.90)
\]

\[ z_{k,i} \geq 0, \quad 1 \leq k \leq N, \quad 1 \leq i \leq \bar{L}_k \]

This problem also has a water-filling solution (from the KKT optimality conditions):

\[
    z_{k,i} = \left( \mu^{-1} w_{k,i} - \lambda_{k,i}^{-1} \right)^+ \quad (9.91)
\]

where \( \mu^{-1} \) is the water-level chosen to satisfy the power constraint with equality. Note that for \( w_{k,i} = 1 \), Equation (9.91) becomes the classical capacity-achieving water-filling solution [50, 56] [see also Equation (4.201)].


**9.3.3.3 Minimization of $|E|$**

The minimization of the determinant of the MSE matrix was considered in [55]. This particular criterion is easily accommodated in the above framework as a Schur-concave function of the diagonal elements of the MSE matrix $E$. For the carrier-noncooperative case, simply consider the global MSE matrix defined as $E = \text{diag}(E_1, \ldots, E_N)$. Using the fact that $X \succeq Y \Rightarrow |X| \geq |Y|$, it follows that $|E|$ is minimized for the choice of the equalizer matrix given by Equation (9.73). From Equation (9.74), it is clear that $|E|$ does not change if the precoder matrices $B_k$s are post-multiplied by a unitary matrix (a rotation).

Therefore, we can always choose a rotation matrix so that $E$ is diagonal without loss of optimality (see also [15]), and then

$$|E| = \prod_j \lambda_j(E) = \prod_j |E|_{jj} \quad (9.92)$$

Therefore, the minimization of $|E|$ is equivalent to the minimization of the (unweighted) product of the MSEs as in Section 9.3.3.2.

**9.3.3.4 Maximization of mutual information**

The maximization of mutual information can be used to obtain a capacity achieving solution [56] (see also Appendix in Chapter 4)

$$\max_Q I = \log |I + R_n^{-1}HQH^H| \quad (9.93)$$

where $Q$ is the precoder covariance matrix. Using the fact that $|I + XY| = |I + YX|$ and that $Q = BB^H$ [from Equation (9.68)], the mutual information can be expressed as [59]:

$$I = -\log |E| \quad (9.94)$$

and therefore, the maximization of $I$ is equivalent to the minimization of $|E|$ treated in Section 9.3.3.3. Hence, the minimization of the unweighted product of the MSEs, the minimization of the determinant of the MSE matrix, and the maximization of the mutual information are all equivalent criteria with the solution given by a channel-diagonalizing structure and the classical capacity-achieving water-filling for the power allocation:

$$z_{k,i} = \left(\mu^{-1} - \lambda_{k,i}^{-1}\right)^+ \quad (9.95)$$

**9.3.3.5 Minimization of the MAX-MSE**

In general, average BER is dominated by the substream with highest MSE. It makes sense then to minimize the maximum of the MSEs (MAX-MSE). Then the objective function is

$$f_0 \left\{ \{\text{MSE}_{k,i}\} \right\} = \max_{k,i} \{\text{MSE}_{k,i}\} \quad (9.96)$$

The function $f_0 (\{x_i\}) = \max, \{x_i\}$ is a Schur-convex function [49]. So, the objective function in Equation (9.96) is Schur-convex on each carrier $k$. Therefore, Equations (9.78)–(9.80), the optimal solution has a nondiagonal MSE matrix $E_k$ with equal diagonal elements given by Equation (9.84), which have to be minimized (scalarized problem). After minimizing the MSEs, we must still obtain the optimal rotation matrices so that the diagonal elements of the MSE matrices $E_k$s are identical. The scalarized problem in convex form (the objective is linear and the constraints are all
This problem has a multilevel water-filling solution (from the KKT optimality conditions):

\[ z_{k,i} = \left( \bar{\mu}^{1/2} - \lambda_{k,i}^{1/2} \right) \left( -\sum_{l=1}^{N} \lambda_{l,i}^{1/2} \right) \]  

where \( \{\bar{\mu}_k^{1/2}\} \) are multiple water levels chosen to satisfy the constraints on \( t \) and the power constraint all with equality. For the case of single beamforming (i.e., \( L_k = 1 \)), the solution simplifies to:

\[ z_k = \lambda_k^{-1} P_T \sum_{l=1}^{N} \lambda_l^{-1} \]  

For the single-carrier case (or multicarrier cooperative approach), problem (9.97) simplifies to the minimization of the unweighted ARITH-MSE considered in Section 9.3.3.1 with solution \( z_i = (\mu^{-1/2} - \lambda_i^{-1})^+ \).

### 9.3.3.6 Maximization of the ARITH-SINR

Since, in Equations (9.78)–(9.80), we assumed \( \text{MSE}_{k,i} \geq \text{MSE}_{k,i+1} \), the SINRs are in increasing order \( \text{SINR}_{k,i} \leq \text{SINR}_{k,i+1} \). The objective function to be minimized for the maximization of the (weighted) arithmetic mean of the SINRs (ARITH-SINR) is

\[ \tilde{f}_0 (\{\text{SINR}_{k,i}\}) = -\sum_{k,i} (w_{k,i} \text{SINR}_{k,i}) \]  

which can be expressed as a function of the MSEs using Equation (9.77) as:

\[ f_0 (\{\text{MSE}_{k,i}\}) = \tilde{f}_0 (\{\text{MSE}_{k,i}^{-1} - 1\}) = -\sum_{k,i} w_{k,i} (\text{MSE}_{k,i}^{-1} - 1) \]  

The function \( f_0 (\{x_i\}) = -\sum_i w_i (x_i^{-1} - 1) \) (assuming \( x_i \geq x_{i+1} > 0 \)) is minimized when the weights are in increasing order \( w_i \leq w_{i+1} \), and it is then a Schur-concave function [49]. So, the objective function of Equation (9.101) is Schur-concave on each carrier \( k \). Therefore, by Equation (9.78)–(9.80), the diagonal structure is optimal and the SINRs are given by Equation (9.83). The problem expressed in convex form (it is actually an LP since the objective and the constraints are all linear) is:

\[ \max_{\{z_{k,i}\}} \sum_{k,i} w_{k,i} \lambda_{k,i} z_{k,i} \quad \text{s.t.} \quad \sum_{k,i} z_{k,i} \leq P_T \]  

\[ z_{k,i} \geq 0, \quad 1 \leq k \leq N, \quad 1 \leq i \leq \bar{L}_k \]
9.3.3.7 Maximization of the GEOM-SINR

The objective function to be minimized for the maximization of the (weighted) geometric mean of the SINRs (GEOM-SINR) is:

$$\tilde{f}_0 \left( \{\text{SINR}_{k,i} \} \right) = - \prod_{k,i} \left( \text{SINR}_{k,i} \right)^{w_{k,i}}$$

(9.103)

which can be expressed as a function of the MSEs using Equation (9.77) as

$$\tilde{f}_0 \left( \{\text{MSE}_{k,i} \} \right) = \tilde{f}_0 \left( \{\text{MSE}_{k,i}^{-1} - 1 \} \right) = - \prod_{k,i} \left( \text{MSE}_{k,i}^{-1} \right)^{w_{k,i}}$$

(9.104)

The maximization of the product of the SINRs is equivalent to the maximization of the sum of the SINRs expressed in decibels. The function

$$f_0 \left( \{x_i \} \right) = - \prod_i \left( x_i^{-1} - 1 \right)^{w_i}$$

(assuming $0.5 \geq x_i \geq x_i + 1 > 0$) is minimized when the weights are in increasing order $w_i \leq w_{i+1}$, and it is then a Schur-concave function [49]. So, the objective function (9.104) is Schur-concave on each carrier $k$, provided that $\text{MSE}_{k,i} \leq 0.5 \forall k, i$ (this is a mild assumption since a MSE greater than 0.5 is unreasonable for a practical communication system). Therefore, by Equations (9.78)–(9.80), the diagonal structure is optimal, and the SINRs are given by Equation (9.83). The problem expressed in convex form (the weighted geometric mean is a concave function [64, 65]) is:

$$\max \left\{ z_{k,i} \right\} \prod_{k,i} \left( \lambda_{k,i} z_{k,i} \right)^{\bar{w}_{k,i}}$$

s.t.

$$\sum_{k,i} z_{k,i} \leq P_T$$

(9.105)

$$z_{k,i} \geq 0, \ 1 \leq k \leq N, \ 1 \leq i \leq \bar{L}_k$$

where $\bar{w}_{k,i} = w_{k,i} / (\sum_{l,j} w_{l,j})$, and it is assumed that $\lambda_{k,i} > 0 \forall k, i$ (otherwise, the problem has the trivial solution $z_{k,i} = 0 \forall k, i$). The solution is easily obtained from the KKT optimality conditions as

$$z_{k,i} = \bar{w}_{k,i} P_T$$

(9.106)

For a uniform weighting $w_{k,i} = 1$, the problem reduces to the maximization of the geometric mean subject to the arithmetic mean:

$$\max \left\{ z_{k,i} \right\} \prod_{k,i} ^{1/\bar{L}_T} (\sum_{k,i} z_{k,i})$$

s.t.

$$1/\bar{L}_T \sum_{k,i} z_{k,i} \leq P_T / \bar{L}_T$$

(9.107)

$$z_{k,i} \geq 0$$

where $\bar{L}_T = \sum_{k=1}^{N} \bar{L}_k$. From the arithmetic–geometric mean inequality ($\prod_k x_k \leq \frac{1}{N} \sum_k x_k$ (with equality if and only if $x_k = x_l \forall k, l$) [60], it follows that the optimal solution is a uniform power allocation:

$$z_{k,i} = P_T / \bar{L}_T$$

(9.108)

The uniform power distribution is commonly used due to its simplicity.

9.3.3.8 Maximization of the HARM-SINR

For maximization of the harmonic mean of the SINRs (HARM-SINR) in the case of single beamforming, the objective function to be minimized is:

$$f_0 \left( \{\text{SINR}_{k,i} \} \right) = \sum_{k,i} \frac{1}{\text{SINR}_{k,i}}$$

(9.109)
which can be expressed as a function of the MSEs using Equation (9.77) as:

\[ f_0 \left( \{ \text{MSE}_{k,i} \} \right) \sum_{k,i} \frac{\text{MSE}_{k,i}}{1 - \text{MSE}_{k,i}} \]  

(9.110)

The function \( f_0 (\{x_i\}) = \sum x_i / (1 - x_i) \) (for \( 0 \leq x_i < 1 \)) is a Schur-convex function [49].

So, the objective function (9.110) is Schur-convex on each carrier \( k \). Therefore, by Equations (9.78)–(9.80), the optimal solution has a nondiagonal MSE matrix \( E_k \) with equal diagonal elements given by Equation (9.84), which have to be minimized. The scalarized problem in convex form is:

\[
\min \left\{ \{ t_k \}, \{ z_{k,i} \} \right\} \sum_k t_k \frac{1}{1 - t_k}
\]

s.t. \( 1 > t_k \geq \frac{1}{L_k} \left( L_k - \bar{L}_k \right) + \sum_{i=1}^{L_k} \frac{1}{1 + \lambda_{k,i} z_{k,i}} \)

\[
\sum_{k,i} z_{k,i} \leq P_T
\]

\[
z_{k,i} \geq 0, \quad 1 \leq k \leq N, \quad 1 \leq i \leq \bar{L}_k
\]

(9.111)

The problem has a multilevel water-filling solution

\[
z_{k,i} = \left( \bar{\mu}_{k,1/2} \lambda_{k,1/2}^{1/2} - \lambda_{k,i}^{1/2} \right)^+ 
\]

(9.112)

where \( \{ \bar{\mu}_{k,1/2} \} \) are multiple water levels chosen to satisfy the lower constraints on the \( t_k \)s and the power constraint, all with equality, and also the constraint \( \bar{\mu}_{k,1/2} = v \frac{L_k^{1/2}}{1 - q} \), where \( v \) is a positive parameter. For the case of single beamforming (i.e., \( L_k = 1 \)), the solution reduces to \( z_k = \lambda_k^{-1/2} (P_T / \sum_{i} \lambda_i^{-1/2}) \).

For the single-carrier case (or multicarrier cooperative approach), the problem simplifies to that considered in Section 9.3.3.1.

### 9.3.3.9 Maximization of the MIN-SINR

The objective function to be minimized for the maximization of the minimum of the SINRs (MIN-SINR) is

\[
\tilde{f}_0 \left( \{ \text{SINR}_{k,i} \} \right) = - \min_{k,i} \{ \text{SINR}_{k,i} \} 
\]

(9.113)

This design criterion is equivalent to the minimization of the maximum MSE treated with detail in Section 9.3.3.5. In [54], the same criterion was used, imposing a channel diagonal structure.

### 9.3.3.10 Maximization of the PROD-(1 + SINR)

Let us start with the following maximization:

\[
\max \prod_{k,i} (1 + \text{SINR}_{k,i}) 
\]

(9.114)

Using Equation (9.77), this maximization can be equivalently expressed as the minimization of \( \prod_{k,i} \text{MSE}_{k,i} \) as in Equation (9.87) with \( w_{k,i} = 1 \), as the minimization of the determinant of the MSE matrix (Section 9.3.3.3), and as the maximization of the mutual information (Section 9.3.3.4) with the solution given by the capacity-achieving expression Equation (9.95). This result is natural since maximizing the logarithm of (9.114) is tantamount to maximizing the mutual information \( I = \sum_{k,i} \log (1 + \text{SINR}_{k,i}) \).
9.3.3.11 Minimization of the ARITH-BER

The minimization of the average BER or of the arithmetic mean of the BERs (ARITH-BER) can be considered as the best criterion (assuming that after the linear processing at the receiver, each substream is detected independently). The objective function is:

\[
\tilde{f}_0 \left( \{ \text{BER}_{k,l} \} \right) = \sum_{k,l} \text{BER}_{k,l} \tag{9.115}
\]

which can be expressed as a function of the MSEs using Equation (9.77) as:

\[
f_0 \left( \{ \text{MSE}_{k,l} \} \right) = \sum_{k,l} \text{BER} \left( \text{MSE}_{k,l}^{-1} - 1 \right) \tag{9.116}
\]

The function \( f_0 \left( \{ x_i \} \right) = \sum_l \text{BER} \left( x_i^{-1} - 1 \right) \) (assuming \( \theta \geq x_i > 0 \), for sufficiently small \( \theta \) such that \( \text{BER} \left( x_i^{-1} - 1 \right) \leq 2 \times 10^{-2}, \forall i \)) is a Schur-convex function [49].

The objective function (9.116) is Schur-convex on each carrier \( k \) (assuming the same constellation/coding on all substreams of the \( k \)th carrier), provided that \( \text{BER}_{k,l} \leq 2 \times 10^{-2} \) (interestingly, for BPSK and QPSK constellations, this is true for any value of the BER). Therefore, by Equations (9.78)–(9.80), the optimal solution has a nondiagonal MSE matrix \( \mathbf{E}_k \) with equal diagonal elements given by Equation (9.84), which have to be minimized. The scalarized problem in convex form is

\[
\min_{\{t_{k,l}\}, \{z_{k,l}\}} \sum_k \alpha_k Q \left( \sqrt{\beta_k (t_{k,i}^{-1} - 1)} \right)
\]

s.t. \( \theta \geq t_k \geq \frac{1}{L_k} \left( L_k - \bar{L}_k \right) + \sum_{i=1}^{\bar{L}_k} \frac{1}{1 + \lambda_{k,i} z_{k,i}} \)

\[
1 \leq k \leq N
\]

\[
\sum_{k,l} z_{k,i} \leq P_T
\]

\[
z_{k,i} \geq 0, \quad 1 \leq k \leq N, \quad 1 \leq i \leq \bar{L}_k \tag{9.117}
\]

The upper bound \( \theta \) on the MSEs is explicitly included to guarantee the convexity of the BER function and, therefore, of the whole problem. For a general case with \( L_k > 1 \) and \( N > 1 \), problem (51) does not have a simple closed-form solution, and one has to resort to general-purpose iterative methods such as interior-point methods (see Section 9.3). For the single-carrier case (or multicarrier cooperative approach), the problem simplifies to the ARITH-MSE criterion considered in Section 9.3.3.1, plus the rotation matrix to make the diagonal elements of the MSE matrix equal.

9.3.4 Constraints in multicarrier systems

We can, for example, add constraints on the dynamic range of the power amplifier at each transmit antenna element, as was done in [68]. Consider a Schur-concave objective function and assume for simplicity \( L_k = L \forall k \). From the optimal structure in Equation (9.79) \( \mathbf{B}_k = \mathbf{U}_{b_k} \sum_{l=1}^{nT} \mathbf{b}_k \), the total average transmitted power (in units of energy per symbol period) by the \( i \)th antenna is

\[
P_i = \frac{1}{N} \sum_{k=1}^{N} \sum_{l=1}^{L} \| \mathbf{b}_k \|^2_{i,l} = \frac{1}{N} \sum_{k=1}^{N} \sum_{l=1}^{L} \sigma_{\mathbf{b}_k,l}^2 \left[ \mathbf{U}_{b_k} \right]_{i,l}^2 \tag{9.118}
\]

which is linear in the variables \( \{ \sigma_{\mathbf{b}_k,l}^2 \} \) (for the carrier-cooperative scheme, \( P_i = (1/N) \sum_{k=1}^{N} \sum_{l=1}^{L} \| \mathbf{b}_k \|^2_{i,l} \)). Therefore, the following constraints are linear as \( \alpha_l^L \leq P_i \leq \alpha_l^U \), \( 1 \leq i \leq nT \) where \( \alpha_l^L \) and \( \alpha_l^U \) are the lower and upper bounds for the \( i \)th antenna. Similarly, it is straightforward
to set limits on the relative dynamic range of a single element in comparison with the total power for the whole array [68] as \( \rho_l P_{array} \leq P_i \leq \rho_u P_{array} \) \( 1 \leq i \leq n_T \), where \( \rho_l \) and \( \rho_u \) are the relative bounds, and \( P_{array} = \sum_{i=1}^{n_T} P_i \) is the total power that is also linear in \( \{\sigma^2_{B_k l}\} \).

We also show how the PAPR (see Chapter 7) can be taken into account into the design of the beam vectors using a convex optimization framework. The PAR is defined as:

\[
\text{PAR} \triangleq \max_{0 \leq t \leq T_s} \frac{A^2(t)}{\sigma^2}
\]

(9.119)

where \( T_s \) is the symbol period, \( A(t) \) is the zero-mean transmitted signal, and \( \sigma^2 = \mathbb{E}[A^2(t)] \). Since the number of carriers is usually large (\( N \geq 64 \)), \( A(t) \) can be accurately modelled as a Gaussian random process (central-limit theorem) with zero mean and variance \( \sigma^2 \). Using this assumption, the probability that the PAR exceeds certain threshold or, equivalently, the probability that the instantaneous amplitude exceeds a clipping value \( A_{\text{clip}} \) is

\[
\Pr\{|A(t)| > A_{\text{clip}}\} = 2Q\left(\frac{A_{\text{clip}}}{\sigma}\right)
\]

(9.120)

The clipping probability of an OFDM symbol is then

\[
P_{\text{clip}}(\sigma) = 1 - \left(1 - 2Q\left(\frac{A_{\text{clip}}}{\sigma}\right)\right)^{2N}
\]

(9.121)

In other words, in order to have a clipping probability lower than \( P \) with respect to the maximum instantaneous amplitude \( A_{\text{clip}} \), the average signal power must satisfy

\[
\sigma \leq \sigma_{\text{clip}}(P) = \frac{A_{\text{clip}}}{Q^{-1}\left(1-(1-P)^{1/2N}\right)^{1/2}}
\]

(9.122)

When using multiple antennas for transmission, the previous equation has to be satisfied for all transmit antennas. Those constraints can be easily incorporated in any of the convex designs derived in Section 9.3.3 with a Schur-concave objective function. Using Equation (9.118), the constraint is

\[
\frac{1}{N} \sum_{k=1}^{N} \sum_{l=1}^{L} \sigma^2_{B_k l} \left| U_{B_k l} H_{l}, i \right|^2 \leq \sigma^2_{\text{clip}} 1 \leq i \leq n_T
\]

(9.123)

which is linear in the optimization variables \( \{\sigma^2_{B_k l}\} \). Such a constraint has two effects in the solution: (i) The power distribution over the carriers changes with respect to the distribution without the constraint, and (ii) the total transmitted power drops as necessary.

### 9.3.5 Performance examples

For the numerical results, we use the multicarrier modulation OFDM (64 carriers). We consider the multicarrier MIMO model used throughout the section. Perfect CSI is assumed at both sides of the communication link. The frequency selectivity of the channel is modeled using the power delay profile as specified in [70] [see Figure 9.11(a)], which corresponds to a typical large open space indoor environment for non-line-of-sight (NLOS) conditions with 150 ns average r.m.s. delay spread and 1050 ns maximum delay (the sampling period is 50 ns) [69]. The spatial correlation of the MIMO channel is modeled according to [71] (which corresponds to a reception hall) specified by the correlation matrices of the envelope of the channel fading at the transmit and receive side given in Figure 9.11(b), where the base station is the receiver (uplink) (see [71] for details of the model). It provides a large open indoor environment with two floors, which could easily illustrate a conference hall or a shopping galleria scenario. The matrix channel generated was normalized so that \( \sum_n \mathbb{E}[|H_{i j}(n)|^2] = 1 \). The SNR is defined as the transmitted power normalized with the noise variance. The following design criteria have been compared: ARITH-MSE, GEOM-MSE,
MAX-MSE (equivalently, MIN-SINR or MAX-BER), GEOM-SINR, HARM-SINR, and ARITH-BER (benchmark). The utilization of the Chernoff upper bound instead of the exact BER function gives indistinguishable results and is therefore not presented in the simulation results. Unless otherwise specified, carrier-noncooperative approaches are considered. The performance in terms of outage BER (averaged over the channel substreams), i.e. the BER that can be guaranteed with some probability or, equivalently, the BER that is not achieved with some small outage probability. In particular, we consider the BER with an outage probability of 5%. The results are presented in Figures 9.12–9.16.

Figure 9.11 (a) Power delay profile type C for HIPERLAN/2. (b) Envelope correlation matrices at the base station (BS) and at the mobile station (MS) © 2000, IEEE.
Figure 9.12 BER (at an outage probability of 5%) versus SNR when using QPSK in a $2 \times 2$ MIMO channel with $L = 1$ for the specified design criteria (without carrier cooperation).

Figure 9.13 BER (at an outage probability of 5%) versus SNR when using 16-QAM in a $4 \times 2$ MIMO channel (two transmit and four receive antennas) with $L = 1$ for the specified design criteria (without carrier cooperation).
Figure 9.14 (a) Probability of clipping and BER (at an outage probability of 5 \%) when using QPSK in a $2 \times 2$ MIMO channel with $L = 1$ for the ARITH-MSE criterion with and without PAR constraints (without carrier cooperation) as a function of $\mu$ (for SNR = 8 dB and $P_{\text{clip}} \leq 10^{-2}$). $\lambda_{\text{clip}} = \mu \sqrt{P_T/n_T}$. 

(a) Prob. clipping (SNR=8.00 dB) in a MIMO(2,2) channel

(b) Outage BER (QPSK) in a MIMO(2,2) channel
Figure 9.14 (b) Probability of clipping and BER (at an outage probability of 5%) when using QPSK in a 2 × 2 MIMO channel with $L = 1$ for the ARITH-MSE criterion with and without PAR constraints (without carrier cooperation) as a function of the SNR (for $\mu = 4$ and $P_{\text{clip}} \leq 10^{-2}$). $A_{\text{clip}} = \mu \sqrt{P_T / n_T}$. 
Figure 9.15 BER (at an outage probability of 5%) versus SNR when using QPSK in a 4 × 4 MIMO channel with $L = 2$ for the specified design criteria (without carrier cooperation).

Figure 9.16 BER (at an outage probability of 5%) versus SNR when using QPSK in a 2 × 2 MIMO channel with $L = 1$ for the specified design criteria with (coop) and without (noncoop) carrier cooperation.
REFERENCES


68. ETSI (2001) Broadband radio access networks (BRAN); HIPERLAN type 2; physical (PHY) layer, ETSI TS 101 475 V1.2.2, pp. 1–41.


10 Cognitive Radio

10.1 ENERGY-EFFICIENT COGNITIVE RADIO

Within this section we discuss some practical solutions and results for the adaptation of receiver parameters in order to improve performance and reduce energy consumption. Depending on the channel state, different receiver parameters can be changed for these purposes. Table 10.1 presents a summary of these options. Figure 10.1 presents a generic block diagram of such a receiver. The discussion in this section is based on [1] © 1999, IEEE. This, so called context aware radio, capable of observing the channel and network parameters and making autonomously decisions on the best transceiver configuration, will be referred to as software or cognitive radio.

In the simulation environment, the sender and receiver entities are instrumental in tracking energy consumption. The sender entity keeps track of energy consumption due to the RF transmission and link layer computation, such as forward error correction (FEC) encoding. The receiver entity tracks energy consumption due to the RF receiver and computation tasks such as FEC decoding and equalization. In all of the simulation, the RF section is assumed to dissipate 0.1 W during sleep, 0.6 W during receive and 1.8 W during transmit mode. The data on dissipation are taken from a commercial (General Electric Company’s Plessy) radio for wireless local area networks (WLANs). The details of the energy estimates for the computation tasks such as the FEC and equalization are described later in the section.

10.1.1 Frame length adaptation

When the BER due to random noise and interference changes, so does the frame error rate, which in turn determines the application level throughput, or goodput (excluding the header). Regardless of the transmission rate $R$, with large frame length the goodput of the link can drop to zero if every packet gets corrupted. The radio will still be transmitting (retransmissions) and therefore wasting battery energy. Reducing frame length can improve the goodput and therefore energy efficiency by reducing the probability of frame errors and the need for excessive numbers of retransmissions. However, the relative overhead of the frame header also increases, thereby offsetting the improvement in goodput. Therefore, an optimum frame length exists that maximizes goodput for a given BER as shown in the measurements in Figure 10.2. These measurements have been obtained with a peer-to-peer wireless link using the 900 MHz WaveLAN (WLAN) radios under varying channel interference introduced using an AWGN noise generator. A link layer header of eight bytes has been used.
Table 10.1 Adaptation of radio parameters to channel degradations

<table>
<thead>
<tr>
<th>Channel degradations</th>
<th>Adaptive radio parameters</th>
<th>Adaptation mechanism</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control Interference</td>
<td>Low level</td>
<td>FEC/ARQ</td>
</tr>
<tr>
<td></td>
<td>Low level</td>
<td>Frame length</td>
</tr>
<tr>
<td></td>
<td>High level</td>
<td>Processing gain</td>
</tr>
<tr>
<td></td>
<td>Bursty</td>
<td>Frame length</td>
</tr>
<tr>
<td>Flat fading</td>
<td></td>
<td>Frame length</td>
</tr>
<tr>
<td>Frequency selective fading</td>
<td></td>
<td>Channel impulse response</td>
</tr>
</tbody>
</table>

Figure 10.1 Simulation environment for adaptive radio.

Conventionally, the relationship in Figure 10.2 is used to adapt the frame length to maximize the goodput for a given BER. However, if the frame length adaptation is applied for reducing energy consumption under a given constraint on goodput, the radio selects the frame length that maximizes the BER required to meet the goodput constraint, thereby minimizing the required transmit energy per bit $E_b/N_0$. The reduction in required $E_b/N_0$ is exploited to reduce the transmitter power to obtain energy saving. Note that in most radios the transmitter energy consumption dominates. This adaptation approach is illustrated in Figure 10.3, which is derived from the measurements in Figure 10.2 and shows the relative energy versus frame length for different goodput constraints.

For a given goodput constraint, the radio selects the frame length that minimizes energy according to the corresponding curve in Figure 10.3. The radio monitors the BER and adjusts the transmit power to hold the BER constant at the maximum allowable value. For example, for the best case in Figure 10.3, if the desired goodput is 150 kbit/s, which is 7.5% of the transmission rate $R_t$ in this experiment, then a 30% reduction in battery energy consumption is achieved by reducing the frame size from the 1500 bytes Ethernet standard to 100 bytes.
The above discussion holds for the general case but the amount of energy saving will depend on the $E_b/N_0$ required for a particular modulation. The results in Figures 10.2 and 10.3 are valid for quadrature phase shift keying (QPSK modulation).

**Figure 10.2** Goodput results.

**Figure 10.3** Energy versus frame length [1] © 1999, IEEE.

10.1.2 **Frame length adaptation in flat fading channels**

The simulations are now in a flat fading channel, where errors occur in bursts as a function of the relative motion between the transmitter and receiver. The burstiness of the channel is modeled with a
two-state Discrete Time Markov Channel (DTMC) shown in Figure 10.4, with the following channel parameters [1]:

- $\rho = R/R_{\text{rms}}$ is the ratio of the Rayleigh fading envelop $R$ to the local root mean square (RMS) level.
- The average number of level crossings in a positive direction per second is given by $N = (2\pi)^{1/2} f_m \rho e^{-\rho^2} N = (2\pi)^{1/2} f_m \rho e^{-\rho^2}$
- The average time $T$, for which the received signal is below a specified level $R$, is given by $T = (e^{\rho^2} - 1)/(f_m \rho (2\pi)^{1/2})$
- Assuming steady-state conditions, the probability that we will find a given channel in the Good or Bad condition is

$$\mu = \begin{bmatrix} \mu_0 \\ \mu_1 \end{bmatrix} = \begin{bmatrix} P(\text{Good}) \\ P(\text{Bad}) \end{bmatrix} = \begin{bmatrix} 1/N - T \\ T \\ \frac{1}{1/N} \end{bmatrix} = \begin{bmatrix} e^{-\rho^2} \\ 1 - e^{-\rho^2} \end{bmatrix}$$

- BER = $P(\text{Good}) \cdot \text{BER}_G + P(\text{Bad}) \cdot \text{BER}_B$.
- The Good state BER is a function of the signal to noise ratio (SNR) at the receiver (degraded by such things as path loss, local interferers, thermal noise, etc.).
- The Bad state BER is fixed at 0.5. The Good state BER is allowed to vary over a wide range, $10^{-2}$ to $10^{-8}$.
- $\rho$ is set to $-20$ dB throughout the simulation. This is a conservative assumption that the radio has a 20 dB fading margin.
- The second variable of interest is the speed of the mobile, which indicates the burstiness of the channel.
- Selective acknowledgment (SACK) has been chosen as the error control scheme with a link layer header of eight bytes.
- The carrier frequency is set at 900 MHz.
- The transmission rate $R_t$ is set at 625 kbits/s.

Simulation results are shown in Figure 10.5 for frame lengths ranging from 50–1500 bytes. The results are based on the greedy data as the data source, with $L_n = (L_f)/R_t$ – the ratio of the frame length to the time interval between fades.

The achievable throughputs are shown in Figure 10.6. From these figures we can see that after the region of flat performance, the throughput decreases exponentially as $L_n$ increases, and energy
Figure 10.5 Energy per useful bit versus normalized frame length. Frame length size in bytes is annotated alongside the corresponding curves. The carrier frequency is 900 Hz and the transmission rate is 625 kbit/s. $\rho = -20$ dB, $\text{BER}_G = 10^{-5}$, and $\text{BER}_B = 0.5$.

Figure 10.6 Throughput versus normalized frame length.
increases as $L_n$ increases. The energy with $L = 50$ bytes becomes noticeably higher than the energy for $L > 50$ bytes, due to the increased header overhead in each frame. For $L > 50$ bytes, the energy and the throughput are relatively insensitive to the value of $L$ and stay flat within a constant range for $L_n < 0.1$. Specifically, with $L_n < 0.1$, the energy stays between 2–2.5 $\mu$J/bit and the throughput stays high, between 80–90% of $R_t$. The radio maintains an energy-efficient link with a high application throughput by adapting the frame length within the following constraint in a flat fading channel

$$50 < L < 0.1 \frac{R_t}{f_m} = 0.1 \frac{\lambda R_t}{v}$$  (10.1)

where $\lambda$ is the carrier wavelength.

### 10.1.3 The adaptation algorithm

The adaptation algorithm uses the frequency tracking loop in the radio receiver to estimate the speed $v$. Given the speed, the carrier frequency and the transmission rate, the upper bound on $L$ is then computed using Equation (10.1). If, in addition to a throughput constraint, the delay is constrained, then the upper bound on $L$ may be lower and can be obtained from the delay simulations as shown in Figure 10.7. The value of the lower bound in Equation (10.1) is determined by simulation and depends on the frame header size and the given throughput requirement (assumed to be eight bytes and 85% of $R_t$, respectively, in the above case). Within the bounds on $L$ for minimizing energy under flat fading conditions, the actual value of $L$ is chosen based on the optimum for adapting to the channel interference, as illustrated in Figure 10.3 (energy vs. $L$). If the optimum $L$ for adapting to channel interference is greater than the upper bound in Equation (10.1), then $L$ is set to this upper bound.

The issue of frame length and data rate adaptation in a cellular network will be discussed in more detail in Chapter 15.

### 10.1.4 Energy-efficient adaptive error control

The above results in frame length adaptation are for uncoded transmissions based on SACK. Additional benefit is expected if FEC is introduced. The combination of FEC with automatic repeat request (ARQ), is known as hybrid ARQ.

![Figure 10.7 Average delay per useful packet versus normalized frame length.](image-url)
Traditionally, the focus has been on satisfying a given throughput and/or delay requirement. As has already been pointed out, in portable multimedia devices, an additional requirement is to meet the throughput and delay requirements with minimum battery energy for a given media type. Simulations have been performed with RS block codes of rate 0.7 and the SACK retransmission protocol with a link layer header of eight bytes. Figure 10.8 shows the energy cost for implementing the RS coding on a StrongARM embedded microprocessor (http://www.develop.intel.com), which implements the link layer control in adaptive radio.

Results for the Gilbert–Elliot channel model from Figure 10.4 are presented in Figures 10.9–10.11 with the parameters shown in the figures. Using this data, the adaptive algorithm chooses an error
control scheme that minimizes battery energy for transmission over the wireless link, while trading off QoS parameters of the link, such as throughput and delay over various channel conditions, traffic types and packet sizes. The simulations are performed for two packet lengths:

- 53 bytes ATM cell size packets;
- 1500 bytes Ethernet frame size IP packets.

Figure 10.10 Data transmission with ATM at 50 km/h. The carrier frequency is at 900 MHz, $R_t = 625$ kbit/s, $\rho = -20$ dB and $BER_B = 0.5$. © 1999, IEEE.

Figure 10.11 Data transmission with IP at 50 km/h. The carrier frequency is at 900 MHz, $R_t = 625$ kbit/s, $\rho = -20$ dB and $BER_B = 0.5$.5
10.1.4.1 Error control for speech transmission

Figure 10.9 shows the effects of altering the Good state BER for real time data. Data is generated by the source at 32 kbits/s and forwarded to the sender with a delay of less than 50 ms. A delay greater than 50 ms is perceptually unacceptable for an interactive speech session and results in the packets being dropped. Figure 10.9 shows only small ATM packets, since larger packets do not meet the delay constraint. For a low Good state BER, no data is lost and both SACK and SACK with Reed–Solomon perform comparably. For energy-efficient speech transmission, the radio selects SACK error control for low BER (below $10^{-3}$). For higher BER, the radio adapts the error control to include FEC.

10.1.4.2 Error control for data transmission

The plots in Figures 10.10 and 10.11 show the trend in energy consumption and delay as the channel quality increases from a Good state BER of $10^{-3}$, as in packet cellular, to $10^{-8}$, as in WLAN applications. In Figure 10.10, for small packets (ATM), hybrid FEC/ARQ consumes less battery energy for the poor channel case (e.g. $10^{-3}$ BER). As the channel improves, the energy for SACK without FEC becomes lower, but the delay can be higher (e.g. BER $10^{-4}$). This is because the radio consumes the same amount of battery energy for the FEC, whereas the energy for SACK (without FEC) decreases because fewer retransmissions are required. In data transmission, a higher delay can be traded off for longer battery life. Therefore, for low BER (less than $10^{-3}$) the radio selects SACK and for high BER $\geq 10^{-3}$ the hybrid FEC/SACK is selected.

10.1.4.3 IP packets

Figure 10.11 shows that FEC/ARQ reduces the average delay, but it also consumes more energy compared to SACK. If the delay constraint is high enough, the radio minimizes battery energy by selecting SACK. The reason that the hybrid FEC/ARQ does not help to reduce the battery energy is because the flat fading effects in the channel become more pronounced with larger packets. In the short packet case, the channel noise effects are more dominant than fading and FEC provides improvements. If the delay constraint is too tight, the radio selects the hybrid error control. In summary: for small data packets, the radio switches between SACK and FEC/ARQ based on the BER; for large packets, the radio adapts the error control based on the delay constraint.

10.1.5 Processing gain adaptation

In the presence of interference, adaptation of FEC and frame length can improve the system performance when the BER does not degrade appreciably beyond $10^{-3}$. At this BER, the corresponding $E_b/N_0$ is 5–10 dB. In certain channel conditions, especially in shared bands, the interference level can be significantly higher, resulting in a signal to interference ratio (SIR) less than zero, implying a higher interference level than in the transmitted signal. Spread spectrum techniques, as discussed in Chapter 5, provide processing gain at the receiver, which can be used to provide protection in channels that exhibit negative SIRs. Transceiver IC, based on the architecture from Figure 10.12, is used to produce the experimental results presented in this section. An adaptation interface for this radio has been built to enable the link layer to control the processing gain (PG) [1].

The radio has a fixed chip rate of 2 Mchips/s and the processing gain can be adapted to 12, 15 and 21 dB for date rates of 128, 64 and 16 kbit/s, respectively. The power dissipation of this radio in transmit, receive and sleep modes is 2.5, 0.6 and 0.2 W, respectively. This experiment was performed indoors, and the radio had no RAKE equalization. The performance improvements are only due to the processing gain.

The experimental results are shown in Figure 10.13, in which the measured application level throughput (goodput) and energy consumption are plotted against the SIR experienced in a channel
Figure 10.12 Adaptive processing gain direct sequence spread spectrum radio [1] © 1999, IEEE.

Figure 10.13 Energy consumption versus number of states in the MLSE estimated with a 3.3 V standard cell library in the 0.35 μm CMOS process. The operating data rate is 1 Mbit/s.

with a narrowband interference. Curves for a processing gain of 12 dB are indicative of the performance one expects with a commercial radio based on IEEE Standard 802.11, such as WLAN. At this low processing gain, the throughput with commercial radios decreases as the SIR falls below −4 dB and the energy consumed per user bit sharply rises. By adapting the processing gain, the radio minimizes the energy by switching to a higher processing gain.
Table 10.2  SIR thresholds used for processing gain adaptation

<table>
<thead>
<tr>
<th>Threshold (dB)</th>
<th>Processing gain (dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SIR &gt; -4</td>
<td>12</td>
</tr>
<tr>
<td>-4 &gt; SIR &gt; -6</td>
<td>15</td>
</tr>
<tr>
<td>-6 &gt; SIR &gt; -12</td>
<td>21</td>
</tr>
</tbody>
</table>

Figure 10.14  Energy per useful bit.

10.1.5.1 Receiver algorithm

The radio measures the SIR from the receive signal and sets the processing gain based on the thresholds in Table 10.2. When the SIR drops to -5 dB the radio adapts by changing the processing gain from 12 to 15 dB, thereby decreasing the energy consumed by 86% as shown in Figure 10.14. In Figure 10.14, the achieved user throughput is 100 kbit/s for the lowest processing gain, and 9.6 kbit/s for the highest processing gain.

The tradeoff between data rate and processing gain is characteristic of spread spectrum, where the product of processing gain and data rate remains a constant and equals the occupied RF bandwidth.

To achieve a throughput of 1 Mbit/s required for most multimedia applications, and still provide a sufficiently high processing gain to adapt to a very low SIR value (e.g. -12 dB), a significant amount of bandwidth (i.e. hundreds of megahertz) is required. Since such a wide bandwidth is not always available, a possible solution is to use non-contiguous bands and frequency hop (FH) across these bands. Current research is aimed at developing such a solution using a multiband radio that can achieve a tunable range from 25–2500 MHz. At a high data rate of 1 Mbit/s, the FH radio can experience ISI due to the frequency selectivity of the channel as the symbol duration becomes less than the delay spread of the channel. To combat that ISI, equalization in the physical layer can be employed in conjunction with the adaptive link layer control to improve the link performance.

10.1.6 Trellis-based processing/adaptive maximum likelihood sequence equalizer

At high data rates, the degradation due to ISI in a frequency selective channel cannot be mitigated adequately by adapting the link layer alone. Equalization at the physical layer is required for the
delays $D_1$ and $D_2$ introduced by the packet detector and trace-back operation in the Viterbi detector

Figure 10.15 Maximum likelihood sequence equalizer modem architecture.

In this section, a maximum likelihood sequence equalizer (MLSE) has been selected for the adaptive radio because it has two well-known advantages over other equalizer algorithms. It also can be easily related to the other trellis-based algorithms presented so far in the book.

The energy consumption of the MLSE architecture shown in Figure 10.15 has been determined for a low-power 3.3 V 0.35 μm CMOS technology.

10.1.7 Hidden Markov channel model

DTMC models, such as the Gilbert–Elliot model, are a special case of the more general hidden Markov model (HMM) which is used in this section for performance evaluation. An HMM is a doubly stochastic process with an underlying stochastic process that is not observable but hidden. By simply observing the output sequence, the state that generated the output cannot be determined at any given time. The HMM has been used to obtain extremely accurate models for the mobile channel. The link performances over frequency selective channels are simulated using an HMM that is more accurate than the two-state Gilbert model.

A two-state HMM, shown in Figure 10.16 is used and is trained to match the error gap distribution in the error sequence, obtained through extensive simulation at the physical layer based on Jakes’s...
Figure 10.17 Energy consumption versus number of states in the MLSE estimated with a 3.3 V standard library on 0.35 µm in the CMOS process. The operating data rate is 1 Mbit/s.

model. In the error sequence, the position of an error event is marked with a ‘1’, while the position of an error-free event is marked with a ‘0’. An error gap is defined as the distance (number of positions) between two consecutive 1s in the error sequence. The output of the HMM consists of error gap values which can then be inverted to obtain the actual error sequence.

Figure 10.17 shows the energy cost for the MLSE with different numbers of states $M$ in the Viterbi decoder. The results can be easily extrapolated to any type of trellis-based algorithm.

- The carrier frequency is set at 2.4 GHz.
- The transmission rate is 1 Mbit/s.
- The carrier frequency is set at 2.4 GHz.
- The typical urban (TU) channel model is used for the mobile frequency selective channel to generate the error sequence for the hidden Markov model.
- The greedy data source is used with an eight-byte link layer header.

### 10.1.8 Link layer performance with inadequate equalization

Adequate equalization occurs when the number of states $M$ satisfies

$$\log_2(M) + 1 = P + L_{\text{mod}} - 1$$

(10.2)

where $P$ is the number of channel coefficients and $L_{\text{mod}}$ is the memory of the modulation. In the simulation, Gaussian minimum shift keying (GMSK) is used as the modulation with a $BT = 0.3$ that results in $L_{\text{mod}} = 2$. As $M$ increases, so does the energy consumption of the MLSE implementation. $M$ is selected to be smaller than required by Equation (10.2).

For the TU channel, $P = 5$, but $M$ is set inadequately to 2, i.e. a two-state MLSE. The objective of this simulation is to determine the tradeoff in the overall link energy, throughput and delay with a lower complexity equalizer that requires less energy consumption. Figure 10.18 shows the simulation results for frame length values of $L = 50, 500$ and 1500 bytes and speed values of $v = 5$ and 100 km/h. As expected, the performance degrades for the cases with larger frame lengths and higher speeds.
Figure 10.18  Energy, throughput and delay versus frame length for two-state MLSE for the mobile frequency selective TU channel. $\diamond$ denotes $v = 5$ km/h, and $\bullet$ denotes $v = 100$ km/h. The annotation to the data points indicates the frame length in bytes. SNR is 17 dB. The carrier frequency is 2.4 GHz. (a) Energy per useful bit versus normalized frame length; (b) throughput versus normalized frame length; (c) average delay per useful packet versus normalized frame length.
If the equalizer removes most of the ISI, then the channel appears to the receiver like an AWGN channel with some remaining level of burstiness due to mobility, similar to the behavior of a flat fading channel. To determine the effectiveness of the two-state equalizer, the results here are compared with the results for adapting frame length in a flat fading channel. As shown in Figure 10.18, the energy is minimal with a frame length of 50 bytes. The value of the energy consumption, throughput and delay are significantly worse than that obtained in a flat fading channel using the bounds on $L$ as in Equation (10.1).

For the two-state equalizer with $L = 50$ bytes and $v = 5$ km/h, a user throughput of 35% is achieved with delay of 70 ms, as compared to a user throughput of 73% and a delay of 2.5 ms achieved with an adaptive frame length for the flat fading channel (Figures 10.6 and 10.7). The energy required is $5 \mu$J/bit, 100% higher than that in Figure 10.5. We can conclude that using an equalizer that is smaller than required to satisfy Equation (10.2) does not provide an efficient tradeoff between equalizer energy consumption and transmit energy.

### 10.1.9 Link layer performance with adequate equalization

In this case, $M$ is chosen to satisfy Equation (10.2), i.e. sufficient equalization is provided for the TU channel. To meet Equation (10.2), an MLSE with 32 states ($M = 32$) is used. The results are shown in Figure 10.19.

Although the equalizer is more complex and consumes more energy, the overall system energy reduces since retransmissions are reduced. Compared to the case with $M = 2$ in Figure 10.18,

- the energy consumption is reduced by 100%;
- the energy can be minimized for the high speed case to 2.5 $\mu$J/bit with $L = 50$ bytes, and to 2.1 $\mu$J/bit for the low speed case with $L = 500$ bytes.

The energy, as well as the throughput and delay, now match those obtained by applying the adaptive frame length in a flat fading channel (Figures 10.6 and 10.7). We can now conclude that in a mobile frequency selective channel, an energy-efficient link requires an equalizer with an adequate number of states, as determined by Equation (10.2).

For given system parameters, simulations can be performed to obtain results such as those in Figure 10.19 to determine optimum frame sizes that minimize energy.

#### 10.1.9.1 Implementation

The radio monitors the CIR to determine the type of fading. A single peak in the CIR indicates flat fading. On the other hand, the occurrence of several peaks indicates frequency selective fading. The number of coefficients $P$ in the CIR is used to determine the number of states required in the MLSE, using Equation (10.2) with $L_{\text{mod}} = 2$. The speed is estimated from the frequency tracking loop and is used to determine the optimum frame length $L$ from simulation results as indicated above. The optimum frame length at different speeds can be intuitively explained by the limited ability of the equalizer to track the rapidly varying received amplitude in each of the CIR coefficients and the Doppler shift that causes an offset in the carrier frequency.

With a large frame size at high speed, the equalizer fails to maintain adequate tracking performance for the increased time duration of the given frame. At lower speed, the channel variation diminishes and therefore the equalizer can maintain tracking of the channel for longer frame sizes.

#### 10.1.9.2 Channel and frequency tracking performance

The ability of the MLSE to equalize ISI degrades if it does not track the fast amplitude variation and a large Doppler shift in the channel. A conventional tracking subsystem has been implemented in the MLSE as shown in Figure 10.15, and consists of a phase-locked loop (PLL) and an adaptive channel
Figure 10.19 Energy, throughput and delay versus frame length for the 32-state MLSE for the mobile frequency selective TU channel. ◆ denotes $v = 5 \text{ km/h}$ and ◆ denotes $v = 100 \text{ km/h}$. The annotation to the data points indicates the frame length in bytes. SNR is 17 dB. The carrier frequency is 2.4 GHz. (a) Energy per useful bit versus frame length; (b) throughput versus frame length; (c) average delay per useful packet versus frame length.

tracking loop (ACTL). The PLL tracks the Doppler frequency and carrier offset in the down-converted RF signal. The ACTL uses the least mean squared (LMS) algorithm to track the time variations in the channel impulse response.

The tracking performance with the PLL and ACTL can be degraded due to the delays $D_1$ and $D_2$ introduced by the packet detector and traceback operation in the Viterbi detector as discussed in Chapter 6.
**10.1.9.3 Per-survivor processing (PSP)**

To maintain performance of the tracking loops, PSP can be implemented in the MLSE. Figure 10.20 shows the performance of the MLSE with, and without, PSP operating at

- 1 Mbit/s over a TU channel;
- a delay spread of 5 µs at 100 km/h;
- a carrier frequency offset of 1200 Hz.

Without the PSP, the BER increases by nearly two orders of magnitude at an SNR of 20 dB.
Figure 10.21 System architecture for maintaining QoS in mobile wireless multimedia networking.

10.1.9.4 System integration

The adaptive radio can be integrated in the system architecture of Figure 10.21. The architecture ties the adaptive link layer and physical layer in the radio with the rest of the network protocol stack. It is based on a multilevel QoS framework where the lower layers can adapt to channel variations without continually requiring renegotiation with the application layer. At the top of the protocol stack are reactive applications that specify their requirements to the QoS manager as a set of multiple values corresponding to a set of allowable operating points with different degrees of acceptability. The application adapts its behavior by reacting to events from a QoS manager indicating a change in the level of QoS being provided by the lower layers. Each QoS level is defined by the average sustained throughput, delay and packet loss rate.

10.1.9.5 Adaptive steps

When the channel estimator indicates a degradation in QoS parameters, the link controller first attempts to maintain the QoS level by adapting the error control and frame length control. If the interference levels are too high to be handled by the link layer, the processing gain is adapted by the modem controller. If none of the adaptations are sufficient to maintain the current QoS, the level of service quality for one or more applications is reduced, and an event indicating this is passed up the protocol stack. The application layer may respond by adjusting parameters such as the speech codec compression ratio to be compatible with the drop in QoS. The reverse sequence of events takes place when channel conditions improve. Similarly, the network layer itself may respond to events from the link layer indicating changes in current QoS level by, for example, performing a connection route optimization. These issues will be discussed later in Chapter 15.

The radio adaptation is based on the channel state information measured in the physical layer that includes:

- SIR;
- the channel impulse response;
- the Doppler frequency.
The SIR determines the interference level in the channel, while the CIR determines the type of fading in the channel. The CIR is measured by a 48-tap complex matched filter (MF) in the adaptive equalizer (Figure 10.15) that is matched to a training sequence inserted in the preamble of the packet.

The speed of the user is inferred from the Doppler frequency which is determined by the frequency tracking loop in Figure 10.15. The deviation from the average value of the phase error ($\phi_k$) is proportional to the Doppler frequency. These three parameters allow the link controller to distinguish between a degradation due to fading or interference and, if due to fading, the speed of the mobile. This information enables the radio to appropriately adapt the frame length, error control, processing gain and equalization for the given channel condition.

### 10.1.9.6 Self-describing packets

The changes in the frame size, error control and processing gain used in the payload of each frame must be communicated by the sender node to enable the receiver to decode the packets. Communicating these parameters can result in a high signaling overhead and defeat many of the gains achieved by adaptive control. An alternative is to make each packet self-describing, such that no synchronization between the send and receive nodes is needed.

The physical layer header is encoded consistently from packet to packet and includes the information which tells the receiver how to decode the remainder of the packet. The physical layer header can, in this way, be decoded rapidly and consistently in the radio hardware while using a strong code, since it constitutes a relatively small amount of extremely important information. Even one bit error means the entire packet must be discarded; therefore, it should be heavily protected. Figure 10.22 shows the structure of the physical layer header used by the radio. After a preamble needed to lock onto the signal, the header bit map has single-bit flags that indicate which of the remaining header fields are actually present. In other words, all other fields are optional so that the overhead is kept to a minimum.

The remaining fields are used for:

1. The physical layer destination and origin addresses;
2. A time stamp field is inserted at the transmitter and used for synchronizing the MAC protocol;
3. The physical layer requires a number of parameters to successfully decode the incoming packet. These are specified in the modem field. For example, this field contains the processing gain to use for the body of the packet;
4. A service field which specifies the error control information for the receiving link layer to decode the rest of the packet;
5. A bit to indicate whether this is a beacon transmission or not (as from a base station);
6. A byte count and CRC for the header.

![Figure 10.22 Packet format for maintaining QoS in mobile wireless multimedia networking.](image)
10.2 A COGNITIVE RADIO ARCHITECTURE FOR LINEAR MULTIUSER DETECTION

10.2.1 A unified architecture for linear multiuser detection and dynamic reconfigurability

From Chapter 5, in a CDMA multiuser system, the output of the matched filter for the $k$th user can be expressed as

$$y_k = A_k b_k + \sum_{j \neq k} A_j b_j \rho_{jk} + n_k \quad k = 1, \ldots, K \quad (10.3)$$

The MF outputs for the users in the system in vector form become

$$y = R\mathbf{A} \mathbf{b} + \mathbf{n} \quad (10.4)$$

10.2.1.1 Linear multiuser schemes

In general, a linear multiuser detector performs a linear transformation of the received vector $\mathbf{y}$ and makes the decision for user $k$ as

$$\hat{b}_k = \text{sgn}(\mathbf{L} y)_k \quad (10.5)$$

where $\mathbf{L}$ is the appropriate transformation. From Chapter 5 we already know that for the MF, $\mathbf{L}$ takes the form

$$\mathbf{L}_{\text{MF}} = \mathbf{I} \quad (10.6)$$

where $\mathbf{I}$ is the identity matrix. For a decorrelator (DC),

$$\mathbf{L}_{\text{DC}} = \mathbf{R}^{-1} \quad (10.7)$$

For an approximate decorrelator (AD)

$$\mathbf{L}_{\text{AD}} = \mathbf{I} - \delta \mathbf{J} \quad (10.8)$$

where $\mathbf{J}$ is derived on the basis of the assumption that $\mathbf{R}$ is strongly diagonal as

$$(\mathbf{I} - \delta \mathbf{J})^{-1} = \mathbf{I} - \delta \mathbf{J} + o(\delta)$$

For the MMSE detector we have

$$\mathbf{L}_{\text{MMSE}} = (\mathbf{R} + \sigma^2 \mathbf{W}^{-1})^{-1} \quad (10.9)$$

where $\mathbf{W} = \mathbf{A}^T \mathbf{A}$.

10.2.1.2 ‘Modified’ filter $h_k(t)$

The DC can also be considered as a matched filter with impulse response

$$h_k(t) = \sum_{j=1}^{K} R_{kj}^+ s_j(t) \quad (10.10)$$

where $R_{kj}^+$ denotes $(\mathbf{R}^{-1})_{kj}$ The AD can be realized as

$$h_k(t) = s_k(t) - \sum_{j \neq k} \rho_{jk} s_j(t) \quad (10.11)$$
The MMSE detector as

$$h_k(t) = \sum_{j=1}^{K} (R + \sigma^2W^{-1})_{kj} s_j(t)$$

(10.12)

where \((R + \sigma^2W^{-1})_{kj}\) again denotes \([R + \sigma^2W^{-1}]_{kj}\). For the MF,

$$h_k(t) = s_k(t)$$

(10.13)

Any of the linear multiuser detectors described above can be realized by appropriately choosing the filter taps of the ‘modified’ MF, \(h_k(t)\). This forms the basis of the software radio architecture for linear multiuser detection.

### 10.2.1.3 Variable QoS

When the measure of quality is the BER achieved by the user, the relative performance of these receivers (see Chapter 5) may be classified as

$$\text{BER}_{MF} \geq \text{BER}_{AD} \geq \text{BER}_{DC} \geq \text{BER}_{MMSE}$$

(10.14)

Thus, reconfiguring the detectors among the MF and the above structures allows the option of a variable QoS from moderate (for the MF) to very high (for the MMSE detector). As an example, a user may switch to an MF configuration when using voice traffic, while the MMSE mode may be preferred for data traffic. We can also account for different data rates by deriving appropriate single-user linear filters for the multirate schemes.

### 10.2.1.4 Cognitive radio architecture for linear multiuser detection

Figure 10.23 shows a possible generic software radio architecture for linear multiuser detection. The architecture includes:

- channel processing (such as translation from IF to baseband);
- environment processing (e.g. estimation of signal and interference parameters and correlations);
- matched filtering and information bit-stream processing (e.g. FEC or convolutional decoding, soft decisions, etc.).

Figure 10.23 Functional architecture of a cognitive radio for linear multiuser detection.
These functionalities are partitioned into two core technologies based on processing speed requirements. These two technologies are based on FPGA and DSP devices. The key idea behind the software architecture is to reconfigure the MF dynamically according to the desired QoS (corresponding to one of the appropriate linear detectors).

### 10.2.1.5 Logical partitioning of the architecture

Essentially, each user’s radio could possibly have only one variable filter-tap receiver implemented using Xilinx FPGAs. All classes of the above receivers require two common generic operations:

- estimation of path delay of the users;
- the generation of PN sequences \( \{ s_k(t) \} \) of the users in the system.

The MF is probably the simplest in that, for any user \( k \), it just uses the information from these two generic operations in determining the timing offset of the PN sequence \( s_k(t) \) for the specific user.

For the AD, the complexity is slightly higher than that of the MF in that the ‘modified’ matched filter taps are adjusted according to the formulation given by Equation (10.10).

As seen in Figure 10.24, the additional functionality required here is the crosscorrelation values \( \{ \rho_{kj} \} \) and also the signature of sequences of all the users \( \{ s_k(t) \} \). The functional operations required for the DC are additionally

- the computation of the inverse matrix of crosscorrelations \( R^{-1} \);
- the column vector corresponding to the \( k \)th user, i.e. \( R_{k}^+ \).

The MMSE receiver incurs additional complexity over the DC in that it also requires estimates of the received signal powers of the users in the system (matrix \( W \)).

The issue of estimating the path delays or the received signal powers of the users in the system should be included.

The software radio architecture does have the functionality required to do both the estimation operations, similar to that used in conventional radio designs. The software radio architecture should be versatile in that it can easily allow a variety of signal processing algorithms to be implemented for accomplishing the required estimation.

![Figure 10.24 Logical partitioning of functionality in a cognitive radio receiver for linear multiuser detection](https://i.imgur.com/3Q5Q5Q.png)

Figure 10.24 Logical partitioning of functionality in a cognitive radio receiver for linear multiuser detection [2] © 1999, IEEE.
10.2.1.6 Testbed example

In the testbed (Figure 10.25) described in [2], the core of the hardware is based around an Aptix MP3 board with up to 12 FPGA components (up to 432,000 programmable gates) for sample level processing tasks with rates of up to 30.0 MHz. Information rate tasks (Viterbi decoding, deinterleaving, etc.) are performed by the VMEbus-based Pentak 4270 Quad ’C40 DSP Processor board. As a baseband front end, the testbed has two sets of dual (I and Q) A/D converters for input (Analog Devices AD 9762XR 12-bit, 41 MHz converters, and Pentak 6472 10-bit, 70 MHz converters) and one set of D/A converters (Analog Devices AD 9042ST 12-bit, 100 MHz).

Analog Devices converters are connected to the FPGA board through the custom adapters. Pentak converters are connected to the DSP via a multiband digital receiver (Pentak 4272) acting as a channelizer in Figure 10.23. The channelizer performs:

- frequency down-conversion;
- low pass filtering;
- decimation of the sampled baseband signal.

It is used for selection of the service bandwidth (i.e. the tuning band) from among those available in the sampled signal. The multiband digital receiver used as a channelizer has two narrowband receivers with a dynamic range of 1 kHz – 1 MHz, and one wideband receiver with dynamic range 2 MHz – 35 MHz.

It is capable of supporting a wide range of output sample rates.

A Sun workstation accesses the VMEbus through a Bit3 Sun-Sbus to VMEbus adapter and is used as a primary data stream source as well as development host. Development tools are centered around Signal Processing Worksystem (SPW).

SPW is a computer-aided design (CAD) tool that allows for the simulation and design of the complex communication systems based on block diagrams. It has a rich library of common communication blocks as well as facilities for creation of custom-coded blocks. Once entered, the block diagram of the target system can be simulated by using SPW’s signal flow simulator.

If these floating-point simulations are producing satisfactory results, the block diagram can be partitioned into DSP and hardware parts. The part of the design that is targeted for DSP implementation is prepared by SPW’s code generation system (CGS) (or MultiProx in case of partitioning into multiple DSPs) and is directly downloadable into the Quad TMS320C40 floating-point DSP board.

SPW’s hardware design system (HDS) is used to model the behavior of a fixed-point part of the design. Again, SPW’s signal flow simulator is used to verify the fixed-point model functionality. After the design is verified, the corresponding hardware description language (HDL) code is automatically

Figure 10.25 Testbed block diagram.
generated. This code is then synthesized (and/or simulated by the event-driven HDL simulator like Synopsys VSS) by an appropriate set of tools (e.g. Synopsys design compiler). The resulting design is further processed by the Xilinx XACT tool in order to generate the FPGA chip layout and routing, thereby producing the configuration bit stream. This configuration bit stream defines the combinatorial circuitry, flip-flops, interconnect structure and the I/O buffers inside a particular FPGA device.

Aptix tools are used to interconnect the FPGAs, connect FPGAs and DSPs through a parallel I/O board, and for routing of debugging signals to the control probes of the logic analyzer.

### 10.2.1.7 Partitioning of the architecture

The software radio architecture for linear multiuser detection is partitioned into two core technologies, namely FPGA and DSP devices. This partitioning is usually driven by the required functionality of the radio device and also the processing speed requirements.

The algorithmic complexity of the linear multiuser receivers increases with an increase in performance. Specifically, the complexity of the signal processing algorithms corresponding to the MF, AD, DC and MMSE receivers can be classified as

\[
C_{\text{MF}} < C_{\text{AD}} < C_{\text{DC}} < C_{\text{MMSE}}
\]

For a system with \( K \) users, processing gain \( N \), and oversampling factor \( O_s \), the floating-point complexity in terms of the number of multiply-accumulate (MAC) operations is

\[
C_{\text{MF}} = K(NO_s + 1)
\]

\[
C_{\text{AD}} = K(NO_s + 1) + (K - 1)NO_s + (K - 1)\log_2 K
\]

\[
C_{\text{DC}} = K(NO_s + 1) + 2K(K - 1)NO_s + (2/3)K^3
\]

\[
C_{\text{MMSE}} = K(NO_s + 1) + 2K(K - 1)NO_s + 2K^2
\]

\[+ (4/3)K^3 + C(\text{amp}) \]  

(10.16)

\( C(\text{amp}) \) denotes the complexity due to amplitude estimation that is incurred in the MMSE receiver (which is not explicitly considered here).

Figure 10.26 presents the number of MAC operations. We can evaluate the number of users that can be supported as a function of achievable information data rates (for different receiver structures) where the active constraint is a limitation in the complexity or the processing speed of the DSP device. In Figure 10.27, a simple illustration is shown for a 50 MHz floating-point TMS320C40 DSP. In all

![Figure 10.26 Processing speed constraints: MAC operations versus number of users.](image-url)
10.2.1.8 The cognitive radio architecture

Reconfigurable linear multiuser detection is implemented by partitioning the resources between FPGAs and DSPs as shown in Figure 10.28. The FPGA segment of the architecture includes:

- the PN sequence generators;
- two reconfigurable blocks that determine the filter taps of the appropriate linear receivers;
- an online estimate module;
- the correlator.

DSP implementations, serious constraints on the achievable data rates are imposed when the number of users increases, especially when we operate with more complex receivers.
The motivation for the particular selection of the constituents that comprise the FPGA segment is that the functionality provided by each constituent here can be easily handled by the processing speeds of the FPGA hardware. The two reconfigurable blocks contain the core of the sample level processing for each of the multiuser detectors and are actually implemented in separate FPGA components to facilitate on-the-fly reconfigurability.

**FPGA**

This segment enables us to reconfigure one of the blocks while the other is running and therefore, by oversizing the hardware, avoid loss of data during switchover to a different receiver structure. The rest of the sample level processing logic (PN sequence generators, control multiplexer and correlator) is implemented in a separate FPGA component since it does not require reprogrammability.

The online estimation block is used to perform timing estimates of the incoming signals, which are then used, along with the reference PN sequences, to compute the appropriate filter taps. There is also a provision for refining the online estimates by using more sophisticated offline algorithms. An FPGA component is used as an interface between the DSP segment and APTIX board.

**DSP**

The more algorithmically complex operations are partitioned into the DSP. This segment is implemented using a quad TMS320C40 DSP board. The operations include:

- offline estimation procedures;
- information bit-stream processing;
- control and reconfiguration management.

The offline estimation block includes:

- estimation of the received signals powers for the MMSE detector;
- computation of the inverse of the matrix of crosscorrelations (in the case of both the DC and the MMSE detectors);
- it can also refine estimates of the online estimation block in the FPGA segment.

The information bit-stream processing functions vary from error control techniques, such as FEC or convolutional decoding, to soft decision decoding.

The control and reconfiguration management block in the DSP segment of the architecture basically determines the variable QoS that can be achieved by dynamic reconfiguration of the receiver structures in the FPGA segment.

The reconfiguration of the blocks in the FPGA segment is directly controlled by the control and reconfiguration management block. The subsequent switching amongst receivers is achieved by a control multiplexer M (shown in Figure 10.28) that is also controlled by the control and reconfiguration management block. The DSP segment of the device acts to achieve the variable QoS requirements of the specific type of service. Implicit in the control and reconfiguration management block is also the capability to interact with higher layer protocols/stacks to facilitate the QoS demand of a specific data stream.

At the link layer, the actions of the DSP segment of the device also encompass environment processing such as sensing interference levels. The DSP devices have access to the configuration bit streams for each of the receiver structures and can download the particular configuration bit stream as and when required.

Table 10.3 shows the relative hardware complexity of the different linear multiuser detection schemes in terms of the required number of configurable logic blocks (CLBs). A CLB, in the case of Xilinx 4000 series FPGA, comprises a pair of flip-flops and two independent (Boolean) logical four-input function generators.

The complexity for each detector increases with both increasing number of users $K$ and increasing precision in quantization ($Q_r$ denotes the number of bits used in quantization). The MF complexity
remains invariant to the number of users in the system and depends only on the precision of quantization.

Even though the DC (and the MMSE) achieves better QoS, it comes at the expense of an exponential increase in complexity (with an increasing number of users) over the AD detector. This directly maps to an increase in processing power requirements of the FPGA segment of the software radio architecture. This again motivates switching to lower order receivers when the QoS requirements are moderate.

The complex operations required to be performed in the DSP segment of the architecture involve matrix inversion and offline estimation of amplitudes, etc.

### 10.2.2 Experimental results

First we consider floating-point implementations of the linear multiuser receivers. In all experiments, as described in [2], the transmitter powers of the users are controlled perfectly so that they are all received at the same power level. The floating-point results presented here agree very well with the analytical results on the performance of these receivers from Chapter 5. To show the flexibility of the software radio, consider two sets of experiments.

#### 10.2.2.1 Example 1

Fix the number of users in the system at $K = 15$.

In Figure 10.29, at low SNRs, the range of QoS achievable is quite limited. At higher SNRs, the DC and MMSE provide a BER gain of up to three orders of magnitude compared to either the MF or the AD. The AD is again slightly better than the MF for the set of operating points considered here.

For the case when $N = 128$, it is seen that the dynamic range of QoS achievable is greater among the detectors at high SNRs. The AD provides up to an order of magnitude better performance than the MF.

#### 10.2.2.2 Example 2: QoS achieved for different numbers of users

When the number of users in the system increases, the users can still operate at the same SNR, but can switch to a higher complexity detector to maintain the same QoS.

As an example, it is seen in Figure 10.30 that when $N = 64$, if users in a system desire a BER of $\approx 10^{-3}$, they can operate on

- an MF receiver (up to five users);
- an AD receiver (up to ten users);
- the DC receiver (up to 30 users)

without changing their received power level.

If some of the users require more stringent BERs, such as $\approx 10^{-5}$, they could operate entirely on a DC realization of the ‘modified’ MF.
When the processing gain is $N = 128$, it is seen that there is a greater range in achievable capacity by switching to the DC and MMSE receivers.

### 10.2.3 The effects of quantization

Consider fixed-point operations using two-, three-, and four-bit quantization. The experimental results for the MF are shown in Figure 10.31, and for the approximate DC in Figure 10.32. All the detectors
experience a degradation in performance, with the MF being the least sensitive and the DC being the most sensitive to quantization effects. The encouraging note is that the use of even a six-bit precision quantizer seems to pull performance close to that of the floating-point reference results.

10.2.4 The effect on the ‘near–far’ resistance

Figure 10.33 shows the average BER achieved by the MF, AD and DC receivers for the case of four-bit quantization as a function of interference powers. The desired user’s power is fixed, while the
interfering powers are increased. The MF and AD are not near–far resistant, and show degradations in performance as the interfering powers increase. The AD shows a more graceful degradation in performance relative to the matched filter. The DC, which is theoretically near–far resistant, however, fails to maintain this property in the presence of quantization effects. However, the BER performance is still superior to the other two detectors. More details on practical solutions can be found in [1–64].
RECONFIGURABLE ASIC ARCHITECTURE

So far in this chapter we have discussed adaptive and reconfigurable schemes used for improving the system performance or reducing energy consumption. In this section, we extend this topic to include reconfiguration from one multiple access technology to another, mainly TDMA, OFDM and CDMA options. There are a number of different solutions to this problem [1–61] © 1999, IEEE. In this section we discuss the case where the reconfiguration is performed on the level of Application Specific Integrated Circuit (ASIC) implementation. In particular, we will have a closer look into an architecture that can be used to realize any one of several functional blocks needed for the physical layer implementation of data communication systems operating at symbol rates in excess of 125 Msymbols/s.

Multiple instances of a chip based on this architecture, each operating in a different mode, can be used to realize the entire physical layer of high-speed data communication systems based on different multiple access schemes. The presentation in this section is based on [950].

The architecture features the following modes (functions):

- real and complex FIR/IIR filtering;
- least mean square (LMS)-based adaptive filtering;
- discrete Fourier transforms (DFT); and
- direct digital frequency synthesis (DDFS) at up to 125 Msamples/s.

All of the modes are mapped onto regular data paths with minimal configuration logic and routing. Multiple chips operating in the same mode can be cascaded to allow for larger blocks.

Baseband signal processing requirements are measured in the units gigaoperations per second (GOPS), and are beyond the capabilities of present day DSP solutions (e.g. the TI TMS320 family [66] or the Motorola 56000 family). Designers of such systems must incur the high cost and extended development time associated with custom ASIC solutions, even at the prototype phase of their work.
Table 10.4  ASIC, DSP, hybrid and flexible ASIC advantages and disadvantages

<table>
<thead>
<tr>
<th></th>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASIC</td>
<td>Low power</td>
<td>Inflexible</td>
</tr>
<tr>
<td></td>
<td>Small area</td>
<td>Expensive</td>
</tr>
<tr>
<td></td>
<td>High performance</td>
<td>Long TTM</td>
</tr>
<tr>
<td></td>
<td></td>
<td>High skill level</td>
</tr>
<tr>
<td>DSP</td>
<td>Flexible</td>
<td>Large area</td>
</tr>
<tr>
<td></td>
<td>Quick TTM</td>
<td>Low performance</td>
</tr>
<tr>
<td></td>
<td>Late changes possible</td>
<td>High performance</td>
</tr>
<tr>
<td></td>
<td>Medium skill level</td>
<td></td>
</tr>
<tr>
<td>Hybrid</td>
<td>Medium</td>
<td>Inflexible</td>
</tr>
<tr>
<td></td>
<td>TTM</td>
<td>Large area</td>
</tr>
<tr>
<td></td>
<td>Some flexibility</td>
<td>High area</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Low performance</td>
</tr>
<tr>
<td>ASIC + DSP</td>
<td>Low power</td>
<td>High skill level</td>
</tr>
<tr>
<td></td>
<td>TTM</td>
<td>New paradigm</td>
</tr>
<tr>
<td></td>
<td>Medium area</td>
<td></td>
</tr>
<tr>
<td></td>
<td>High performance</td>
<td></td>
</tr>
</tbody>
</table>

TTM – time to market

Traditionally, the designers have had to settle for the usual ASIC versus DSP tradeoff. An ASIC provides low power and high performance, as shown in Table 10.4, but it is inflexible to change and takes considerable engineering time to build. The DSP approach provides design flexibility and short time to market, but it is power hungry and incapable of satisfying the increasing computational demands of high-speed communication systems. The norm has been to take the hybrid approach, which is to utilize ASIC blocks to handle computations that are beyond the capabilities of a DSP, and DSP code for flexibility and time to market. This approach was demonstrated in the previous sections of this chapter. However, such a hybrid quite often leads to a system that processes the disadvantages of both the ASIC and the DSP.

One approach that overcomes this dilemma is to develop highly flexible VLSI data paths that are specifically designed and optimized for a single class of functions or tasks [65]. Such a class of circuits could be thought of as ASIC/FPGA hybrids, and they will be referred to as ASIC/FPGA. They would combine highly optimized processing units with programmable interconnects which can be changed in real time. However, unlike traditional FPGAs, the minimal functional block is not general purpose and the routing options are not global, rather, they are local and optimized for the set of applications envisioned for the architecture. Such a circuit is also fundamentally different from the minimally programmable arrays of custom computing elements used to solve very specific problems [67, 68]. The result is a highly optimized yet programmable VLSI circuit that can easily compete with ASICs in terms of performance, but at the same time provides a high degree of flexibility.

Another attempt to combine high speed with flexibility, generally known as reconfigurable computing (RC), has recently seen active development [67, 68]. Most reported RC implementations attempt to integrate an FPGA-type programmable block with a general purpose microprocessor [69] on the same die [70] or at the board level [954]. The programmable block is then configured to improve computational efficiency of the microprocessor. Thus, RC implementations can be thought of as a DSP/FPGA hybrid, while the example architecture is an ASIC/FPGA hybrid.

The first approach has the advantage of being maximally flexible, but it cannot achieve the very high speeds required for the target application of the example architecture, since it offers no speed advantages over a regular FPGA.

Other efforts are directed at making FPGAs faster at the expense of decreased flexibility, and could some day make the ASIC/FPGA tradeoff unnecessary [71]. A design has been reported that uses a high-speed reconfigurable ASIC data path that is quite similar to the one discussed in this section but is limited to different versions of the same algorithm [71].

Finally, a major research effort has resulted in a chip that attempts a three-way hybrid (DSP/FPGA/ASIC), but stops short of implementing sufficient computational power for the applications envisioned for the example architecture [72].
Table 10.5  Modes of operation

<table>
<thead>
<tr>
<th>Mode</th>
<th>Parameters</th>
<th>ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>Real FIR</td>
<td>Max. number of taps = 64</td>
<td>RFIR</td>
</tr>
<tr>
<td>Dual real FIR</td>
<td>Max. number of taps = 32</td>
<td>R2FIR</td>
</tr>
<tr>
<td>Complex FIR</td>
<td>Max. number of taps = 16</td>
<td>CFIR</td>
</tr>
<tr>
<td>Real IIR</td>
<td>Max. number of taps = 64</td>
<td>RIIR</td>
</tr>
<tr>
<td>Adaptive FIR</td>
<td>Max. number of taps = 8</td>
<td>AFIR</td>
</tr>
<tr>
<td>Digital frequency synthesis</td>
<td>fmax. = 60 MHz; fmin = 4 kHz; Df = 2 Hz; SFDR = 72 dB</td>
<td>DDFS</td>
</tr>
<tr>
<td>Discrete Fourier transformation</td>
<td>Max. block size = 32</td>
<td>GFFT</td>
</tr>
<tr>
<td></td>
<td>Min. block size = 4</td>
<td></td>
</tr>
</tbody>
</table>

Figure 10.34  Block diagram of an OFDM transmitter.

Figure 10.35  Block diagram of a QAM receiver.

At this time, no reported design has the combination of high speed, low power and flexibility of the example architecture. This example is a highly versatile VLSI architecture targeted at the high-speed data communications market. The architecture is highly regular and operates at data rates of up to 125 Msamples/s. Most importantly, it can be reconfigured in different modes in order to realize many of the functional blocks required in the transmitters and receivers of high-speed data communication systems. All modes utilize the same I/O pins and have been mapped onto a common, highly regular, data path with minimal control and configuration circuitry. A summary of the modes supported by the example architecture is provided in Table 10.5.

10.3.1  Motivation and present art

Figures 10.34 and 10.35 illustrate how multiple instances of a chip, based on the example architecture, can be used to realize different blocks within an OFDM transmitter and a QAM receiver. Each of the shaded blocks in these figures represents a single instance of the chip under a different configuration.
A chip based on the example architecture is assigned the acronym RADComm, for Reconfigurable ASIC for Data Communications.

The computational requirements are annotated in gigaoperations per second (GOPS, where an operation is defined as a single real multiplication) for a system operating at the maximum data rate.

10.3.2 Alternative implementations

The need for a high-performance flexible architecture becomes obvious when the limitations of the classical FPGA and DSP solutions are considered. The RC designs discussed above cannot implement all the desired features at the required data rates, and thus will not be used in this comparison. Let us compare the example architecture to the standard FPGA and DSP solutions using three basic metrics, namely, required sources, power consumption and ease of implementation.

The example architecture is synthesized onto a 0.25 mm/2.5 V CMOS technology, and is compared to the present state-of-the-art FPGA and DSP solutions available from leading manufacturers [65].

The FPGA field is represented by the largest and most advanced chip from the Xilinx Virtex series, XVC1000. The XVC1000 [73] features over one million gates partitioned into 8464 logic blocks and is manufactured in 0.22 mm/2.5 V technology [74]. DSPs are represented by the top-of-the-line processor from TI, TMS320C6201B. This processor operates at up to 233 MHz and is based on a 0.18 mm/1.8 V technology [75].

The example architecture is very flexible and easily scalable. For the purposes of illustration, we will assume an implementation with 64 computational elements (defined subsequently) operating on 16-bit data and capable of realizing the features in Table 10.5 at a rate of 125 Ms/s. Based on results obtained from the synthesis of the VHDL code, this implementation requires 70 kGates, occupies an area of just 4 mm$^2$, and consumes approximately 1.6 W power. We will make comparisons on two representative modes of operation: a 16-tap complex FIR (CFIR) and a 16-point DFT (GFFT) [76]. The example architecture is ideally suited for implementing a complex FIR filter and is less suitable for computing the DFT. It will be shown that the example architecture is superior to the FPGA and the DSP solutions for both of these modes.

10.3.3 Example architecture versus an FPGA

The CFIR mode utilizes 100 % of the computational resources in the example architecture. Over 90 % of the computational core is taken up by fast 16 × 16 bit multipliers, making the size and performance of a multiplier a critical parameter. The fastest 16 × 16 multiplier that can be implemented in an XCV1000 operates at 59 MHz and requires 96 logic blocks [77]. Thus, the desired data rates simply cannot be achieved using an FPGA (pipelining will not be considered since it would also increase RADComm’s performance).

Let us decrease the requirements in order to compare power consumption and implementation complexity.

A fully utilized XCV1000 can just implement 64 multipliers, leaving no room for additional logic. Thus, two FPGAs would be needed to realize a 16-tap complex filter operating at 59 MHz. This greatly complicates the design for two reasons:

1. A single chip solution becomes a board level problem.
2. Complicated partitioning of logic elements is needed, with special attention paid to delays introduced by inter-chip connections.

A first estimate based on the number of logic blocks required to implement the needed functionality, gives a power dissipation of 6.4 W when operating at 59 Ms/s (13.5 W at 125 Ms/s). The development time required for an FPGA design of this scope is considerably longer than the time required to configure and use a chip based on the example architecture. Moreover, a chip based on this architecture can be switched to a different mode in just a few clock cycles, whereas an FPGA of this size takes a long time to reconfigure (over 750 kcycles, or 15 ms).
Figure 10.36 RADComm versus FPGA and DSP for complex filtering [65] © 2000, IEEE.

<table>
<thead>
<tr>
<th>Simple to use</th>
<th>Single chip</th>
<th>Fast reconfiguration</th>
</tr>
</thead>
<tbody>
<tr>
<td>RADComm</td>
<td>⬤</td>
<td>⬤</td>
</tr>
<tr>
<td>FPGA</td>
<td>⬤</td>
<td></td>
</tr>
<tr>
<td>DSP</td>
<td>⬤</td>
<td>⬤</td>
</tr>
</tbody>
</table>

10.3.4 DSP against the example architecture

This comparison is not as straightforward as the comparison to an FPGA. The TMS320C6201B operates at a high frequency of 233 MHz but can only perform two multiplications per cycle and incurs delay in memory access. TI benchmarks indicate that the DSP requires $2N$ cycles per symbol for an $N$-tap complex FIR filter [76]. Thus, the TMS320C6201B could only process 7.3 Ms/s for a 16-tap complex filter. Again, the desired data rates are considerably above those achievable. Indeed, it would take 17 of these processors to provide the needed computational resources. One TMS320C6201B consumes over 1.9 W when operating at 233 MHz [20]. Thus, a total of 32 W would be used for filtering at the desired data rate. Clearly, a board full of these very expensive and power-hungry chips is not an acceptable solution. The advantages of the example architecture operating in the CFI mode over the existing solutions are summarized in Figure 10.36 and Table 10.6.

10.3.5 Computation of a complex 16-point DFT – the Goertzel FFT mode

The DFT can be computed using a very efficient radix-4 FFT algorithm. This algorithm does not map well onto the example computational core and is not used in this architecture. No such constraints exist for an FPGA or a DSP, and these solutions have a potential advantage over the example architecture. The DFT operating at the desired data rate uses 1386 logic blocks (16 %) in an XCV1000, and consumes 3.3 W power. The TMS320C6201B DSP requires ten cycles per symbol to compute 16-point FFTs, and could only process the input at up to 23 Ms/s, requiring five processors to achieve the desired data rate and consuming 9.5 W power. Despite using a less efficient algorithm, the example architecture operating in the GFFT mode has significant advantages over the existing solutions, as summarized in Figure 10.37 and Table 10.7.

The main challenge in designing such a highly versatile architecture, capable of supporting high symbol rates (over 125 Msymbols/s), is to identify suitable realizations of each function (mode of operation) such that all functions could be mapped onto a computational fabric with a minimal amount of routing and control logic. Indeed, chips exits that implement one or two of the features in
Table 10.7 RADComm, FPGA and DSP for DFT computation [65] © 2000, IEEE

<table>
<thead>
<tr>
<th>Simple to use</th>
<th>Single chip</th>
<th>Fast reconfiguration</th>
</tr>
</thead>
<tbody>
<tr>
<td>RADComm</td>
<td>•</td>
<td></td>
</tr>
<tr>
<td>FPGA</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>DSP</td>
<td>•</td>
<td></td>
</tr>
</tbody>
</table>

Figure 10.37 RADComm versus FPGA and DSP for DFT computation © 2000, IEEE.

Table 10.5 (e.g. [78]), but no existing architecture can implement all of them at high sampling rates. Identification of such an architecture together with the selection of the algorithms to implement each function will be presented hereafter.

At first sight, the architectures’ modes of operation do not reveal a great deal of similarity between them. For example, the conventional architecture for DFT [79] is quite different from adaptive filters using the LMS algorithm [80], which in turn have little in common with architectures used for the implementation of direct digital frequency synthesizers [81–83]. The main reason for these differences is the optimization of each architecture for the implementation of a single function. The first task is iteratively to consider various implementations for each mode to identify those that have the greatest computational overlap with all other modes. Once the implementations have been selected, we can define a single computational element that forms the least common denominator for all functions listed in Table 10.5. This rather simple computational unit, hereafter referred to as a tap, is shown in Figure 10.38. Having identified the tap, a computational core is created by grouping a number of
10.3.6 Fixed coefficient filters

The transfer function of an FIR filter

\[ y(n) = \sum_{k=0}^{N-1} x(n-k)C_k \]

can be implemented in either direct or transposed forms [964]. The transposed form implementation, shown in Figure 10.40, allows the critical path to be reduced to a single ‘multiply/add,’ and is a natural choice for the high-speed operations envisioned for this architecture. The transposed form FIR filter structure can be mapped onto the example core directly, as shown in Figure 10.41. For all fixed coefficient filters (Table 10.5), the coefficients are shifted in serially from an external source.
during the chip startup sequence. If a filter requires more taps than are available in a single chip, an unlimited number of chips can be cascaded.

10.3.7 Real FIR/correlator

The architecture can be configured as a single 64-tap structure (RFIR) by routing the output of each row to the input of the row below (e.g. output of row RR to input of row II, Figure 10.39). If a lower filter order is acceptable, a single chip based on the example architecture can be configured to process two data streams (in-phase and quadrature) simultaneously (R2FIR), with 32 taps each. In this mode, one filter is realized using rows RR and II, while the second is realized using rows RI and IR. A single 16-tap complex FIR filter can be implemented by using the chip in the CFIR mode.

Since filtering is a linear operation, the filter operating on complex values can be decomposed into four parallel real-valued filters, with the final output obtained by combining the four outputs

\[ x_c = (x_Rc_R - x_Ic_I) + i(x_Rc_I + x_Ic_R) \]

where \( x \) is the input and \( c \) is the coefficient vector, see Figure 10.39). A pair of adders is used to combine the outputs of rows (RR, II) and (RI, IR).

10.3.8 Real IIR/correlator

The example architecture can also be configured to operate as a single 64-tap, real-valued infinite impulse response filter (RIIR). This configuration is almost identical to that of the RFIR mode described above. To realize an IIR filter, we simply feedback the output of the FIR filter to an additional adder at the input, as shown in Figure 10.42.

10.3.9 Cascading fixed coefficient filters

An unlimited number of chips based on the example architecture can be cascaded to achieve filters with a larger number of taps. The cascading is accomplished by routing the outputs of each chip to the X0 inputs of the next chip, as shown in Figure 10.43 for the RFIR and R2FIR modes (cascading routing for the CFIR and IIR modes is very similar). All of the chips load the coefficients from the same bus, requiring only one external source (ROM) for the entire filter. Each chip uses its sequential position in the cascade to determine the starting and ending times for shifting in the coefficients. The unused inputs on all chips are set to ‘0.’

10.3.10 Adaptive filtering

The adaptive filtering mode can operate on both real and complex data. For complex operations, the filter coefficient adaptation is based on the popular LMS algorithm [5], summarized in Figure 10.44. The adaptive filter consists of two distinct parts – the filtering circuit, Equation (10.17), and the coefficient update circuit, Equation (10.18). In the example architecture, this functionality is implemented by allocating half of the taps to the update circuit, and the rest to the filter circuit. From
Equation (10.18) we observe that \( \mu e(n) \) is a term common to all of the new coefficients.

\[
y(n) = \sum_{j=1}^{N} C_j x(n - j)
\]  
\( (10.17) \)

\[
C_j(n + 1) = C_j(n) + \mu x^*(n - j)e(n)
\]  
\( (10.18) \)

Figure 10.43 Cascading for (a) RFIR; and (b) R2FIR. Unused I/Os are not shown.

Figure 10.44 The LMS adaptive algorithm.

For this reason, in Figure 10.45, \( \mu e(n) \) is fed to all the update taps. The \( x^*(n - j) \) values are obtained by shifting the values on \( x(n) \) into the coefficient registers. At time \( n \), the \( j \)th tap will have \( x(n - j) \) stored in the coefficient register. The complex conjugate is obtained by multiplying \( \mu e(n) \) by \(-1\) in the RI row. The new coefficient value, \( C_j(n) \), is obtained by feeding the output of the II row to the adder input of the RR row (dashed line). The filtering part of the circuit is identical to the CFIR mode discussed above.
10.3.11 Direct digital frequency synthesis

There exist a number of different methods for digital generation of a sinusoid of variable frequency. Most DDFS designs used today store precomputed samples of a sinusoid in a ROM lookup table and output these samples at different rates [10]. This method requires a large ROM to achieve acceptable spectral purity and does not map well on any flexible computational core. An alternative method for generation of a sinusoid is based on trigonometric definitions and properties of the sine and cosine functions. This method, known as coordinate rotation (CORDIC) [81–83] (see Figure 10.46), requires very few constant coefficients and is more suitable for implementation on a flexible ASIC architecture such as the one used in this example.
10.3.12 CORDIC algorithm [83]

Signal $e^{j\omega t}$ at sampling instants $t = nT_s$ can be represented as $e^{j\omega(n-1)T_s}e^{j\omega T_s}$. So, to synthesize a signal of frequency $\omega$ with sampling rate $1/T_s$, each previous sample should be multiplied by $e^{j\omega T_s}$ (rotated for phase, the $\omega T_s$). In this case, the sine and cosine of an angle are calculated using a cascade of $N$ ‘subrotation’ stages. The $k$th stage rotates the input complex number, considered as a two-element vector $(x_k, y_k)$, by $\pm \delta/2^k(\delta = \pi/2)$ radians depending on the $k$th bit of the phase control word ($W$) [see Equation (10.19)]. By changing $W$, we can rotate an initial vector by an angle in the range $[0 \cdots \pi - \delta/2^{N+1}]$ in increments of $\delta/2^N$ radians.

\[
(x_k, y_k) = \begin{cases} 
 x_{k-1} - T_k y_{k-1}, y_{k-1} + T_k x_{k-1} & \text{when } W[k] = 1 \\
 x_{k-1} - T_k y_{k-1}, y_{k-1} - T_k x_{k-1} & \text{when } W[k] = 0 
\end{cases} \tag{10.19}
\]

where $T_k = \tan(\delta/2^k)$.

The standard CORDIC algorithm suffers from two major problems, namely, low frequency resolution and high power consumption. The particular realization of the DDFS in the example architecture introduces two modifications to the conventional CORDIC algorithm that circumvent both of these problems. Modifications involve the fine frequency resolution and the feedback stages shown in Figure 10.46, which depicts the top-level block diagram of the modified CORDIC. A detailed discussion of these enhancements to the standard CORDIC algorithm and their effect on the DDFS performance are given in [966]. The algorithm is easily decomposed into a sum of products formulation, and is ideally suited for implementation on the example core.

The detailed mapping of the precomputed stages and the feedback section on the proposed computational core is shown in Figure 10.47. Only a small accumulator is needed in addition to the example computational core to implement the CORDIC algorithm.

A DDFS implemented using the example core can generate 125 Ms/s quadrature sinusoids in the frequency range of 4 kHz to 40 MHz in 2-Hz steps while maintaining SFDR above 72 dB [81].

![Figure 10.47 Mapping of the DDFS architecture.](image-url)
10.3.13 Discrete Fourier transform

The discrete Fourier transform (DFT) can be represented as

$$X[k] = \sum_{n=0}^{N-1} x(n)W_N^{kn}$$  \hspace{1cm} (10.20)

where $W_N = e^{-j(2\pi/N)}$.

Two approaches can be used: direct evaluation, and the Cooley–Tukey FFT algorithm [84]. Most dedicated DFT processors use the FFT algorithm because of its computational efficiency for large block sizes. Two major problems prevent the use of the FFT algorithm in the example architecture; FFT requires complex global routing, and the routing must change significantly to process different DFT sizes.

10.3.14 Goertzel algorithm

The example architecture implements a form of direct DFT evaluation known as the Goertzel algorithm [84]. The Goertzel algorithm calculates each DFT point using a simple recursive circuit, shown in Figure 10.48. The major advantages of this algorithm are ‘in-place’ computation, no external memory requirements, a highly regular structure, and easy cascadability that allows multiple chips to process a single large DFT block. If multiplication by $W_N^{-k}$ is viewed as a rotation by $2\pi k/N$ radians, the Goertzel algorithm is very similar to the CORDIC algorithm discussed in the previous section. The similarity can be extended to the set of CORDIC precompute stages by ‘unrolling’ the recursive loop. $N$ recursive cycles in Figure 10.48 can also be computed using $N$ multiplication stages, with the output of the $k$th stage being the input of the $(k + 1)$th stage, as shown in Figure 10.49. The main difference between the two algorithms is that the same coefficient is used for all the stages of the DFT computation.

![Figure 10.48 Goertzel algorithm for computing $X[k]$](image1)

![Figure 10.49 Unrolling the Goertzel recursive stage](image2)
The DFT algorithm operates on blocks of data at a time and therefore requires proper scheduling of resources to process the serial input data. Since the algorithm is first-order recursive, only the value $y_k(n-1)$ is needed to compute $y_k(n)$. Thus, only the $n$th stage is used at time $n$ to compute $y_k(n)$. This property is exploited by scheduling the data and coefficients on each stage so that $y_k(n)$, $k = 0 \ldots N-1$ can be computed simultaneously. The following schedule allows full utilization of the core; the coefficient in the $k$th stage at time $n$ is given by $W_n^{-k}$. An example of this schedule for a four-point FFT using four columns of taps is given in Table 10.8, where $W$ refers to the coefficient and $A$ refers to the input to the adder (see Figure 10.49). The input data stream is given by

### Table 10.8 Scheduling for the Goertzel algorithm

<table>
<thead>
<tr>
<th>Time $n$</th>
<th>Tap number / $(W, A)$</th>
<th>Out $X$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>$W^0, x(1)$</td>
<td>$W^3, v(2)$, $W^2, v(3)$, $W^1, 0$</td>
</tr>
<tr>
<td>1</td>
<td>$W^1, x(1)$</td>
<td>$W^0, x(2)$, $W^3, v(3)$, $W^2, 0$ $V[1]$</td>
</tr>
<tr>
<td>2</td>
<td>$W^2, x(1)$</td>
<td>$W^1, x(2)$, $W^0, x(3)$, $W^3, 0$ $V[2]$</td>
</tr>
<tr>
<td>3</td>
<td>$W^3, x(1)$</td>
<td>$W^2, x(2)$, $W^1, x(3)$, $W^0, 0$ $V[3]$</td>
</tr>
<tr>
<td>0</td>
<td>$W^0, y(1)$</td>
<td>$W^3, x(2)$, $W^2, x(3)$, $W^1, 0$ $X[0]$</td>
</tr>
<tr>
<td>1</td>
<td>$W^1, y(1)$</td>
<td>$W^0, y(2)$, $W^3, x(3)$, $W^2, 0$ $X[1]$</td>
</tr>
<tr>
<td>2</td>
<td>$W^1, y(2)$</td>
<td>$W^0, y(3)$, $W^3, 0$ $X[2]$</td>
</tr>
<tr>
<td>3</td>
<td>$W^1, y(3)$</td>
<td>$W^0, y(3)$, $W^3, 0$ $X[3]$</td>
</tr>
<tr>
<td>0</td>
<td></td>
<td>$W^1, 0$ $Y[0]$</td>
</tr>
</tbody>
</table>

Figure 10.50 Output in DDFS mode [65] © 2000, IEEE.
[\psi(0), \ldots, \psi(3), x(0), \ldots, x(3), y(0), \ldots, y(3)]. The highlighted part of the table shows the processing of \( x(n) \).

This scheduling can be easily implemented on the example core. Coefficient values are simply read from the ROM in sequential order and shifted in. The adder inputs \((A)\) are stored in the adder registers of rows RR and RI. The correct load/hold pattern for these registers is achieved using a simple \( N \)-bit shift register. A single ‘1’ circulates in the shift register. Whenever the \( k \)th bit is ‘1,’ the adder registers in rows RR and RI of the \( k \)th stage load the current value of the input, \( x(n) \); if the bit is ‘0,’ the value is unchanged. This shift register, and the two registers to hold values \( x(0) \) and \( x(1) \) (see Figure 10.49), make up all the overhead of this algorithm. The algorithm implicitly allows for taking DFTs of blocks smaller than the number of stages. Assuming that the coefficient values are stored sequentially in a ROM, to compute DFTs of size \( L \), \((L < N)\), the ROM address is incremented in steps of \( m = N/L \) (\( m \) is an integer).

Using this method, the desired value, \( X[k] \), is generated by the first \( L \) stages. Processing of these values by the remaining \( N - L \) stages does not change them. This can be verified by observing that each stage multiplies the input by \( W_{N}^{-k} \), and \( N - L \) stages result in a multiplication by \( W_{N}^{-k(N-L)} \).

Using the definition of \( W_{L} \), we obtain

\[
W_{L}^{-k(N-L)} = e^{i2\pi k(N-L)/L} = e^{-i2\pi k N/L} e^{i2\pi k (N/L)} = e^{-i2\pi m} = 1
\]

Figure 10.50 shows the PSD of a sinusoid generated by a chip configured in the DDFS mode [65].

REFERENCES


REFERENCES


74. Virtex 2.5 V programmable gate arrays, XILINX databook, 1999. rev. 1.9.


Cooperative Diversity in Cognitive Wireless Networks

11.1 SYSTEM MODELING

It was shown in Chapter 4 that multiple transmit antennas provide spatial diversity. Unfortunately, this is not easy to implement in the uplink of a cellular system, due to the size of the mobile unit. In order to overcome this limitation, yet still emulate transmit antenna diversity, an alternative form of spatial diversity is being considered, where diversity gains are achieved via the cooperation of in-cell users. That is, in each cell, each user may have a ‘partner.’ Each of the two partners would be responsible for transmitting not only their own information, but also the information of their partner, which they receive and detect. Spatial diversity would be achieved through the use of the partner’s antenna. This is complicated by the fact that the interuser channel is noisy and by the fact that both partners have information of their own to send; this is not a simple relay problem.

The basic premise in this concept is that both users have information of their own to send, denoted by $U_i$ for $i = 1, 2$, and would like to cooperate in order to send this information to the receiver at the highest rate possible. To distinguish this main/final receiver from the receiving units of the mobiles, we will refer to it as the BS, even though the user cooperation idea is equally applicable to ad hoc networks too. The channel model we use is depicted in Figure 11.1. Each mobile receives an attenuated and noisy version of the partner’s transmitted signal and uses that, in conjunction with its own data, to construct its transmit signal. The BS receives a noisy version of the sum of the attenuated signals of both users. The mathematical formulation of the model is [2]:

$$
Y_0(t) = h_{10}X_1(t) + h_{20}X_2(t) + n_0(t)
$$
$$
Y_1(t) = h_{21}X_2(t) + n_1(t)
$$
$$
Y_2(t) = h_{12}X_1(t) + n_2(t)
$$

where $Y_0(t)$, $Y_1(t)$, and $Y_2(t)$ are the baseband models of the received signals at the BS, user 1 and user 2, respectively, during one symbol period. Also, $X_i(t)$ is the signal transmitted by user $i$, for $i = 1, 2$, and $n_i(t)$ are the additive channel noise terms at the BS, user 1, and user 2, for $i = 0, 1, 2$, respectively. The fading coefficients, $\{h_{ij}\}$, remain constant over at least one symbol period, and observed over time form independent stationary ergodic stochastic processes, resulting in frequency-nonelective fading.
A critical assumption here is that there is no contribution from $X_2(t)$ in $Y_2(t)$, even though they are actually both present at the terminal belonging to user 2. Since $X_2(t)$ does not go through any fading before it reaches the antenna of user 2, unlike $Y_2(t)$, it may appear that it will have a detrimental effect on the reception of $Y_2(t)$. Provided that user 2 knows the relevant antenna gains, it is assumed that canceling the effects of $X_2(t)$ on $Y_2(t)$ is possible, and thus the model gives an accurate representation. A similar argument can be made in the case of user 1, regarding the effects of $X_1(t)$ on the reception of $Y_1(t)$. In practice, to isolate the transmitted signal from the received one, it may be necessary to use two separate channels, two colocated antennas, or some other means. For example, the CDMA implementations of this concept make use of spreading codes to create two separate channels, thus eliminating the need for echo cancellation. Also, time division among the mobiles has been investigated in [3, 4], where it was shown that cooperation continues to provide full diversity.

In the sequel we also use additional assumptions: (i) the transmitted signals $X_i(t)$ have an average power constraint of $P_i$ for $i = 1, 2$; (ii) the noise terms $n_i(t)$ are white zero-mean complex Gaussian random processes with spectral height $N_i/2$ for $i = 0, 1, 2$; and (iii) the fading coefficients $h_{ij}$ are zero-mean complex Gaussian random variables with variance $\xi_{ij}^2$ (Rayleigh fading). It is also assumed that the BS can track the variations in $h_{10}$ and $h_{20}$, user 1 can track $h_{21}$, and user 2 can track $h_{12}$, implying that all the decoding is done with the knowledge of the fading parameters. Due to the reciprocity of the channel, it is also assumed that $h_{21}$ and $h_{12}$ are equal. Finally, for simplicity of analysis and exposition, though with no loss in generality, a synchronous system is assumed.

### 11.1.1 System capacity

The mathematical model we use is a discrete time version of the model described in (i) and is given by:

\[
Y_0 = h_{10}X_1 + h_{20}X_2 + n_0 \\
Y_1 = h_{21}X_2 + n_1 \\
Y_2 = h_{12}X_1 + n_2
\]
with \( n_0 \sim N(0, \sigma_0^2) \), \( n_1 \sim N(0, \sigma_1^2) \) and \( n_2 \sim N(0, \sigma_2^2) \). In general, we assume that \( \sigma_1^2 = \sigma_2^2 \). The system is causal and transmission is done through blocks of length \( N \), therefore the signal of user 1 at time \( j, j = 1, \ldots, N \), can be expressed as \( X_1(U_1, Y_1(j - l), Y_1(j - 2), 1/4, Y_1(1)) \), where \( U_1 \) is the message that user 1 wants to transmit to the BS at that particular block. Similarly, for user 2, we have \( X_2(U_2, Y_2(j - l), Y_2(j - 2), 1/4, Y_2(1)) \).

The transmission is done for \( B \) blocks of length \( N \), where both \( B \) and \( N \) are large. The mobiles will cooperate based on the signals they receive in the previous block.

We assume mobile 1 divides its information \( U_1 \) into two parts: \( U_{10} \), to be sent directly to the BS, and \( U_{12} \), to be sent to the BS via mobile 2. Mobile 1 then structures its transmit signal so that it is able to send the above information as well as some additional cooperative information to the BS. This is done according to \( X_1 = X_{10} + X_{12} + e_1 \) and divides its total power accordingly \( P_1 = P_{10} + P_{12} + P_{e1} \) where \( e_1 \) refers to the part of the signal exchanged to transmit the cooperative information. Thus, \( X_{10} \) is allocated power \( P_{10} \) for sending \( U_{10} \) at rate \( R_{10} \) directly to the BS, \( X_{12} \) is allocated power \( P_{12} \) for sending \( U_{12} \) to user 2 at rate \( R_{12} \), and \( e_1 \) is allocated power \( P_{e1} \) for sending cooperative information to the BS.

Recall that we transmit for \( 12 \), should be such that \( U_{12} \) can be perfectly decoded by mobile 2. This perfect reconstruction at the partner forms the basis for cooperation. Mobile 2 structures its transmit signal \( X_2 \) and divides its total power \( P_2 \) in a similar fashion.

Recall that we transmit for \( B \) blocks of length \( N \). Cooperation in block \( i \) is achieved by constructing signals \( e_1 \) and \( e_2 \) based on \((U_{12}(i - 1), U_{21}(i - 1))\), both of which are now known at mobile 1 and mobile 2. The receiver waits until all the \( B \) blocks have been received and starts decoding from the last block. An achievable rate region with user cooperation is obtained by first considering the above cooperative strategy with constant attenuation factors, and then incorporating the randomness. It is assumed that each block of length \( N \) is long enough to observe the ergodicity of the fading distributions.

An achievable rate region for the system given in (2) is the closure of the convex hull of all rate pairs \((R_1, R_2)\) such that \( R_1 = R_{10} + R_{12} \) and \( R_2 = R_{20} + R_{21} \) with

\[
\begin{align*}
R_{12} &< E \left\{ C \left( \frac{h_{12}^2 P_{12}}{h_{12}^2 P_{10} + \sigma_1^2} \right) \right\} \\
R_{21} &< E \left\{ C \left( \frac{h_{21}^2 P_{21}}{h_{21}^2 P_{20} + \sigma_2^2} \right) \right\} \\
R_{10} &< E \left\{ C \left( \frac{h_{10}^2 P_{10}}{\sigma_0^2} \right) \right\} \\
R_{20} &< E \left\{ C \left( \frac{h_{20}^2 P_{20}}{\sigma_0^2} \right) \right\} \\
R_{10} + R_{20} &< E \left\{ C \left( \frac{h_{10}^2 P_{10} + h_{20}^2 P_{20}}{\sigma_0^2} \right) \right\} \\
R_{10} + R_{20} + R_{12} + R_{21} &< E \left\{ C \left( \frac{h_{10}^2 P_{10} + h_{20}^2 P_{20} + 2 h_{10} h_{20} \sqrt{P_{e1} P_{e2}}}{\sigma_0^2} \right) \right\} \quad (11.3)
\end{align*}
\]

for some power assignment satisfying \( P_1 = P_{10} + P_{12} + P_{e1}, P_2 = P_{20} + P_{21} + P_{e2} \). The function \( C(y) = (1/2) \log(1 + y) \) is the capacity of an additive white Gaussian noise (AWGN) channel with signal-to-noise ratio (SNR) \( y \) and \( E \) denoted expectation with respect to the fading parameters \( h_{ij} \).

The proof of Equation (11.3) is based on [2,8].

Equation (11.3) is shown in Figures 11.2 and 11.3, for different scenarios of channel quality. For the no-cooperation case, the users ignore the signals \( Y_2 \) and \( Y_1 \), which are equivalent to the multiple-access channel capacity region. The noiseless interuser channel \( (\sigma_1^2 = \sigma_2^2 = 0) \) is referred to as ideal cooperation, and is used mostly as an upper bound for the performance of any cooperation scheme.
Figure 11.2 Capacity region when the two users face statistically equivalent channels toward the BS.

Figure 11.3 Capacity region when the two users face statistically dissimilar channels toward the BS.
From Figure 11.2, we can see that when the channels from the users to the BS have similar quality ($h_{10}$ and $h_{20}$ have the same mean) and the channel between the users is better ($h_{12}$ has larger mean), the cooperation scheme greatly improves the achievable rate region. As the interuser channel degrades and the severity of the interuser fading increases, performance approaches that of no cooperation.

When the user–BS links of the two users experience fading with different means, cooperation again improves the achievable rate region, as shown in Figure 11.3. In this case, the user with more fading benefits most from the cooperation. The equal rate point ($R_1 = R_2$) or the maximum rate sum point ($R_1 + R_2$) is increased considerably with cooperation.

In Figures 11.2 and 11.3, the point where any achievable rate curve intersects the $Y$ axis corresponds to user 2 becoming a relay for user 1, and the point where the curve intersects the $X$ axis corresponds to user 1 becoming a relay for user 2.

Relay channels and their extensions form the basis for the study of cooperative diversity. Because relaying and cooperative diversity essentially create a virtual antenna array, work on multiple-antenna systems, or multiple-input, multiple-output (MIMO) systems, discussed in Chapter 4, is of course relevant.

The classical relay channel models, a class of three-terminal communication channels, are originally examined by van der Meulen [10]. Cover and El Gamal [1] treat certain discrete memoryless and additive white Gaussian noise relay channels, and they determine channel capacity for the class of physically degraded relay channels. In general, degraded means that the destination receives a corrupted version of what the relay receives, all conditioned on the relay transmit signal. More generally, they develop lower bounds on capacity, i.e. achievable rates, via three structurally different random coding schemes: facilitation, in which the relay does not actively help the source, but rather, facilitates the source transmission by inducing as little interference as possible; cooperation, in which the relay fully decodes the source message and retransmits, jointly with the source, a bin index (in the sense of Slepian–Wolf coding [11, 12]) of the previous source message (see also Section 2.5); and observation, in which the relay encodes a quantized version of its received signal, using ideas from source coding with side information [11–14] (see also Chapter 2.5). In general, cooperation yields highest achievable rates when the source–relay channel quality is very high, and observation yields highest achievable rates when the relay–destination channel quality is very high. Various extensions to the case of multiple relays have appeared in the work of Schein and Gallager [15, 16], Gupta and Kumar [17, 18], and Gastpar et al. [19, 20]. For channels with multiple information sources, Kramer and Wijngaarden [21] consider a multiple-access channel in which the sources communicate to a single destination and share a single relay.

Work by King [22], Carleial [23], and Willems and coworkers [24–26] examines multiple-access channels with generalized feedback. Here, the generalized feedback allows the sources essentially to act as relays for one another. This model relates most closely to the wireless channels we have in mind in this chapter. The constructions in [23, 24] can be viewed as two-terminal generalizations of the cooperation scheme in [1]; the construction [22] may be viewed as a two-terminal generalization of the observation scheme in [1]. Sendonaris et al. introduce multipath fading into the model of [23, 24], calling their approaches for this system model user cooperation diversity [2, 27, 28]. For ergodic fading, they illustrate that the adapted coding scheme in [24] enlarges the achievable rate region.

### 11.1.2 Probability of outage

If the attenuation factors vary slowly and can be approximated as constants over the $B$ blocks of length $N$, then over these $B$ blocks we can achieve rates dictated by the current values of $h_{10}$, $h_{20}$, $h_{12}$, and $h_{21}$. We assume $N$ (the block length) and $B$ (the number of blocks) are large enough to achieve capacity in the case of constant attenuation factors. However, in order to talk about the achievable rate region in (iii), we need to have even longer block lengths and observe different realizations of our fading amplitudes. When the delay requirements prevent us from having these longer block lengths, the rates achieved will be random quantities based on the current realizations of the fading amplitudes. Some wireless services have minimum requirements on the supported data rates, below which the service is unsustainable. Therefore, we observe an outage if the random rates that we can
achieve fall below a certain level, which we will call the service sustainability rate, and consider the probability of outage as a performance criterion [9].

In particular, we consider the equal rate point ($R_1 = R_2 = R$) and calculate the probability of outage versus the service sustainability rate $r$ for the cooperation and the no-cooperation schemes. The probability of outage, $P_{\text{out}} = \Pr(R < r)$, provides us with the probability that the current realization of our slowly fading parameters $h_{10}$, $h_{20}$, $h_{12}$, and $h_{21}$ will not be able to support an equal transmission rate of $r$ for the particular scheme under consideration. The results are given in Figure 11.4. We observe that for all service sustainability rates, the probability of outage for the cooperation scheme is smaller than the probability of outage under no cooperation. This is true despite the fact that the increase in achievable rate due to cooperation is moderate for the scenario depicted in Figure 11.4, as can be seen from Figure 11.2 ($E[h_{12}] = 0.63$). This demonstrates that even in cases when it does not significantly increase achievable rates, user cooperation is still able to increase robustness against channel variations. We, of course, expect the robustness to improve even more as the interuser channel quality ($E[h_{12}]$) improves.

11.1.3 Cellular coverage

Our expectation of increased cell coverage is based on a reduction in the required average received power. Assume the mobile is at a distance $d$ km from the BS. Let $P_{\text{tx}}$ denote the average power transmitted by the mobile and $P_{\text{rx}}$ denote the average power received at the BS (in decibels). Then, in a typical wireless channel, we have (in decibels) $P_{\text{rx}} = P_{\text{tx}} - PL(d)$ where $PL(d)$ is the mean path loss at distance $d$ km. A common model for $PL(d)$ is the Hata model (see Chapter 14) with $PL(d) = B_1 + B_2 \log d$, where $B_1$ and $B_2$ are functions of transmitter and receiver antenna height, carrier frequency, and type of environment. Therefore $\log d = (P_{\text{tx}} - P_{\text{rx}} - B_1)/B_2$. For a fixed $P_{\text{rx}}$, the maximum $d$ is achieved when $P_{\text{tx}}$ is at its maximum, denoted by $P_{\text{tx}}^{\text{max}}$. This is the maximum power that the mobile can transmit. Solving $d_{\text{max}}(P_{\text{rx}}) = (P_{\text{tx}}^{\text{max}} - P_{\text{rx}} - B_1)/B_2$. The coverage of the cell is thus given by $d_{\text{max}}$. The only quantity of the right-hand side that is variable is $P_{\text{tx}}$.

Assume that two different transmission schemes require different received average powers in order to operate successfully. That is, let scheme 1 require $P_{\text{rx}}^{(1)}$ and scheme 2 require $P_{\text{rx}}^{(2)}$. Define $\alpha$ to be their ratio, that is in decibels $10 \log \alpha = P_{\text{rx}}^{(1)} - P_{\text{rx}}^{(2)}$. From the above, we have $\log d_{\text{max}}^{(1)} - \log d_{\text{max}}^{(2)} = (P_{\text{rx}}^{(1)} - P_{\text{rx}}^{(2)})/B_2 = -(10 \log \alpha)/B_2$ resulting in $d_{\text{max}}^{(1)}/d_{\text{max}}^{(2)} = \alpha^{-10}/B_2$. A typical set of values for $B_1$
and $B_2$ is $B_1 = 17.3$ and $B_2 = 33.8$ (obtained for a medium-sized city environment, with carrier frequency of 900 MHz, transmit antenna height of 50 m, receive antenna height of 1 m, and a net antenna gain of 6 dB). Thus, the ratio of the coverage of scheme 1 versus scheme 2, as a function of the ratio of the required received average powers, is $d_{\text{max}}^{(1)}/d_{\text{max}}^{(2)} = \alpha^{-1/3.38}$.

11.2 COOPERATIVE DIVERSITY PROTOCOLS

11.2.1 System and channel models

In this section we divide the available bandwidth into orthogonal channels and allocate these channels to the transmitting terminals, allowing the protocols to be readily integrated into existing networks. By removing the interference between the terminals at the destination radio substantially simplifies the receiver algorithms and the outage analysis for purposes of exposition.

Current limitations in radio implementation preclude the terminals from full-duplex operation, i.e. transmitting and receiving at the same time in the same frequency band because of severe attenuation over the wireless channel, and insufficient electrical isolation between the transmit and receive circuitry. Thus, to ensure half-duplex operation, we further divide each channel into orthogonal subchannels. Figure 11.5 illustrates our channel allocation for an example time-division approach with two terminals.

The critical assumptions in this section are different levels of synchronization between the terminals in order for cooperative diversity to be effective. As suggested by Figure 11.5 and the modeling discussion to follow, we consider the scenario in which the terminals are block, carrier, and symbol synchronous. Given some form of network block synchronization, carrier and symbol synchronization for the network can build upon the same between the individual transmitters and receivers. For details on network synchronization see www.wiley.com/go/glisic1.

In characterizing the channel models, we modify notation (11.2) and focus on the message of the 'source' terminal $T_s$, which potentially employs terminal $T_r$ as a 'relay,' in transmitting to the 'destination' terminal $T_d$. We utilize a baseband-equivalent, discrete-time channel model for the continuous-time channel, and we consider $N$ consecutive uses of the channel, where $N$ is large.

For direct transmission, we model the channel as:

$$y_d[n] = h_{s,d}x_s[n] + n_d[n]; \quad n = 1, \ldots, N/2,$$

Figure 11.5 Example time-division channel allocations for (a) direct transmission with interference, (b) orthogonal direct transmission, and (c) orthogonal cooperative diversity. Throughout this section we focus on orthogonal transmissions of the form (b) and (c).
where \( x_i[n] \) is the source transmitted signal, and \( y_d[n] \) is the destination received signal. The other terminal transmits for \( n = N/2 + 1, \ldots, N \) as depicted in Figure 11.5(b). Thus, in the baseline system, each terminal utilizes only half of the available degrees of freedom of the channel.

For cooperative diversity, we model the channel during the first half of the block as:

\[
y_r[n] = h_{s,r} x_r[n] + n_r[n] \]

\[
y_d[n] = h_{s,d} x_r[n] + n_d[n]; \quad n = 1, \ldots, N/4
\]

where \( x_i[n] \) is the source transmitted signal and \( y_r[n] \) and \( y_d[n] \) are the relay and destination received signals, respectively. For the second half of the block, we have:

\[
y_r[n] = h_{r,d} x_r[n] + n_d[n]; \quad n = N/4 + 1, \ldots, N/2
\]

where \( x_i[n] \) is the relay transmitted signal and \( y_d[n] \) is the destination received signal. The representation in the second half of the block is similar, with the roles of the source and relay reversed, as depicted in Figure 11.5(c). Note that half the degrees of freedom are allocated to each source terminal for transmission to its destination, and a quarter of the degrees of freedom are available for communication to its relay.

Channel coefficients \( h_{i,j} \) capture the effects of path-loss, shadowing, and frequency nonselective fading, and \( n_i[n] \) captures the effects of receiver noise and other forms of interference in the system, where \( i \in \{s, r\} \) and \( j \in \{r, d\} \). Statistically, we model \( h_{i,j} \) as zero-mean, independent, circularly symmetric complex Gaussian random variables with variances \( \sigma_i^2 \). Furthermore, we model \( n_j[n] \) as zero-mean mutually independent, circularly symmetric, complex Gaussian random sequences with variance \( N_0 \).

Two important parameters of the above system are the SNR without fading and the spectral efficiency. If the transmitting terminals have an average power constraint in the continuous-time channel model of \( P_c \) joules per second, this translates into a discrete-time power constraint of \( P = 2P_c/W \) since each terminal transmits in half of the available degrees of freedom, under both direct transmission and cooperative diversity. Thus, the channel model is parameterized by the SNR variables \( \text{SNR}|h_{i,j}|^2 \), where \( \text{SNR} := 2P_c/(N_0W) = P/N_0 = \gamma \) is the common SNR without fading.

In addition to SNR, transmission schemes are further parameterized by the rate \( r \) bits per second, or spectral efficiency \( R := 2r/W \) b/s/Hz attempted by the transmitting terminals. Nominal, one could parameterize the system by the pair \((y, R)\); however, our results lend more insight, and are substantially more compact, when we use either of the pairs \((Y, R)\) or \((\gamma, R)\), where

\[
Y := \frac{y}{2^R - 1}, \quad \gamma = \frac{R}{\log(1 + y\sigma_i^2)}
\]

11.2.2 Cooperative diversity protocols

11.2.2.1 Amplify-and-forward

For amplify-and-forward transmission, the appropriate channel model is (11.5–11.6). The source terminal transmits its information as \( x_i[n] \), say, for \( n = 1, \ldots, N/4 \). During this interval, the relay processes \( y_r[n] \), and relays the information by transmitting \( x_r[n] = \beta y_r[n - N/4] \) for \( n = N/4 + 1, 1/4, N/2 \). To remain within its power constraint (with high probability), an amplifying relay must use gain \( \beta \leq \sqrt{P/(|h_{s,r}|^2P + N_0)} \) where we allow the amplifier gain to depend upon the fading coefficient \( h_{s,r} \) between the source and relay, which the relay estimates to high accuracy. This scheme can be viewed as repetition coding from two separate transmitters, except that the relay transmitter amplifies its own receiver noise. The destination can decode its received signal \( y_d[n] \) for \( n = 1, \ldots, N/2 \) by first appropriately combining the signals from the two subblocks using one of a variety of combining techniques; in the sequel, we focus on a suitably designed matched filter, or maximum-ratio combiner.
11.2.2 Decode-and-forward

For decode-and-forward transmission, the appropriate channel model is again Equations (11.5) and (11.6). The source terminal transmits its information as $x_s[n]$, for $n = 0, \ldots, N/4$. During this interval, the relay processes $y_r[n]$ by decoding an estimate $\hat{x}_s[n]$ of the source transmitted signal.

Under a repetition-coded scheme, the relay transmits the signal:

$$x_r[n] = \hat{x}_s[n - N/4] \quad \text{for} \quad n = N/4 + 1, \ldots, N/2.$$  

Decoding at the relay can take on a variety of forms. For example, the relay might fully decode, i.e. estimate without error, the entire source codeword, or it might employ symbol-by-symbol decoding and allow the destination to perform full decoding. These options allow for trading off performance and complexity at the relay terminal. Note that we focus on full decoding in the sequel. Again, the destination can employ a variety of combining techniques; we focus in the sequel on a suitably modified matched filter.

11.2.2.3 Selection relaying in cognitive radio

Fixed decode-and-forward is limited by direct transmission between the source and relay. However, since the fading coefficients are known to the appropriate receivers, $h_{s,r}$ can be measured to high accuracy by the cooperating terminals; thus, they can adapt their transmission format according to the realized value of $h_{s,r}$.

If the measured $|h_{s,r}|^2$ falls below a certain threshold, the source simply continues its transmission to the destination, in the form of repetition or more powerful codes. If the measured $|h_{s,r}|^2$ lies above the threshold, the relay forwards what it received from the source, using either amplify-and-forward or decode-and-forward, in an attempt to achieve diversity gain.

11.2.2.4 Incremental relaying in cognitive radio

Fixed and selection relaying can make inefficient use of the degrees of freedom of the channel, especially for high rates, because the relays repeat all the time. As an alternative, incremental relaying protocols can be used that exploit limited feedback from the destination terminal, e.g. a single bit indicating the success or failure of the direct transmission. We nominally allocate the channels according to Figure 11.5(b). First, the source transmits its information to the destination at spectral efficiency $R$. The destination indicates success or failure by broadcasting a single bit of feedback to the source and relay, which we assume is detected reliably by at least the relay. If the source-destination SNR is sufficiently high, the feedback indicates success of the direct transmission, and the relay does nothing. If the source-destination SNR is not sufficiently high for successful direct transmission, the feedback requests that the relay amplify-and-forward what it received from the source. In the latter case, the destination tries to combine the two transmissions.

11.2.3 Outage probabilities

As a function of the fading coefficients viewed as random variables, the mutual information for a protocol is a random variable denoted by $I$; in turn, for a target rate $R$, $I < R$ denotes the outage event, and $\Pr[I < R]$ denotes the outage probability.

11.2.3.1 Direct transmission

For direct transmission, the source terminal transmits over the channel (11.4). The maximum average mutual information between input and output in this case, achieved by independent and identically distributed (i.i.d.) zero-mean, circularly symmetric complex Gaussian inputs, is given by $I_D = \log(1 + y|h_{s,d}|^2)$ as a function of the fading coefficient $h_{s,d}$. The outage event for spectral
efficiency $R$ is given by $I_d < R$ and is equivalent to the event $|h_{s,d}|^2 < 1/Y$. For Rayleigh fading, i.e., $|h_{s,d}|^2$ exponentially distributed with parameter $\sigma_{s,d}^2$, the outage probability satisfies

$$p_{ID}^{\text{out}}(y, R) := \Pr[I_D < R] = \Pr[|h_{s,d}|^2 < \frac{1}{Y}] = 1 - \exp\left(-\frac{1}{Y\sigma_{s,d}^2}\right) \sim \frac{1}{\sigma_{s,d}^2} \cdot \frac{1}{Y}$$

(11.8)

### 11.2.3.2 Amplify-and-forward

The amplify-and-forward protocol produces an equivalent one-input, two-output complex Gaussian noise channel with different noise levels in the outputs. The maximum average mutual information between the input and the two outputs, achieved by i.i.d. complex Gaussian inputs, is given by (see Appendix 11.2):

$$I_{AF} = \frac{1}{2} \log(1 + y|h_{s,d}|^2 + f(y|h_{s,r}|^2, y|h_{r,d}|^2))$$

(11.9)

as a function of the fading coefficients, where $f(u, v) := uv/(u + v + 1)$.

The outage event for spectral efficiency $R$ is given by $I_{AF} < R$ and is equivalent to the event:

$$|h_{s,d}|^2 + \frac{1}{y} f(y|h_{s,r}|^2, y|h_{r,d}|^2) < \frac{2^R - 1}{y}$$

(11.10)

For Rayleigh fading, i.e., $|h_{i,j}|^2$ independent and exponentially distributed with parameters $\sigma_{i,j}^{-2}$, analytic calculation of the outage probability becomes involved, but we can approximate its high-SNR behavior as:

$$p_{AF}^{\text{out}}(y, R) := \Pr[I_{AF} < R] \sim \left(\frac{1}{2\sigma_{s,d}^2} \frac{\sigma_{s,r}^2 + \sigma_{r,d}^2}{\sigma_{s,r}^2, \sigma_{r,d}^2}\right) \left(\frac{2^R - 1}{y}\right)^2$$

(11.11)

where we have utilized the result of Equality 1 in Appendix 11.1, with:

$$u = |h_{s,d}|^2, v = |h_{s,r}|^2, w = |h_{r,d}|^2, \lambda_u = \sigma_{s,d}^{-2}, \lambda_v = \sigma_{s,r}^{-2}, \lambda_w = \sigma_{r,d}^{-2}$$

$$g(\varepsilon) = (2^R - 1)\varepsilon, t = y, h(t) = 1/t.$$ 

### 11.2.3.3 Decode-and-forward

In this case we examine a particular decoding structure at the relay. Specifically, we require the relay to decode fully the source message; examination of symbol-by-symbol decoding at the relay becomes involved because it depends upon the particular coding and modulation choices. The maximum average mutual information for repetition-coded decode-and-forward can be readily shown to be:

$$I_{DF} = \frac{1}{2} \min \left\{ \log(1 + y|h_{s,r}|^2), \log(1 + y|h_{s,d}|^2 + y|h_{r,d}|^2) \right\}$$

(11.12)

as a function of the fading random variables. The first term represents the maximum rate at which the relay can reliably decode the source message, while the second term represents the maximum rate at which the destination can reliably decode the source message given repeated transmissions from the source and destination. Requiring both the relay and destination to decode the entire codeword without error results in the minimum of the two mutual pieces of information in (11.12).

The outage event for spectral efficiency $R$ is given by $I_{df} < R$ and is equivalent to the event:

$$\min \left\{ |h_{s,r}|^2, |h_{s,d}|^2 + |h_{r,d}|^2 \right\} < \frac{2^R - 1}{y}.$$ 

(11.13)
For Rayleigh fading, the outage probability for repetition-coded decode-and-forward can be computed according to

\[ p_{DF}^{out}(y, R) : = \Pr[I_{DF} < R] = \Pr\left[ |h_{s,r}|^2 < g(y) \right] + \right. \\
+ \Pr\left[ |h_{s,r}|^2 \geq g(y) \right] \Pr\left[ |h_{s,d}|^2 + |h_{r,d}|^2 < g(y) \right] \]  
\hspace{1cm} (11.14)

where \( g(y) = [2^{2R} - 1]/y \). We examine the large \( y \) behavior of Equation (11.14) by computing the limit

\[ \frac{1}{g(y)} p_{DF}^{out}(y, R) = \frac{1}{g(y)} \Pr\left[ |h_{s,r}|^2 < g(y) \right] \]

\[ \to 1/2\sigma_{s,r}^2 \]  
\hspace{1cm} (11.15)

as \( y \to \infty \), using the results of Results 1 and 2 in Appendix 11.1. So,

\[ p_{DF}^{out}(y, R) \sim \frac{1}{2\sigma_{s,r}^2} \frac{2^{2R} - 1}{y} \]  
\hspace{1cm} (11.16)

The \( 1/y \) behavior in (11.16) indicates that fixed decode-and-forward does not offer diversity gains for large SNR, because requiring the relay to decode fully the source information limits the performance of decode-and-forward to that of direct transmission between the source and relay.

### 11.2.3.4 Selection relaying

As an example analysis, we determine the performance of selection decode-and-forward. In the case of repetition coding at the relay, we have

\[ I_{SDF} = \begin{cases} 
\frac{1}{2} \log(1 + 2y |h_{s,d}|^2), & |h_{s,r}|^2 < g(y) \\
\frac{1}{2} \log(1 + y |h_{s,d}|^2 + |h_{r,d}|^2), & |h_{s,r}|^2 \geq g(y) 
\end{cases} \]  
\hspace{1cm} (11.17)

where \( g(y) = [2^{2R} - 1]/y \). This threshold is motivated by our discussion of direct transmission. The first case in Equation (11.17) corresponds to the relay’s not being able to decode and the source’s repeating its transmission; here, the maximum average mutual information is that of repetition coding from the source to the destination, hence the factor \( 2y \). The second case in Equation (11.17) corresponds to the relay’s ability to decode and repeat the source transmission; here, the maximum average mutual information is that of repetition coding from the source and relay to the destination.

The outage event for spectral efficiency \( R \) is given by \( I_{SDF} < R \) and is equivalent to the event:

\[ \left( \left\{ |h_{s,r}|^2 < g(y) \right\} \cap \left\{ 2|h_{s,d}|^2 < g(y) \right\} \right) \]

\[ \cup \left( \left\{ |h_{s,r}|^2 \geq g(y) \right\} \cap \left\{ |h_{s,d}|^2 + |h_{r,d}|^2 < g(y) \right\} \right) \]  
\hspace{1cm} (11.18)

The first (second) event in (11.18) corresponds to the first (second) case in Equation (11.17). Because the events in Equation (11.18) are mutually exclusive, the outage probability becomes a sum:

\[ p_{SDF}^{out}(y, R) : = \Pr[I_{SDF} < R] \]

\[ = \Pr\left[ |h_{s,r}|^2 < g(y) \right] \Pr\left[ 2|h_{s,d}|^2 < g(y) \right] \]

\[ + \Pr\left[ |h_{s,r}|^2 \geq g(y) \right] \Pr\left[ |h_{s,d}|^2 + |h_{r,d}|^2 < g(y) \right] \]  
\hspace{1cm} (11.19)
and we may readily compute a closed-form expression for (11.xx). For comparison with our other protocols, we examine the large SNR behavior of Equation (11.19) by computing the limit:

$$\frac{1}{g^2(y)} p_{\text{SDF}}^{\text{out}}(y, R) = \frac{1}{g(y)} \Pr \left[ |h_{s,r}|^2 < g(y) \right] + \frac{1}{g^2(y)} \Pr \left[ |h_{r,d}|^2 < g(y) \right]$$

\[
= \frac{1}{g(y)} \Pr \left[ |h_{s,r}|^2 \geq g(y) \right] + \frac{1}{g^2(y)} \Pr \left[ |h_{r,d}|^2 + |h_{t,d}|^2 < g(y) \right]
\]

as \( y \to \infty \), using the results of Results 1 and 2 of Appendix 11.1. Thus, we conclude that the large SNR performance of selection decode-and-forward is identical to that of fixed amplify-and-forward.

### 11.2.4 Performance bounds for cooperative diversity

If we suppose that the source and relay know each other’s messages \textit{a priori}, then instead of direct transmission, each would benefit from using a space-time code for two transmit antennas. In this sense, the outage probability of conventional transmit diversity (see Chapter 4) represents an optimistic lower bound on the outage probability of cooperative diversity.

#### 11.2.4.1 Transmit diversity bound

To utilize a space-time code for each terminal, we allocate the channel as in Figure 11.5 (b). Both terminals transmit in all the degrees of freedom of the channel, so their transmitted power is \( P/2 \), half that of direct transmission. The spectral efficiency for each terminal remains \( R \).

For transmit diversity, we model the channel as:

$$y_d[n] = [h_{s,d} \ h_{r,d}] \begin{bmatrix} x_{s}[n] \\ x_{r}[n] \end{bmatrix} + n_d[n]; \quad n = 0, \ldots, N/2 \quad (11.21)$$

An optimal signaling strategy, in terms of minimizing outage probability in the large SNR regime, is to encode information using \([x_s, x_r]^T\) i.i.d. complex Gaussian, each with power \( P/2 \) (see Appendix 11.3). Using this result, the maximum average mutual information as a function of the fading coefficients is given by:

$$I_T = \log \left( 1 + \frac{1}{2} \left[ |h_{s,d}|^2 + |h_{r,d}|^2 \right] \right) \quad (11.22)$$

The outage event \( I_T < R \) is defined by \( |h_{s,d}|^2 + |h_{r,d}|^2 < (2^R - 1)/(y/2) \).

By using the result of Fact 2 in Appendix 11.1, for \( |l_{i,j}|^2 \) exponentially distributed with parameters \( \sigma_{i,j}^{-2} \), the outage probability becomes:

$$p_{T}^{\text{out}}(y, R) := \Pr[I_T < R] \sim \frac{2}{\sigma_{i,d}^2 \sigma_{r,d}^2} \left( \frac{2^R - 1}{y} \right)^2 \quad (11.23)$$

### 11.2.4.2 Orthogonal transmit diversity bound

The transmit diversity bound (11.23) does not take into account the half-duplex constraint. To capture this effect, we constrain the transmit diversity scheme to be orthogonal. When the source and relay can cooperate perfectly, an equivalent model to Equation (11.21), incorporating the relay orthogonality
constraint, consists of parallel channels:

\[ y_d[n] = h_{s,d}x_d[n] + n_d[n], \quad n = 0, \ldots, N/4 \]

\[ y_r[n] = h_{r,d}x_r[n] + n_r[n], \quad n = N/4 + 1, \ldots, N/2 \]  

(11.24)

This pair of parallel channels is utilized half as many times as the corresponding direct transmission channel, so the source must transmit at twice the spectral efficiency in order to achieve the same spectral efficiency as direct transmission.

For each fading realization, the maximum average mutual information can be obtained using independent complex Gaussian inputs. Allocating a fraction \( \alpha \) of the power to \( x_r \), and the remaining fraction \( \bar{\alpha} := (1 - \alpha) \) of the power to \( x_d \), the average mutual information is given by:

\[ I_p = \frac{1}{2} \log \left( \left( 1 + 2\alpha y |h_{s,d}|^2 \right) \left( 1 + 2\bar{\alpha} y |h_{r,d}|^2 \right) \right) \]  

(11.25)

The outage event \( I_P < R \) is equivalent to the outage region \( \alpha |\lambda_{s,d}|^2 + \bar{\alpha} |\lambda_{r,d}|^2 + 2\alpha \bar{\alpha} y |h_{s,d}|^2 |h_{r,d}|^2 < (2^R - 1)/2y \), and for high-SNR in Rayleigh fading, using the result of Equality 2 in Appendix 11.1, with \( u = \alpha \lambda_{s,d}^2 \), \( v = \bar{\alpha} \lambda_{r,d}^2 \), \( \lambda_u = 1/(\alpha \sigma_{s,d}^2) \), \( \lambda_v = 1/((\bar{\alpha} \sigma_{r,d}^2) \varepsilon = (2^R - 1)/(2y), t = 2^R - 1 \), we have:

\[ p_{P|Y}^e(y, R) := \Pr[I_P < R] \sim \frac{1}{4\alpha \bar{\alpha} \sigma_{s,d}^2 \sigma_{r,d}^2} \cdot \frac{2^R [2R \ln(2) - 1] + 1}{y^2} \]  

which is minimized for \( \alpha = 1/2 \), yielding:

\[ p_{P|Y}^e(y, R) \sim \frac{1}{\sigma_{s,d}^2 \sigma_{r,d}^2} \cdot \frac{2^R [2R \ln(2) - 1] + 1}{y^2} \]  

(11.27)

### 11.2.4.3 Incremental relaying

The protocols operate at spectral efficiency \( R \) when the source-destination transmission is successful, and spectral efficiency \( R/2 \) when the relay repeats the source transmission. Thus, we examine outage probability as a function of SNR and the expected spectral efficiency \( \bar{R} \).

For incremental amplify-and-forward, the outage probability as a function of SNR and \( R \) is given by:

\[ p_{IAF}^e(y, R) = \Pr[I_D < R] \Pr[I_AF < R/2 | I_D < R] = \Pr[I_AF < R/2] \]

\[ = \Pr \left[ |h_{s,d}|^2 + \frac{1}{y} f \left( y |h_{s,d}|^2, y |h_{r,d}|^2 \right) < g(y) \right] \]  

(11.28)

where \( I_D, I_AF \) and \( f(\bullet, \bullet) \) are given in Section 11.2.3 and \( g(y) = [2^R - 1]/y = 1/Y \). The second equality follows from the fact that the intersection of the direct and amplify-and-forward outage events is exactly the amplify-and-forward outage event at half the rate. Furthermore, the expected spectral efficiency can be computed as:

\[ \bar{R} = R \Pr \left[ |h_{s,d}|^2 \geq \frac{1}{Y} \right] + \frac{R}{2} \Pr \left[ |h_{s,d}|^2 < \frac{1}{Y} \right] \]

\[ = R \exp \left( -\frac{1}{Y} \right) + \frac{R}{2} \left[ 1 - \exp \left( -\frac{1}{Y} \right) \right] \]  

(11.29)

where the second equality follows from substituting standard exponential results for \( |h_{s,d}|^2 \).

A fixed value of \( \bar{R} \) can arise from several possible \( R \), depending upon the value of SNR. We define a function \( h^{-1}_Y(\bar{R}) := \min h^{-1}_Y(\bar{R}) \) to capture a useful mapping from \( \bar{R} \) to \( R \). For fair comparison with protocols without feedback, we characterize a modified outage expression in the large-SNR regime. Specifically, we compare outage of fixed and selection relaying protocols
to the modified outage $p_{IAF}^{out}(y,\tilde{h}_r^{-1}(\tilde{R}))$. For large SNR, we have:

$$p_{IAF}^{out}(y,\tilde{h}_r^{-1}(\tilde{R})) \sim \left( \frac{1}{2\sigma^2_{s,d}} \right) \left( \frac{2\tilde{R} - 1}{y} \right)^2$$

(11.30)

where we have combined the results of Equality 1 and 3 in Appendix 11.1.

### 11.3 DISTRIBUTED SPACE–TIME CODING

#### 11.3.1 System description

The next unavoidable question within this chapter is that of how well the concept of space–time coding can be implemented in a distributed way where the pull of transmitting antennas is formed by simultaneously using antennas from different users that are not collocated at the same point in the network.

We illustrate transmission protocol for such a system by again using the TDD scheme, although the same concept could be implemented with frequency-division duplexing (FDD) mode as well. In a TDD system, each frame is subdivided into consecutive time slots as shown in Figure 11.6 which is obtained by further elaboration of Figure 11.5. In the first slot, $S$ (source) transmits and $R$ (relay) receives. In the second slot, $S$ and $R$ transmit simultaneously. $D$ (destination) can be in a listening mode in both time slots or only in the second slot. Here we assume that in the second time slot, $S$ and $R$ transmit using a block Alamouti scheme (see, for example, Chapter 4), as if they were the two antennas of a single node. The extension to other space–time coding techniques is almost straightforward. Of course there are differences between conventional space time coding (STC) and the distributed (DSTC) concept. First of all, regenerative relays might make decision errors, so that the symbols transmitted from $R$ could be affected by errors. Then, the links between $S$ and $D$ and between $R$ and $D$ do not have the same statistical properties. Also, even if $S$ and $R$ are synchronous, their packets might arrive at $D$ at different times, as $S$ and $R$ are not colocated.

As before, we denote with $h_{s,d}^{S,R}(i)$, $h_{s,r}^{S,R}(i)$, and $h_{r,d}^{S,R}(i)$, the channel impulse responses between $S$ and $D$, $S$ and $R$, and $R$ and $D$, respectively, during the time slot indexed with $\kappa_n$; $i$ is the block index. Each block of symbols $s(i)$ has size $M$ and it is linearly encoded, so as to generate the $N$-size vector $x_s(n) = Fs(n)$, where $F$ is the $N \times M$ precoding matrix. All channels are finite-impulse response (FIR) of (maximum) order $L_h$ and time invariant over at least a pair of consecutive blocks. So, a CP (cyclic prefix) of length $L \geq L_h$ is inserted at the beginning of each block, to facilitate elimination of interblock interference (IBI), synchronization, and channel equalization at the receiver.

As shown in Figure 11.6, during the first time slot, $S$ sends, consecutively, the two $N$-size information symbol blocks $s(i)$ and $s(i+1)$. The blocks are linearly encoded using the precoding matrix

![Figure 11.6](image-url)

Figure 11.6 Structure of transmit slots: (a) time slots and block indices; (b) information blocks sent by $S$; and (c) information blocks sent by $R$. 


are introduced in order to have a degree of freedom in the power distribution between the receiver, the \( n \)-size vectors \( {\mathbf{y}}_d(n) \) and \( {\mathbf{y}}_r(n) \) received from \( D \) and \( R \) are, respectively

\[
{\mathbf{y}}_d(n) = {\mathbf{H}}_{sd}^{k_d} {\mathbf{F}}_d s(n) + {\mathbf{n}}_d(n), \quad n = i, i + 1
\]

\[
{\mathbf{y}}_r(n) = {\mathbf{H}}_{sr}^{k_r} {\mathbf{F}}_r s(n) + {\mathbf{n}}_r(n), \quad n = i, i + 1
\]

(11.31)

where \( {\mathbf{n}}_d(n) \) and \( {\mathbf{n}}_r(n) \) are the additive-noise vectors in \( R \) and \( D \), respectively. The channel matrices \( {\mathbf{H}}_{sd}^{k_d} \) and \( {\mathbf{H}}_{sr}^{k_r} \) thanks to the insertion of the CP, are \( N \times N \) circulant Toeplitz matrices, with entries \( {\mathbf{H}}_{sd}^{k_d}(i, j) = h_{sd}^{k_d}((i - j) \bmod N) \) and \( {\mathbf{H}}_{sr}^{k_r}(i, j) = h_{sr}^{k_r}((i - j) \bmod N) \), respectively. Because of their circulant and Toeplitz structure, \( {\mathbf{H}}_{sd}^{k_d} \) and \( {\mathbf{H}}_{sr}^{k_r} \) are diagonalized as follows:

\[
{\mathbf{H}}_{sd}^{k_d} = {\mathbf{W}}{\mathbf{A}}_{sd}^{k_d} {\mathbf{W}}^H, \quad {\mathbf{H}}_{sr}^{k_r} = {\mathbf{W}}{\mathbf{A}}_{sr}^{k_r} {\mathbf{W}}^H,
\]

where \( {\mathbf{W}} \) is the \( N \times N \) IFFT matrix with \( {\mathbf{W}}_{kl} = e^{j2\pi kl/N} / \sqrt{N} \), whereas \( {\mathbf{A}}_{sd}^{k_d} \) and \( {\mathbf{A}}_{sr}^{k_r} \) are the \( N \times N \) diagonal matrices, whose entries are \( a_{sd}^{k_d}(k, k) = \sum_{l=0}^{L_h-1} h_{sd}^{k_d}(l)e^{-j2\pi kl/N} \) and \( a_{sr}^{k_r}(k, k) = \sum_{l=0}^{L_h-1} h_{sr}^{k_r}(l)e^{-j2\pi kl/N} \), respectively.

The relay node decodes the received vectors and provides the estimated vectors \( \hat{s}(i) \) and \( \hat{s}(i + 1) \).

During the successive time slot, \( S \) and \( R \) transmit simultaneously, using the Alamouti block code. Specifically, with reference to Figure 11.6, in the first half of the second time slot, \( S \) transmits \( x_r(i + 2) = \alpha_1 {\mathbf{F}}_s(i) \) and \( R \) transmits \( x_r(i + 2) = \alpha_2 {\mathbf{F}}_s(i + 1) \). In the second half, \( S \) transmits \( x_r(i + 3) = \alpha_1 {\mathbf{G}}_s^+ (i + 1) \) while \( R \) transmits \( x_r(i + 3) = -\alpha_2 {\mathbf{G}}_s^+ (i) \). To guarantee maximum spatial diversity, the two matrices \( \mathbf{G} \) and \( \mathbf{F} \) are related to each other by \( \mathbf{G} = \mathbf{J}\mathbf{F}^T \), as in [31], where \( \mathbf{J} \) is a time-reversal matrix. To guarantee maximum spatial diversity, we introduce also the unitary matrix \( \mathbf{Q}^{k_r} \), satisfying the relationships

\[
Q^{k_r}Q^{k_r} = I_{2n}, \quad Q^{k_r}A_{k_r} = I_{2n} \quad \text{and} \quad Q^{k_r}A_{k_r} = I_{2n}
\]

(11.33)

where \( \tilde{\mathbf{F}} := \mathbf{W}^H \mathbf{F} \) and \( \tilde{\mathbf{G}} := \mathbf{W}^T \mathbf{G} \). For the sake of simplicity, we assume that orthogonal frequency-division multiplexing (OFDM) is performed at both \( S \) and \( R \) nodes, so that \( N = M, \mathbf{F} = \mathbf{W} \), and thus, \( \tilde{\mathbf{F}} = \mathbf{I}_n \) and \( \mathbf{G} = \mathbf{W} \). We also introduce the orthogonal matrix:

\[
A_{k_r} := \left( \begin{array}{cc}
\alpha_1 A_{sd}^{k_d} & \alpha_2 A_{sr}^{k_r} \\
-\alpha_2 A_{sd}^{k_d} & \alpha_1 A_{sr}^{k_r}
\end{array} \right)
\]

(11.34)

such that \( A_{k_r}A_{k_r} := I_2 \otimes \tilde{\mathbf{A}}_{k_r}^2 \), where \( \tilde{\mathbf{A}}_{k_r} := \alpha_1 \left| A_{sd}^{k_d} \right|^2 + \alpha_2 \left| A_{sr}^{k_r} \right|^2 \), whereas \( \otimes \) denotes the Kronecker product. We introduce also the unitary matrix \( Q^{k_r} := A_{k_r}(I_2 \otimes \tilde{\mathbf{A}}_{k_r}^{-1}) \), satisfying the relationships

\[
Q^{k_r}Q^{k_r} = I_{2n}, \quad Q^{k_r}A_{k_r} = I_{2n} \quad \text{and} \quad Q^{k_r}A_{k_r} = I_{2n}
\]

Exploiting the above equalities and multiplying...
the vector \( \mathbf{u} := [(W^H \mathbf{y}_d(i + 2))^T, (W^T \mathbf{y}_d^*(i + 3))^T]^T \) by the matrix \( Q^{k+2H} \), we get:

\[
\begin{bmatrix}
\mathbf{r}(i) \\
\mathbf{r}(i + 1)
\end{bmatrix} := Q^{k+2H} \mathbf{u} = \begin{bmatrix}
\tilde{A}_{sd}^{k+2} & -\tilde{A}_{sd}^{k+2} \\
\tilde{A}_{rd}^{k+2} & \tilde{A}_{rd}^{k+2}
\end{bmatrix} \mathbf{s} + \begin{bmatrix}
\tilde{A}_{rd}^{k+2} \\
-\tilde{A}_{sd}^{k+2}
\end{bmatrix} \hat{\mathbf{s}} + \mathbf{n}
\]

where

\[
\begin{align*}
\mathbf{s} & := [\mathbf{s}(i)^T, \mathbf{s}(i + 1)^T]^T, \\
\hat{\mathbf{s}} & := [\tilde{\mathbf{s}}(i)^T, \tilde{\mathbf{s}}(i + 1)^T]^T, \\
\tilde{\mathbf{A}}_{sd}^{k+2} & := \alpha_1 \tilde{\mathbf{A}}_{sd}^{k+1} \tilde{\mathbf{A}}_{sd}^{-1/2}, \\
\tilde{\mathbf{A}}_{rd}^{k+2} & := \alpha_2 \tilde{\mathbf{A}}_{rd}^{k+1} \tilde{\mathbf{A}}_{rd}^{-1/2}, \\
\hat{\mathbf{n}} & := [\hat{\mathbf{n}}^T(i), \hat{\mathbf{n}}^T(i + 1)]^T = Q^{k+2H} [\mathbf{n}^T(i + 2), \mathbf{n}^T(i + 3)]^T.
\end{align*}
\]

If the two transmit antennas use the same power, i.e. \( \alpha_1 = \alpha_2 \), and there are no decision errors at the relay node, i.e. \( \hat{\mathbf{s}}(n) = s(n), \ n = i, i + 1 \), the previous equations reduce to the classical block Alamouti equations discussed in Chapter 4.

Since \( Q^{k+2H} \) is unitary, if \( \mathbf{n} \) is white, \( \hat{\mathbf{n}} \) is also white, with covariance matrix \( \mathbf{C}_n = \sigma^2_n \mathbf{I}_{2N} \). Furthermore, since all matrices \( \mathbf{A} \) appearing in Equation (11.35) are diagonal, the system Equation (11.35) of 2N equations can be decoupled into N independent systems of two equations in two unknowns, each equation representing a single subcarrier. Introducing the vectors \( \mathbf{r}_k := [r_k(i), r_k(i + 1)]^T, \mathbf{s}_k := [s_k(i), s_k(i + 1)]^T, \mathbf{\tilde{s}}_k := [\tilde{s}_k(i), \tilde{s}_k(i + 1)]^T, \) and \( \mathbf{\bar{n}}_k := [\bar{n}_k(i), \bar{n}_k(i + 1)]^T \), representing the \( k \)th subcarrier, with \( k = 0, \ldots, N - 1 \) [for simplicity of notation, we drop the block index and set \( \tilde{\mathbf{A}}_{sd} = \tilde{\mathbf{A}}_{sd}^{k+2}(k, k), \tilde{\mathbf{A}}_{rd} = \tilde{\mathbf{A}}_{rd}^{k+2}(k, k) \)], Equation (11.35) is equivalent to

\[
\mathbf{r}_k = \begin{bmatrix}
\tilde{\mathbf{A}}_{sd}^{k+2} & -\tilde{\mathbf{A}}_{sd}^{k+2} \\
\tilde{\mathbf{A}}_{rd}^{k+2} & \tilde{\mathbf{A}}_{rd}^{k+2}
\end{bmatrix} \mathbf{s}_k + \begin{bmatrix}
\tilde{\mathbf{A}}_{rd}^{k+2} \\
-\tilde{\mathbf{A}}_{sd}^{k+2}
\end{bmatrix} \hat{\mathbf{s}}_k + \mathbf{n}_k,
\]

where \( \mathbf{r}_k \) represents a sufficient statistic for the decision on the transmitted symbol vector \( \mathbf{s}_k \).

Due to synchronization error, in general, the blocks transmitted from \( S \) and \( R \) arrive at \( D \) at different times. However, if the difference in arrival times \( \tau_d \) is incorporated in the CP used from both \( S \) and \( R \), \( D \) is still able to get \( N \) samples from each received block, without IBI.

If due to synchronization error, the block coming from \( S \) arrives with a delay of \( L_d \) samples, with respect to arrival instant from \( R \), the only difference, with respect to the case of perfect synchronization, is that the transfer function \( \tilde{\mathbf{A}}_{rd}(k) \) in Equation (11.37) will be substituted by \( \tilde{\mathbf{A}}_{sd}(k)e^{-j2\pi L_d k/N} \). Such a substitution does not affect the useful term, but only affects the interfering term. The price paid for this robustness is the increase of the CP length \( L \), which, in its turn, reflects into a rate loss. However, this loss can be made small by choosing a block length \( N \) much greater than \( L \), or by selecting only relays that are relatively close to the source, so as to make the relative delay small.

The ML detector would assume the knowledge, at the destination node, of the set of error probabilities \( p_{e1}(k) \) and \( p_{e2}(k) \) at the relay, with \( k = 0, \ldots, N - 1 \). If this knowledge is not available, a suboptimum scalar detector can be implemented, instead of the ML detector. In the case of BPSK modulation the decision on the transmitted symbol \( sk(n) \) can be simply obtained as:

\[
\hat{s}(n) = \text{sign}(\text{Re}[\mathbf{r}(n)]), \quad n = i, i + 1
\]

For high signal-to-noise ratio (SNR) at the relay (i.e., when \( R \) makes no decision errors), the symbol-by-symbol decision in \( D \) becomes optimal and, thus, the decoding rule (11.38) provides the same performance as the optimal receiver. When the decision errors at the relay side cannot be neglected, the suboptimal receiver introduces a floor in the BER curve, because the symbol-by-symbol decision treats the wrong received symbols as interference. A cognitive radio should make the choice between the decoding rules as a tradeoff between performance and computational complexity (power consumption), taking into account the need for the ML detector to make available, at the destination node, the error probabilities of the relay node.
11.3.2 BER analysis in DSTC

For error-free Source to Relay link, using the same derivations introduced in Section 11.3.1, a symbol-by-symbol detector is the optimal detector, and the signal-to-noise ratio on the kth symbol in the nth block is:

$$\text{SNR}_k(n) = \gamma_k(n) = \frac{A^2}{\sigma_n^2} \left( \left| \Lambda_{id}^{k}(k, k) \right|^2 + \alpha \left| \Lambda_{rd}^{k+1}(k, k) \right|^2 + (1 - \alpha) \left| \Lambda_{rd}^{k+2}(k, k) \right|^2 \right)$$  \hspace{1cm} (11.39)

with $k = 0, 1, \ldots, N - 1$; $n = i, i + 1$, and $\alpha = \alpha^2$. We assume that the variance of the channel impulse-response coefficients is proportional to $1/d_r^2$, with $r \geq 1$, where $d$ is the distance of the link and calculate the BER for fast and slow fading channel.

In fast-fading $\Lambda_{id}^{k}(k, k)\), $\Lambda_{rd}^{k+1}(k, k)$, and $\Lambda_{rd}^{k+2}(k, k)$ are independent. The error probability for BPSK, conditioned to a given channel realization, is given by:

$$P_e(k) = \frac{1}{2} \text{erfc}(\sqrt{0.5\gamma_k})$$  \hspace{1cm} (11.40)

where $\gamma_k$ is given by Equation (11.39). For each subcarrier $k$, $\gamma_k$ is given by the sum of three statistically independent random variables, each one distributed according to a $\chi^2$ pdf with two degrees of freedom. Thus, the BER $P_b$ averaged over the channel realizations is [32]:

$$P_b = \frac{1}{2} \sum_{k=1}^{Q} \pi_k \left[ 1 - \sqrt{\frac{\gamma_k}{1 + \gamma_k}} \right]$$  \hspace{1cm} (11.41)

where $Q = 3$, and

$$\pi_k := \prod_{i \neq k=1}^{Q} \frac{\gamma_1}{\gamma_1 - \gamma_k}; \quad \gamma_1 := \frac{A^2}{\sigma_n^2} \frac{\sigma_n^2}{d_r^2}; \quad \gamma_2 := \frac{A^2}{\sigma_n^2} \frac{\alpha \sigma_n^2}{d_r^2}; \quad \gamma_3 := \frac{A^2}{\sigma_n^2} \frac{(1 - \alpha) \sigma_n^2}{d_r^2};$$

$$\sigma_n^2 = \sigma^2_h (L_h + 1).$$

The optimal value of $\alpha$ can be found by minimizing Equation (11.41).

In slow-fading $\Lambda_{id}^{k}(k, k)\), $\Lambda_{rd}^{k+1}(k, k)$, and $\Lambda_{rd}^{k+2}(k, k)$, the $\gamma_k$ on the kth bit of the received block is:

$$\gamma_k(n) = \frac{A^2}{\sigma_n^2} \left( (1 + \alpha) \left| \Lambda_{id}^{k}(k, k) \right|^2 + (1 - \alpha) \left| \Lambda_{rd}^{k+2}(k, k) \right|^2 \right)$$  \hspace{1cm} (11.42)

and the BER averaged over the channel realizations is given by (11.41), with $Q = 2$ and

$$\gamma_1 := \frac{A^2}{\sigma_n^2} \frac{(1 + \alpha) \sigma_n^2}{d_r^2}; \quad \gamma_2 := \frac{A^2}{\sigma_n^2} \frac{(1 - \alpha) \sigma_n^2}{d_r^2};$$

For source to relay link with errors the decision on the transmitted symbol $s_k(i)$ is performed by taking a hard decision on

$$r_k(i) = \left| \Lambda_{id}^{k+1}(k, k) \right|^2 s_k(i) + \left| \Lambda_{rd}^{k+1}(k, k) \right|^2 \tilde{s_k}(i) + \left| \Lambda_{id}^{k+2}(k, k) \right|^2 \tilde{s_k}(i + 1) - \left| \Lambda_{rd}^{k+2}(k, k) \right|^2 \tilde{s_k}(i + 1) + \hat{e}_k(i)$$  \hspace{1cm} (11.43)

where $\Lambda_{id}^{k+1}(k, k)\), $\Lambda_{rd}^{k+1}(k, k)$, and $\Lambda_{rd}^{k+2}(k, k)$ are independent. In the sequel we use the following notation for the possible events: $H_1 = \{s_k(i) = s_k(i); \tilde{s_k}(i + 1) = s_k(i + 1)\}; \quad H_2 = \{s_k(i) = -s_k(i); \tilde{s_k}(i + 1) = -s_k(i + 1)\}; \quad H_3 = \{s_k(i) = \tilde{s_k}(i); \tilde{s_k}(i + 1) = -s_k(i + 1)\}; \quad H_4 = \{s_k(i) = -\tilde{s_k}(i); \tilde{s_k}(i + 1) = s_k(i + 1)\}; \quad H_5 = \{s_k(i) = \tilde{s_k}(i); \tilde{s_k}(i + 1) = s_k(i + 1)\}; \quad H_6 = \{s_k(i) = -\tilde{s_k}(i); \tilde{s_k}(i + 1) = -s_k(i + 1)\}; \quad H_7 = \{s_k(i) = s_k(i + 1); \tilde{s_k}(i + 1) = -s_k(i + 1)\}; \quad H_8 = \{s_k(i) = \tilde{s_k}(i); \tilde{s_k}(i + 1) = s_k(i + 1)\}; \quad H_9 = \{s_k(i) = -\tilde{s_k}(i); \tilde{s_k}(i + 1) = -s_k(i + 1)\}; \quad \epsilon_{1,1} = \{s_k(i) = A; \tilde{s_k}(i + 1) = A\}; \quad \epsilon_{1,-1} = \{s_k(i) = A; \tilde{s_k}(i + 1) = -A\}; \quad \epsilon_{-1,1} = \{s_k(i) = -A; \tilde{s_k}(i + 1) = -A\}; \quad \epsilon_{-1,-1} = \{s_k(i) = -A; \tilde{s_k}(i + 1) = A\}$. With $p_{e_1}(k)$ and $p_{e_2}(k)$ the conditional (to a given channel realization) error probabilities, at the relay node, on $s_k(l)$ and $s_k(2)$, respectively, we
have  \( P(H_1) = (1 - p_{11}(k))(1 - p_{22}(k)) \),  \( P(H_2) = p_{11}(k)(1 - p_{22}(k)) \),  \( P(H_3) = (1 - p_{11}(k))p_{22}(k) \), and  \( P(H_2) = p_{11}(k)p_{22}(k) \). Denoting, for simplicity,  \( u_i = R_{R_k(i)} \), and assuming that the information symbols are i.i.d. binary phase-shift keying (BPSK) symbols that may assume the values \( \delta \) where

\[
\text{and}
\]

\[
\beta_k
\]

\[
\text{Setting } \bar{\eta}_k := \text{Re}\{\bar{h}_k(i)\}, \text{we have:}
\]

\[
P[u_i > 0/\varepsilon_{1,1}, H_j] = P\left\{ -\left| \lambda_{d}^{k_1,2} \right|^2 - \left| \lambda_{r}^{k_1,2} \right|^2 \right\} A + \bar{\eta}_r > 0 \right\} = \frac{1}{2} \text{erfc}\left( \frac{A\beta_k}{\sigma_n} \right)
\]

where \( \beta_k \) is given by:

\[
\beta_k = \sqrt{\alpha_1^2 \left| \lambda_{\sigma_d}^{k_1}(k, k) \right|^2 + \alpha_2^2 \left| \lambda_{\sigma_d}^{k_2}(k, k) \right|^2}
\]

Using this approach, we find:

\[
P[u_i > 0/\varepsilon_{1,1}, H_1] = \frac{1}{2} \text{erfc}\left( \frac{A\beta_k}{\sigma_n} \right)
\]

\[
P[u_i > 0/\varepsilon_{1,1}, H_1] = \frac{1}{2} \text{erfc}\left( \frac{A\beta_k}{\sigma_n} \right)
\]

\[
P[u_i > 0/\varepsilon_{1,1}, H_2] = \frac{1}{2} \text{erfc}\left( \frac{A\beta_k}{\sigma_n} \right)
\]

\[
P[u_i > 0/\varepsilon_{1,1}, H_2] = \frac{1}{2} \text{erfc}\left( \frac{A\beta_k}{\sigma_n} \right)
\]

\[
P[u_i > 0/\varepsilon_{1,1}, H_3] = \frac{1}{2} \text{erfc}\left( \frac{A(\beta_k + \gamma_k)}{\sigma_n} \right)
\]

\[
P[u_i > 0/\varepsilon_{1,1}, H_3] = \frac{1}{2} \text{erfc}\left( \frac{A(\beta_k - \gamma_k)}{\sigma_n} \right)
\]

\[
P[u_i > 0/\varepsilon_{1,1}, H_4] = \frac{1}{2} \text{erfc}\left( \frac{A(\delta_k + \gamma_k)}{\sigma_n} \right)
\]

\[
P[u_i > 0/\varepsilon_{1,1}, H_4] = \frac{1}{2} \text{erfc}\left( \frac{A(\delta_k - \gamma_k)}{\sigma_n} \right)
\]

where \( \delta_k \) and \( \gamma_k \) are defined as:

\[
\gamma_k = \frac{2\alpha_1\alpha_2\text{Re}\left\{ \lambda_{\sigma_d}^{k_1}(k, k)\lambda_{\sigma_d}^{k_2}(k, k) \right\}}{\sqrt{\alpha_1^2 \left| \lambda_{\sigma_d}^{k_1}(k, k) \right|^2 + \alpha_2^2 \left| \lambda_{\sigma_d}^{k_2}(k, k) \right|^2}}
\]

\[
\delta_k = \frac{\alpha_1^2 \left| \lambda_{\sigma_d}^{k_1}(k, k) \right|^2 - \alpha_2^2 \left| \lambda_{\sigma_d}^{k_2}(k, k) \right|^2}{\sqrt{\alpha_1^2 \left| \lambda_{\sigma_d}^{k_1}(k, k) \right|^2 + \alpha_2^2 \left| \lambda_{\sigma_d}^{k_2}(k, k) \right|^2}}
\]
Figure 11.7 Average BER versus $\alpha$: (a) without errors at the relay; (b) with errors at the relay.

Repeating this same approach, we are able to find

$$P_e(k) = \frac{1}{2} \text{erfc} \left( \frac{A\beta_k}{\sqrt{\sigma_n^2}} \right)(1 - P_{e1}(k))(1 - P_{e2}(k)) + \frac{1}{2} \text{erfc} \left( \frac{A\delta_k}{\sqrt{\sigma_n^2}} \right)$$

where, for $n = i + 2, i + 3$

$$P_{e1}(k) = P_{e2}(k) = \frac{1}{2} \text{erfc} \left( \frac{A}{\sqrt{\sigma_n^2(k)}} \right)$$

with $\sigma_n^2(k) := \sigma_n^2 / |\Lambda_{k}^2(k, k)|^2$, $k_{i+2} = k_{i+3}$.

The average BER can then be obtained by averaging Equation (11.45) over the channel realizations.

As an example, Figure 11.7, shows average BER versus $\alpha$ (setting $\alpha = \alpha_1 = 1 - \alpha_2$), for different values of the distance $d_{sd}$ (and thus, of $y_R$) between $R$ and $D$ (all distances are normalized with respect to the distance $d_{sd}$ between $S$ and $D$). Figure 11.7(a), shows the ideal case where there are no errors at the relay node, and Figure 11.7(b), shows average BER as a function of $\alpha$, but for different values of the $y_R$ at the relay node. We can observe that, as $y_R$ decreases, the system tends to allocate less power to the relay node (the optimal value of $\alpha$ is greater than 0.5), as the relay node becomes less and less reliable.

11.4 GENERALIZATION OF DISTRIBUTED SPACE–TIME-CODING BASED ON COOPERATIVE DIVERSITY

11.4.1 System and channel model

In this section we focus on a wireless network with a set $M = 1, 2, \ldots, m$ of transmitting terminals. Each transmitting source terminal $s \in M$ has information to transmit to a single destination terminal, denoted $d(s) \notin M$, potentially using terminals $M - \{s\}$ as relays. Thus, there are $m$ cooperating terminals communicating to $d(s)$. If algorithms require the relays to decode fully the source message,
COOPERATIVE DIVERSITY IN COGNITIVE WIRELESS NETWORKS

Figure 11.8 The two phases of cooperative diversity transmission.

Figure 11.9 Repetition-based medium-access control

the decoding set $D(s)$ is the set of relays that can decode the message of source $s$. In the case of amplify-and-forward cooperative diversity, $D(s) = M - \{s\}$.

Both classes of algorithms consist of two transmission phases, as illustrated in Figure 11.8. In the first phase, the source broadcasts to its destination and all potential relays. During the second phase of the algorithms, the other terminals relay to the destination, either on orthogonal subchannels in the case of repetition-based cooperative diversity, or simultaneously on the same subchannel in the case of space–time-coded cooperative diversity.

An example of channel and subchannel allocations for repetition-based cooperative diversity, in which relays either amplify what they receive or fully decode and repeat the source signal, is shown in Figure 11.9.

Channel and subchannel allocations for space-time-coded cooperative diversity, in which relays utilize a suitable space–time code in the second phase and can therefore transmit simultaneously on the same subchannel, is illustrated in Figure 11.10. Again, transmission between source $s$ and destination $d(s)$ utilizes $1/m$ of the total degrees of freedom in the channel. In contrast to noncooperative transmission and repetition-based cooperative diversity transmission, each terminal, employing space–time-coded cooperative diversity transmits in $\frac{1}{m}$ the total degrees of freedom in the channel. It is important to keep track of these ratios when normalizing power and bandwidth in the sequel.

During the first phase, each potential relay $r \in M - \{s\}$ receives

$$y_r[n] = a_{r,x}x_s[n] + z_r[n]$$

(11.46)
The destination receives signals during both phases. During the first phase, the received signal at $d(s)$ is

$$y_d(s)[n] = a_{r,d(s)} x_r[n] + z_{d(s)}[n]$$  \hspace{1cm} (11.47)

in the corresponding subchannel. During the second phase, the equivalent channel models are different for repetition-based and space–time-coded cooperative diversity. For repetition-based cooperative diversity, the destination receives separate retransmissions from each of the relays $r \in M - \{s\}$ and the received signal at $d(s)$ is

$$y_d(s)[n] = a_{r,d(s)} x_r[n] + z_{d(s)}[n]$$  \hspace{1cm} (11.48)

in the corresponding subchannel, where $x_r[n]$ is the transmitted signal of relay $r$. For space–time-coded cooperative diversity, all of the relay transmissions occur in the same subchannel and superimpose at the destination, so that:

$$y_d(s)[n] = \sum_{r \in D(s)} a_{r,d(s)} x_r[n] + z_{d(s)}$$  \hspace{1cm} (11.49)

in the corresponding subchannel.

Parameter $a_{i,j}$ captures the effects of path loss, shadowing, and frequency nonselective fading, and $z_j[n]$ represents the effects of receiver noise and other forms of interference in the system. Statistically, we model $a_{i,j}$ as zero-mean, independent, circularly symmetric complex Gaussian random variables with variances $1/\lambda_{i,j}$, so that the magnitudes $|a_{i,j}|$ are Rayleigh distributed ($|a_{i,j}|^2$ are exponentially distributed with parameter $\lambda_{i,j}$), and the phases of $a_{i,j}$ are uniformly distributed on $[0, 2\pi)$. $z_j[n]$ is modeled as zero-mean mutually independent, circularly symmetric, complex Gaussian random sequences with variance $N_0$.

For a continuous-time channel with total bandwidth $W$ hertz available for transmission, the discrete-time model contains $W$ two-dimensional symbols per second (2D/s). If the transmitting terminals have an average power constraint in the continuous-time channel model of $P_c$ joules per second ($J/s$), this translates into a discrete-time power constraint of $P = mP_c/W$ $J/2D$, since each terminal transmits in a fraction $1/m$ of the available degrees of freedom for noncooperative transmission and repetition-based cooperative diversity. The channel model is parameterized by the SNR random variables $\text{SNR}|a_{i,j}|^2$, where $\text{SNR} = mP_c/N_0W = P/N_0$ is the SNR without fading. For space–time-coded cooperative diversity, the terminals transmit in half the available degrees of freedom, so the discrete-time power constraint becomes $2P/m$. 

![Figure 11.10 Space–time-coded channel allocations across frequency and time for $m$ transmitting terminals. For source $s$, $D(s)$ denotes the set of decoding relays participating in a space–time code during the second phase.](image)
In addition to SNR, transmission schemes are further parameterized by the spectral efficiency \( R \) bits per second per hertz (b/s/Hz) attempted by the transmitting terminals. Throughout this section, \( R \) is the transmission rate normalized by the number of degrees of freedom utilized by each terminal under noncooperative transmission, not by the total number of degrees of freedom in the channel. In addition a normalized rate \( R_{\text{norm}} \) defined as

\[
R_{\text{norm}} = \frac{R}{\log(1 + \text{SNR} \sigma^2_{d(s)})} \tag{11.50}
\]

will be used. For an AWGN channel with bandwidth (W/m) and SNR given by \( \text{SNR} R \sigma^2_{d(s)} \), \( R_{\text{norm}} < 1 \) is the spectral efficiency normalized by the maximum achievable spectral efficiency, i.e. channel capacity. Results under (SNR, R) exhibit a tradeoff between the normalized SNR gain and spectral efficiency of a protocol, while results under (SNR, \( R_{\text{norm}} \)) exhibit a tradeoff between the diversity order and normalized spectral efficiency of a protocol. The latter tradeoff, called the diversity-multiplexing tradeoff, was developed originally in the context of multiple-antenna systems.

### 11.4.2 Cooperative diversity based on repetition

Since the channel average mutual information \( I_{\text{rep}} \) is a function of, the coding scheme, the rule for including potential relays into the decoding set \( D(s) \), and the fading coefficients of the channel, it too is a random variable. The event \( I_{\text{rep}} < R \) when this mutual information random variable falls below some fixed spectral efficiency \( R \) is referred to as an outage event, and the probability of an outage event, \( \text{Pr}[I_{\text{rep}} < R] \), is referred to as the outage probability of the channel. Since \( D(s) \) is a random set, we have

\[
\text{Pr}[I_{\text{rep}} < R] = \sum_{D(s)} \text{Pr}[D(s)] \text{Pr}[I_{\text{rep}} < R | D(s)] \tag{11.51}
\]

For repetition coding, the random codebook at the source is generated independent and identically distributed (i.i.d.) circularly symmetric, complex Gaussian; each of the relays employs the exact same codebook as the source. Conditioned on \( D(s) \) being the decoding set, the mutual information between \( s \) and \( d(s) \) is:

\[
I_{\text{rep}} = \frac{1}{m} \log \left( 1 + \text{SNR} |a_{s,d(s)}|^2 + \text{SNR} \sum_{r \in D(s)} |a_{r,d(s)}|^2 \right) \tag{11.52}
\]

\( \text{Pr}[I_{\text{rep}} < R | D(s)] \) involves \(|D(s)|+1 \) independent fading coefficients, so we expect it to decay asymptotically in proportion to \( \text{SNR} |D(s)|+1 \). In the sequel, we develop the following high-SNR approximation:

\[
\text{Pr}[I_{\text{rep}} < R | D(s)] \sim \left[ \frac{2mR - 1}{\text{SNR}} \right]^{\lvert D(s) \rvert + 1} \times \lambda_{r,d(s)} \prod_{r \in D(s)} \lambda_{r,d(s)} \times \frac{1}{(|D(s)|+1)!} \tag{11.53}
\]

(11.52) is expressed in such a way that the first term captures the dependence upon SNR and the second term captures the dependence upon \( \{\lambda_{i,j}\} \).

To prove (11.52) we start with the following claim;

**Claim 1:** If \( u_k, k = 1, 2, \ldots, m \), are positive, independent random variables with \( \lim_{\varepsilon \to 0} p_{u_k}(\varepsilon u) \geq \lambda_k \) and \( p_{u_k}(\varepsilon u) \leq \lambda_k \) then

\[
\lim_{\varepsilon \to 0} \frac{1}{e^m} \text{Pr} \left[ \sum_{k=1}^{m} u_k < \varepsilon \right] = \frac{1}{m!} \prod_{k=1}^{m} \lambda_k \tag{11.54}
\]

As an example the exponential distribution satisfies both above requirements. More generally, however, this result suggests that many of our results hold for a much larger class of PDFs, and, in particular, depend mainly upon properties of the PDFs near the origin.
Proof: If \( s_n = \sum_{k=1}^{m} u_k, n \leq m \) then
\[
\Pr \left[ \sum_{k=1}^{m} u_k < \epsilon \right] = \Pr[s_m < \epsilon] = \int_{0}^{\epsilon} p_{s_m}(s)ds = \epsilon \int_{0}^{1} p_{s_m}(\epsilon w)dw
\]
(11.55)
where the last equality results from the change of variables \( w = s/\epsilon \). Thus, it is sufficient for us to compute the limit
\[
\lim_{\epsilon \to 0} \frac{1}{\epsilon^{(m-1)}} \int_{0}^{1} p_{s_m}(\epsilon w)dw
\]
(11.56)
By exploiting Fatou’s lemma we get for lower-bound of liminf:
\[
\liminf_{\epsilon \to 0} \frac{1}{\epsilon^{(m-1)}} \int_{0}^{1} p_{s_m}(\epsilon w)dw \geq \int_{0}^{1} \left\{ \liminf_{\epsilon \to 0} \frac{1}{\epsilon^{(m-1)}} p_{s_m}(\epsilon w) \right\} dw
\]
(11.57)
\( s_m = s_{m-1} + u_m \), and by independence the PDF of \( s_m \) is the convolution of the PDFs of \( s_{m-1} \) and \( u_m \). Specifically, since \( u_m \) is positive, we have:
\[
p_{s_m}(s) = \int_{0}^{s} p_{s_{m-1}}(s - r)p_{u_m}(r)dr = s \int_{0}^{1} p_{s_{m-1}}(s(1 - y))p_{u_m}(sy)dy
\]
(11.58)
where the last equality results from the change of variables \( y = r/s \).
Letting
\[
A_m(w) = \liminf_{\epsilon \to 0} \frac{1}{\epsilon^{(m-1)}} p_{s_m}(\epsilon w)
\]
(11.59)
and substituting into Equation (11.58), again exploiting Fatou’s lemma, we obtain the recursion
\[
A_m(w) = \liminf_{\epsilon \to 0} \frac{1}{\epsilon^{(m-2)}} p_{s_{m-1}}(\epsilon w(1 - y)) + \left[ \liminf_{\epsilon \to 0} p_{u_m}(\epsilon wy) \right] dy \geq \lambda_m w \int_{0}^{1} A_{m-1}(w(1 - y))dy
\]
(11.60)
where the last inequality follows from the initial assumption \( \liminf_{\epsilon \to 0} p_{u_m}(\epsilon u) \geq \lambda_k \) and substitution of \( A_{m-1}(w(1 - y)) \). Beginning with \( A_1(w) \geq \lambda_1 \), the recursion (11.60) yields:
\[
A_m(w) \geq \frac{1}{(m-1)!} w^{(m-1)} \prod_{k=1}^{m} \lambda_k
\]
(11.61)
As a result, Equation (11.57) with (11.59) and (11.61) yields
\[
\liminf_{\epsilon \to 0} \frac{1}{\epsilon^{m}} \Pr[s_m < \epsilon] \geq \frac{1}{m!} \prod_{k=1}^{m} \lambda_k.
\]
(11.62)
To upper-bound the limsup, we obtain a recursive upper bound for the PDF of \( s_m \). Letting \( B_m(w, \epsilon) = p_{s_m}(\epsilon w) \), we have
\[
B_m(w, \epsilon) = \epsilon w \int_{0}^{1} p_{s_{m-1}}(\epsilon w(1 - y))p_{u_m}(\epsilon wy)dy \leq \epsilon \lambda_m w \int_{0}^{1} B_{m-1}(w(1 - y), \epsilon)dy
\]
(11.63)
where the equality comes from the convolution (11.58), and the inequality follows from the initial condition \( p_{u_m}(\epsilon u) \leq \lambda_k \) and substitution of \( B_{m-1}(w(1 - y)) \). Beginning with \( B_1(w, \epsilon) \leq \lambda_1 \), Equation (11.63) yields an upper bound very similar to the lower bound in (11.61), namely
\[
B_m(w, \epsilon) \leq \epsilon^{(m-1)} w^{(m-1)} \frac{1}{(m-1)!} \prod_{k=1}^{m} \lambda_k
\]
(11.64)
Then
\[
\limsup_{\varepsilon \to 0} \frac{1}{\varepsilon (m - 1)} \int_0^1 p_{m}(\varepsilon w) \, dw \leq \limsup_{\varepsilon \to 0} \frac{1}{\varepsilon (m - 1)} \int_0^1 B_m(w, \varepsilon) \, dw \leq \frac{1}{(m - 1)!} \prod_{k=1}^m \lambda_k
\]  \quad (11.65)

Together with the fact that, in general, \( \lim \inf \leq \lim \sup \), (11.62), and (11.65) yield the desired result (11.54).

Now we use the result of Claim 1 to obtain a large SNR approximation for \( \Pr[I_{\text{rep}} < R|D(s)] \), the conditional outage probability for repetition decode-and-forward cooperative diversity for source \( s \) given a set of decoding relays \( D(s) \). As in Equation (11.52), \( I_{\text{rep}} \) is of the form:
\[
I_{\text{rep}} = \frac{1}{m} \log \left( 1 + \text{SNR} \sum_{k=1}^m u_k \right)
\]  \quad (11.66)
where \( u_k \) are independent exponential random variables with parameters \( \lambda_k \), \( k = 1, 2, \ldots, m \). After some algebraic manipulations, the outage probability reduces to exactly the same form as in Claim 1:
\[
\Pr[I_{\text{rep}} < R|D(s)] = \Pr \left[ \sum_{k=1}^m u_k < \varepsilon \right]
\]  \quad (11.67)
with \( \varepsilon = (2^m R - 1)/\text{SNR} \to 0 \) as \( \text{SNR} \to \infty \). Thus, Claim 1 and continuity yield, for large SNR, the approximation:
\[
\Pr[I_{\text{rep}} < R|D(s)] \sim \left[ \frac{2^m R - 1}{\text{SNR}} \right]^m \frac{1}{m!} \prod_{k=1}^m \lambda_k
\]  \quad (11.68)
If, with the channel allocation illustrated in Figure 11.9, the relays employ independently generated codebooks, corresponding to utilizing parallel channels, the mutual information would become a sum of logarithmic terms:
\[
I_{\text{rep}} = \frac{1}{m} \sum_{r \in |D(s)|} \log \left( 1 + \text{SNR} |a_{s,r}|^2 \right)
\]  \quad (11.69)
instead of the log-sum in Equation (11.52), which is larger than (11.52). This means that parallel channel coding is more bandwidth efficient than repetition coding, as we might expect.

In order to calculate the outage probability defined by Equation (11.51) we now consider the term \( \Pr[D(s)] \), the probability of a particular decoding set. As one rule for selecting from the potential relays, we can require that a potential relay fully decodes the source message in order to participate in the second phase. Since the realized mutual information between \( s \) and \( r \) for i.i.d. complex Gaussian codebooks is given by \( \log \left( 1 + \text{SNR} |a_{s,r}|^2 \right) / m \), under this rule we have:
\[
\Pr[r \in D(s)] = \Pr[|a_{s,r}|^2 > (2^m R - 1)/\text{SNR}] = \exp \left[ -\lambda_{s,r} (2^m R - 1)/\text{SNR} \right]
\]
Moreover, since each potential relay makes its decision independently under the above restrictions, and the fading coefficients are independent in our model, we have
\[
\Pr[D(s)] = \prod_{r \in D(s)} \exp \left[ -\lambda_{s,r} (2^m R - 1)/\text{SNR} \right]
\]
\[
\times \prod_{r \notin D(s)} \left( 1 - \exp \left[ -\lambda_{s,r} (2^m R - 1)/\text{SNR} \right] \right) \sim \left[ \frac{2^m R - 1}{\text{SNR}} \right]^{-|D(s)|+1} \prod_{r \notin D(s)} \lambda_{s,r}
\]  \quad (11.70)
Note that any selection means by which \( \Pr[r \in D(s)] \sim 1 \) and \( (1 - \Pr[r \in D(s)]) \mu 1/\text{SNR} \), for \( \text{SNR} \) large, independently for each \( r \), will result in similar asymptotic behavior for \( \Pr[D(s)] \). Substituting
Figure 11.11 Outage probabilities for repetition-based cooperative diversity (numeric integration of the outage probability (solid lines) and calculation of the outage probability approximation (11.71) (dashed lines) versus SNR for different network sizes $m = 1, 3, 5, 7, 9$)

(11.53) and (11.70) into (11.51) results in

$$\Pr[I_{\text{rep}} < R] \sim \left[ \frac{2^{mR} - 1}{\text{SNR}} \right]^m \times \prod_{D(s)} \lambda_{s,d(s)} \times \prod_{r \in D(s)} \lambda_{r,d(s)} \prod_{r \notin D(s)} \lambda_{s,r} \times \frac{1}{(|D(s)| + 1)!} \quad (11.71)$$

Figure 11.11 compares the results of numeric integration of the actual outage probability to computing the approximation (11.71), for an increasing number of terminals with $\lambda_{i,j} = 1$. As result (11.71) and Figure 11.11 indicate, repetition decode-and-forward cooperative diversity achieves full spatial diversity of order $m$, the number of cooperating terminals, for sufficiently large SNR. However, the SNR loss due to bandwidth inefficiency is exponential in $m$.

In order further to simplify the summation in (11.71), we note that for a given decoding set $D(s)$,

either $r \in D(s)$, in which case $\lambda_{r,d(s)}$ appears in the corresponding term in (11.71), or $r \notin D(s)$, in which case $\lambda_{s,r}$ appears in the corresponding term in (11.71). We, therefore, define

$$\lambda_r = \min \{ \lambda_{r,d(s)}, \lambda_{s,r} \}, \quad \bar{\lambda}_r = \max \{ \lambda_{r,d(s)}, \lambda_{s,r} \} \quad (11.72)$$

and $\bar{\lambda}_s = \lambda_{s,d(s)}$. Then the product dependent upon $\{ \lambda_{i,j} \}$ is bounded by

$$\bar{\lambda}^m \leq \lambda_{r,d(s)} \prod_{r \in D(s)} \lambda_{r,d(s)} \prod_{r \notin D(s)} \lambda_{s,r} \leq \bar{\lambda}^m \quad (11.73)$$

where $\bar{\lambda}$ is the geometric mean of the $\lambda_r$ and $\bar{\lambda}$ is the geometric mean of the $\bar{\lambda}_i$, for $i \in M$. One can see that the upper and lower bounds in (11.73) are independent of $D(s)$. We also note that the bounds in (11.73) coincide, i.e., $\bar{\lambda} = \bar{\lambda}_i$ if, though not only if, $\bar{\lambda}_i = \bar{\lambda}_j$ for all $i \in M$. Viewing $\lambda_{i,j}$ as a measure of distance between terminals $i$ and $j$, the class of planar network geometries that satisfy this condition are those in which all the relays lie with arbitrary spacing along the perpendicular bisector between the source and destination.

Substituting (11.73) into (11.71), gives the following simplified asymptotic bounds for outage probability:

$$\Pr[I_{\text{rep}} < R] \geq \left[ \frac{2^{mR} - 1}{\text{SNR} / \bar{\lambda}} \right]^m \sum_{D(s)} \frac{1}{(|D(s)| + 1)!} \quad (11.74)$$

$$\Pr[I_{\text{rep}} < R] \leq \left[ \frac{2^{mR} - 1}{\text{SNR} / \bar{\lambda}} \right]^m \sum_{D(s)} \frac{1}{(|D(s)| + 1)!} \quad (11.75)$$
11.4.3 Cooperative diversity using space–time coding

In this case (11.51) becomes

$$\Pr[I_{stc} < R] = \sum_{D(s)} \Pr[D(s)] \Pr[I_{stc} < R | D(s)]$$  \hspace{1cm} (11.76)

Conditional on $D(s)$ being the decoding set, the mutual information between $s$ and $d(s)$ for random codebook-generated i.i.d. circularly symmetric, complex Gaussian at the source and all potential relays can be shown to be

$$I_{stc} = \frac{1}{2} \log \left( 1 + \frac{2}{m} \text{SNR} |a_{s,d(s)}|^2 \right) + \frac{1}{2} \log \left( 1 + \frac{2}{m} \text{SNR} \sum_{r \in D(s)} |a_{r,d(s)}|^2 \right)$$  \hspace{1cm} (11.77)

Equation (11.77) represents the sum of the mutual information for two ‘parallel’ channels, one from the source to the destination, and one from the set of decoding relays to the destination. Again, $\Pr[I_{stc} < R | D(s)]$ involves $|D(s)| + 1$ independent fading coefficients, so we expect it to decay asymptotically in proportion to $1/\text{SNR}^{(|D(s)|+1)}$. In the sequel we develop the following high-SNR approximation

$$\Pr[I_{stc} < R | D(s)] \sim \left[ \frac{2^R - 1}{2 \text{SNR}/m} \right]^{|D(s)|+1} \times \prod_{r \in D(s)} \lambda_{r,d(s)} \times A_{|D(s)|}(2^R - 1)$$  \hspace{1cm} (11.78)

where

$$A_n(t) = \frac{1}{(n-1)!} \int_0^1 \frac{w^{n-1}(1-w)}{(1+tw)} \text{d}w, \hspace{1cm} n > 0$$  \hspace{1cm} (11.79)

and $A_0(t) = 1$. Equation (11.78) is expressed in such a way that the first term captures the dependence upon SNR and the second term captures the dependence upon $\{\lambda_i, j\}$.

To prove Equation (11.78) we first notice that in (11.77), $I_{stc}$ is of the form

$$I_{stc} = \frac{1}{2} \log \left( 1 + \frac{2}{m} \text{SNR}u_m \right) + \frac{1}{2} \log \left( 1 + \frac{2}{m} \text{SNR} \sum_{k=1}^{m-1} u_k \right)$$  \hspace{1cm} (11.80)

where again $u_k$ are independent exponential random variables with parameters $\lambda_k$, $k = 1, 2, \ldots, m$. If we use notation:

$$s_{m-1} = \sum_{k=1}^{m-1} u_k, t = (2^R - 1), \text{ and } \epsilon = (2^R - 1)/(2 \text{SNR}/m)$$

then

$$\Pr[I_{stc} < R | D(s)] = \Pr \left[ u_m + s_{m-1} + \frac{2}{m} \text{SNR}u_m s_{m-1} < \epsilon \right]$$

$$= \int_0^\epsilon \Pr \left[ u_m < \frac{\epsilon - s}{1 + (2 \text{SNR}/m)s} \right] p_{s_{m-1}}(s) ds = \epsilon \int_0^1 \Pr \left[ u_m < \frac{\epsilon(1-w)}{1+tw} \right] p_{u_m}(\epsilon w) dw$$

$$= \epsilon \int_0^1 \left[ 1 - \exp \left( -\lambda_m \frac{\epsilon(1-w)}{1+tw} \right) \right] p_{u_m}(\epsilon w) dw$$  \hspace{1cm} (11.81)

Equation (11.81) follows from the change of variables $w = s/\epsilon$, and the last equality follows from substituting the CDF for $u_m$. We now compute

$$\lim_{\epsilon \to 0} \frac{1}{\epsilon^m} \Pr[I_{stc} < R | D(s)] = \frac{1}{(m-2)!} \prod_{k=1}^{m} \lambda_m \int_0^1 \left[ \frac{1-w}{1+tw} \right] u_m^{(m-2)} dw$$  \hspace{1cm} (11.82)
which, provides the large-SNR approximation

$$\Pr[I_{stc} < R | D(s)] \sim \left[ \frac{2^R - 1}{2\text{SNR}/m} \right]^m \frac{1}{(m-2)!} \prod_{k=1}^m \lambda_m \times \int_0^1 \left[ \frac{1 - w}{1 + (2^R - 1)w} \right] w^{(m-2)} \, dw$$

(11.83)

To lower-bound the \(\lim \inf\), we use Fatou’s lemma in Equation (11.81) to obtain:

$$\liminf_{\varepsilon \to 0} \frac{1}{\varepsilon^m} \Pr[I_{stc} < R | D(s)] \geq \int_0^1 \left\{ \liminf_{\varepsilon \to 0} \frac{1}{\varepsilon} \left[ 1 - \exp \left( -\lambda_m \frac{\varepsilon(1-w)}{1+t} \right) \right] \right\} \lambda_m \left[ \frac{1 - w}{1 + tw} \right] A_{m-1}(w) \, dw$$

$$\geq \frac{1}{(m-2)!} \prod_{k=1}^m \lambda_k \int_0^1 \left[ \frac{1 - w}{1 + tw} \right] w^{(m-2)} \, dw$$

(11.84)

where the first equality follows from properties of exponentials and substitution of \(A_{m-1}(w)\) from Equation (11.58), and the second equality follows from the result (11.61) in the proof of Claim 1. To upper-bound the \(\lim sup\), we derive

$$\limsup_{\varepsilon \to 0} \frac{1}{\varepsilon^m} \Pr[I_{stc} < R | D(s)] \leq \limsup_{\varepsilon \to 0} \int_0^1 \left\{ \frac{1}{\varepsilon} \left[ 1 - \exp \left( -\lambda_m \frac{\varepsilon(1-w)}{1+t} \right) \right] \right\} \lambda_m \left[ \frac{1 - w}{1 + tw} \right] B_{m-1}(w, \varepsilon) \, dw$$

$$\leq \limsup_{\varepsilon \to 0} \int_0^1 \left\{ \lambda_m \left[ \frac{1 - w}{1 + tw} \right] \right\} \left\{ \frac{1}{\varepsilon^{m-2}} \right\} \frac{w^{(m-2)}}{(m-2)!} \prod_{k=1}^{m-1} \lambda_k \, dw$$

$$= \frac{1}{(m-2)!} \prod_{k=1}^{m-1} \lambda_k \int_0^1 \left[ \frac{1 - w}{1 + tw} \right] w^{(m-2)} \, dw$$

(11.85)

where the first inequality follows from substitution of \(B_{m-1}(w, \varepsilon)\) from (11.63), the second inequality follows from the fact that \(1 - \exp(-x) \leq x\) for all \(x \geq 0\), and the third inequality follows from the result (11.64) in Claim 1. Taken together with the fact that \(\lim inf < \lim sup\), Equation (11.84) and (11.85) yield the desired result, Equation (11.82).

The remaining term in Equation (11.76) to be considered is \(\Pr[D(s)]\), the probability of a particular decoding set. As before, we require a potential relay to decode the source message fully in order to participate in the second phase, a necessary condition for the mutual information expression (11.77) to be correct. Since the realized mutual information between \(s\) and \(r\) for i.i.d. complex Gaussian codebooks is given by

$$\frac{1}{2} \log \left( 1 + \frac{2}{m}\text{SNR} |a_{s,r}|^2 \right)$$

under this condition we have

$$\Pr[r \in D(s)] = \Pr \left[ |a_{s,r}|^2 > \frac{2^R - 1}{2\text{SNR}/m} \right] = \exp \left[ -\lambda_{s,r} \frac{2^R - 1}{2\text{SNR}/m} \right]$$

(11.87)

Now, since each potential relay makes its decision independently, and the fading coefficients are independent, we have

$$\Pr[D(s)] = \prod_{r \in D(s)} \exp \left[ -\lambda_{s,r} \frac{2^R - 1}{2\text{SNR}/m} \right] \times \prod_{r \notin D(s)} \left( 1 - \exp \left[ -\lambda_{s,r} \frac{2^R - 1}{2\text{SNR}/m} \right] \right)$$

$$\sim \left[ \frac{2^R - 1}{2\text{SNR}/m} \right]^{m-|D(s)|-1} \times \prod_{r \in D(s)} \lambda_{s,r}.$$  

(11.88)
Figure 11.12 Outage probability of space–time-coded cooperative diversity.

Combining (11.78) and (11.88) into (11.76), we obtain

$$\Pr [I_{stc} < R] \sim \left[ \frac{2^{2R} - 1}{2SNR/m} \right]^m \times \sum_{D(s)} \lambda_{s,d(s)} \times \prod_{r \in D(s)} \lambda_{r,d(s)} \prod_{r \notin D(s)} \lambda_{s,r} \times A_{|D(s)|} \left(2^{2R} - 1 \right)$$  \hspace{1cm} (11.89)

Figure 11.12 presents the results of numeric integration of the actual outage probability and the approximation (11.89), for an increasing number of terminals with $\lambda_i, j = 1$. As the result (11.89) and Figure 11.12 indicate, space–time-coded cooperative diversity achieves full spatial diversity of order $m$, the number of cooperating terminals, for sufficiently large SNR. In contrast to repetition-based algorithms, the SNR loss for space–time-coded cooperative diversity is only linear in $m$.

As in Equations (11.74) and (11.75), we want further to simplify the summation in (11.89). The product dependent upon $\{\lambda_{i,j}\}$ can again be bounded as in Equation (11.73). To avoid dealing with (11.79), we exploit the bounds

$$\frac{1}{(n + 1)! (1 + t)} \leq A_n (t) \leq \frac{1}{n!}$$ \hspace{1cm} (11.90)

Combining Equations (11.73) and (11.90) into (11.89), we get the following simplified asymptotic bounds for outage probability:

$$\Pr [I_{stc} < R] \geq \left[ \frac{2^{2R} - 1}{2SNR/(m\lambda_i)} \right]^m \sum_{D(s)} \frac{1}{(|D(s)| + 1)!}$$ \hspace{1cm} (11.91)

$$\Pr [I_{stc} < R] \leq \left[ \frac{2^{2R} - 1}{2SNR/(m\lambda_i)} \right]^m \sum_{D(s)} \frac{1}{|D(s)|!}$$ \hspace{1cm} (11.92)

**APPENDIX 11.1 ASYMPTOTIC CDF APPROXIMATIONS**

All results from Section 11.2 are of the form

$$\lim_{t \to a} \frac{P_{u(t)}(g_1(t))}{g_2(t)} = c$$ \hspace{1cm} (A11.1.1)
where \( t \) is a parameter of interest; \( P_{u(t)}(g_1(t)) \) is the CDF of a certain random variable \( u(t) \) that can, in general, depend upon \( t; g_1(t) \) and \( g_2(t) \) are two (continuous) functions; and \( t_0 \) and \( \epsilon \) are constants. (A11.1.1) implies the approximation \( P_{u(t)}(g_1(t)) \sim c g_2(t) \) for \( t \) close to \( t_0 \).

**Result 1:** Let \( u \) be an exponential random variable with parameter \( \lambda_u \). Then, for a function \( g(t) \) continuous about \( t = t_0 \) and satisfying \( g(t) \to 0 \) as \( t \to t_0 \)

\[
\lim_{t \to t_0} \frac{1}{g(t)} P_u(g(t)) = \lambda_u
\]

(A11.1.2)

**Result 2:** Let \( w = u + v \), where \( u \) and \( v \) are independent exponential random variables with parameters \( \lambda_u \) and \( \lambda_v \), respectively. Then the CDF

\[
P_w(w) = \begin{cases} 
1 - \left( \frac{\lambda_v}{\lambda_u - \lambda_v} \right) e^{-\lambda_u w} + \left( \frac{\lambda_u}{\lambda_u - \lambda_v} \right) e^{-\lambda_v w}, & \lambda_u \neq \lambda_v \\
1 - (1 + \lambda w) e^{-\lambda}, & \lambda_u = \lambda_v = \lambda
\end{cases}
\]

(A11.1.3)

satisfies

\[
\lim_{\epsilon \to 0} \frac{1}{\epsilon^2} P_w(\epsilon) = \frac{\lambda_u \lambda_v}{2},
\]

(A11.1.4)

and

\[
\lim_{t \to t_0} \frac{1}{g^2(t)} P_u(g(t)) = \frac{\lambda_u \lambda_v}{2}
\]

(A11.1.5)

if a function \( g(t) \) is continuous about \( t = t_0 \) and satisfies \( g(t) \to 0 \) as \( t \to t_0 \).

**Equality 1:** Let \( u, v, \) and \( w \) be independent exponential random variables with parameters \( \lambda_u, \lambda_v, \) and \( \lambda_w \), respectively. Let \( \epsilon \) be positive, and let \( g(\epsilon) > 0 \) be continuous with \( g(\epsilon) \to 0 \) and \( \epsilon / g(\epsilon) \to c \) \(< \infty \) as \( \epsilon \to 0 \). Then for \( f(x, y) = (xy)/(x + y + 1) \)

\[
\lim_{t \to t_0} \frac{1}{g^2(t)} \Pr[u + \epsilon f(v/\epsilon, w/\epsilon) < g(\epsilon)] = \frac{\lambda_u (\lambda_v + \lambda_w)}{2}
\]

(A11.1.6)

and

\[
\lim_{t \to t_0} \frac{1}{g^2(h(t))} \Pr[u + h(t) f(v/h(t), w/h(t)) < g(h(t))] = \frac{\lambda_u (\lambda_v + \lambda_w)}{2}
\]

(A11.1.7)

if \( h(t) \) is continuous about \( t = t_0 \) and satisfies \( h(t) \to 0 \) as \( t \to t_0 \).

The following lemma will be useful in the proof of **Equality 1**.

**Lemma 1:** For the following set of parameters: positive \( \delta; r_\delta := \delta f(v/\delta, w/\delta) \), where \( v \) and \( w \) are independent exponential random variables with parameters \( \lambda_v \) and \( \lambda_w \), respectively; continuous \( h(\delta) > 0 \) with \( h(\delta) \to 0 \) and \( \delta / h(\delta) \to d < \infty \) as \( \delta \to 0 \); probability \( \Pr[r_\delta < h(\delta)] \) satisfies

\[
\lim_{\delta \to 0} \frac{1}{h(\delta)} \Pr[r_\delta < h(\delta)] = \lambda_v + \lambda_w.
\]

(A11.1.8)

**Proof of Lemma 1:** First we look at the lower bound

\[
\Pr[r_\delta < h(\delta)] = \Pr[1/v + 1/w + \delta/(vw) > 1/h(\delta)] \\
> \Pr[1/v + 1/w > 1/h(\delta)] \geq \Pr[\max(1/v, 1/w) > 1/h(\delta)]
\]

(A11.1.9)

\[
= 1 - \Pr[v \geq h(\delta)] \Pr[w \geq h(\delta)] = 1 - \exp[-(\lambda_v + \lambda_w) h(\delta)]
\]

so, by using **Result 1** we have

\[
\lim_{\delta \to 0} \frac{1}{h(\delta)} \Pr[r_\delta < h(\delta)] \geq \lambda_v + \lambda_w
\]

(A11.1.10)
To prove the other direction, let $l > 1$ be a fixed constant:

$$\Pr[r_s < h(\delta)] = \Pr[1/v + 1/w + \delta/(vw) > 1/h(\delta)]$$

$$= \int_0^\infty \Pr \left[ \frac{1}{v} > \frac{1/h(\delta) - 1/w}{1 + \delta/w} \right] p_w(w)dw$$

(A11.1.11)

$$\leq \Pr[w < lh(\delta)] + \int_{lh(\delta)}^\infty \Pr \left[ \frac{1}{v} > \frac{1/h(\delta) - 1/w}{1 + \delta/w} \right] p_w(w)dw$$

But

$$\Pr[w < lh(\delta)]/h(\delta) \leq \lambda_w l$$

(A11.1.12)

which takes care of the first term of Equation (A11.1.11). To bound the second term of (A11.1.11), let $k > l$ be another fixed constant, and note that

$$\int_{lh(\delta)}^\infty \Pr \left[ \frac{1}{v} > \frac{1/h(\delta) - 1/w}{1 + \delta/w} \right] p_w(w)dw =$$

$$= \int_{lh(\delta)}^\infty \Pr \left[ \frac{1}{v} > \frac{1/h(\delta) - 1/w}{1 + \delta/w} \right] p_w(w)dw + \int_{lh(\delta)}^{kh(\delta)} \Pr \left[ \frac{1}{v} > \frac{1/h(\delta) - 1/w}{1 + \delta/w} \right] p_w(w)dw$$

$$\leq \Pr \left[ \frac{1}{v} > \frac{1 - 1/k}{h(\delta) + \delta/k} \right] + \lambda_w \int_{lh(\delta)}^{kh(\delta)} \Pr \left[ \frac{1}{v} > \frac{1/h(\delta) - 1/w}{1 + \delta/w} \right] p_w(w)dw$$

where the first term in the bound of Equation (A11.1.13) follows from the fact that $\Pr[1/v > (1/h(\delta) - 1/w)/(1 + \delta/w)]$ is nonincreasing in $w$, and the second term in the bound of (A11.1.13) follows from the fact that $p_w(w) = \lambda_w \exp(-\lambda_w w) \leq \lambda_w$. Now, the first term of (A11.1.13) satisfies

$$\Pr \left[ \frac{1}{v} > \frac{1 - 1/k}{h(\delta) + \delta/k} \right] / h(\delta) \leq \frac{1 + \delta/(kh(\delta))}{1 - 1/k}$$

(A11.1.14)

and, by a change of variable $w' = w/h(\delta)$, the second term of (A11.1.13) satisfies

$$\frac{1}{h(\delta)} \int_{lh(\delta)}^{kh(\delta)} \Pr \left[ \frac{1}{v} > \frac{1/h(\delta) - 1/w}{1 + \delta/w} \right] dw =$$

$$= h(\delta) \int_1^{\ell} \Pr \left[ \frac{1}{v} > \frac{1/h(\delta) - 1/w}{1 + \delta/w} \right] dw'$$

$$\leq h(\delta) \int_1^{\ell} \lambda_w \left( \frac{1 + \delta/(w'h(\delta))}{1 - 1/w'} \right) dw'$$

(A11.1.15)

where $B(\delta, h(\delta), k, l)$ remains finite for any $k > l > 1$ as $\delta \to 0$.

Combining Equation (A11.1.12), (A11.1.14), and (A11.1.15), gives

$$\frac{1}{h(\delta)} \Pr[r_s < h(\delta)] \leq \lambda_w l + \lambda_v \left( \frac{1 + \delta/(kh(\delta))}{1 - 1/k} \right) + h(\delta)B(\delta, h(\delta), k, l)$$

(A11.1.16)

and

$$\limsup_{\delta \to 0} \frac{1}{h(\delta)} \Pr[r_s < h(\delta)] \leq \lambda_w l + \lambda_v \left( \frac{1 + d/k}{1 - 1/k} \right)$$

since $\lim_{\delta \to 0} B(\delta, h(\delta), k, l) < \infty$ and, by assumption, $h(\delta) \to 0$ and $\delta/h(\delta) \to d$ as $\delta \to 0$. The constants $k > l > 1$ are arbitrary. In particular, $k$ can be chosen arbitrarily large, and $l$ arbitrarily close to 1. Hence,

$$\limsup_{\delta \to 0} \frac{1}{h(\delta)} \Pr[r_s < h(\delta)] \leq \lambda_w + \lambda_v$$

(A11.1.17)

Combining Equation (A11.1.10) with (A11.1.17), the lemma is proved.
Proof of Equality 1:

\[
\Pr[u + \varepsilon f(v/\varepsilon, w/\varepsilon) < g(\varepsilon)] = \Pr[u + r_\varepsilon < g(\varepsilon)] = \int_0^{g(\varepsilon)} \Pr[r_\varepsilon < g(\varepsilon) - u]p_u(u)\,du
\]

\[
= g(\varepsilon) \int_0^{g(\varepsilon)} \Pr[r_\varepsilon < g(\varepsilon)(1 - u')]\lambda_u e^{-\lambda_u g(\varepsilon)u'}\,du'
\]

\[
= g^2(\varepsilon) \int_0^1 (1 - u') \frac{\Pr[r_\varepsilon < g(\varepsilon)(1 - u')]}{g(\varepsilon)(1 - u')} \lambda_u e^{-\lambda_u g(\varepsilon)u'}\,du'
\]

where in the second equality we have used the change of variables \(u' = u/g(\varepsilon)\). But by Lemma 1 with \(\delta = \varepsilon\) and \(h(\delta) = g(\delta)(1 - u')\), the quantity in brackets approaches \(\lambda_u + \lambda_v\) as \(\varepsilon \to 0\), so we expect

\[
\lim_{\varepsilon \to 0} \frac{1}{g^2(\varepsilon)} \Pr[u + r_\varepsilon < g(\varepsilon)] = \lambda_u(\lambda_v + \lambda_w) \int_0^1 (1 - u')\,du' = \frac{\lambda_u(\lambda_v + \lambda_w)}{2} (A11.1.19)
\]

To verify Equation (A11.1.19), fully we must utilize the lower and upper bounds developed in Lemma 1. Using the lower bound (A11.1.10), (A11.1.18) satisfies

\[
\lim_{\varepsilon \to 0} \inf \frac{1}{g^2(\varepsilon)} \Pr[u + r_\varepsilon < g(\varepsilon)] \geq \lim_{\varepsilon \to 0} \int_0^1 \frac{1 - \exp[-(\lambda_v + \lambda_u)g(\varepsilon)(1 - u')]}{g(\varepsilon)} \lambda_u e^{-\lambda_u g(\varepsilon)u'}\,du' = \lambda_u(\lambda_v + \lambda_w) \int_0^1 (1 - u')\,du' = \frac{\lambda_u(\lambda_v + \lambda_w)}{2} (A11.1.20)
\]

where the first equality results from the Dominated Convergence theorem [29] after noting that the integrand is both bounded by and converges to the function \(\lambda_u(\lambda_v + \lambda_w)(1 - u')\). Using the upper bound (A11.1.17), Equation (A11.1.18) satisfies

\[
\lim_{\varepsilon \to 0} \sup \frac{1}{g^2(\varepsilon)} \Pr[u + r_\varepsilon < g(\varepsilon)] \leq \lim_{\varepsilon \to 0} \sup(\lambda_u/(1 - 1/k) + \lambda_w) \int_0^1 (1 - u')\lambda_u e^{-\lambda_u g(\varepsilon)u'}\,du' + \lim_{\varepsilon \to 0} \sup g(\varepsilon)D(\varepsilon, g(\varepsilon), k, l) (A11.21)
\]

where the last equality results from the fact \(\varepsilon/g(\varepsilon)\to c\) and the fact that \(D(\varepsilon, g(\varepsilon), k, l) := \int_0^1 (1 - u')^2 B(\varepsilon, g(\varepsilon)(1 - u'), k, l)\lambda_u e^{-\lambda_u g(\varepsilon)u'}\,du'\) remains finite for all \(k > l > 1\) even as \(\varepsilon \to 0\).

Again, the constants \(k > l > 1\) are arbitrary. In particular, \(k\) can be chosen arbitrarily large, and \(l\) arbitrarily close to 1. Hence,

\[
\lim_{\varepsilon \to 0} \sup \frac{1}{g^2(\varepsilon)} \Pr[u + r_\varepsilon < g(\varepsilon)] \leq \frac{\lambda_u(\lambda_v + \lambda_w)}{2} (A11.22)
\]

Combining Equation (A11.21) and (A11.22) completes the proof. \(\square\)

Equality 2: Let \(u\) and \(v\) be independent exponential random variables with parameters \(\lambda_u\) and \(\lambda_v\), respectively. Let \(\varepsilon\) be positive and let \(g(\varepsilon) > 0\) be continuous with \(g(\varepsilon) \to 0\) as \(\varepsilon \to 0\). Then for

\[
h(\varepsilon) := \varepsilon^2[(g(\varepsilon)/\varepsilon + 1) \ln(g(\varepsilon)/\varepsilon + 1) - g(\varepsilon)/\varepsilon] (A11.1.23)
\]

we have

\[
\lim_{\varepsilon \to 0} \frac{1}{h(\varepsilon)} \Pr[u + v + u\varepsilon/\varepsilon < g(\varepsilon)] = \lambda_u\lambda_v (A11.1.24)
\]

and

\[
\lim_{\varepsilon \to 0} \frac{1}{h(\varepsilon(t))} \Pr[u + v + u\varepsilon(t)/\varepsilon(t) < g(\varepsilon(t))] = \lambda_u\lambda_v (A11.1.25)
\]

if \(\varepsilon(t)\) is continuous about \(t = t_0\) with \(\varepsilon(t) \to 0\) as \(t \to t_0\).
Proof: First, we write CDF in the form

\[ \Pr[u + v + uv/\varepsilon < g(\varepsilon)] = \int_{0}^{\infty} \Pr[u + v + uv/\varepsilon < g(\varepsilon) | v = v] p_v(v) \, dv \]

\[ = \int_{0}^{g(\varepsilon)} \Pr \left[ u < \frac{g(\varepsilon) - v}{1 + v/\varepsilon} \right] \lambda_v e^{-\lambda_v v} \, dv \]

where the last equality follows from the change of variables \( w = v/g(\varepsilon) \). To upper-bound Equation (A11.1.26), we use the identities \( 1 - e^{-x} \leq x \) for all \( x \geq 0 \) and \( e^{-x} \leq 1 \) for all \( y \geq 0 \), so that (A11.1.26) becomes

\[ \Pr[u + v + uv/\varepsilon < g(\varepsilon)] \leq g^2(\varepsilon) \lambda_u \lambda_v \int_{0}^{1} \frac{1 - w}{1 + g(\varepsilon)w/\varepsilon} \, dw \]

\[ = \lambda_u \lambda_v g^2(\varepsilon) \frac{(g(\varepsilon)/\varepsilon + 1) \ln(g(\varepsilon)/\varepsilon + 1) - g(\varepsilon)/\varepsilon}{(g(\varepsilon)/\varepsilon)^2} = \lambda_u \lambda_v h(\varepsilon) \]

and

\[ \limsup_{\varepsilon \to 0} \frac{1}{h(\varepsilon)} \Pr[u + v + uv/\varepsilon < g(\varepsilon)] \leq \lambda_u \lambda_v. \] (A11.1.27)

To lower-bound Equation (A11.1.26), we use the concavity of \( 1 - e^{-x} \), i.e. for any \( t > 0 \), \( 1 - e^{-t} \geq ((1 - e^{-t})/t)x \), for all \( x \leq t \) and the identity \( e^{-y} \geq 1 - y \) for all \( y \geq 0 \), so that Equation (A11.26) becomes

\[ \Pr[u + v + uv/\varepsilon < g(\varepsilon)] \]

\[ \geq g(\varepsilon) \int_{0}^{1} \left[ \frac{1 - e^{-\lambda_u g(\varepsilon)}}{\lambda_u g(\varepsilon)} \right] \frac{\lambda_u g(\varepsilon)(1 - w)}{1 + wg(\varepsilon)/\varepsilon} \lambda_v (1 - \lambda_v g(\varepsilon)w) \, dw \]

\[ = \lambda_u \lambda_v g^2(\varepsilon) \left( \frac{1 - e^{-\lambda_u g(\varepsilon)}}{\lambda_u g(\varepsilon)} \right) \int_{0}^{1} \left[ \frac{1 - w}{1 + wg(\varepsilon)/\varepsilon} \right] (1 - \lambda_v g(\varepsilon)w) \, dw \]

\[ \geq \lambda_u \lambda_v g^2(\varepsilon) \left( \frac{1 - e^{-\lambda_u g(\varepsilon)}}{\lambda_u g(\varepsilon)} \right) (1 - \lambda_v g(\varepsilon)) \times \int_{0}^{1} \frac{1 - w}{1 + wg(\varepsilon)/\varepsilon} \, dw \]

\[ = \lambda_u \lambda_v g^2(\varepsilon) \left( \frac{1 - e^{-\lambda_u g(\varepsilon)}}{\lambda_u g(\varepsilon)} \right) (1 - \lambda_v g(\varepsilon)) \times \frac{(g(\varepsilon)/\varepsilon + 1) \ln(g(\varepsilon)/\varepsilon + 1) - g(\varepsilon)/\varepsilon}{(g(\varepsilon)/\varepsilon)^2} \]

\[ = \lambda_u \lambda_v \left( \frac{1 - e^{-\lambda_u g(\varepsilon)}}{\lambda_u g(\varepsilon)} \right) (1 - \lambda_v g(\varepsilon)) h(\varepsilon) \]

and

\[ \liminf_{\varepsilon \to 0} \frac{1}{h(\varepsilon)} \Pr[u + v + uv/\varepsilon < g(\varepsilon)] \]

\[ \geq \lambda_u \lambda_v \lim_{\varepsilon \to 0} \left( \frac{1 - e^{-\lambda_u g(\varepsilon)}}{\lambda_u g(\varepsilon)} \right) (1 - \lambda_v g(\varepsilon)) = \lambda_u \lambda_v \] (A11.1.28)

Since the bounds in Equations (A11.1.27) and (A11.1.28) are equal, Equality 2 is proved.
Equality 3: For \( f_t(s) \to g(s) \) pointwise as \( t \to t_0 \), and \( f_t(s) \) monotone increasing in \( s \) for each \( t \) let \( h_t(s) \) be such that \( h_t(s) \leq s \), \( h_t(s) \to s \) pointwise as \( t \to t_0 \), and \( h_t(s) \) is monotone decreasing in \( s \) for each \( t \). For \( \tilde{h}_t^{-1}(r) := \min h_t^{-1}(r) \) we have

\[
\lim_{t \to t_0} f_t(\tilde{h}_t^{-1}(r)) = g(r). \tag{A11.1.29}
\]

Proof: Since \( h_t(s) \leq s \) for all \( t \), we have \( r \leq \tilde{h}_t^{-1}(r) \), and consequently \( f_t(r) \leq f_t(\tilde{h}_t^{-1}(r)) \) because \( f_t(\cdot) \) is monotone increasing. Thus,

\[
\liminf_{t \to t_0} f_t(\tilde{h}_t^{-1}(r)) \geq g(r). \tag{A11.1.30}
\]

For upper bound fix \( \delta > 0 \). Lemma 2 shows that for each \( r \) there exists \( t^* \) such that \( \tilde{h}_t^{-1}(r) \leq r/(1 - \delta) \) for all \( t \) such that \( |t - t_0| < |t^* - t_0| \). So

\[
f_t(\tilde{h}_t^{-1}(r)) \leq f_t(r/(1 - \delta)) \quad \text{and} \quad \limsup_{t \to t_0} f_t(\tilde{h}_t^{-1}(r)) \leq g(r/(1 - \delta)).
\]

Since \( \delta \) can be made arbitrarily small we have

\[
\limsup_{t \to t_0} f_t(\tilde{h}_t^{-1}(r)) \leq g(r). \tag{A11.1.31}
\]

Combining (A11.1.30) with (A11.1.31), we obtain the desired result.

The following Lemma is used in the proof of the upper bound of Equality 3.

Lemma 2: For \( h_t(s) \) such that \( h_t(s) \leq s \), \( h_t(s) \to s \) pointwise as \( t \to t_0 \), and \( h_t(s) \) is monotone decreasing in \( s \) for each \( t \), define \( \tilde{h}_t^{-1}(r) := \min h_t^{-1}(r) \). For each \( r_0 > 0 \) and any \( \delta > 0 \), there exists \( t^* \) such that \( \tilde{h}_t^{-1}(r) \leq r_0/(1 - \delta) \) for all \( t \) such that \( |t - t_0| < |t^* - t_0| \).

Proof: Fix \( r_0 > 0 \) and \( \delta > 0 \), and select \( s_0 \) such that \( s_0 > r_0/(l - \delta) \).

Because \( h_t(s)/s \to 1 \) point-wise as \( t \to t_0 \), for each \( s > 0 \) and any \( \delta > 0 \), there exists a \( t^* \) such that \( h_t(s) > s(1 - \delta) \), all \( t : |t - t_0| < |t^* - t_0| \).

Also, since \( h_t(s)/s \) is monotone decreasing in \( s \), if \( t^* \) is sufficient for convergence at \( s_0 \), then it is sufficient for convergence at all \( s \leq s_0 \). Thus, for any, \( s_0 > 0 \) and \( \delta > 0 \) there exists a \( t^* \) such that \( h_t(s) \geq s(1 - \delta) \), all \( s \leq s_0 \), \( t : |t - t_0| < |t^* - t_0| \).

In the rest of the proof, we only consider \( s \leq s_0 \) and \( t \) such that \( |t - t_0| < |t^* - t_0| \). Consider the interval \( I = [r_0, r_0/(l - \delta)] \), and note that \( s \in I \) implies \( s < s_0 \). Since \( h_t(s) < s \), we have \( h_t(r_0) < r_0 \).

Also, since \( h_t(s) > s(1 - \delta) \) by the above construction, we have \( h_t(r_0/(l - \delta)) > r_0 \). By continuity, \( h_t(s) \) assumes all intermediate values between \( h_t(r_0) \) and \( h_t(r_0/(l - \delta)) \) on the interval \( (r_0, r_0/(l - \delta)) \) [30, Theorem 4.23]; in particular, there exists an \( s_1 \in (r_0, r_0/(l - \delta)) \) such that \( h_t(x_1) = r_0 \). The result follows from \( \tilde{h}_t^{-1}(r) \leq x \leq r_0/(1 - \delta) \), where the first inequality follows from the definition of \( \tilde{h}_t^{-1}(\cdot) \) and the second inequality follows from the fact that \( x \in I \).

APPENDIX 11.2 AMPLIFY-AND-FORWARD MUTUAL INFORMATION

We write the equivalent channel (11.5–6), with relay processing, in vector form as

\[
y_d[n] = h x_t[n] + B[n]
\]
where

\[
y_d[n] = \begin{bmatrix} y_d[n] \\ y_d[n + N/4] \end{bmatrix}; \quad h = \begin{bmatrix} \frac{h_{s,d}}{\sqrt{P_r}} \\
\frac{h_{r,d}}{\sqrt{P_s}} h_{s,r} \end{bmatrix}
\]

\[
B = \begin{bmatrix} 0 & 1 & 0 \\
\frac{h_{r,d}}{\sqrt{P_s}} & 0 & 1 \end{bmatrix}; \quad n[n] = \begin{bmatrix} n_s[n] \\
n_d[n] \\
n_d[n + N/4] \end{bmatrix}
\]

The source signal has power constraint \(E[x_s] \leq P_s\), and relay amplifier has constraint

\[
\beta \leq \sqrt{\frac{P_r}{|h_{s,r}|^2 P_s + N_r}} \tag{A11.2.1}
\]

and the noise has covariance \(E[nn^T] = \text{diag}(N_r, N_d, N_d)\). Since the channel is memoryless, the average mutual information satisfies

\[
I_{AF} \leq I(x_s; y_d) \leq \log \det(I + (P_s h h^T)(B E[nn^T]B^T)^{-1})
\]

with equality for \(x_s\), zero-mean, circularly symmetric complex Gaussian. Noting that

\[
h h^T = \begin{bmatrix} |h_{s,d}|^2 & h_{s,d}^2(h_{r,d} \beta h_{s,r})' \\
|h_{r,d}^2 h_{r,d} \beta h_{s,r}| & |h_{r,d} \beta h_{s,r}|^2 \end{bmatrix}
\]

\[
B E[nn^T]B^T = \begin{bmatrix} N_d & 0 \\
0 & |h_{r,d} \beta|^2 N_r + N_d \end{bmatrix}
\]

we have

\[
\det(I_2 + (P_s h h^T)(B E[nn^T]B^T)^{-1}) = 1 + \frac{P_s |h_{s,d}|^2}{N_d} + \frac{P_s |h_{r,d} \beta h_{s,r}|^2}{(|h_{r,d} \beta|^2 N_r + N_d)} \tag{A11.2.2}
\]

Because (A11.2.2) is increasing in \(\beta\), the amplifier power constraint (A11.2.1) should apply, yielding,

\[
I_{AF} = \log(1 + |h_{s,d}|^2 y_{s,d} + f(\frac{|h_{s,r}|^2}{y_{s,r}}; |h_{r,d}|^2 y_{r,d})
\]

where \(y_{a,b} = SNR_{a,b}\) and is \(f(\cdot, \cdot)\) given in Section 11.2.3

**APPENDIX 11.3 INPUT DISTRIBUTIONS FOR TRANSMIT DIVERSITY BOUND**

An equivalent channel model for the two-antenna case can be summarized as

\[
y[n] = \begin{bmatrix} h_1 & h_2 \\
h \end{bmatrix} \begin{bmatrix} x_1[n] \\
x_2[n] \end{bmatrix} + n[n] \tag{A11.3.1}
\]

where \(h\) represents the fading coefficients and \(x[n]\) the transmit signals from the two transmit antennas, and \(n[n]\) is a zero-mean, white complex Gaussian process with variance \(N_0\) that captures the effects of noise and interference. If \(Q = E[xx^T]\) is the covariance matrix for the transmit signals, then the power constraint on the inputs may be written in the form \(\text{tr}(Q) \leq P\).
We want to find a distribution on the input vector $\mathbf{x}$, subject to the power constraint, that minimizes outage probability, i.e.,

$$\min_{p_X \text{ s.t. } \text{tr}(Q) \leq P} \Pr[I(\mathbf{x}; y | h = a) < R].$$  \hfill (A11.3.2)

The optimization (A11.3.2) can be restricted to optimization over zero-mean, circularly symmetric complex Gaussian inputs, because Gaussian codebooks maximize the mutual information for each value of the fading coefficients $a$, i.e.

$$\min_{Q \text{ s.t. } \text{tr}(Q) \leq P} \Pr \left[ \log \left( 1 + \frac{\mathbf{h}Q\mathbf{h}^T}{N_0} \right) < R \right]$$  \hfill (A11.3.3)

We now argue that $Q$ diagonal is sufficient, even if the components of $a$ are independent but not identically distributed. We write $\mathbf{h} = \tilde{\mathbf{h}} \Sigma$, where $\tilde{\mathbf{h}}$ is a zero-mean, i.i.d. complex Gaussian vector with unit variances and $\Sigma = \text{diag}(\sigma_1, \sigma_2)$. Thus, the outage probability in (A11.3.3) may be written as:

$$\Pr \left[ \log \left( 1 + \frac{\tilde{\mathbf{h}}\Sigma Q\Sigma^T \tilde{\mathbf{h}}^T}{N_0} \right) < R \right].$$

Now consider an eigendecomposition of the matrix $\Sigma Q \Sigma^T = UD^T U^T$, where $U$ is unitary and $D$ is diagonal. Using the fact that the distribution of $\tilde{\mathbf{h}}$ is rotationally invariant, i.e., $\tilde{\mathbf{h}} U$ has the same distribution as $\tilde{\mathbf{h}}$ for any unitary $U$, we observe that the outage probability for covariance matrix $\Sigma Q \Sigma^T$ is the same as the outage probability for the diagonal matrix $D$. For $D = \text{diag}(d_1, d_2)$, the outage probability can be written in the form

$$\Pr \left[ d_1 |h_1|^2 + d_2 |h_2|^2 < \frac{2R - 1}{\text{SNR}} \right],$$

which, using Result 2, decays in proportion to $1/(\text{SNR}^2 \det D)$ for large SNR if $d_1, d_2 \neq 0$. Thus, minimizing the outage probability for large SNR is equivalent to maximizing

$$\det D = \det \Sigma Q \Sigma^T = \sigma_1^2 \sigma_2^2 (Q_{1,1} Q_{2,2} - |Q_{1,2}|^2)$$  \hfill (A11.3.4)

such that $Q_{1,1} + Q_{2,2} \leq P$. Clearly, (A11.3.4) is maximized for $Q_{1,1} = Q_{2,2} = P/2$ and $Q_{1,2} = Q_{2,1} = 0$. Thus, zero-mean, i.i.d. complex Gaussian inputs minimize the outage probability in the high-SNR regime.

REFERENCES


REFERENCES


12

Cognitive UWB Communications

12.1 INTRODUCTION

In this chapter we discuss cognitive ultra wide band (CUWB) systems, capable of monitoring and adapting to the amount and type of interference in the network, like advanced personal area networks (PAN). In this concept, cognition is used to identify presence and location on the spectral scale of the interfering signal, generated by other networks coexisting in the same band, and create a corresponding notch to suppress the interference. This enables frequency reuse for different networks and significant increase in overall spectrum efficiency. The scheme can be used to improve significantly performance of UWB systems, e.g. high speed Bluetooth, in the presence of interference from mobile communication systems such as GSM and WCDMA. It is also effective in the presence of WLAN systems, which are nowadays based on OFDMA technology (e.g., IEEE802.11, 16e, 20) or military communications where the interference is generated by intentional jamming. We also discuss the effectiveness of the scheme to suppress MC CDMA, which is a candidate technology for 4G mobile communications.

Recently WiMedia Alliance has announced that UWB technology has been accepted as the basic standard for high speed Bluetooth, see http://www.wimedia.org/en/events/index.asp?id=events. The basic principles of UWB technology have been already discussed in Chapter 8. In addition, the online source 0 gives historical perspective to ultra wide band (UWB) technologies. It lists the early UWB references and patents from the 60s and 70s. In 0, a comprehensive overview of ultra wide band wireless systems is given. It discusses the Federal Communications Commission (FCC) allocation of the 7.5 GHz (3.1–10.6 GHz) unlicensed band for UWB devices. Potential UWB modulation schemes, multiple access issues, single versus multiband implementation, and link budgets are also discussed. Paper 0 is a very frequently referred to as giving a brief introduction to the basics of impulse radio systems. It describes the characteristics of impulse radio and gives analytical estimates of the multi-access capability under idealistic channel conditions.

Channel capacity and channel models are studied in [1–8]. Performance of PPM and on–off keying (OOK) binary block-coded modulation formats using maximal ratio combining rake receiver are studied analytically in 0. The tradeoff between receiver complexity and performance is examined. Several suboptimal receivers in indoor multipath AWGN channels have been employed. Results indicate
that the robust performance may require an increase in rake complexity. This implies allocation of more rake fingers and tracking of the strongest multipaths to help in the selection combining. Rake performance for a pulse-based high data rate UWB system in an Intel Labs indoor channel model is addressed in 0. It is noted that at low input SNR values (0–10 dB) and a small number of rake fingers, it is more beneficial to add rake taps for energy capture rather than for intersymbol interference (ISI) mitigation. In the presence of channel estimation errors, equal gain combining can be more robust than maximal ratio combining, and therefore yield better performance. In order to quantify the tradeoff between rake receiver energy capture and diversity order 0 presents partly quasi-analytical and partly experimental analysis suited to dense multipath propagation environments. Numerical results show that a diversity level of less than 50 is adequate in typical indoor office conditions.

In 0, a method to evaluate the bit error rate (BER) performance of time hopping (TH) PPM in the presence of multiuser interference and AWGN channel is proposed. Gaussian quadrature rules are used in this approach. Paper 0 concentrates on the signal design for binary UWB communications in dense multipath channels. The aim is to find signals with good distance properties leading to nice BER performance that both depend on the time shift parameter $\tau$. Performance of UWB correlation receivers for equal mean power Gaussian monocycles is studied in 0. Channel conditions vary among ideal single-user AWGN, non-ideal synchronous, multipath fading and multiple access interference. It is shown that the pulse shape has a notable impact on the correlation receiver performance. The effects can be seen in the autocorrelation function, especially in the mainlobe. The autocorrelation is highly related to the SNR gain of the output and to interference resistance properties. Special characteristics of the Gaussian monocycles include: (i) higher order derivatives have higher SNR gain in single user and asynchronous multiple access channels but are less robust to interference than are lower order derivatives, (ii) narrower pulses have higher SNR gain in asynchronous multiple access channel at the cost of inferior interference resistance ability. Exact bit error rate performance of TH-PPM UWB systems in the presence of multiple access interference (MAI) is analyzed and simulated in 0. Furthermore, it is shown that with a moderate number of MAI sources, the standard Gaussian approximation becomes inaccurate at high SNRs.

Main principles for multi-access in UWB systems are discussed in 0. Reference 0 is one of the first public and widely cited papers outlining the potential of time-hopping impulse radio multi access communications. It describes the basic building blocks of the impulse transmitter and receiver and their mathematical formulations. It also shows an example for the bit error versus user capacity estimate at variable data rates. Finally, some drawbacks of high time-resolution impulse radio systems are mentioned: (i) the need for up to thousands of rake fingers in the multipath receiver, and (ii) complex initial clock acquisition. In 0, a quite comprehensive overall description of the time-hopping UWB system physical layer issues is given. Achievable transmission rates and multiple access capacities are estimated for analog and digital modulation formats. Numerical results indicate that the digital implementation has the potential for nearly one order of magnitude higher user densities than the analog one.

Reference 0 is focused on multiuser detection (MUD) possibilities for direct sequence UWB systems. It is demonstrated that the adaptive minimum mean squared error (MMSE) MUD receiver outperforms the rake receiver both in energy capture and in the interference rejection sense. Studied interference sources are narrowband IEEE 802.11a interference and wideband multiuser UWB interference. Ideally, a MMSE receiver can achieve AWGN bit error rate within a 1–2 dB margin even in dense multipath channels. In heavily loaded conditions the penalty of 6 dB is experienced, but at the same time the rake receivers suffer from unbearable error floors. Iterative partial parallel multiuser interference cancellation (PIC) is applied to the UWB multiuser system in 0. Matched filter, maximum-likelihood, and linear minimum mean squared error receivers are also used in the performance comparison. In this chapter, multiuser detection is combined with error control coding. The UWB system includes only one pulse per symbol and an AWGN channel is assumed. Numerical results show that it is possible to attain the coded single user BER bound for 8–15 users in a heavily loaded system without any processing gain. As the number of users increases and the bandwidth-to-pulse repetition frequency decreases, MAI is expected to affect system capacity and performance adversely. As a consequence, a framework for the design of multiuser detectors for UWB multiple access communications systems is presented in 0. Optimum multiuser detector is also proposed.
Coexistence of UWB system with some other radio systems is studied in 0. This means the evaluation of interference caused by the UWB system to other radio systems and vice versa. The coexisting radio concepts are GSM900, UMTS/WCDMA, and GPS. Several short Gaussian-based UWB pulses are employed. According to the numerical results, convenient selection of pulse waveform and width leads to interference resistance up to a certain limit. The pulse shape is in interaction with the data rate. High-pass filtered waveforms are preferred in the case of short UWB pulses, whereas generic Gaussian ones are favorable if long pulses are utilized. Interference caused by narrowband systems is the most detrimental to UWB if it is located at the UWB system’s nominal center frequency. In the GPS band, the DS based UWB system interfered less than the time hopping system.

Channel estimation for time hopping UWB communications is discussed in 0. The above survey of issues in UWB communications indicates a need for special care in interference avoidance or interference suppression due to extremely wide signal bandwidth and the possibility of interfering with other systems operating in the same bandwidth. One way to deal with the problem is to design the pulse shape in such a way that the signal has no significant spectral component in the occupied frequency bands. Pulse shapes respecting the FCC spectral mask are proposed in [28–30]. The drawback of such solutions is the need for over sampling and lack of flexibility in the case of the interfering signal not being stationary, as in military applications. Another approach is to use adaptive interference suppression as in the schemes summarized above. The solution discussed in this chapter belongs to the latter category. We will demonstrate the advantages of the approach with a number of numerical results. The solution is adaptive and can be implemented with no over-sampling, unlike other schemes.

Within the concept of cognitive radio [32–38] the scheme presents a cognitive ultra wide band (CUWB) systems, capable of monitoring and adapting to the amount and type of interference in the network, as with advanced personal area networks (PAN). In this concept, cognition is used to identify presence and location on the spectral scale of the interfering signal and to create a corresponding notch to suppress the interference.

### 12.2 SIGNAL AND INTERFERENCE MODELS

In general the signal transmitted by the desired user is modeled as:

\[ s(t) = \sum_i b(t - iNT_f - (1 - a_i)\Delta) \cos \omega_c t \quad (12.1) \]

where

\[ b(t) = \sum_{n=0}^{N-1} g(t - NT_f - h(n)T_c) \quad (12.2) \]

The signal can be also transmitted in the baseband with no carrier. In Equation (12.2), \( g(t) \) represents its basic pulse shape (monocycle pulse) and \( T_f \) represents the frame duration during which there is only one pulse \( T_c \) seconds wide. The sequence \( h(n) \) is the user’s time-hopping code and its elements are integers taking values in the range \( 0 \leq h(n) \leq N - 1 \). The parameter \( T_c \) is the duration of an addressable time bin within a frame. In other words, the right hand side of Equation (12.2) consists of a block of \( N \) time-hopped monocycles. \( a_i \) represents information bits (0,1). Equation (12.1) says that, if \( a_i \) were all zero, the signal would be a repetition of \( b(t) \)-shaped blocks with period \( NT_f \). \( \Delta \) may be viewed as the time shift impressed by a unit data symbol on the monocycles of a block. It is clear that the choice of \( \Delta \) affects the detection process and can be exploited to optimize system performance. In summary, the transmitted signal consists of a sequence of \( b(t) \)-shaped position-modulated blocks.

The code sequence restarts at every data symbol. This ‘short-code’ assumption is made here for the sake of simplicity and is in keeping with some trends in the design of third-generation CDMA cellular systems. Longer codes are conceivable and perhaps more attractive but lead to more complex channel estimation schemes.
The OFDM interference, generated for example by WLAN user is modeled as:

\[ j(t) = \sum_{i=0}^{N-1} d_i J_i \cos \left( 2\pi \left( f_c + \Delta f_c + \frac{i}{T_j} \right) t + \varphi_i \right) \]  

(12.3)

where \( N \) is the number of channels; \( d_i \) is FDM interference information bits; \( f_c + \Delta f_c \) are the first channel carrier frequency; \( J_i \) is OFDM interference amplitudes; \( \varphi_i \) is the channel phase at the receiver input, and \( T_j \) the bit interval.

The MC CDMA interference can be modeled as

\[ j(t) = K \sum_{k=1}^{K} \sum_{i=0}^{N-1} d_k(t) c_k(i) J_{k,i} \cos \left( 2\pi \left( f_c + \Delta f_c + \frac{i}{T_j} \right) t + \varphi_{k,i} \right) \]  

(12.4)

where \( N \) is the number of channels; \( d_k(t) \) is \( k \)th user interference bit; \( c_k(i) \) is the PN sequence of \( k \)th user and \( i \)th channel; \( K \) is the number of users; \( f_c + \Delta f_c \) is the first channel carrier frequency; \( J_{k,i} \) is the interference amplitude; \( \varphi_{k,i} \) is the channel phase at the receiver input, and \( T_j \) is the bit interval.

### 12.3 RECEIVER STRUCTURE AND PERFORMANCE

The receiver block diagram is shown in Figure 12.1. There are two interference rejection circuits, A and B. The first one (A) processes a signal if logic one is sent, and the other (B) processes a signal for logic zero. When several time-hopping signals are simultaneously received over a channel with \( L_c \) paths, the composite waveform at the output of the receiver antenna may be written as:

\[ r(t) = \sum_{l=1}^{L_c} (\gamma_l I_l s(t - \tau_l) \cos \omega_c t + \gamma_l Q_l s(t - \tau_l) \sin \omega_c t + n(t) + j(t) \]  

(12.5)

where \( n(t) \) is noise, and \( j(t) \) is total interference, \( \gamma_l = \gamma I_l + j \gamma Q_l \) is the complex attenuation and \( \tau_l \) is the delay in \( l \)th path. If we consider signal sampled at chip interval \( T_c \) we have:

\[ r(k) = rI(k) + j rQ(k) \]  

(12.6)
where
\[ r_I(t) = r(t) \cos \omega_c t \] (12.6a)
\[ r_Q(t) = r(t) \sin \omega_c t \] (12.6b)
\[ k = \frac{t}{T_c} \] (12.7)

Detection variable in the lth rake receiver finger is:
\[ D_{Il}(i) = D\hat{A}_{Il}(i) - D\hat{B}_{Il}(i) \] (12.8)
\[ D\hat{Q}_{Il}(i) = DAQ_{Il}(i) - DBQ_{Il}(i) \] (12.9)
\[ DA_{Il}(i) = \sum_{k=iNfTc - \frac{\tau_l}{T_c}}^{(i+1)NfTc - \frac{\tau_l}{T_c}} r_I(k)b \left( k - iNfTc - \frac{\tau_l}{T_c} \right) \] (12.10)
\[ DA\hat{Q}_{Il}(i) = \sum_{k=iNfTc - \frac{\tau_l}{T_c}}^{(i+1)NfTc - \frac{\tau_l}{T_c}} r_Q(k)b \left( k - iNfTc - \frac{\tau_l}{T_c} \right) \] (12.11)
\[ DB_{Il}(i) = \sum_{k=iNfTc - \frac{\tau_l}{T_c}}^{(i+1)NfTc - \frac{\tau_l}{T_c}} r_I(k)b \left( k - iNfTc - \frac{\tau_l}{T_c} - \frac{\Delta_j}{T_c} \right) \] (12.12)
\[ DB\hat{Q}_{Il}(i) = \sum_{k=iNfTc - \frac{\tau_l}{T_c}}^{(i+1)NfTc - \frac{\tau_l}{T_c}} r_Q(k)b \left( k - iNfTc - \frac{\tau_l}{T_c} - \frac{\Delta_j}{T_c} \right) \] (12.13)

where \( i \) represents the bit index. Now, the signal at the output of RAKE combiner is:
\[ d(i) = \sum_{l=1}^{L_c} (D\hat{I}_{Il}(i)T\hat{I}_{Il}(i) + D\hat{Q}_{Il}(i)T\hat{Q}_{Il}(i)) \] (12.14)

The weight of I branch in lth finger is:
\[ T\hat{I}_{Il}(i) = D\hat{A}_{Il}(i) + DB_I(i) \] (12.15)
and the weight of Q branch in lth finger is:
\[ T\hat{Q}_{Il}(i) = DAQ_{Il}(i) + DBQ_{Il}(i) \] (12.16)

### 12.3.1 Interference rejection circuit model

Interference rejection at UWB radio system may be performed by transversal filter employing an LMS algorithm. Basically, in the first step, the interfering signal is estimated in the presence of a UWB signal, which at that stage is considered as an additional noise. The estimated interference \( \hat{j} \) is subtracted from the overall input signal \( r \), creating the input signal \( r' = r - \hat{j} = s + n + j - \hat{j} = s + n + \Delta j \) to a standard UWB receiver. In order to predict the interference signal, sampling is performed at frame rate, and the adaptation of filter weights using the LMS algorithm is performed at bit rate. It is already known that the changes of the symbol in the interfering signal will disrupt the estimation process. Curve 3 in Figure 12.2, shows the detection variable at the output of the transversal filter when there is PSK interference (with an interference-to-signal ratio of 40 dB) at the input of the receiver together with the useful UWB signal and Gaussian noise. The presence of impulse interference may be seen in the figure. In this chapter we propose a new structure for the interference rejection, which successfully rejects this impulse interference.
Figure 12.2 Detection variable. 1, no interference; 2, interference with $J/S = 40$ dB, with interference rejection circuit; 3, interference with $J/S = 40$ dB, with transversal filter using classical LMS algorithm; PSK interference, $M = 4$, $\Delta = 5$ ns, $v_{bJ} = 100$ MSymbol/s, $v_{bTH} = 5$ Mbit/s, $T_{frame} = 10$ ns, $\Delta_f = 800$ MHz.

Curve 2, in Figure 12.2, shows the signal at the output of the proposed interference rejection structure in the case when there is PSK interference with an interference-to-signal ratio of 40 dB at the input of the receiver. It can be noted that the interference is rejected and the detection variable is very similar to the one when there is no interference (curve 1).

More details on the operating principle of the interference rejection circuit are shown in Figure 12.3. We consider the case of a bit interval having 20 frames with 5 samples per frame ($M = 2$). Central samples within each frame carry the same information about the useful signal and the interference symbols and, since each sample belongs to a different frame, all those samples originate from different instances of time. Similarly, samples from different frames equally distant from the central sample carry the correlated interference signal. Therefore, an equivalent signal may be formed in the following way: the $M$th equivalent signal sample is the sum of $M$ samples from each frame. Adaptation of filter weights using a LMS algorithm and interference prediction is performed using the equivalent signal samples.

As was already mentioned, the changes of the symbol in the interfering signal will disrupt the estimation process so that forward and backward predictions are used simultaneously. When the symbol change occurs, the filter with less disruption (smaller error at its output) is used to deliver the estimates. This process is described in the sequel in more detail. For additional insight into the problem the reader is also referred to [31]. Possible moments for the transition to happen are shown in Figure 12.3 and are denoted by $1^\circ$, $2^\circ$, $3^\circ$, and $4^\circ$. Therefore, at frame rate BPF(F) (backward prediction
Figure 12.3 Illustration of the algorithm operation.
filter) and FPF\(^{(F)}\) (forward prediction filter), filters are operating with weights being forwarded from the LMS algorithm adapted by the equivalent signal. BPF\(^{(F)}\) and FPF\(^{(F)}\) filters have the same weights during the useful signal bit interval, i.e. within all the frames belonging to the same bit interval. So, the interference is predicted using the same weights computed using the equivalent signal. If, in cases 1\(^{o}\) and 2\(^{o}\), we discard samples belonging to the BPF\(^{(F)}\) filter, we will also discard the interference transition influence on the prediction. For cases 3\(^{o}\) and 4\(^{o}\), samples that belong to the FPF\(^{(F)}\) should be discarded. This sample discarding is performed in the selector \(S\) at the outputs of BPF\(^{(F)}\) and FPF\(^{(F)}\) filters, based on the error signal.

Therefore, if there is no interference signal transition during the sampling within one frame, the equivalent signal will be formed using all the samples from that frame. Also, the same equivalent central sample (\(SX0^{(B)} = SX0^{(F)}\)) is passed to both BPF\(^{(B)}\) and FPF\(^{(B)}\) LMS algorithms. On the other hand, if there is interference signal transition during the sampling within one frame, the equivalent signal will be formed using samples from FPF\(^{(F)}\) (cases 1\(^{o}\) and 2\(^{o}\)) or BPF\(^{(F)}\) (cases 3\(^{o}\) and 4\(^{o}\)), and there will be two different equivalent central samples.

For the described interference rejection circuit we have:

(i) The first part of the interference rejection circuit processes data at frame level, and at each frame the following input signal processing is performed:

At filter A, the signal is sampled very close in time to the useful signal. For \(-M \leq m \leq M\) we have:

\[
A_{m}^{l}(n) = \sum_{k=n}^{(n+1)\frac{T_{f}}{T_{c}}-\frac{n}{T_{c}}} r I(k) g \left( k - n \frac{T_{f}}{T_{c}} - \frac{T_{f}}{T_{c}} - h(n) - m \right) \quad (12.17)
\]

\[
A_{m}^{l}(n) = A_{m}^{l}(n) + jAQ_{m}^{l}(n)
\]

\[
A_{m}^{l}(n) = \sum_{k=n}^{(n+1)\frac{T_{f}}{T_{c}}-\frac{n}{T_{c}}} r I(k) g \left( k - n \frac{T_{f}}{T_{c}} - \frac{T_{f}}{T_{c}} - h(n) - m \right) \quad (12.18)
\]

\[
A_{m}^{l}(n) = \sum_{k=n}^{(n+1)\frac{T_{f}}{T_{c}}-\frac{n}{T_{c}}} r Q(k) g \left( k - n \frac{T_{f}}{T_{c}} - \frac{T_{f}}{T_{c}} - h(n) - m \right) \quad (12.19)
\]

For filter B we have:

\[
B_{m}^{l}(n) = \sum_{k=n}^{(n+1)\frac{T_{f}}{T_{c}}-\frac{n}{T_{c}}} r I(k) g \left( k - n \frac{T_{f}}{T_{c}} - \frac{T_{f}}{T_{c}} - h(n) - m - \frac{\Delta}{T_{c}} \right) \quad (12.20)
\]

\[
B_{m}^{l}(n) = \sum_{k=n}^{(n+1)\frac{T_{f}}{T_{c}}-\frac{n}{T_{c}}} r Q(k) g \left( k - n \frac{T_{f}}{T_{c}} - \frac{T_{f}}{T_{c}} - h(n) - m - \frac{\Delta}{T_{c}} \right) \quad (12.21)
\]

\[
B_{m}^{l}(n) = \sum_{k=n}^{(n+1)\frac{T_{f}}{T_{c}}-\frac{n}{T_{c}}} r I(k) g \left( k - n \frac{T_{f}}{T_{c}} - \frac{T_{f}}{T_{c}} - h(n) - m - \frac{\Delta}{T_{c}} \right) \quad (12.22)
\]

After that, variables \(C1\) and \(C2\), which the operation of one side of filter \(X\) (A and B) is based on, are determined:

\[
CX1^{l}(n) = \begin{cases} 0, \left( \sum_{m=-M}^{1} X_{m}^{l}(n)W_{m}(i) \right)^{2} \geq A_{1} \left( \sum_{m=-M}^{M} X_{m}^{l}(n)W_{m}(i) \right)^{2} \quad (12.23) \\ 1, \left( \sum_{m=-M}^{1} X_{m}^{l}(n)W_{m}(i) \right)^{2} < A_{1} \left( \sum_{m=-M}^{M} X_{m}^{l}(n)W_{m}(i) \right)^{2} \end{cases}
\]

\[
CX2^{l}(n) = \begin{cases} 0, \left( \sum_{m=-M}^{1} X_{m}^{l}(n)W_{m}(i) \right)^{2} \geq A_{2} \left( \sum_{m=-M}^{M} X_{m}^{l}(n)W_{m}(i) \right)^{2} \quad (12.24) \\ 1, \left( \sum_{m=-M}^{1} X_{m}^{l}(n)W_{m}(i) \right)^{2} < A_{2} \left( \sum_{m=-M}^{M} X_{m}^{l}(n)W_{m}(i) \right)^{2} \end{cases}
\]
where $A_i$ ($i = 1, 2$) are constants (for $A_i \to \infty$, the selector selects all the samples and the structure operates as a traditional LMS algorithm). These gains are introduced because of the decrease of noise influence on the irregular selections.

(ii) The second part of the interference rejection circuit operates at bit interval level $T_b = NT_f$, and for each $i$th bit we have the following signals:

$$ SX_m^l(i) = \sum_{n=iN}^{(i+1)N} X_m^l(n)CX_1^l(n) \quad -M \leq m \leq -1 $$  \hfill (12.25)

$$ SX_m^l(i) = \sum_{n=iN}^{(i+1)N} X_m^l(n)CX_2^l(n) \quad 1 \leq m \leq M $$  \hfill (12.26)

$$ SX_1^l(i) = \sum_{n=iN}^{(i+1)N} X_0^l(n)CX_1^l(n) $$  \hfill (12.27)

$$ SX_2^l(i) = \sum_{n=iN}^{(i+1)N} X_0^l(n)CX_2^l(n) $$  \hfill (12.28)

The error signal used for $w$ coefficients adaptation is:

$$ EX_1^l(i) = SX_1^l(i) - \sum_{m=-M}^{-1} SX_m^l(i)W_m(i) $$  \hfill (12.29)

$$ EX_2^l(i) = SX_2^l(i) - \sum_{m=1}^{M} SX_m^l(i)W_m(i) $$  \hfill (12.30)

The filter output signal, being led to the RAKE combiner, is:

$$ DX_l^l(i) = EX_1^l(i) + EX_2^l(i) $$  \hfill (12.31)

The adaptation algorithm is defined as:

$$ W_m(i+1) = W_m(i) + \frac{\mu EX_1^l(i)(SX_m^l(i))^\ast}{\sum_{j=-M}^{-1} (SX_j^l(i))^\ast} \quad -M \leq m \leq -1 $$  \hfill (12.32)

$$ W_m(i+1) = W_m(i) + \frac{\mu EX_2^l(i)(SX_m^l(i))^\ast}{\sum_{j=1}^{M} (SX_j^l(i))^\ast} \quad 1 \leq m \leq M $$  \hfill (12.33)

Finally, the interference rejection circuit is made symmetrical in the following way:

$$ W_m(i+1) = \frac{(W_m(i+1) + W_{-m}(i+1)^\ast)}{2} \quad 1 \leq m \leq M $$  \hfill (12.34)

$$ W_m(i+1) = W_{-m}(i+1)^\ast \quad -1 \leq m \leq -M $$  \hfill (12.35)

WLAN systems that are based on OFDMA technology are widely used, and therefore they could interfere with UWB systems. OFDM interference, unlike other interference types, requires filters of a shorter length in order to be successfully rejected. Thus, OFDM interference is considered separately, and for its rejection a new structure is designed. Basic principles of this new OFDM structure operation are described below.
(ii) The second part of the interference rejection circuit operates at bit interval level. The filter output signal, being led to RAKE combiner, is:

\[ CX_1^l(n) = CX_2^l_m(n) = \begin{cases} 
0, & \left(\left(X_m^l(n)W_m(i)\right)\right)^2 \geq A_1 \left(\left(X_m^l(n)W_m(i)\right)\right)^2 \\
\geq A_2 \left(\left(X_m^l(n)W_m(i)\right)\right)^2 \\
1 & \text{otherwise}
\end{cases} \]

where \( A_i (i = 1, 2) \) are constants (for \( A_i \rightarrow \infty \), the selector selects all the samples and the structure operates as traditional LMS algorithm). These gains are introduced because of the decrease of noise influence on the irregular selections.

(i) The first part of the interference rejection circuit processes data at frame level, and parameters \( A_m^l(n) \) and \( B_m^l(n) \) are calculated from Equations (12.17–22), where \(-M \leq m \leq M\). After that, variables \( C_1 \) and \( C_2 \), which the operation of one side of filter \( X \) (A and B) is based on, are determined:

\[ CX_1^l_m(n) = CX_2^l_m(n) = \begin{cases} 
0, & \left(\left(X_m^l(n)W_m(i)\right)\right)^2 \geq A_1 \left(\left(X_m^l(n)W_m(i)\right)\right)^2 \\
\geq A_2 \left(\left(X_m^l(n)W_m(i)\right)\right)^2 \\
1 & \text{otherwise}
\end{cases} \]

The error signal used for coefficients adaptation is:

\[ EX_1^l_m(i) = SX_1^l_m(i) - SX_2^l_m(i)W_m(i) \]

\[ EX_2^l_m(i) = SX_2^l_m(i) - SX_1^l_m(i)W_m(i) \]

The filter output signal, being led to RAKE combiner, is:

\[ DX^l_m(i) = EX_1^l_m(i) + EX_2^l_m(i) \]

The adaptation algorithm is defined as:

\[ W_m^l(i + 1) = W_m^l(i) + \frac{\mu EA_1W_m^l(i)\left(SA_m^l(i)\right)^*}{\left(SA_m^l(i)\right)^2} + \frac{\mu EB_1W_m^l(i)\left(SB_m^l(i)\right)^*}{\left(SB_m^l(i)\right)^2} \]

\[ -M \leq m \leq -1 \]

\[ W_m^l(i + 1) = W_m^l(i) + \frac{\mu EA_2W_m^l(i)\left(SA_m^l(i)\right)^*}{\left(SA_m^l(i)\right)^2} + \frac{\mu EB_2W_m^l(i)\left(SB_m^l(i)\right)^*}{\left(SB_m^l(i)\right)^2} \]

\[ 1 \leq m \leq M \]
Finally, the interference rejection circuit is made symmetrical in the following way:

\[
W_m(i+1) = \frac{(W_m(i+1) + W_m'(i+1)^*)}{2} \quad 1 \leq m \leq M
\]

\[
W_m(i+1) = W_m(i+1)^* \quad -M \leq m \leq -1
\]  

The signal at the output of RAKE combiner is:

\[
d(i) = \sum_{l=1}^{L_x} \sum_{m=1}^{M} \left[ \left( \text{Re} \left\{ D A_m(i) \right\} - \text{Re} \left\{ D B_m(i) \right\} \right) \cdot \text{Re} \left\{ T_m(i) \right\} + \left( \text{Im} \left\{ D A_m(i) \right\} \cdot \text{Im} \left\{ T_m(i) \right\} \right) \right]
\]  

where

\[
T_m(i) = \frac{D A_m(i) + D B_m(i)}{|D A_m(i) + D B_m(i)|}
\]  

\(DX_m(i)\) for filter A (X = A) is denoted by \(DA_m(i)\), and \(DX_m(i)\) for filter B (X = B) is denoted by \(DB_m(i)\).

The error probability per bit is:

\[
P_e = \frac{1}{N_i} \sum_{i=1}^{N_i} P_e(i)
\]  

where

\[
P_e(i) = \begin{cases} 
\frac{1}{2} \text{erfc} \left( \sqrt{\frac{SNR(i)}{2}} \right) & \text{if } \sum_{j=1}^{N_a} d^j(i) \geq 0 \\
1 - \frac{1}{2} \text{erfc} \left( \sqrt{\frac{SNR(i)}{2}} \right) & \text{if } \sum_{j=1}^{N_a} d^j(i) < 0
\end{cases}
\]  

and \(N_i\) is the bit ensemble size (measured in number of information bits) used for averaging the result, and \(N_a\) is the number of ensemble members. Estimated signal to noise ratio per bit is:

\[
SNR(i) = \frac{\left( \frac{1}{N_a} \sum_{j=1}^{N_a} d^j(i) \right)^2}{\frac{1}{N_a} \sum_{j=1}^{N_a} (d^j(i))^2 - \left( \frac{1}{N_a} \sum_{j=1}^{N_a} d^j(i) \right)^2}
\]  

where \(d^j(i)\) is the \(j\)th ensemble member.

### 12.4 PERFORMANCE EXAMPLES

Figure 12.4 presents the results for BER as a function of signal to noise ratio SNR, in the presence of different PSK/QAM-type interfering signals. Additional parameters for the signals are: filter length \(M = 4\), \(\Delta = 5\) ns, \(v_{b,j} = 100\) MSymbol/s, \(v_{STH} = 5\) Mbit/s, \(\Delta f_c = 800\) MHz and \(T_{frame} = 10\) ns. One can see: (a) fair agreement of simulation and numerical results, (b) the performance results are close to a no interference case although the interference of a level of 40 dB above the UWB signal is present, and (c) there is also a slight degradation of the performance when the interfering signal constellation size is increased.
Figure 12.4 Error probability as a function of signal to noise ratio. Error probability based on Monte-Carlo simulation: a, no interference, without interference rejection filter; b, no interference, with interference rejection filter; c, PSK interference, $J/S = 40$ dB, with interference rejection filter; d, QPSK interference, $J/S = 40$ dB, with interference rejection filter; e, 16QAM interference, $J/S = 40$ dB, with interference rejection filter; f, 64QAM interference, $J/S = 40$ dB, with interference rejection filter; g, 256QAM interference, $J/S = 40$ dB, with interference rejection filter; Error probability based on estimated detection variable signal to noise ratio: b$_1$, the same parameters as b; d$_1$, the same parameters as d; f$_1$, the same parameters as f. $\Delta f_c = 800$ MHz, $M = 4$, $\Delta = 5$ ns, $v_{b,J} = 100$ MSymbol/s, $v_{b,T_H} = 5$ Mbit/s, $T_{frame} = 10$ ns.

Figure 12.5 presents the results for BER as a function of interference to signal ratio $J/S$, in the presence of different PSK/QAM type interfering signals. Additional parameters for the signals are: filter length $M = 4$, SNR = 7 dB, $\Delta = 5$ ns, $v_{b,J} = 100$ MSymbol/s, $v_{b,T_H} = 5$ Mbit/s, $\Delta f_c = 800$ MHz and $T_{frame} = 10$ ns. One can see that when $J/S$ becomes larger than zero (5 dB) the BER increases rapidly if there is no interference suppression (curves A). The performance is very much similar if a standard LMS algorithm is used (curves C). On the other hand the new structure is performing significantly better (curves B). There is again a slight degradation of the performance when the interfering signal constellation size is increased.
Figure 12.5 Error probability as a function of interference to signal ratio. A, without interference rejection; B, with interference rejection circuit; C, with classical LMS interference rejection filter; a, PSK interference; b, QPSK interference; c, 16QAM interference; d, 64QAM interference; e, 256QAM interference; $\Delta f_c = 800$ MHz, $M = 4$, SNR = 7 dB, $\Delta = 5$ ns, $v_{bJ} = 100$ MSymbol/s; $v_{bTH} = 5$ Mbit/s, $T_{\text{frame}} = 10$ ns.

Figure 12.6 presents the results for BER as a function of interference symbol duration $T_j/T_c$ in the presence of different PSK/QAM type interfering signals. Additional parameters for the signals are $J/S = 30$ dB, SNR = 7 dB, $M = 4$, $\Delta = 5$ ns, $v_{bTH} = 5$ Mbit/s, $\Delta f_c = 800$ MHz and $T_{\text{frame}} = 10$ ns. One can see that BER decreases when $T_j/T_c$ increases. There is again a slight degradation of the performance when the interfering signal constellation size is increased.

Figure 12.7 presents the results for BER as a function of interference symbol duration $T_j/T_c$ in the presence of MC CDMA type interfering signals for a different number of subcarriers $N$ and number of users $K$.

Additional parameters for the signals are $J/S = 30$ dB, SNR = 7 dB, $M = 4$, $\Delta = 5$ ns, $v_{bTH} = 5$ Mbit/s, $T_{\text{frame}} = 10$ ns, and $\Delta f_c = 800$ MHz. One can see again that BER decreases when $T_j/T_c$ increases. The performance is improved if the number of subcarriers is decreased and if the number
of users is increased for the same overall power of the interfering signal. This can be explained by the fact that a sum of MC CDMA signals will create an equivalent multicarrier signal with lower number of dominant subcarriers that are suppressed more effectively by the filter because the LMS algorithm better adjusts the filter weights. This is demonstrated in Figures 12.9 and 12.10 for an OFDM signal.

Figure 12.8 shows the error probability as a function of the interference rejection structure length. Curves labeled with A show the results of non-OFDM interference rejection using the new structure. It can be seen that there is a optimal structure length, and that it does not depend on interference constellation size. OFDM interference rejection results are labeled with B. The new structure results are represented with thin lines, while thick lines show the results for the new OFDM structure. In the case of high power interference (curves b) the new structure has optimal length $2M = 2$ regardless of interference constellation size. The new OFDM structure, in the case of such strong interference, has a much better performance for high structure length (optimal length is $2M = 8$). In the case of low power interference (curves a) both the new and the new OFDM signal suppression structure have similar performances for the same structure length ($2M = 8$).
Figure 12.7 Error probability as a function of MC CDMA interference bit duration. a, \( N = 64, K = 10 \); b, \( N = 32, K = 5 \); c, \( N = 32, K = 10 \); d, \( N = 32, K = 20 \); e, \( N = 16, K = 10 \), MC CDMA interference: \( J/S = 30 \) dB, \( \text{SNR} = 7 \) dB, \( M = 4, \Delta = 5 \) ns, \( \nu_{bTH} = 5 \) Mbit/s, \( T_{\text{frame}} = 10 \) ns, \( \Delta f_c = 800 \) MHz.

Figure 12.9 presents the results for BER as a function of interference symbol duration \( T_j/T_c \) and the number of subcarriers \( N \) in the presence of OFDM type interfering signals. Additional parameters for the signals are \( J/S = 30 \) dB, \( \text{SNR} = 7 \) dB, \( M = 4, \Delta = 5 \) ns, \( \nu_{bTH} = 5 \) Mbit/s, \( T_{\text{frame}} = 10 \) ns, \( \Delta f_c = 800 \) MHz and 16QAM per subcarrier. One can see that BER decreases when \( T_j/T_c \) increases up to \( T_j/T_c \approx 300 \) for \( N \leq 8 \). Beyond that point there is no reduction in BER if \( T_j/T_c \) is further increased. There is a significant degradation of the performance when the number of subcarriers in the OFDM signal is \( N \geq 16 \).

Figure 12.10 presents the results for BER as a function of interference symbol duration \( T_j/T_c \) in the presence of OFDM type interfering signals. Additional parameters for the signals are \( J/S = 60 \) dB, \( \text{SNR} = 13 \) dB, \( M = 4, \Delta = 5 \) ns, \( \nu_{bTH} = 5 \) Mbit/s, \( T_{\text{frame}} = 10 \) ns, \( \Delta f_c = 800 \) MHz and \( N = 64 \). One can see again that BER decreases when \( T_j/T_c \) increases. It can be seen that the interfering signal constellation size has a very slight influence on the system’s performance, regardless of the interference rejection circuit. Both new structures perform much better than does the classical LMS algorithm, but for higher \( T_j/T_c \) (\( \approx 6000 \)) the new OFDM structure brings additional performance improvement for OFDM interference rejection compared with the new structure.
Figure 12.8 Error probability as a function of filter length. a, $J/S = 30$ dB; b, $J/S = 60$ dB; A, Non OFDM interference; B, OFDM interference; thin lines, new structure; thick lines, new OFDM structure; SNR = 13 dB, $\Delta = 5$ ns, $v_{bT/H} = 5$ Mbit/s, $T_{\text{frame}} = 10$ ns, $\Delta f_c = 800$ MHz.

Figure 12.9 Error probability as a function of OFDM interference bit duration and the number of subcarriers. OFDM/16QAM interference: $\Delta f_c = 800$ MHz, $J/S = 30$ dB, SNR = 7 dB, $M = 4$, $\Delta = 5$ ns, $v_{bT/H} = 5$ Mbit/s, $T_{\text{frame}} = 10$ ns.
Figure 12.10 Error probability as a function of OFDM interference bit duration. A, new OFDM structure; B, new structure; C, classical LMS interference rejection filter; $N = 64$, $\Delta f_c = 800$ MHz, $J/S = 60$ dB, SNR = 13 dB, $M = 4$, $\Delta = 5$ ns, $v_{bT H} = 5$ Mbit/s, $T_{frame} = 10$ ns.

REFERENCES


REFERENCES


Positioning in Wireless Networks

13

13.1 MOBILE STATION LOCATION IN CELLULAR NETWORKS

13.1.1 Introduction

In this section we present a general mathematical framework that covers all the positioning techniques for processing absolute and/or relative distance measurements between the mobile station and multiple base transceiver stations. Based on this scheme, a general measure of positioning accuracy is introduced and then analyzed in the special cases defined by the three most feasible techniques. A geometrical interpretation of the formulas that define the location accuracy and a comparison between the techniques from the geometric conditioning point of view are also given.

The Federal Communications Commission (FCC) mandated US cellular operators, in 1996, to locate mobile phones calling the emergency number 911 by October 2001 [1, 2]. The European Union has been taking steps towards similar regulation [3]. Emergency services, beside commercial applications such as vehicle fleet management, intelligent transport systems, and location-based billing [4, 5], are also examples of applications of mobile station (MS) positioning. The first studies on positioning of mobile phones were published in the 1970s [6]. Since that time, several techniques for locating the MS by measuring attenuation, direction of arrival, and delay of the radio signals exchanged between the MS and multiple base transceiver stations (BTSs) have been proposed (see [7] for an overview). Simulation results of phase-ranging and pulse-ranging techniques are presented in [8] and [9]. Methods for locating mobiles by processing attenuation measurements are presented in [10]–[14]. In [15], methods involving time-of-arrival (TOA) and angle-of-arrival (AOA) measurements in a code-division, multiple-access, network are analyzed. Positioning through AOA measurements is discussed in [16]. Performance of tracking algorithms that process absolute and/or relative propagation delay measurements are presented using simulations in [17]–[19] and verified experimentally in [20] and [21–23].

Standardization activities for MS positioning are presented in [24–26]. Along the years, the standardization group T1P1.5 has considered four alternatives:

(i) the network-assisted GPS method [27], which calculates the MS location by using GPS technology;

(ii) time advance [28];
(iii) enhanced observed time difference (E-OTD) [29];
(iv) uplink TOA [30].

The network-assisted GPS method is beyond the scope of this section and the remaining methods will be the focus of this discussion. The more general goal of this section is to analyze the accuracy of the multilateration techniques when used in cellular networks. Such analysis is carried out in a general manner so that it applies to the third and incoming fourth generation of universal mobile telecommunications systems [31].

The time advance, enhanced OTD, and uplink TOA methods estimate the MS coordinates by processing with multilateration techniques absolute distances (ADs) and/or relative distances (RDs) between the MS and multiple BTSs. A general analysis of the multilateration techniques that process only ADs (circular multilateration) or only RDs (hyperbolic multilateration) can be found in [32–35]. The technique that combines ADs and RDs (referred to as mixed multilateration) is of major interest for cellular applications.

In this section we present the basics of the time advance, enhanced OTD, and uplink TOA methods as a background for later analysis. The mathematical formulation of the generic multilateration problem and its linear weighted least squares (WLS) solution are also presented. The WLS location estimate is then used to derive an explicit expression for the positioning accuracy measure. The accuracy of circular, hyperbolic, and mixed multilateration is geometrically interpreted.

13.1.2 MS location estimation using AD and RD measurements

The time advance method uses the existing timing advance (TA) parameter, which is introduced to avoid overlapping of bursts transmitted by the MS during a call in TDMA systems [28, 38, 39]. For positioning purposes, the TA is considered to be an estimate of the absolute distance between MS and serving BTS, and is used to implement the circular multilateration technique. In a two-dimensional scenario, as assumed throughout this section, ADs from $N \geq 3$ different stations are needed to find the MS coordinates at the intersection of $N$ circumferences centered at the BTSs with radii equal to the AD measurements.

In a TDMA-based network, the TA is estimated by the serving BTS only when the MS is in connected mode (i.e., the MS is communicating with the serving BTS using a dedicated channel) [39]. As a consequence, ADs from multiple BTSs can only be measured by sequentially forcing the communication to be handed over from one BTS to another until all the $N$ BTSs have been accessed. In practice, when a positioning handover occurs, the TA is estimated by the new serving BTS, but the handover request is not responded to and the connection returns to the previous serving BTS [28].

The E-OTD method is based on three parameters: observed time difference (OTD), real time difference (RTD), and geometric time difference $GTD = RTD − OTD$. If a burst is transmitted by BTS$_1$ and BTS$_2$ at the instants $t_{Rx1}$ and $t_{Rx2}$ respectively, and received by the MS at instants $t_{Rx1}$ and $t_{Rx2}$ respectively, then the RTD is $t_{Rx2} − t_{Rx1}$ and the OTD is $t_{Rx2} − t_{Rx1}$. The GTD is a scaled measure of the relative distance between the MS and the pair {BTS$_1$, BTS$_2$}; in fact, $GTD = RTD − OTD = (t_{Rx1} − t_{Rx1}) − (t_{Rx2} − t_{Rx2}) = (d_1 − d_2)/c = RD/c$, with $c$ being the speed of light and $d_1 = c(t_{Rx1} − t_{Rx1})$ i $d_2 = c(t_{Rx2} − t_{Rx2})$ being the lengths of the propagation path between the MS and BTS$_1$, BTS$_2$, respectively [29]. The possible positions of an MS observing a constant GTD value are located on a hyperbola having foci at BTS$_1$ and BTS$_2$. In a two-dimensional scenario, the MS position is calculated via hyperbolic multilateration at the intersection of at least two hyperbolas; thus $N \geq 3$ BTSs are needed to implement this technique. Because of the implementation [39], the enhanced OTD method takes one of the $N$ available BTSs as reference BTS and uses it to calculate all the $N − 1$ RDs. As a consequence, linear dependence between multiple equations is avoided.

13.1.3 The circular, hyperbolic, and mixed multilateration

Let \{BTS$_1$, \ldots, BTS$_N$\} be the $N$ base stations available for locating the MS. BTS$_1$ is the reference station eventually used to evaluate RDs. Let $S = \{1, \ldots, N\}$ be an ordered set of indexes identifying
the BTSs. Within $S$, the following ordered subsets are defined as in Figure 13.1, where:

- $S_c = \{\text{indexes identifying the BTSs involved in AD measurements}\}$;
- $S_h = \{\text{indexes identifying the BTSs involved in RD measurements}\}$;
- $S_m = S_c \cap S_h = \{\text{indexes identifying the BTSs involved both in AD and RD measurements}\}$;
- $\bar{S}_c = S_c - S_m = \{\text{indexes identifying the BTSs involved only in AD measurements}\}$;
- $\bar{S}_h = S_h - S_m = \{\text{indexes identifying the BTSs involved only in RD measurements}\}$.

The number of elements in each subset is $N_c = \text{size}\{S_c\}$, $N_h = \text{size}\{S_h\}$, $N_m = \text{size}\{S_m\}$, $\bar{N}_c = \text{size}\{\bar{S}_c\}$ and $\bar{N}_h = \text{size}\{\bar{S}_h\}$. Moreover $S = S_c \cup S_h = \bar{S}_c \cup \bar{S}_h \cup S_m$ and $N = N_c + N_h - N_m = \bar{N}_c + \bar{N}_h + N_m$.

The general multilateration problem is described by a set of equations in which each AD defines a circumference and each RD defines a hyperbola. The known BTS coordinates are $(x_1, y_1), \ldots, (x_n, y_n)$, and the unknown MS coordinates are $(x, y)$. The AD measured between the MS and the BTS $i (i \in \bar{S}_c)$ is $d_i$; the RD measured between the MS and the pair of stations $\{\text{BTS}_1, \text{BTS}_l\} (l \in \bar{S}_h - \{1\})$ is $r_{1l}$; the AD and RD involving the BTS identified by $S_m$ are $d_p \ (p \in S_m)$ and $r_{1q} (q \in S_m - \{1\})$, respectively.

$$\begin{align*}
\sqrt{(x - x_i)^2 + (y - y_i)^2} &= d_i, & i \in \bar{S}_c \\
\sqrt{(x - x_i)^2 + (y - y_i)^2} - \sqrt{(x - x_j)^2 + (y - y_j)^2} &= r_{1l}, & l \in \bar{S}_h - \{1\} \\
\sqrt{(x - x_p)^2 + (y - y_p)^2} &= d_p, & p \in S_m \\
\sqrt{(x - x_i)^2 + (y - y_i)^2} - \sqrt{(x - x_q)^2 + (y - y_q)^2} &= r_{1q}, & q \in S_m - \{1\}
\end{align*}$$

(13.1)

The circular, hyperbolic, and mixed multilateration can be derived from Equation (13.1) by properly defining $S_c$, $S_h$, and $S_m$. In circular multilateration all the available BTSs are used only to measure ADs, and no RD measurements are available; thus $\bar{S}_h$ and $S_m$ are empty and $\bar{S}_c$ coincides with $S$: $S_m = S_h = \bar{S}_h = 0$; $S_c = S_e = S = \{1, \ldots, N\}$. In hyperbolic multilateration all the available BTSs are used to measure only RDs. No ADs are used. BTS$_1$ is only used to measure RDs; thus $S_m$ is empty and BTS$_1$ belongs to subset $\bar{S}_h$, which in turn coincides with $S_h$: $S_m = S_e = \bar{S}_c = 0$; $\bar{S}_h = S_h = S = \{1, \ldots, N\}$. Finally, in mixed multilateration all the available BTSs are used to measure
RDs. BTS₁ is also used to measure ADs; thus BTS₁ belongs to \( S_h \) and is the only element in \( S_m \), which coincides with \( S_c \) (BTS₁ \( \notin S_h \)). Since no BTSs are used only to measure ADs, \( S_c \) is empty: \( S_h = \{1, \ldots, N\}; S_m = S_c = \{1\}; S_h = \{2, \ldots, N\}; S_c = 0 \).

### 13.1.4 WLS Solution of the Location Problem

In real applications, Equation (13.1) should be further modified to include the presence of noise. In this section, \( x \) and \( y \) are estimated by using a linear WLS algorithm [41, 48], following the same approach of [35] and [37]. To apply the WLS estimation, Equation (13.1) is linearized by assuming that an a priori estimate of the MS position, \( P_{(0)} = (x^{(0)}, y^{(0)}) \), is available. \( P_{(0)} \) could be determined, for instance, from a previous iteration of the WLS algorithm or by classical calculus to find the intersection of a number of circumferences defined by Equation (13.1). If \( P_{(0)} \) is sufficiently close to \( P = (x, y) \), the equations resulting from the \( i \)th AD measurement and the \( l \)th RD measurement can be accurately represented by their linear approximation in a neighborhood of \( P_{(0)} \):

\[
\begin{align*}
\Delta x &= x - x^{(0)}; \\
\Delta y &= y - y^{(0)}; \\
u_{i,x} &= \left( x_i - x^{(0)} \right) / d_i^{(0)}; \\
u_{i,y} &= \left( y_i - y^{(0)} \right) / d_i^{(0)}; \\
d_i^{(0)} &= \sqrt{(x_i - x^{(0)})^2 + (y_i - y^{(0)})^2}; \\
r_{il}^{(0)} &= d_i^{(0)} - d_l^{(0)}; \\
w_{1l,x} &= u_{1,x} - u_{l,x}; \\
w_{1l,y} &= u_{1,y} - u_{l,y}.
\end{align*}
\]

A graphical representation is given in Figure 13.2 where \( \hat{u}_i = u_{i,x} \hat{x} + u_{i,y} \hat{y} \) is a unit vector originated at the MS and directed toward BTSᵢ (\( \hat{x} \) and \( \hat{y} \) are the unit vectors in the \( x \) and \( y \) directions) and \( w_{1l,x} \) and \( w_{1l,y} \) are the \( x \)-\( y \) components of the difference vector \( w_{1l} = \hat{u}_i - \hat{u}_l \).

Stacking the linearized Equations (13.2) as in Equation (13.1) results in a linear system \( Ax = b \) defined by the \( 2 \times 1 \) unknown vector \( x \), the \( (\tilde{N}_c + \tilde{N}_h + 2N_m - 1) \times 2 \) design matrix \( A \), and the

![Figure 13.2 Definition of the basic vectors: \( w_{1l}, \hat{u}_i, \hat{u}_l \). The area of the parallelogram of vertices \( P^{(0)}, P^{(l)}, \bar{P}^{(0)}, P^{(1)} \) is \( A_{1j} \). The area of the parallelogram of vertices \( P^{(0)}, P^{(l)}, P^{(1)}, P^{(m)} \) is \( B_{lm} \).](image)
Equation (13.3), and the location error along the directions to derive an explicit expression of

\[ D = \text{typical coherence time of the mobile radio channel.} \]

Under this hypothesis, ADs and RDs are averages of ‘raw’ propagation delays estimated during periods longer than the 

\[ \bar{\Delta} \]

with

\[ A \in \mathbb{R}^{2 \times 2} \]

is a \( \bar{N}_c \times 2 \) matrix with the x–y components of \( \bar{u}_i (i \in \bar{S}_c) \) on its columns. Analogously, \( W_b = [w_{1,i}, w_{2,i}]_{i \in \bar{S}_b} \), \( U_m = [u_{p,x}, u_{p,y}]_{p \in \bar{S}_m} \) and \( W_m = [w_{1,q}, w_{2,q}]_{q \in \bar{S}_m} \). The vectors

\[ D = \begin{bmatrix} d_1 \cdots d_{\bar{N}_c} \end{bmatrix} \]

\[ R_b = \begin{bmatrix} r_{11} \cdots r_{1q} \end{bmatrix} \]

\[ D_m = \begin{bmatrix} d_{p1} \cdots d_{pq} \end{bmatrix} \]

\[ R_m = \begin{bmatrix} r_{q1} \cdots r_{qq} \end{bmatrix} \]

are obtained by stacking the measurements in a single column. The WLS solution of the linear problem

\[ Ax = b \]

that minimizes the scalar cost function

\[ J(x) = (Ax - b)^T Q_b^{-1} (Ax - b) \]

is defined in Equation (13.3), and \( Q_b \) is the covariance matrix of \( b \). [41, 48].

### 13.1.5 Accuracy measure

The covariance matrix of \( \bar{x} \) is \( Q_{\bar{x}} = \text{Cov}([\bar{x}]) = G^{-1} \). The diagonal entries of \( Q_{\bar{x}} \) are the variances of the location error along the directions \( x \) and \( y \). The square root of their sum, corresponding to the square root of the trace of \( Q_{\bar{x}} \), is the accuracy measure considered hereafter: \( M = \sqrt{\text{Tr}(Q_{\bar{x}})} \). In order to derive an explicit expression of \( M, Q_{\bar{x}} = G^{-1} = (A^T Q_b^{-1} A)^{-1} \) must be evaluated; \( A \) is defined in Equation (13.3), and \( Q_b \) is derived below.

Let \( e_i \) and \( f_i \) be the measurement errors affecting the \( i \text{th AD} \) measurement \( d_i \) and the \( i \text{th RD} \) measurement \( r_{i1} \), respectively. In TDMA-based systems, such errors can be assumed uncorrelated. ADs and RDs are averages of ‘raw’ propagation delays estimated during periods longer than the typical coherence time of the mobile radio channel. Under this hypothesis, \( Q_b \) has the following block-diagonal structure (matrices \( 0 \) are properly sized matrices with all zero entries):

\[ Q_b = \begin{bmatrix}
S_{c,c} & 0 & 0 & 0 \\
0 & S_{f,f} & 0 & 0 \\
0 & 0 & S_{c,m} & 0 \\
0 & 0 & 0 & S_{f,m}
\end{bmatrix} \] (13.4)

where

\[ S_{c,c} = \text{diag}(\sigma_{c,i}^2)_{i \in \bar{S}_c} \]

\[ S_{f,f} = \text{diag}(\sigma_{f,j}^2)_{j \in \bar{S}_b} \]

\[ S_{c,m} = \text{diag}(\sigma_{c,p}^2)_{p \in \bar{S}_m} \]

\[ S_{f,m} = \text{diag}(\sigma_{f,q}^2)_{q \in \bar{S}_m} \]

\[ \sigma_{c,i}^2 \text{ are variances of the AD measurement errors.} \]

\[ \sigma_{f,j}^2 \text{ are variances of the RD measurement errors.} \]

\[ A \text{ and } Q_b \text{ defined in Equation (13.3) and (13.4) can be used to derive } G = A^T Q_b^{-1} A \]

\[ G = \begin{bmatrix}
G_{11} & G_{12} \\
G_{12} & G_{22}
\end{bmatrix} \] (13.5)

with

\[ G_{11} = \sum_{i \in \bar{S}_c} \sigma_{c,i}^{-2} u_{1,i}^2 \]

\[ + \sum_{p \in \bar{S}_m} \sigma_{c,p}^{-2} u_{1,p}^2 \]

\[ + \sum_{i \in \bar{S}_b} \sigma_{f,i}^{-2} w_{1,i}^2 \]

\[ + \sum_{q \in \bar{S}_m} \sigma_{f,q}^{-2} w_{1,q}^2 \]

\[ G_{12} = \sum_{i \in \bar{S}_c} \sigma_{c,i}^{-2} u_{1,i} u_{1,y} \]

\[ + \sum_{p \in \bar{S}_m} \sigma_{c,p}^{-2} u_{1,p} u_{1,y} \]

\[ + \sum_{i \in \bar{S}_b} \sigma_{f,i}^{-2} w_{1,i} w_{1,y} \]

\[ + \sum_{q \in \bar{S}_m} \sigma_{f,q}^{-2} w_{1,q} w_{1,y} \]

\[ G_{22} = \sum_{i \in \bar{S}_c} \sigma_{c,i}^{-2} u_{2,i}^2 \]

\[ + \sum_{p \in \bar{S}_m} \sigma_{c,p}^{-2} u_{2,p}^2 \]

\[ + \sum_{i \in \bar{S}_b} \sigma_{f,i}^{-2} w_{2,i}^2 \]

\[ + \sum_{q \in \bar{S}_m} \sigma_{f,q}^{-2} w_{2,q}^2 \] (13.6)
Inversion of $G$ leads to the following general expression of the accuracy measure:

$$M = \sqrt{\text{Tr}(Q_x^2)} = \sqrt{\frac{G_{11} + G_{22}}{\text{Det}(G)}}$$

(13.7)

where

$$G_{11} + G_{22} = \sum_{i \in S_c} \sigma_{e,i}^{-2} + \sum_{p \in S_m} \sigma_{e,p}^{-2} + \sum_{l \in S_h - \{1\}} \sigma_{f,l}^{-2} |w_{1l}|^2 + \sum_{q \in S_m - \{1\}} \sigma_{f,q}^{-2} |w_{1q}|^2$$

(13.8)

and

$$\text{Det}(G) = \left[ \sum_{i \in \tilde{S}_c} \sigma_{e,i}^{-2} u_{i,x}^2 + \sum_{p \in S_m} \sigma_{e,p}^{-2} u_{p,x}^2 + \sum_{l \in \tilde{S}_h - \{1\}} \sigma_{f,l}^{-2} w_{1l,x}^2 + \sum_{q \in \tilde{S}_m - \{1\}} \sigma_{f,q}^{-2} w_{1q,x}^2 \right]$$

$$\cdot \left[ \sum_{i \in \tilde{S}_c} \sigma_{e,i}^{-2} u_{i,y}^2 + \sum_{p \in S_m} \sigma_{e,p}^{-2} u_{p,y}^2 + \sum_{l \in \tilde{S}_h - \{1\}} \sigma_{f,l}^{-2} w_{1l,y}^2 + \sum_{q \in \tilde{S}_m - \{1\}} \sigma_{f,q}^{-2} w_{1q,y}^2 \right]$$

$$- \left[ \sum_{l \in \tilde{S}_c} \sigma_{e,l}^{-2} u_{l,x} u_{l,y} + \sum_{p \in S_m} \sigma_{e,p}^{-2} u_{p,x} u_{p,y} + \sum_{l \in \tilde{S}_h - \{1\}} \sigma_{f,l}^{-2} w_{1l,x} w_{1l,y} \right]$$

$$\cdot \left[ \sum_{j \in \tilde{S}_c} \sigma_{e,j}^{-2} u_{j,x} u_{j,y} + \sum_{r \in S_m} \sigma_{e,r}^{-2} u_{r,x} u_{r,y} \right]$$

$$+ \sum_{q \in \tilde{S}_m - \{1\}} \sigma_{f,q}^{-2} w_{1q,x} w_{1q,y}$$

$$+ \sum_{m \in \tilde{S}_h - \{1\}} \sigma_{f,m}^{-2} w_{1m,x} w_{1m,y}$$

(13.9)

The structure of $M$ shows that the location accuracy is affected by the measurement accuracy ($\sigma_{e,i}, \sigma_{f,i}$) and the reciprocal position of the MS and the BTSs (represented by the $x$-$y$ components of vectors $\mathbf{u}_i$ and $\mathbf{w}_{ij}$). These two contributions can be separated only if all the (uncorrelated) measurement errors have the same standard deviation.

### 13.1.6 Circular multilateration

In this case we have $S_h = S_m = \tilde{S}_h = 0$ and $\tilde{S}_c = S_c$. Using this in Equation (13.7) gives:

$$M_c^2 = \sum_{i,j \in S_c} \frac{\sigma_{e,i}^{-2} \sigma_{e,j}^{-2}}{\text{Det}(G_c)}$$

(13.10)

$\text{Det}(G_c)$ can be obtained from Equation (13.9) by retaining only the sums over indexes $i$ and $j$:

$$\text{Det}(G_c) = \sum_{i,j \in S_c} \frac{\sigma_{e,i}^{-2} \sigma_{e,j}^{-2}}{\text{Det}(G_c)}$$

(13.11)

The cross product between two vectors $\mathbf{a} = a_x \mathbf{x} + a_y \mathbf{y}$ and $\mathbf{b} = b_x \mathbf{x} + b_y \mathbf{y}$ can be written as $\mathbf{a} \times \mathbf{b} = (a_y b_x - a_x b_y) \mathbf{z}$ ($\mathbf{z} = \mathbf{x} \times \mathbf{y}$ is the unit vector in the $z$ direction). With this observation in mind, the quantity between parentheses in Equation (13.11) becomes $\text{Det}(G_c) = u_{i,x} u_{j,x} \mathbf{z} \cdot (\mathbf{u}_i \times \mathbf{u}_j)$, where $\cdot$ represents the scalar product. $\text{Det}(G_c)$ can be further modified by excluding from the double sum the terms occurring when $j = i$ since $\mathbf{u}_i \times \mathbf{u}_i = \mathbf{0}$. $S_c$ (as well as $S_h$ and $S_m$) is an ordered set of indexes; thus the formula $\sum_i \sum_{j \neq i} a_i b_j = \sum_i \sum_{j > i} (a_i b_j + a_j b_i)$, valid for $i$ and $j$ spanning the same set of
ordered values, can be used and \( \det(\mathbf{G_e}) \) becomes:

\[
\det(\mathbf{G_e}) = \sum_{i \in S_e} \sum_{j \in S_e, j > i} \sigma_{e,i}^{-2} \sigma_{e,j}^{-2} |\mathbf{u}_i \times \mathbf{u}_j|^2
\]  

(13.12)

An interpretation of \( \det(\mathbf{G_e}) \) in Equation (13.12) comes from the geometric definition of cross product (see Figure 13.2). In a polar reference system \((\rho, \theta)\) centered at the MS position, each unit vector \(\mathbf{u}_i\) can be expressed as \(\mathbf{u}_i = \cos \theta_i \mathbf{x} + \sin \theta_i \mathbf{y}\), with \(\theta_i\) being the angle of \(\mathbf{u}_i\) measured counterclockwise from \(\mathbf{x}\). By definition, \(|\mathbf{u}_i \times \mathbf{u}_j|^2 = (|\mathbf{u}_i||\mathbf{u}_j||\sin \theta_ij|)^2\), where \(\theta_{ij} = \theta_i - \theta_j\); thus \(|\mathbf{u}_i \times \mathbf{u}_j|^2\) is the area of the parallelogram determined by \(\mathbf{u}_i\) and \(\mathbf{u}_j\). \(A_{ij} = |\mathbf{u}_i||\mathbf{u}_j||\sin \theta_{ij}|\), raised to the second power. Introducing, \(A_{ij}\) in Equation (13.12), we have

\[
M_e^2 = \frac{\sum_{i \in S_e} \sum_{j \in S_e, j > i} \sigma_{e,i}^{-2} \sigma_{e,j}^{-2} |\mathbf{u}_i \times \mathbf{u}_j|^2}{\sum_{i \in S_e} \sum_{j \in S_e} \sigma_{e,i}^{-2} \sigma_{e,j}^{-2} A_{ij}^2}
\]  

(13.13)

Equation (13.13) shows the dependence of the positioning accuracy on the AD measurements’ accuracy \((\sigma_{e,i}, \sigma_{e,j})\) and on the geometric conditioning \((A_{ij})\). Only if \(\sigma_{e,i} = \sigma_{e,j} \Delta \sigma_{e}\) the two contributions can be separated and \(M_e = \sigma_{e} \text{ GDOP}_e\) can be written as a product of \(\sigma_{e}\) and the geometric dilution of precision GDOP_e:

\[
\text{GDOP}_e = \sqrt{N_e \sum_{i \in S_e} \sum_{j \in S_e, j > i} A_{ij}^2}
\]  

(13.14)

For a given \(N_e\), the location accuracy improves if GDOP_e is small, or equivalently if the areas \(A_{ij}\)s in the denominator of Equation (13.14) are large. Geometrically, this means that the unit vectors \(\mathbf{u}_i\), \(\mathbf{u}_j\), and then the BTSs, must be widely (angularly) separated with respect to the MS. Notice that since \(|\mathbf{u}_i| = 1\), \(A_{ij} = |\sin(\theta_i - \theta_j)|\), thus GDOP depends only on the angular distribution of the BTSs around the MS and not on the distance between MS and BTSs, which might affect \(\sigma_{e}\), for instance.

### 13.1.7 Hyperbolic multilateration

In this case we have \(S_m = S_e = \bar{S}_e = 0\) and \(\bar{S}_h = \bar{S}_h\) so that Equation (13.7) gives

\[
M_h^2 = \frac{\sum_{l \in S_h \setminus \{m\}} \sigma_{f,l}^{-2} |w_{1l}|^2}{\det(\mathbf{G}_h)}.
\]  

(13.15)

\(\det(\mathbf{G}_h)\) can be obtained from Equation (13.9) by retaining only the sums over indexes \(l\) and \(m\). Introducing the area of the parallelogram determined by \(w_{11}\) and \(w_{1m}, B_{lm}\) (see Figure 13.2), we have

\[
M_h^2 = \frac{\sum_{l \in S_h \setminus \{m\}} \sigma_{f,l}^{-2} |w_{1l}|^2}{\sum_{l \in S_h \setminus \{m\}} \sum_{m \in S_h \setminus \{l\}} \sigma_{f,l}^{-2} \sigma_{f,m}^{-2} |w_{1l} \times w_{lm}|^2}
\]  

(13.16)

If \(\sigma_{f,l} = \sigma_{f,m} \Delta \sigma_{f}\), we have \(M_h = \sigma_{f} \text{ GDOP}_h\) where

\[
\text{GDOP}_h = \sqrt{\frac{\sum_{l \in S_h \setminus \{m\}} \sigma_{f,l}^{-2} |w_{1l}|^2}{\sum_{l \in S_h \setminus \{m\}} \sum_{m \in S_h \setminus \{l\}} B_{lm}^2}}
\]  

(13.17)

is the GDOP for the hyperbolic multilateration.
In the denominator of Equation (13.20), the area \( A \) GDOP for the mixed multilateration can be defined only if \( AD \) and \( RD \) measurement errors have definitions of \( \theta \) and \( \pi \), and can be interpreted as follows: the condition giving GDOP \( \Delta \) can be derived from Equation (13.9) by retaining the sums over indexes \( l, m \) and by substituting \( p = r = 1 \). Introducing the definitions of \( A_{1m} \) and \( B_{im} \) as before, the expression for \( M_m^2 \), becomes

\[
M_m^2 = \frac{\sigma_{\epsilon,1}^{-2} + \sum_{l \in S_h - \{1\}} \sigma_{f,l}^{-2} \left| w_{1l} \right|^2}{\det(G_m)}
\]

(13.18)

because \( S_m = S_c = \{1\}, S_h = S_h - \{1\} \) and \( S_c = S_c - S_m = 0 \). \( \det(G_h) \) can be derived from Equation (13.9) by retaining the sums over indexes \( p, r \) and by substituting \( p = r = 1 \). Introducing the definitions of \( A_{1m} \) and \( B_{im} \) as before, the expression for \( M_m^2 \), becomes

\[
M_m^2 = \frac{\sigma_{\epsilon,1}^{-2} + \sum_{l \in S_h - \{1\}} \sigma_{f,l}^{-2} \left| w_{1l} \right|^2}{\sum_{l \in S_h - \{1\}} \sum_{m \neq l} \sigma_{f,m}^{-2} \left| w_{1l} \times w_{1m} \right|^2 + \sum_{m \in S_h - \{1\}} \sigma_{\epsilon,1}^{-2} \sigma_{f,m}^{-2} A_{1m}^2}
\]

(13.19)

A GDOP for the mixed multilateration can be defined only if \( AD \) and \( RD \) measurement errors have the same standard deviation (\( \sigma_{\epsilon,1} = \sigma_{f,i} = \sigma_m \)).

\[
\text{GDOP}_m = \left[ 1 + \sum_{l \in S_h - \{1\}} \left| w_{1l} \right|^2 \right] \frac{1}{\sum_{l \in S_h - \{1\}} \sum_{m \neq l} B_{lm}^2 + \sum_{m \in S_h - \{1\}} A_{1m}^2}
\]

(13.20)

In the denominator of Equation (13.20), the areas \( B_{lm} \) represent the RDs’ contributions, while the area \( A_{1m} \) represents the contribution of the AD from BTS1.

### 13.1.9 Performance results for three stations

When the minimum number of BTSs is available (\( N = 3 \)), the location techniques are named ‘trilaterations’. \( \text{GDOP}_c \), \( \text{GDOP}_h \) and \( \text{GDOP}_m \) can be expressed as follows:

\[
\text{GDOP}_c^2 = \frac{3}{\left| \mathbf{u}_1 \times \mathbf{u}_2 \right|^2 + \left| \mathbf{u}_1 \times \mathbf{u}_3 \right|^2 + \left| \mathbf{u}_2 \times \mathbf{u}_3 \right|^2} = \frac{3}{A_{12}^2 + A_{13}^2 + A_{23}^2}
\]

(13.14a)

\[
\text{GDOP}_h^2 = \frac{\left| w_{12} \right|^2 + \left| w_{13} \right|^2}{\left| w_{12} \times w_{13} \right|^2} = \frac{\left| w_{12} \right|^2 + \left| w_{13} \right|^2}{B_{23}^2}
\]

(13.17a)

\[
\text{GDOP}_m^2 = \frac{1 + \left| w_{12} \right|^2 + \left| w_{13} \right|^2}{\left| w_{12} \times w_{13} \right|^2 + \left| \mathbf{u}_1 \times \mathbf{u}_2 \right|^2 + \left| \mathbf{u}_1 \times \mathbf{u}_3 \right|^2} = \frac{1 + \left| w_{12} \right|^2 + \left| w_{13} \right|^2}{B_{23}^2 + A_{12}^2 + A_{13}^2}
\]

(13.20a)

and analyzed in the polar reference system (\( \rho, \theta \)) introduced earlier. Assuming \( \theta_1 = 0 \), the GDOPs become functions of \( \theta_2, \theta_3 \in [0, 2\pi] \) and can be plotted graphically. These plots can be used to determine under what conditions GDOP \( \rightarrow \infty \) (e.g., even with noiselless measurements, a unique solution of the problem does not exist) or GDOP = GDOP\(_{\text{min}} \) (e.g., the problem is optimally conditioned).

The results generalized to any \( \theta_1 \) value are summarized in Figures 13.3 and 13.4 and Table 13.1 and can be interpreted as follows:

In the case of circular trilateration: (i) \( \text{GDOP}_c \rightarrow \infty \) when \( \theta_2 \in \theta_1 + [0, \pi] \), and \( \theta_3 \in [\pi - \theta_2, \pi + \theta_2] \). These conditions are met when the three BTSs lie along a straight line passing through the MS (in fact, \( \mathbf{u}_1, \mathbf{u}_2, \) and \( \mathbf{u}_3 \) are parallel and all the cross products in the denominator of GDOP, are null). It can be easily shown, moreover, that the alignment between all the BTSs and the MS is the condition giving GDOP \( \rightarrow \infty \) in the general case with \( N \geq 3 \).

(ii) \( \text{GDOP}_c = \text{GDOP}_{c,\text{min}} = 1.15 \) when \( \theta_2 \in \theta_1 + [\pi/3, 2\pi/3, 4\pi/3, 5\pi/3] \) and \( \theta_3 \in [\pi - \theta_2, 2\pi - \theta_2, 3\pi - \theta_2] \), meaning that the BTSs are angularly separated from the MS viewpoint by
Figure 13.3 Infinite GDOP configurations when AT = 3, x, o, and Δ identify the BTSs and the center of the circumference the MS. (a) and (b) Circular, mixed, and hyperbolic trilateration (GDOP → ∞ if MS and BTSs are aligned; α can be either zero or π). (c) Hyperbolic trilateration (GDOP → ∞ if at least two BTSs are aligned with the MS; α can assume any value).

Figure 13.4 Minimum GDOP configurations when N = 3, x, o, and Δ identify the BTSs and the center of the circumference the MS. (a) and (b) Circular trilateration (GDOP_c = GDOP_{c, min} if α is either π/3 or 2π/3 rad). (c) Hyperbolic trilateration (GDOP_h = GDOP_{h, min} if α = 0.61π). (d) Mixed trilateration (GDOP_m = GDOP_{m, min} if α = 0.58π).

Table 13.1 Comparison of the GDOP for the trilateration techniques

<table>
<thead>
<tr>
<th>Conditions of GDOP → ∞</th>
<th>GDOP_{min} value</th>
<th>Conditions of GDOP = GDOP_{min}</th>
</tr>
</thead>
<tbody>
<tr>
<td>θ_2 ∈ θ_1 + [0, π]</td>
<td>1.15</td>
<td>θ_2 ∈ θ_1 + {π/3, 2π/3, 4π/3, 5π/3} and</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Circular trilateration</td>
<td>and θ_3 ∈ [θ_2 - π, θ_2, θ_2 + π]</td>
<td>and θ_3 ∈ [π - θ_2, 2π - θ_2, 3π - θ_2]</td>
</tr>
<tr>
<td></td>
<td>θ_1 = θ_2 or θ_1 = θ_3 or</td>
<td>θ_2 ∈ θ_1 + {0.61π, 1.39π} and</td>
</tr>
<tr>
<td></td>
<td>θ_2 = θ_3 or</td>
<td></td>
</tr>
<tr>
<td>Hyperbolic trilateration</td>
<td>0.92</td>
<td></td>
</tr>
<tr>
<td>Mixed trilateration</td>
<td>and θ_3 ∈ [θ_2 - π, θ_2, θ_2 + π]</td>
<td>and θ_3 = 2π - θ_2</td>
</tr>
<tr>
<td></td>
<td>0.88</td>
<td>θ_2 ∈ θ_1 + {0.58π, 1.42π} and</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
either $\pm 2\pi/3$ (for example, $\theta_1 = 0$, $\theta_2 = 2\pi/3$, $\theta_3 = -2\pi/3$) or $\pm \pi/3$ (for example, $\theta_1 = 0$, $\theta_2 = \pi/3$, $\theta_3 = -\pi/3$). In the first case, the BTSs are maximally separated; in the second case, they define the same parallelograms as if they were separated by $\pm 2\pi/3$.

In the case of hyperbolic trilateration: (i) $\text{GDOP}_h \to \infty$ when at least two out of the three angles $\theta_1$, $\theta_2$, $\theta_3$ are equal. In these conditions, at least one vector among $w_{12}$ and $w_{13}$ is null and $B_{23} = 0$. Geometrically, the MS is the origin of half a straight line passing through at least two of the three BTSs (see Figure 13.3). In the general scheme with $N \geq 3$, $\text{GDOP}_h \to \infty$ when the MS is at the origin of half a straight line passing through at least $N$ BTSs. (ii) Numerical minimization of $\text{GDOP}_h$ results in two minima ($\text{GDOP}_{h,\text{min}} \simeq 0.92$) verified when $\theta_2 \in \theta_1 + (0.61\pi, 1.39\pi)$ and $\theta_3 = 2\pi - \theta_2$.

Finally in the case of mixed trilateration: (i) $\text{GDOP}_m \to \infty$ in the same conditions as $\text{GDOP}_c$. (ii) Numerical minimization of $\text{GDOP}_m$ results in two minima ($\text{GDOP}_{m,\text{min}} \simeq 0.88$) verified when $\theta_2 \in \theta_1 + (0.58\pi, 1.42\pi)$ and $\theta_3 = 2\pi - \theta_2$.

### 13.1.10 Performance results for $N$ stations

The combined effect of number and geographical distribution of the BTSs on the GDOP is analyzed in the regular network of hexagonal omnidirectional cells. Two relevant cases are considered. In the first one, the MS is located in a central cell; thus no particular restriction on the geometric conditioning is expected to be introduced by the BTS distribution. In the second case, the MS is located in a border cell of the network where the distribution of the available BTSs introduces serious limitations on the geometric conditioning of the problem. For both cases, the cell of the network where the distribution of the available BTSs introduces serious limitations on the geometric conditioning. In the second case, the MS is located in a border cell of the general scheme.

Table 13.2 reports maximum, minimum, and mean values of $\text{GDOP}_c$, $\text{GDOP}_h$, and $\text{GDOP}_m$ when the MS is located in the central cell. Analogous values for the MS located in the border cell are listed in Table 13.3 [43].

Results in Table 13.1 prove that for $N = 3$ and a given value of $\theta_1$, $\text{GDOP}_b \to \infty$ in an uncountable number of cases while $\text{GDOP}_c$ and $\text{GDOP}_m$ go to infinity in only four cases. Table 13.3 shows that when the MS is in the border cell, $\text{GDOP}_h$ has maximum and mean values always greater than 1 and well above the corresponding values of $\text{GDOP}_c$ and $\text{GDOP}_m$, which in turn are comparable ($\text{GDOP}_{h,\text{mean}} \geq \text{GDOP}_{m,\text{mean}} > \text{GDOP}_{c,\text{mean}}$; $\text{GDOP}_{h,\text{max}} \gg \text{GDOP}_{m,\text{max}} \gg \text{GDOP}_{c,\text{max}}$). From the robustness point of view (of particular interest when $N = 3$ or when the geometry of the problem is poor), the hyperbolic multilateration is thus the least reliable technique because a low number of BTSs and a poor geometric conditioning offer the highest GDOP values. Moreover, when $N = 3$, it is enough that two BTSs out of three are aligned with the MS to make $\text{GDOP}_h$ infinite. The circular technique, on the other hand, is the most robust technique as its GDOP has the lowest values when the geometry of the problem is poor. When $N = 3$, $\text{GDOP}_c \to \infty$ only when the three BTSs are

<table>
<thead>
<tr>
<th>$N$</th>
<th>$\text{GDOP}_c$</th>
<th>$\text{GDOP}_h$</th>
<th>$\text{GDOP}_m$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>min.</td>
<td>max.</td>
<td>mean</td>
</tr>
<tr>
<td>3</td>
<td>1.15</td>
<td>1.53</td>
<td>1.24</td>
</tr>
<tr>
<td>4</td>
<td>1.00</td>
<td>1.03</td>
<td>1.01</td>
</tr>
<tr>
<td>5</td>
<td>0.89</td>
<td>0.91</td>
<td>0.91</td>
</tr>
<tr>
<td>6</td>
<td>0.82</td>
<td>0.83</td>
<td>0.82</td>
</tr>
<tr>
<td>7</td>
<td>0.76</td>
<td>0.77</td>
<td>0.76</td>
</tr>
<tr>
<td>8</td>
<td>0.71</td>
<td>0.73</td>
<td>0.72</td>
</tr>
<tr>
<td>9</td>
<td>0.67</td>
<td>0.68</td>
<td>0.68</td>
</tr>
<tr>
<td>10</td>
<td>0.63</td>
<td>0.66</td>
<td>0.66</td>
</tr>
</tbody>
</table>

© IEEE 2001
Table 13.3 GDOP$_c$, GDOP$_h$, and GDOP$_m$ when the MS is located in the border cell [43]

<table>
<thead>
<tr>
<th>$N$</th>
<th>min.</th>
<th>max.</th>
<th>mean</th>
<th>min.</th>
<th>max.</th>
<th>mean</th>
<th>min.</th>
<th>max.</th>
<th>mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>1.15</td>
<td>1.51</td>
<td>1.23</td>
<td>0.95</td>
<td>8.81</td>
<td>2.44</td>
<td>0.93</td>
<td>1.58</td>
<td>1.34</td>
</tr>
<tr>
<td>4</td>
<td>1.00</td>
<td>1.35</td>
<td>1.07</td>
<td>0.81</td>
<td>5.23</td>
<td>2.11</td>
<td>0.79</td>
<td>0.57</td>
<td>1.21</td>
</tr>
<tr>
<td>5</td>
<td>0.89</td>
<td>1.22</td>
<td>0.95</td>
<td>0.71</td>
<td>4.63</td>
<td>1.99</td>
<td>0.70</td>
<td>1.44</td>
<td>1.13</td>
</tr>
<tr>
<td>6</td>
<td>0.82</td>
<td>1.13</td>
<td>0.86</td>
<td>0.66</td>
<td>4.54</td>
<td>1.76</td>
<td>0.65</td>
<td>1.42</td>
<td>1.05</td>
</tr>
<tr>
<td>7</td>
<td>0.76</td>
<td>0.98</td>
<td>0.80</td>
<td>0.60</td>
<td>4.16</td>
<td>1.54</td>
<td>0.59</td>
<td>1.30</td>
<td>0.99</td>
</tr>
<tr>
<td>8</td>
<td>0.71</td>
<td>0.95</td>
<td>0.75</td>
<td>0.56</td>
<td>3.36</td>
<td>1.44</td>
<td>0.56</td>
<td>1.28</td>
<td>0.94</td>
</tr>
<tr>
<td>9</td>
<td>0.67</td>
<td>0.86</td>
<td>0.70</td>
<td>0.55</td>
<td>3.11</td>
<td>1.38</td>
<td>0.55</td>
<td>1.23</td>
<td>0.91</td>
</tr>
<tr>
<td>10</td>
<td>0.64</td>
<td>0.83</td>
<td>0.67</td>
<td>0.52</td>
<td>3.03</td>
<td>1.33</td>
<td>0.52</td>
<td>1.2</td>
<td>0.89</td>
</tr>
</tbody>
</table>

aligned. In this sense, circular and mixed techniques are equivalent (GDOP$_m \rightarrow \infty$ under the same conditions as GDOP$_c$).

For $N = 3$ and a fixed value of $\theta_1$, GDOP$_c$ has eight minima, while GDOP$_h$ and GDOP$_m$ have only two minima; however, GDOP$_{m,min} < GDOP_{h,min} < 1 < GDOP_{c,min}$, meaning that when $N = 3$, the accuracy for the circular trilateration is always worse than the AD measurements’ accuracy ($M_c = \sigma_c \cdot GDOP_c > \sigma_c$). Analogous conclusions can be drawn from Table 13.2: given $N$, minimum, mean, and maximum values of GDOP calculated when the MS in the central cell satisfy the inequality GDOP$_{m} < GDOP_{h} < GDOP_{c}$. This simply means that when the geometric conditioning is not problematic, the mixed and hyperbolic techniques are the most accurate while the circular technique has the highest GDOP values.

Based on the GDOP analysis above, the mixed multilateration turns out to be the location estimation technique that, among those considered in this section, offers the best compromise between robustness and accuracy.

### 13.2 RELATIVE POSITIONING IN WIRELESS SENSOR NETWORKS

In this section we discuss self-configuration in wireless sensor networks as a general class of estimation problem via the Cramér–Rao bound (CRB). Specifically, we consider sensor location estimation when sensors measure received signal strength (RSS) or time-of-arrival (TOA) between themselves and neighboring sensors. A small fraction of sensors in the network have a known location, whereas the remaining locations must be estimated. CRBs and maximum-likelihood estimators (MLEs) under Gaussian and log-normal models for the TOA and RSS measurements, are derived respectively.

We assume a network in which a small proportion of devices, called reference devices, have a priori information about their coordinates. All devices, regardless of their absolute coordinate knowledge, estimate the range between themselves and their neighboring devices. Such location estimation is called ‘relative location’ because the range estimates collected are predominantly between pairs of devices of which neither has absolute coordinate knowledge. These devices without a priori information we call blindfolded devices. In cellular location estimation systems, described in the previous section, location estimates are made using only ranges between a blindfolded device and reference devices. Relative location estimation requires simultaneous estimation of multiple device coordinates. Greater location estimation accuracy can be achieved as devices are added into the network, even when new devices have no a priori coordinate information and range to just a few neighbors.

Emerging applications for wireless sensor networks will depend on automatic and accurate location of thousands of sensors. In environmental sensing applications ‘sensing data without knowing the
sensor location is meaningless’. In addition, relative location estimation may enable applications such as inventory management, intrusion detection, traffic monitoring, and locating emergency workers in buildings.

To design a relative location system that meets the needs of these applications, several capabilities are necessary. The system requires a network of devices capable of peer-to-peer range measurement, an ad-hoc networking protocol, and a distributed or centralized location estimation algorithm. For range measurement, using received signal strength (RSS) is attractive from the point of view of device complexity and cost but is traditionally seen as a coarse measure of range. Instead, time-of-arrival (TOA) range measurement can be used. In this section, we will show that both RSS and TOA measurements can lead to accurate location estimates in dense sensor networks.

### 13.2.1 Performance bounds

In network self-calibration problems, parameters of all devices in a network must be determined. Information comes both from measurements made between pairs of devices and a subset of devices that know a priori their parameters. A network self-calibration estimator estimates the unknown device parameters. For example, distributed clock synchronization in a network could be achieved by devices observing pair-wise timing offsets when just a small number of devices are synchronous. For details of mutual node synchronization in wireless networks see Chapter 13 in the previous edition of this book [45].

Specifically, consider a vector of device parameters \( \mathbf{y} = [\gamma_1, \ldots, \gamma_n, \ldots, \gamma_{n+m}] \). Each device has one parameter. Devices 1 \( \ldots \) \( n \) are blindfolded devices, and devices \( n + 1 \ldots n + m \) are reference devices. The unknown parameter vector is \( \mathbf{\theta} = [\theta_1, \ldots, \theta_n] \), where \( \theta_i = \gamma_i \), for \( i = 1 \ldots n \). Note that \( \{\gamma_i, i = n + 1 \ldots n + m\} \) are known. Devices \( i \) and \( j \) make pair-wise observations \( X_{i,j} \) with density \( f_{X_{ij}}(X_{i,j}/\gamma_i, \gamma_j) \). We include also the case when devices make incomplete observations since two devices may be out of range or have limited link capacity. Let \( H(i) = \{j: device \ j \ makes \ pair-wise \ observations \ with \ device \ i\} \). By convention, a device cannot make a pair-wise observation with itself, so that \( i \notin H(i) \). By symmetry, if \( j \in H(i) \), then \( i \in H(j) \).

We assume by reciprocity that \( X_{i,j} = X_{j,i} \); thus, it is sufficient to consider only the lower triangle of the observation matrix \( \mathbf{X} = (X_{i,j})_{i<j} \) when formulating the joint likelihood function. In practice, if it is possible to make independent observations on the links from \( i \) to \( j \) and from \( j \) to \( i \), then we assume that a scalar sufficient statistic can be found. Finally, we assume that \( \{X_{i,j}\} \) are statistically independent for \( j < i \). This assumption can be somewhat oversimplified for the RSS case but necessary for analysis. The log of the joint conditional pdf is

\[
I(\mathbf{X}/\mathbf{y}) = \sum_{i=1}^{n+m} \sum_{\substack{j \in H(i) \\cap j < i}} \left( I_{i,j} \right) = \sum_{i=1}^{n+m} \sum_{\substack{j \in H(i) \\cap j < i}} \log f_{X_{ij}}(X_{i,j}/\gamma_i, \gamma_j). \tag{13.21}
\]

The MLE will produce the vector \( \hat{\mathbf{\theta}} = \max_{\mathbf{\theta}} I(\mathbf{X}/\mathbf{y}) \). The CRB on the covariance matrix of any unbiased estimator \( \hat{\mathbf{\theta}} \), including MLE estimator, is \( \text{cov}(\hat{\mathbf{\theta}}) \geq \mathbf{F}_\theta^{-1} \), where the Fisher information matrix (FDM) \( \mathbf{F}_\theta \) is defined as

\[
\mathbf{F}_\theta = -\mathbf{E}_{\mathbf{\theta}}(\nabla_{\mathbf{\theta}} I(\mathbf{X}/\mathbf{y}))^T = \begin{bmatrix} f_{1,1} & \cdots & f_{1,n} \\ \vdots & \ddots & \vdots \\ f_{n,1} & \cdots & f_{n,n} \end{bmatrix}. \tag{13.22}
\]

The diagonal elements \( f_{k,k} \) of \( \mathbf{F} \) given in Equation (13.22) are:

\[
f_{k,k} = E \left( \frac{\partial}{\partial \theta_k} l(\mathbf{X}/\mathbf{\theta}) \right)^2 = E \left( \sum_{j \in H(k)} \left( \frac{\partial}{\partial \theta_k} l_{k,j} \right)^2 \right)
\]

\[
f_{k,k} = \sum_{j \in H(k)} \sum_{p \in H(k)} E \left( \frac{\partial}{\partial \theta_k} l_{k,j} \right) \left( \frac{\partial}{\partial \theta_k} l_{k,p} \right)
\]

The diagonal elements \( f_{k,k} \) of \( \mathbf{F} \) are necessary. The system requires a network of devices capable of peer-to-peer range measurement, an ad-hoc networking protocol, and a distributed or centralized location estimation algorithm. For range measurement, using received signal strength (RSS) is attractive from the point of view of device complexity and cost but is traditionally seen as a coarse measure of range. Instead, time-of-arrival (TOA) range measurement can be used. In this section, we will show that both RSS and TOA measurements can lead to accurate location estimates in dense sensor networks.
Since \( X_{k,j} \) and \( X_{k,p} \) are independent random variables, and \( E \left( \frac{\partial}{\partial \theta_i} l_{k,j} \right) = 0 \), the expectation of the product is only nonzero for \( p = j \). Thus, \( f_{k,k} \) simplifies to the \( k = l \) result in Equation (13.23). The off-diagonal elements similarly simplify:

\[
f_{k,l} = \sum_{j \in H(k)} \sum_{p \in H(l)} E \left( \frac{\partial}{\partial \theta_k} l_{k,j} \right) \left( \frac{\partial}{\partial \theta_l} l_{l,p} \right)
\]

Here, due to independence and zero mean of the two terms, the expectation of the product will be zero unless both \( p = k \) and \( j = l \). Thus, the \( k \neq l \) result in Equation (13.23):

\[
f_{k,l} = \begin{cases} 
- \sum_{j \in H(k)} E \left( \frac{\partial^2}{\partial \theta_k \partial \theta_l} l_{k,j} \right), & k = l \\
-I_{H(k)}(l) E \left( \frac{\partial^2}{\partial \theta_l \partial \theta_l} l_{k,l} \right), & k \neq l 
\end{cases}
\tag{13.23}
\]

where \( I_{H(k)}(l) \) is an indicator function: 1 if \( l \in H(k) \) or 0 otherwise.

Intuitively, as more devices are used for location estimation, the accuracy increases for all the devices in the network. For an \( n \)-device network, there are \( O(n) \) parameters but \( O(n^2) \) variables \( \{X_{i,j}\} \) used for their estimation. The remaining question is, what are sufficient conditions to ensure the CRB decreases as devices are added to the network? For a network of \( n \) blindfolded devices and \( m \) reference devices, consider adding one additional blindfolded device. For the \( n \) and \( (n + 1) \) blindfolded device cases, let \( F \) and \( G \) be the FIMs defined in Equation (13.22), respectively. If \( [G^{-1}]_{ul} \) is the upper left \( n \times n \) block of \( G^{-1} \) and if for the \( (n + 1) \) blindfolded device case

- **Condition (i):** \( (\partial / \partial \theta_{n+1}) l_{k,n+1} = \pm (\partial / \partial \theta_i) l_{k,n+1}, \forall k = 1 \ldots n \)
- **Condition (ii):** device \( n + 1 \) makes pair-wise observations between itself and at least one blindfolded device and at least two devices in total; then it can be shown that two properties hold:
  - **Property (i):** \( F^{-1} - [G^{-1}]_{ul} \geq 0 \) in the positive semi-definite sense, and
  - **Property (ii):** \( \text{tr} F^{-1} > \text{tr} [G^{-1}]_{ul} \).

The Gaussian and log-normal distributions used in the next section meet condition (i). Property (i) implies that the additional unknown parameter introduced by the \( (n + 1) \)st blindfolded device does not impair the estimation of the original \( n \) unknown parameters. Furthermore, property (ii) implies that the sum of the CRB variance bounds for the \( n \) unknown parameters strictly decreases. Thus, when a blindfolded device enters a network and makes pair-wise observations with at least one blindfolded device and at least two devices in total, the bound on the average variance of the original \( n \) coordinate estimates is reduced.

To prove the above properties, compare \( F \), which is the FIM for the \( n \) blindfolded device problem, to \( G \), which is the FIM for the \( n+1 \) blindfolded device case. Partition \( G \) into blocks:

\[
G = \begin{bmatrix} G_{ul} & g_{ul} \\ g_{lu} & g_{lr} \end{bmatrix}
\]

where \( G_{ul} \) is an \( n \times n \) matrix, \( g_{lr} \) is the scalar Fisher information for \( \theta_{n+1} \), and \( g_{ul} = g_{lu}^T \) are \( n \times 1 \) vectors with \( k \)th element:

\[
g_{ur}(k) = I_{H(n+1)}(k) E \left( \frac{\partial}{\partial \theta_k} l_{n+1}^{n+1} \right) \left( \frac{\partial}{\partial \theta_{n+1}} l_{n,k}^{n+1} \right)
\]

\[
g_{lr} = \sum_{j \in H(n+1)} E \left( \frac{\partial}{\partial \theta_k} l_{n+1}^{n+1} \right)^2
\]

Here, we denote the log-likelihood of the observation between devices \( i \) and \( j \) in Equation (13.21) as \( l_{n,i}^{n+1} \) and \( l_{n+1,j}^{n+1} \) for the \( n \) and \( (n + 1) \) blindfolded device cases, respectively. If \( P(X|\gamma_n) \) and \( P^{n+1}(X|\gamma_{n+1}) \) are the joint log-likelihood function in Equation (13.21) for the \( n \) and \( n + 1 \) blindfolded device cases,
respectively, then we have: 

\[ l^{n+1}(X/Y_{n+1}) = \sum_{i=1}^{m+n+1} \sum_{j \in i} l^{n+1}_{i,j} = l^n(X/Y_n) + \sum_{j \in H(n+1)} l^{n+1}_{n+1,j} \]

Since \( l^{n+1}_{n+1,j} \) is a function only of parameters \( \gamma_{n+1} = \theta_{n+1} \) and \( \gamma_j \), then

\[
\frac{\partial^2}{\partial \theta_k \partial \theta_j} \sum_{j \in H(n+1)} l^{n+1}_{n+1,j} = \begin{cases} 
I_{H(n+1)}(k) \frac{\partial}{\partial \theta_k} l^{n+1}_{n+1,k}, & l = k \\
0, & l \neq k.
\end{cases}
\]

Thus, \( G_{nl} = F + \text{diag}(h) \), where \( h = \{ h_1, \ldots, h_n \} \), and \( h_k = I_{H(n+1)}(k) \text{E}((\partial/\partial \theta_k) l^{n+1}_{n+1,k})^2 \). Compare the CRB for the covariance matrix of the first \( n \) devices in the \( n \) and \( n + 1 \) device cases, given by \( F^{-1} \) and \( [G^{-1}]_{nl} \), respectively. Here, \([G^{-1}]_{nl}\) is the upper left \( n \times n \) submatrix of \( G^{-1} \):

\[
\left[ G^{-1} \right]_{nl} = \left[ G_{nl} - g_{ur} g_{lr}^{-1} g_{lr} \right]^{-1} = (F + J)^{-1}
\]

where \( J = \text{diag}(h) - g_{ur} g_{lr}^{-1} g_{lr} \). Both \( F \) and \( J \) are Hermitian. We know that \( F \) is positive semidefinite. Let \( \lambda_k(F), k = 1 \cdots n \) be the eigenvalues of \( F \) and \( \lambda_k(F + J), k = 1 \cdots n \) be the eigenvalues of the sum, both listed in increasing order. Then, if we can show that \( J \) is positive semidefinite, then it is known [46] that

\[
0 \leq \lambda_k(F + J) \leq \lambda_k(F), \quad \forall k = 1, \ldots, n \quad (13.24)
\]

Since the eigenvalues of a matrix inverse are the inverses of the eigenvalues of the matrix

\[
\lambda_k \left( (F + J)^{-1} \right) \leq \lambda_k(F^{-1}), \quad \forall k = 1, \ldots, n.
\]

which proves Property (i). If we can show that \( \text{tr}(J) > 0 \), then \( \text{tr}(F + J) > \text{tr}(F) \), and therefore, sum \( \sum_{k=1}^{n} \lambda_k(F + J) > \sum_{k=1}^{n} \lambda_k(F) \). This, with (13.24), implies that \( \lambda_j(F + J) > \lambda_j(F) \) for at least one \( j \in 1 \cdots n \). So, in addition to (13.25) \( \lambda_j(F + J)^{-1} < \lambda_j(F^{-1}) \), for some \( j \in 1, \ldots, n \), that implies \( \text{tr}(F + J)^{-1} < \text{tr}(F^{-1}) \) which proves Property (ii).

To show positive semidefiniteness and positive trace of \( J \) we notice that the diagonal elements of \( J \), \([J]_{k,k}\) are

\[
[J]_{k,k} = h_k - g_{ur}^{-1}(k) g_{lr}.
\]

(13.26)

If \( k \notin H(n+1) \), then \( h_k = 0 \) and \( g_{ur}(k) = 0 \); thus, \([J]_{k,k} = 0 \). Otherwise, if \( k \in H(n+1) \)

\[
[J]_{k,k} = E \left( \frac{\partial l^{n+1}_{n+1,k}}{\partial \theta_k} \right)^2 - E \left( \frac{\partial l^{n+1}_{n+1,k}}{\partial \theta_k} \right) \left( \frac{\partial l^{n+1}_{n+1,k}}{\partial \theta_{n+1}} \right) \sum_{j \in H(n+1)} E \left( \frac{\partial l^{n+1}_{n+1,j}}{\partial \theta_{n+1}} \right)^2
\]

Because of reciprocity, the numerator is equal to the square of the \( j = k \) term in the sum in the denominator. Thus:

\[
[J]_{k,k} \geq E \left( \frac{\partial l^{n+1}_{n+1,k}}{\partial \theta_k} \right)^2 - E \left( \frac{\partial l^{n+1}_{n+1,k}}{\partial \theta_k} \right) \left( \frac{\partial l^{n+1}_{n+1,k}}{\partial \theta_{n+1}} \right) = 0
\]

The equality will hold if \( k \) is the only member of the set \( H(n+1) \). When Condition (ii) holds, \([J]_{k,k} \) will be strictly greater than zero. Thus, \( \text{tr} J > 0 \). Next, we show that \( J \) is diagonally dominant [46], i.e.

\[
[J]_{k,k} \geq \sum_{j=1}^{n} \left| [J]_{k,j} \right| = \sum_{j=1}^{n} \left| g_{ur}(k) g_{lr}(j) \right| \sum_{j=1}^{n} \left| g_{lr}(j) \right| / g_{lr}
\]
where \([J]_{k,k}\) is given in Equation (13.26). Since \(H(n+1) \neq 0\), thus, \(g_{\ell} > 0\), and an equivalent condition is

\[
g_{\ell} h_k \geq |g_{\text{ar}}(k)| \sum_{j=1}^{n} |g_{\text{ar}}(j)| \tag{13.27}
\]

If \(k \notin H(n+1)\), then \(h_k = 0\) and \(g_{\text{ar}}(k) = 0\) and equality holds. If \(k \in H(n+1)\), then

\[
g_{\ell} h_k = E \left( \left| \frac{\partial t_{k,n+1}^{n+1}}{\partial \theta_k} \right| \right)^2 \sum_{j \in H(n+1)} E \left( \left| \frac{\partial t_{j,n+1}^{n+1}}{\partial \theta_{n+1}} \right| \right)^2
\]

Due to Condition (i)

\[
E \left( \left| \frac{\partial t_{k,n+1}^{n+1}}{\partial \theta_k} \right| \right)^2 = \left| E \left( \frac{\partial t_{k,n+1}^{n+1}}{\partial \theta_{n+1}} \right) E \left( \frac{\partial t_{j,n+1}^{n+1}}{\partial \theta_{j}} \right) \right|
\]

Thus

\[
g_{\ell} h_k = |g_{\text{ar}}(k)| \left[ \sum_{j=1 \atop j \notin H(n+1)}^{n} |g_{\text{ar}}(j)| + \sum_{j<0 \atop j \notin H(n+1)}^{n} E \left( \frac{\partial t_{k,n+1}^{n+1}}{\partial \theta_{n+1}} \frac{\partial t_{j,n+1}^{n+1}}{\partial \theta_{j}} \right) \right]
\]

Since \(g_{\text{ar}}(j) = 0\) if \(j \notin H(n+1)\), we can include in the first sum all \(j \in 1 \ldots n\). Since the second sum is \(\geq 0\), (13.27) is true.

Diagonal dominance implies that \(J\) is positive semidefinite, which proves (13.25). Note that if \(H(n+1)\) includes \(\geq 1\) reference device, the second sum is \(>0\), and the inequality in (13.27) is strictly \(>0\), which implies positive definiteness of \(J\) and assures that the CRB will strictly decrease.

### 13.2.2 Relative location estimation

Here we use the general framework from the previous section for device location estimation using pair-wise RSS or TOA measurements in a wireless network. For a network of \(m\) reference and \(n\) blindfolded devices, the device parameters are \(\gamma = [z_i, \ldots, z_{n+m}]\), where, for a two-dimensional (2D) system, \(z_i = [x_i, y_i]^T\). An extension of these results to 3D is also possible.

The relative location problem corresponds to the estimation of blindfolded device coordinates \(\theta = [\theta_s, \theta_j]\) (where \(\theta_s = [x_s, \ldots, x_n], \theta_j = [y_1, \ldots, y_n]\)) given the known reference coordinates \([x_{n+1}, \ldots, x_{n+m}, y_{n+1}, \ldots, y_{n+m}]\). In the TOA case, \(X_{i,j} = T_{i,j}\) is the measured TOA between devices \(i\) and \(j\) in (seconds), and in the RSS case, \(X_{i,j} = P_{i,j}\) is the measured received power at device \(i\) transmitted by device \(j\) (in milliwatts). As discussed in the previous section, only a subset \(H(k)\) of devices make pair-wise measurements with device \(k\), \((T_{i,j})_{i,j} , \text{ and } (P_{i,j})_{i,j}\) are taken to be upper triangular matrices, and these measurements are assumed statistically independent. In addition, assume that \(T_{i,j}\) is Gaussian distributed with mean \(d_{i,j}/c\) and variance \(\sigma_i^2\), which is denoted:

\[
T_{i,j} \sim N(d_{i,j}/c, \sigma_i^2), \quad d_{i,j} = d(x_i, z_j) = \|x_i - z_j\|^{1/2} \tag{13.28}
\]

where \(c\) is the speed of propagation, and \(\sigma_i^2\) is not a function of \(d_{i,j}\). We assume that \(P_{i,j}\) is log-normal; thus, the random variable \(P_{i,j}\) (dBm) = 10\log_{10} P_{i,j}\) is Gaussian:

\[
\frac{P_{i,j} \text{(dBm)}}{10n_p \log_{10}(d_{i,j}/d_0)} = \frac{\bar{P}_{i,j} \text{(dBm)} + \sigma_{\text{dB}}^2}{\bar{P}_{i,j} \text{(dBm)}} = P_0 \text{(dBm)} - 10n_p \log_{10}(d_{i,j}/d_0) \tag{13.29}
\]

where \(\bar{P}_{i,j}\) (dBm) is the mean power in decibel milliwatts, \(\sigma_{\text{dB}}^2\) is the variance of the shadowing, and \(P_0\) (dBm) is the received power in decibel milliwatts at a reference distance \(d_0\). Typically, \(d_0 = 1\) m, and \(P_0\) is calculated from the free space path loss formula. The path loss exponent \(n_p\) is a function of the environment. For particular environments, \(n_p\) may be known from prior measurements. Although we derive the CRB assuming \(n_p\) is known, it could have been handled as an unknown ‘nuisance’ parameter.
Given (13.28), the density of $P_{i,j}$ is

$$f_{P_i}(P_{i,j}/\gamma) = \frac{10 \log 10}{\sqrt{2\pi} \sigma_{\text{db}}^2} \frac{1}{P_{i,j}} \exp \left[ -\frac{b}{8} \left( \log \frac{d_{i,j}^2}{\hat{d}_{i,j}} \right)^2 \right]$$

$$b = \left( \frac{10 n_p}{\sigma_{\text{db}}^2 \log 10} \right)^2; \quad \hat{d}_{i,j} = d_0 \left( \frac{P_0}{P_{i,j}} \right)^{1/n_p}$$  \hspace{1cm} (13.30)

Here, $\hat{d}_{i,j}$ is the MLE of range $d_{i,j}$ given received power $P_{i,j}$.

For simplicity let us first consider one-dimensional TOA example that could be applied to location estimation on an assembly line. Consider $n$ blindfolded devices and $m$ reference devices with combined parameter vector $\gamma = [x_1, \ldots, x_{m+n}]$. The unknown coordinate vector is $\theta = [x_1, \ldots, x_n]$. Assume all devices make pair-wise measurements with every other device, i.e. $H(k) = \{1, \ldots, k - 1, k + 1, \ldots, m + n\}$. The distribution of the observations is given by (13.28) with $d_{i,j} = |x_i - x_j|$ and $l_{i,j} = (\partial^2 / \partial x_i^2) \log 10$, $\log 10 ((\partial^2 / \partial x_i \partial x_j) \log 10) = -1/\sigma_{\text{db}}^2$, $\forall i \neq j$, which are constant with respect to the random variables $T_{i,j}$. Thus, the FIM, which is calculated using Equation (13.23), is $F = (n + m)I_n - 11^2 / (\sigma_{\text{db}}^2)^2$, where $I_n$ is the $n \times n$ identity matrix, and $I$ is an $n$ by $1$ vector of ones. For $m \geq 1$, $F^{-1} = \sigma_{\text{db}}^2 c^2 (mI_n + 11^2) / (m(n + m))$.

The CRB on the variance of an unbiased estimator for $x_i$ is:

$$\sigma_{\text{db}}^2 x_i = \sigma_{\text{db}}^2 (m + 1) / (m(n + m)) \cdot$$  \hspace{1cm} (13.31)

Equation (13.31) implies that the variance $\sigma_{\text{db}}^2 x_i$ is reduced more quickly by adding reference $(m)$ than blindfolded $(n)$ devices. However, if $m$, is large, the difference between increasing $m$ and $n$ is negligible.

In the case of two-dimensional location estimation we denote by $F_R$ and $F$, the FIMs for the RSS and TOA measurements, respectively. Each device has two parameters, and we can see that the FIM $F$ is the MLE $d_{i,j}$ of range $d_{i,j}$ and $\log_{10} (x_i - x_j)^2 + (y_i - y_j)^2$. We have

$$l_{i,j} = \log \left( \frac{10 \log 10}{\sqrt{2\pi} \sigma_{\text{db}}^2} \frac{1}{P_{i,j}} \right) - \frac{b}{8} \left( \log \frac{d_{i,j}^2}{\hat{d}_{i,j}} \right)^2 \cdot$$  \hspace{1cm} (13.33)

Since $d_{i,j} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$ we have

$$\frac{\partial}{\partial x_i} l_{i,j} = -\frac{b}{2} \left( \log \frac{d_{i,j}^2}{\hat{d}_{i,j}} \right) \frac{x_i - x_j}{d_{i,j}^2}$$  \hspace{1cm} (13.34)

Since $(\partial / \partial x_j) l_{i,j} = -(\partial / \partial x_j) l_{i,j}$, the log-normal distribution of RSS measurements meets Condition (i). The second partials differ based on whether or not $i = j$ and if the partial is taken w.r.t. $y_i$ or $x_i$. For example:

$$\frac{\partial^2 l_{i,j}}{\partial x_i \partial y_j} = -\frac{b}{2} \frac{(x_i - x_j)(y_i - y_j)}{d_{i,j}^2} \left[ -\log \left( \frac{d_{i,j}^2}{\hat{d}_{i,j}} \right) + 1 \right]$$

$$\frac{\partial^2 l_{i,j}}{\partial x_j \partial y_i} = -\frac{b}{2} \frac{(x_i - x_j)(y_i - y_j)}{d_{i,j}^2} \left[ -\log \left( \frac{d_{i,j}^2}{\hat{d}_{i,j}} \right) - 1 \right]$$  \hspace{1cm} (13.35)
Since \( E[\log(d_{i,j}^2/\tilde{d}_{i,j}^2)] = 0 \), the FIM simplifies to take the form:

\[
[F_{Rxx}]_{k,l} = \begin{cases} 
  b \sum_{i \in H(k)} \frac{(y_i - y_j)^2}{\|x_i - x_l\|^2}, & k = l \\
  -b I_{H(k)}(l) \frac{(y_i - y_j)^2}{\|x_i - x_l\|^2}, & k \neq l
\end{cases}
\]

\[
[F_{Rxy}]_{k,l} = \begin{cases} 
  b \sum_{i \in H(k)} \frac{(y_i - x_j)(y_i - x_j)}{\|x_i - x_l\|^2}, & k = l \\
  -b I_{H(k)}(l) \frac{(y_i - x_j)(y_i - x_j)}{\|x_i - x_l\|^2}, & k \neq l
\end{cases}
\]

\[
[F_{Ryy}]_{k,l} = \begin{cases} 
  b \sum_{i \in H(k)} \frac{(y_i - y_j)^2}{\|x_i - x_l\|^2}, & k = l \\
  -b I_{H(k)}(l) \frac{(y_i - y_j)^2}{\|x_i - x_l\|^2}, & k \neq l
\end{cases}
\]

(13.36)

For the TOA case, the derivation is very similar, and the details are omitted for brevity:

\[
[F_{Txx}]_{k,l} = \begin{cases} 
  \frac{1}{\sigma_t^2} \sum_{i \in H(k)} \frac{(y_i - x_j)^2}{\|x_i - x_l\|^2}, & k = l \\
  -\frac{1}{\sigma_t^2} I_{H(k)}(l) \frac{(y_i - x_j)^2}{\|x_i - x_l\|^2}, & k \neq l
\end{cases}
\]

\[
[F_{Txy}]_{k,l} = \begin{cases} 
  \frac{1}{\sigma_t^2} \sum_{i \in H(k)} \frac{(y_i - x_j)(y_i - x_j)}{\|x_i - x_l\|^2}, & k = l \\
  -\frac{1}{\sigma_t^2} I_{H(k)}(l) \frac{(y_i - x_j)(y_i - x_j)}{\|x_i - x_l\|^2}, & k \neq l
\end{cases}
\]

\[
[F_{Tyy}]_{k,l} = \begin{cases} 
  \frac{1}{\sigma_t^2} \sum_{i \in H(k)} \frac{(y_i - y_j)^2}{\|x_i - x_l\|^2}, & k = l \\
  -\frac{1}{\sigma_t^2} I_{H(k)}(l) \frac{(y_i - y_j)^2}{\|x_i - x_l\|^2}, & k \neq l
\end{cases}
\]

(13.37)

Note that \( F_R \propto n_p^2/\sigma_{dB}^2 \) while \( F_T \propto 1/(\sigma_T^2) \). These SNR quantities directly affect the CRB. For TOA measurements, the dependence on the device coordinates is in unitless distance ratios, indicating that the size of the system can be scaled without changing the CRB as long as the geometry is kept the same. However, in the case of RSS measurements, the variance bound scales with the size of the system even if the geometry is kept the same due to the \( d^4 \) terms in the denominator of each term of \( F_R \). These scaling characteristics indicate that TOA measurements would be preferred for sparse networks, but for sufficiently high density, RSS can perform as well as TOA.

If \( \hat{x}_i \) and \( \hat{y}_i \) are unbiased estimates of \( x_i \) and \( y_i \), for the case of TOA measurements, the trace of the covariance of these estimates satisfies:

\[
\sigma_T^2 \triangleq \text{tr} \{ \text{cov}_{\hat{F}_R}(\hat{x}_i, \hat{y}_i) \} = \text{Var}_R(\hat{x}_i) + \text{Var}_R(\hat{y}_i) \geq \left( [F_{Txx} - F_{Txy} F^{-1}_{Tyy} F^{-1}_{Txx}]^{-1} \right)_{i,i} + \left( F_{Tyy} - F_{Txy} F^{-1}_{Txx} F^{-1}_{Txy} \right)_{i,i}
\]

(13.38)

For RSS measurements, \( F_R \) in Equation (13.38) should be replaced with \( F_R \). For the case of one blindfolded device, a simple expression can be derived for both RSS and TOA measurements.

Consider the network having blindfolded device 1 and reference devices 2 \ldots m + 1. This example, with a single pair of unknowns \( x_1 \) and \( y_1 \), is equivalent to many existing location systems, and a bound for the variance of the location estimator has already been derived in for TOA measurements in Section 13.1. In the case of RSS measurements:

\[
\sigma_T^2 \triangleq E \left[ (\hat{x}_1 - x_1)^2 + (\hat{y}_1 - y_1)^2 \right] \geq \frac{F_{Rxx} + F_{Ryy}}{F_{Rxx} F_{Ryy} - F_{Rxy}^2}
\]

from which we obtain

\[
\sigma_T^2 \leq 1 \frac{1}{b} \sum_{i=2}^{m+1} \sum_{j=i+1}^{m+1} \left( \frac{d_{i,j}^2}{\tilde{d}_{i,j}^2} \right)
\]
where the distance $d_{1,i,j}$ is the shortest distance from the point $(x_1, y_1)$ to the line segment connecting device $i$ and device $j$. For the case of TOA measurements, we obtain:

$$\sigma^2_1 = c^2 \sigma_T^2 \left[ \sum_{i=2}^{m} \sum_{j=i+1}^{m+1} \left( \frac{d_{1,i,j}, d_{1,j}}{d_{1,i,j}} \right)^2 \right]^{-1} \quad (13.39)$$

In Section 13.1, the ratio $d_{1,i,j}, d_{1,j}/(d_{1,i,j})$ has been called the geometric conditioning $A_{i,j}$ of device 1 w.r.t. references $i$ and $j$. $A_{i,j}$ is the area of the parallelogram specified by the vectors from device 1 to $i$ and from device 1 to $j$, normalized by the lengths of the two vectors. The geometric dilution of precision (GDOP), which is defined as $\sigma_1/(c \sigma_T)$, is

$$\text{GDOP} = \sqrt{\sum_{i=1}^{m} \sum_{j=i+1}^{m+1} A_{i,j}^2}$$

which matches the result in (13.14). The CRBs are shown in Figure 13.5 when there are four reference devices located in the corners of a 1 by 1 m square. The minimum of Figure 13.5(a) is 0.27. Since the CRB scales with size in the RSS case, the standard deviation of unbiased location estimates in a traditional RSS system operating in a channel with $\sigma_{dB}/n_p = 1.7$ is limited to about 27% of the distance between reference devices. This performance has prevented use of RSS in many existing location systems and is the motive for having many blindfolded devices in the network. Note that in the TOA case, $\sigma_1$ is proportional to $c \sigma_T$ and thus, $c \sigma_T = 1$ was chosen in Figure 13.5(b).

For general $n$ and $m$, we calculate the MLE of $\theta$. In the case of TOA measurements, the MLE is:

$$\hat{\theta}_T = \arg \min_{[z_i]} \sum_{i=1}^{m+n} \sum_{j<i} \left( cT_{i,j} - d(z_i, z_j) \right)^2$$

where $z_i = [x_i, y_i]^T$. The MLE for the RSS case is:

$$\hat{\theta}_R = \arg \min_{[z_i]} \sum_{i=1}^{m+n} \sum_{j<i} \left( \ln \left( \frac{\tilde{d}_{i,j}^2}{d^2(z_i, z_j)} \right) \right)^2$$

Unlike the MLE based on TOA measurements, the RSS MLE is readily shown to be biased. Specifically, for a single reference and single blindfolded device, the range estimate between the two devices is $\tilde{d}_{1,2}$. Using Equation (13.30), the mean of $\tilde{d}_{1,2}$ is given by:

$$E[\tilde{d}_{1,2}] = C d_{1,2}, \text{ where, } C = \exp \left[ \frac{1}{2} \left( \frac{\ln(10)}{10} \frac{\sigma_{dB}}{n_p} \right)^2 \right]$$

Figure 13.5 $\sigma_1$ (in meters) for the example system versus the coordinates of the single blindfolded node for (a) RSS with $\sigma_{dB}/n = 1.7$ or (b) TOA with $c \sigma_T = 1$ m © IEEE 2003.
For typical channels, $C \approx 1.2$, adding 20% bias to the range. Motivated by Equation (13.42), a bias-reduced MLE can be defined as:

$$\hat{\theta}_R = \arg \min \sum_{i=1}^{m+n} \sum_{j \in H(i)} \left( \ln \frac{\hat{d}_{i,j}^2/C^2}{d^2(z_i, z_j)} \right)^2$$  \hspace{1cm} (13.43)

However, there remains residual bias. Estimation results using methodology presented in this section and measurements from [44] are shown in Figures 13.6 and 13.7.

Figure 13.6 True (●#T) and estimated (▼ #E) location using (a) RSS, and (b) TOA data for measured network with four reference devices (X#). Higher errors are indicated by darker text [44] © IEEE 2003.
13.3 AVERAGE PERFORMANCE OF CIRCULAR AND HYPERBOLIC GEOLOCATION

In this section we provide more details on performance analysis of geolocation methods in terms of their theoretical positioning errors. Comparison is established in two different ways: strict and average. As in Section 13.1, in the strict type, methods are examined for a particular geometric configuration of base stations (BSs) with respect to mobile position, which determines a given noise profile affecting the estimates. In the average type, methods are evaluated in terms of the expected covariance matrix of the position error over an ensemble of random geometries, so that comparison is geometry independent.

In this section, the accuracy of four geolocation methods is defined in terms of their position error covariance matrices: (i) $C_{st}$—circular with known clock (ii) $C_{\bar{st}}$—circular with unknown clock (iii) $C_{hc}$—hyperbolic correlated, and (iv) $C_{\bar{hc}}$—hyperbolic uncorrelated. In scenario (i), the available measurements consist of the absolute TOAs of the signals transmitted between the mobile and the beacons. A circular technique is used to compute the position estimate. In scenario (ii), the available measurements consist of the pseudo-TOAs of the signals transmitted between the mobile and the beacons. A circular technique is used to compute the position estimate along with the clock offset. In scenario (iii), the available measurements consist of the TDOAs obtained from differences between pairs of pseudo-TOAs. A hyperbolic technique is used to compute the position estimate. In scenario (iv), the available measurements consist of uncorrelated TDOAs obtained at different times. A hyperbolic technique is used to compute the position estimate.

13.3.1 Signal models and performance limits

For circular methods, the TOA/pseudo-TOA measurements obtained from $N$ beacons can be expressed as a function of the mobile position $x = [x_1, x_2]^T$ as:

$$\hat{t}_n = \tilde{t}_n(x, t_0) + u_n = t_n(x) + t_0 + u_n$$  \hspace{1cm} (13.44)

where $\tilde{t}_n(x) = \frac{1}{c} \|x - x_n\|$, $x_n = [x_{1,n}, x_{2,n}]^T$ is the $n$th beacon position, for $1 \leq n \leq N$, $t_0$ is the clock offset, which has the same value for all $n$, $u_n$ are the measurement errors, and $c$ is the speed of light. Note that $t_0 = 0$ in the case of absolute TOAs, while it becomes an unknown parameter in the case of pseudo-TOAs.
If the measurements be stacked into vector \( \hat{\mathbf{t}} \), we have:

\[
\hat{\mathbf{t}} = \tilde{\mathbf{u}}(x, t_0) + \mathbf{u} = \bar{\mathbf{u}}(x) + t_0 \mathbf{1} + \mathbf{u}
\]  

(13.45)

where \( \mathbf{1} \) is the all-ones column vector. The \( N \times N \) covariance matrix of the noise vector \( \mathbf{u} \) is \( \mathbf{R} = \mathbb{E}[\mathbf{uu}^T] \). It is assumed throughout the section that matrix \( \mathbf{R} \) is known. In practice, \( \mathbf{R} \) is to be estimated, with a consequent degradation in the performance.

For hyperbolic methods the \( (N - 1) \) TDOA measurements are given by:

\[
\hat{\mathbf{d}} = \mathbf{H}\bar{\mathbf{u}}(x, t_0) + \mathbf{v} = \mathbf{Ht}(x) + \mathbf{v}
\]  

(13.46)

with \( \mathbf{H} \) representing any full-rank \( (N - 1) \times N \) matrix such that \( \mathbf{H1} = \mathbf{0} \), where \( \mathbf{0} \) is the all-zeros column vector. Note that this condition implies that the measurement vector \( \hat{\mathbf{d}} \) is not affected by the presence of the nonzero clock offset \( t_0 \) in Equation (13.46). The common example of \( \mathbf{H} \) is \( \mathbf{H} = [1, -1] \), where all TDOAs are obtained with respect to a common pseudo-TOA. This first (or reference) pseudo-TOA corresponds to the all-ones column of \( \mathbf{H} \), and will henceforth be called the pivot BS.

The analysis in this section is general for any \( \mathbf{H} \), unless otherwise stated. The \( (N - 1) \times (N - 1) \) covariance matrix of the noise vector \( \mathbf{v} \) is \( \mathbf{R}_v = \mathbb{E}[\mathbf{vv}^T] \). Only in the ‘hyperbolic-correlated’ scenario, the TDOAs are computed as \( \hat{\mathbf{d}} = \mathbf{H}\bar{\mathbf{u}} \), and then it holds from Equation (13.45) that \( \mathbf{v} = \mathbf{Hu} \), and \( \mathbf{R}_v = \mathbf{R}_\mathbf{v} = \mathbf{HRH}^T \). However, in the ‘hyperbolic-uncorrelated’ scenario, the structures of \( \mathbf{R}_\mathbf{v} \) and \( \mathbf{R}_\mathbf{c} \) are not necessarily related in a simple way. Indeed, in the ‘hyperbolic-uncorrelated’ scenario, matrix \( \mathbf{H} \) in Equation (13.46) only shows how the pairs of BS are selected to obtain the \( N - 1 \) TDOA measurements, but measurements remain independent. In that scenario, the relationship between \( \mathbf{R}_\mathbf{v} \) and \( \mathbf{R} \) for any \( \mathbf{H} \) can be expressed as \( \mathbf{R}_v = \mathbf{R}_\mathbf{v} = (\mathbf{HRH}^T) \circ \mathbf{I} \) where ‘\( \circ \)’ stands for the component-wise Schur–Hadamard product between two matrices. \( \mathbf{R}_\mathbf{v} \) and \( \mathbf{R}_\mathbf{c} \) are the specific versions of \( \mathbf{R}_v \) for the correlated and uncorrelated version, respectively, of the hyperbolic approach.

The performance limits are computed as the performance of the linearized weighted least squares (WLS) estimator, which coincides with the Cramer–Rao bound in the case of Gaussian noise.

As in Equation (13.2) for circular approaches, the linearization of Equation (13.45) at the true position yields

\[
\Delta \hat{\mathbf{i}} = \mathbf{F}\Delta \mathbf{x} + \Delta t_0 \mathbf{1} + \mathbf{u}
\]  

(13.47)

where the \( N \times 2 \) matrix of partial derivatives \( \mathbf{F} \) can be calculated as:

\[
[F]_{i,j} = \frac{\partial t_{n}(x)}{\partial x_i} = \frac{1}{c} \frac{x_i - x_{i,n}}{\| x - x_n \|} = \frac{1}{c} \frac{x_i - x_{i,n}}{\sqrt{(x_1 - x_{1,n})^2 + (x_2 - x_{2,n})^2}}
\]  

(13.48)

with \( i = 1, 2 \), and where \( [\mathbf{M}]_{l',l} \) denotes the \( l \)th row \( l' \)th column element of a matrix \( \mathbf{M} \). The WLS estimator [48] of \( \Delta \mathbf{x} \) and \( \Delta t_0 \) is the minimizer of

\[
(\Delta \hat{\mathbf{i}} - \mathbf{F}\Delta \mathbf{x} - \Delta t_0 \mathbf{1})^T \mathbf{R}^{-1} (\Delta \hat{\mathbf{i}} - \mathbf{F}\Delta \mathbf{x} - \Delta t_0 \mathbf{1})
\]  

(13.49)

which, using standard differential matrix calculus, can be expressed as:

\[
\begin{bmatrix}
\Delta \mathbf{x} \\
\Delta t_0
\end{bmatrix} = (\mathbf{F}_1^T \mathbf{R}^{-1} \mathbf{F}_1)^{-1} \mathbf{F}_1^T \mathbf{R}^{-1} \Delta \hat{\mathbf{i}}
\]  

(13.50)

where \( \mathbf{F}_1 = [\mathbf{F}, \mathbf{1}] \), and the covariance matrix of \( \Delta \mathbf{x} \) is given by \( \mathbf{C}_\mathbf{d} = \mathbf{T}^T (\mathbf{F}_1^T \mathbf{R}^{-1} \mathbf{F}_1)^{-1} \mathbf{T} \) with \( \mathbf{T}^T = [\mathbf{I}, \mathbf{0}] \). Following the same procedure, the covariance matrix of \( \Delta \mathbf{x} \) in the case of known clock \( t_0 \) is

\[
\mathbf{C}_\mathbf{d} = (\mathbf{F}_1^T \mathbf{R}^{-1} \mathbf{F}_1)^{-1}
\]

For hyperbolic approaches, we have the linearization of Equation (13.46) that can be written as:

\[
\Delta \hat{\mathbf{d}} = \mathbf{H}\Delta \mathbf{x} + \mathbf{v}
\]  

(13.51)

and the WLS estimator of \( \Delta \mathbf{x} \) becomes the minimizer of

\[
(\Delta \hat{\mathbf{d}} - \mathbf{H}\Delta \mathbf{x})^T \mathbf{R}_v^{-1} (\Delta \hat{\mathbf{d}} - \mathbf{H}\Delta \mathbf{x})
\]  

(13.52)
given by

$$\Delta x = (F^T H^T R_h^{-1} HF)^{-1} F^T H^T R_h^{-1} \Delta \hat{d}$$  \hspace{1cm} (13.53)$$

with covariance matrix $C_h = (F^T H^T R_h^{-1} HF)^{-1}$ where the subindex in $C_h$ and $R_h$ stands for a generic hyperbolic approach. For the uncorrelated hyperbolic approach, this covariance matrix becomes:

$$C_{h_{\text{uc}}} = (F^T H^T R_h^{-1} HF)^{-1} = \left(F^T H^T \left( (HRH^T) \circ I \right)^{-1} HF \right)^{-1}$$  \hspace{1cm} (13.54)$$

and, in the specific case that the TDOAs are computed as differences between pseudo-TOAs (correlated hyperbolic approach), we have $R_h = R_{hc} = HRH^T$, and the corresponding covariance matrix is given by

$$C_{hc} = (F^T H^T (HRH^T)^{-1} HF)^{-1}. \hspace{1cm} (13.55)$$

### 13.3.2 Performance of location techniques

From the previous section we have for $C_{s_{\text{st}}}^{-1}$ and $C_{hc}^{-1}$:

$$C_{s_{\text{st}}}^{-1} = \left[T^T \left(F^T R^{-1} F_1 \right)^{-1} T \right]^{-1}$$

$$C_{hc}^{-1} = \left(F^T \left(H^T \left( (HRH^T) \circ I \right) \right)^{-1} H \right) F^T$$  \hspace{1cm} (13.56)$$

By forming the product $F_1^T R^{-1} F_1$ in $C_{s_{\text{st}}}^{-1}$, as:

$$F_1^T R^{-1} F_1 = \begin{bmatrix} F^T \\ I^T \end{bmatrix} R^{-1} \begin{bmatrix} F & 1 \\ F & 1 \end{bmatrix} = \begin{bmatrix} F^T R^{-1} F & F^T R^{-1} F_1 \\ I^T R^{-1} F & I^T R^{-1} F_1 \end{bmatrix}$$  \hspace{1cm} (13.57)$$

and using block matrix inversion in Equation (13.57), we have:

$$B_{11} = \begin{bmatrix} F^T R^{-1} F - F^T R^{-1} I^T R^{-1} F_1 \\ I^T R^{-1} I \\ 0 \end{bmatrix}$$  \hspace{1cm} (13.58)$$

We also have from (13.56) that:

$$C_{s_{\text{st}}}^{-1} = \begin{bmatrix} I & 0 \\ F^T R^{-1} F_1 \end{bmatrix} \begin{bmatrix} I \\ 0 \end{bmatrix}^{-1} = B_{11}^{-1}$$

$$B_{11}^{-1} = F^T R^{-1} F - \frac{F^T R^{-1} I^T R^{-1} F_1}{I^T R^{-1} I} = F^T \left(R^{-1} - \frac{R^{-1} I^T R^{-1}}{I^T R^{-1} I} \right) F$$  \hspace{1cm} (13.59)$$

By using the fact that

$$R^{-1} - \frac{R^{-1} I^T R^{-1}}{I^T R^{-1} I} = H^T (HRH^T)^{-1} H$$  \hspace{1cm} (13.59)$$

is true for any matrix $H$ that fulfills $H I = 0$, we have

$$result \hspace{1cm} C_{s_{\text{st}}} = C_{hc}$$  \hspace{1cm} (13.60)$$
Proof of Equation (13.59): If $\Xi$ is a matrix and $G$ is any Gram matrix, the matrix $G^\perp$ is a projection matrix

$$G^\perp = G - G \Xi [\Xi^T G \Xi]^{-1} \Xi^T G$$

that fulfills $G^\perp \Xi = G \Xi - G \Xi [\Xi^T G \Xi]^{-1} \Xi^T G \Xi = 0$.

Hence, for $G = R$ and $\Xi = H^T$, we have $R^\perp = R - RH^T (HRH^T)^{-1} HR$ and $R^\perp H^T = 0$. Consequently, $R^\perp_H$ has rank one (because $H$ is almost full rank) and $R^\perp = \lambda a a^T$, with $Ha = 0$. Therefore, $a = 1$ fulfills the required orthogonality with the columns of $H$ and $R - RH^T (HRH^T)^{-1} HR = \lambda 11^T$.

Multiplying by $R^{-1}$ on both sides

$$R^{-1} - H^T (HRH^T)^{-1} H = \lambda R^{-1} 11^T R^{-1}.$$

Now, as $H 1 = 0$, multiplication by $1^T$ and $1$ on both sides yields

$$1^T R^{-1} 1 = \lambda 1^T R^{-1} 11^T R^{-1} 1.$$

Thus, $\lambda = (1^T R^{-1} 1)^{-1}$, and

$$R^{-1} - H^T (HRH^T)^{-1} H = \frac{R^{-1} 11^T R^{-1}}{1^T R^{-1} 1}$$

which is Equation (13.59).

Next we compare $C_{\beta} = T^T (F^T R^{-1} F) F^{-1} T$ and $C_{\alpha} = (F^T R^{-1} F) F^{-1}$ to prove that $C_{\alpha} > C_{\beta}$ for any geometry or noise profile. We know from Equation (13.58) that

$$C_{\alpha}^{-1} = F^T \left( R^{-1} - \frac{R^{-1} 11^T R^{-1}}{1^T R^{-1} 1} \right) F.$$  \hspace{1cm} (13.61)

Hence

$$C_{\alpha}^{-1} = C_{\alpha}^{-1} - q q^T$$  \hspace{1cm} (13.62)

where

$$q = \frac{F^T R^{-1} 1}{\sqrt{1^T R^{-1} 1}}.$$  \hspace{1cm} (13.63)

Using the matrix inversion lemma in Equation (13.62), we see that $C_{\alpha}$ is more positive definite than $C_{\beta}$ (i.e., $C_{\alpha} - C_{\beta}$ is a positive definite matrix), giving

$$result 2 \quad C_{\beta} = C_{\alpha} + \frac{q q^T}{1 + q^T C_{\alpha} q} \geq C_{\alpha}$$  \hspace{1cm} (13.64)

and we conclude from Equation (13.64) that absolute TOA measurements generate a $C_{\beta}$ smaller than $C_{\alpha}$ for any geometry or noise correlation. However, $C_{\beta}$ and $C_{\alpha}$ are not simply related by a scaling factor.

Strict comparison between $C_{hc}$ and $C_{\beta}$ is not possible, as some scenarios will deliver better performance for $C_{hc}$ and others will not. The judgment as to which method is better, and to what extent, will require the average performance analysis that is presented in the next section.

### 13.3.3 Average performance of location techniques

This section presents an efficient procedure for comparing algorithms in terms of their average covariance matrix, as well as an analytical lower bound in the positive definite sense. The placement of BSs is defined according to a uniform random distribution on a circular area centered at the true mobile position, and depending on a density parameter $\rho$ (BS/km$^2$).

From Section 13.3.1 and Equations (13.54), and (13.55), the covariance matrices of the four geolocation algorithms can be expressed as:

$$C = (F^T A F)^{-1}$$  \hspace{1cm} (13.65)
where $A$ is referred to as the information matrix. As defined in Section 13.3.1, we have for each geolocation algorithm:

$$A_{st} = R^{-1}$$

$$A_{si} = H^T (HRH^T)^{-1} H$$

$$A_{hc} = A_{ii}$$

$$A_{hc} = H^T [(HRH^T) \circ I]^{-1} H.$$  \hspace{1cm} (13.66)

In Equation (13.65), geometrical information is included in matrix $F$ and in the noise power distribution defined by the measurement covariance matrix $R$. From Equation (13.48), the rows of $F$ constitute the directive cosines of the vectors aligned with the paths from the mobile to the base stations (or generic references) with respect to the coordinate axes. Then:

$$F = F(\alpha) = \frac{1}{c} \begin{bmatrix} \cos(\alpha_1) & \sin(\alpha_1) \\ \cos(\alpha_2) & \sin(\alpha_2) \\ \vdots & \vdots \\ \cos(\alpha_N) & \sin(\alpha_N) \end{bmatrix}$$  \hspace{1cm} (13.67)

with $\alpha = [\alpha_1, \alpha_2, \ldots, \alpha_N]^T$, and $\alpha_n$ the angle with respect to the horizontal coordinate axis of the vector from the mobile to the $n$th BS. The diagonal of $R$ contains the noise power of all individual pseudo-TOA measurements, which depends mainly on the distance between the mobile and each BS. So, the specific position of the BSs affects the covariance matrix $C$ in two different ways: in their angular distribution implicit in $F$, and in their distance distribution implicit in $R$. Hence, the geometry-independent average of the position error covariance matrix can be expressed as:

$$\bar{C} = E_G \left[ (F^T A F)^{-1} \right] = E_N E_{\alpha} \left[ (F^T A F)^{-1} \right]$$  \hspace{1cm} (13.68)

with $E_G$ the expectation operator with respect to all possible geometry scenarios, which can be split into the expected value with respect to the angles $\alpha$, ($E_{\alpha}$), and the expected value with respect to the power distribution ($E_N$) (related to the distance distribution). Its analytical computation is too involved due to the particular way in which geometry affects the covariance structure. So, the expression derived in the sequel exploits a rotational property of the 2D location case in such a way that the structure of Equation (13.68) can be studied without computing its expected value analytically.

For convenience, let us define the expectation with respect to the angles in (13.68) as $\bar{C}_N$:

$$\bar{C} = E_N \bar{C}_N$$  \hspace{1cm} (13.69)

where

$$\bar{C}_N = E_{\alpha} \left[ (F^T A F)^{-1} \right].$$  \hspace{1cm} (13.70)

The angles $\alpha$ are independent, uniformly distributed random variables in $[0, 2\pi]$ due to the uniform random placement of BSs in the coverage area. Thus, the infinite set of all possible angular configurations of the BSs can be divided into an infinite number of subsets within which angular configurations differ in a rotation. That is:

$$[\alpha_1, \alpha_2, \ldots, \alpha_N] = \alpha_1 \cdot 1^T + [0, \alpha_2 - \alpha_1, \ldots, \alpha_N - \alpha_1]$$  \hspace{1cm} (13.71)

with the rotated angles $\alpha_i - \alpha_1$, $2 \leq i \leq N$ a set of mutually independent random variables uniformly distributed in $[0, 2\pi]$ and also independent of the equally uniformly distributed leading angle $\alpha_1$. Then, the expectation ($E_\alpha$) in Equation (13.70) can be expressed as the composition of two expectations: the expectation over the rotation $\alpha_1$ applied to the expectation over the subset of rotated angles $SS = \{\alpha_i - \alpha_1, 2 \leq i \leq N\}$. From this division into subsets of the geometry, Equation (13.70) becomes:

$$\bar{C}_N = E_{SS} E_\beta \left[ (F_{SS,\beta}^T A F_{SS,\beta})^{-1} \right]$$  \hspace{1cm} (13.72)

where $F_{SS,\beta}$ denotes $F$ for a particular rotation angle $\beta = \alpha_1$ and subset configuration. From Equation (13.71), we can express the generic subset $F_{SS,\beta}^T$ matrix as $F_{SS,\beta}^T = G_{\beta} F_{SS}^T$, where $F_{SS}^T = F_{SS,0}^T$ denotes
the canonic $\mathbf{F}$ matrix of the subset in the specific case $\beta = 0$, and $\mathbf{G}_\beta$ is a unitary-rotation matrix defined as:

$$
\mathbf{G}_\beta = \begin{bmatrix}
\cos(\beta) & \sin(\beta) \\
-\sin(\beta) & \cos(\beta)
\end{bmatrix}.
$$

(13.73)

Now, Equation (13.72) can be expressed as:

$$
\overline{\mathbf{C}_N} = E_{SS} E_\beta \left( \mathbf{G}_\beta \mathbf{F}_{SS}^T \mathbf{A} \mathbf{F}_{SS} \mathbf{G}_\beta^T \right)^{-1} = E_{SS} E_\beta \left[ \mathbf{G}_\beta \left( \mathbf{F}_{SS}^T \mathbf{A} \mathbf{F}_{SS} \right)^{-1} \mathbf{G}_\beta^T \right]
$$

(13.74)

since $\mathbf{G}_\beta$ is unitary. The expectation over $\beta$ gives

$$
\overline{\mathbf{C}_N} = \frac{1}{2} E_{SS} \text{tr} \left( \left( \mathbf{F}_{SS}^T \mathbf{A} \mathbf{F}_{SS} \right)^{-1} \right) \cdot \mathbf{I}
$$

(13.75)

Proof of (13.75) : Here we prove that for symmetric matrix $\mathbf{A}$ holds

$$
\mathbf{B} = E_\beta [\mathbf{G}_\beta \mathbf{A} \mathbf{G}_\beta^T] = \frac{1}{2} \text{tr}[\mathbf{A}] \cdot \mathbf{I}
$$

where $E_\beta$ denotes the expectation operation over $\beta$. $\mathbf{A}$ is a $2 \times 2$ matrix:

$$
\mathbf{A} = \begin{bmatrix}
a_{11} & a_{12} \\
a_{21} & a_{22}
\end{bmatrix}
$$

and $\mathbf{G}_\beta$ is the previously defined unitary rotation matrix:

$$
\mathbf{G}_\beta = \begin{bmatrix}
\cos(\beta) & \sin(\beta) \\
-\sin(\beta) & \cos(\beta)
\end{bmatrix}
$$

Then, the expectation over the uniformly distributed random variable $\beta$ can be expressed as

$$
E_\beta [\mathbf{G}_\beta \mathbf{A} \mathbf{G}_\beta^T] = \begin{bmatrix}
b_{11} & b_{12} \\
b_{21} & b_{22}
\end{bmatrix}
$$

(13.76)

Now, taking the expectation over the uniform distribution of $\beta$,

$$
b_{11} = \frac{a_{11} + a_{22}}{2}, \quad b_{12} = \frac{a_{12} - a_{21}}{2}, \quad b_{21} = \frac{a_{21} - a_{12}}{2}, \quad b_{22} = \frac{a_{11} + a_{22}}{2}
$$

For $\mathbf{A}$ a symmetric matrix, we finally obtain:

$$
E_\beta [\mathbf{G}_\beta \mathbf{A} \mathbf{G}_\beta^T] = \begin{bmatrix}
\frac{a_{11} + a_{22}}{2} & 0 \\
0 & \frac{a_{11} + a_{22}}{2}
\end{bmatrix} = \frac{1}{2} \text{tr}[\mathbf{A}] \cdot \mathbf{I}
$$

Applying Equation (13.75) in (13.69) to the four algorithms associated with the information matrices $\mathbf{A}$ defined in Equation (13.66), we obtain:

$$
\overline{\mathbf{C}_{st}} = \frac{1}{2} E_{N,SS} \text{tr} \left( \left( \mathbf{F}_{SS}^T \mathbf{R}^{-1} \mathbf{F}_{SS} \right)^{-1} \right) \cdot \mathbf{I}
$$

$$
\overline{\mathbf{C}_{st}} = \frac{1}{2} E_{N,SS} \text{tr} \left( \left( \mathbf{F}_{SS}^T \mathbf{H}^T \left( \mathbf{H} \mathbf{R} \mathbf{H}^T \right)^{-1} \mathbf{F}_{SS} \right)^{-1} \right) \cdot \mathbf{I}
$$

$$
\overline{\mathbf{C}_{ch}} = \overline{\mathbf{C}_{st}}
$$

$$
\overline{\mathbf{C}_{ch}} = \frac{1}{2} E_{N,SS} \text{tr} \left( \left( \mathbf{F}_{SS}^T \mathbf{H}^T \left( \left( \mathbf{H} \mathbf{R} \mathbf{H}^T \right) \circ \mathbf{I} \right)^{-1} \mathbf{H} \mathbf{F}_{SS} \right)^{-1} \right) \cdot \mathbf{I}
$$

(13.76)

As an illustration, numerical results for (13.76) are presented in Figures 13.8 and 13.9 with the same parameters as in [47]. A completely random uniform distribution of the BSs in the visibility circle of the mobile is assumed. It is also assumed that the mobile can observe BSs within a radius of $R$ m where BSs are uniformly distributed with a certain density of $\rho$ BS/km$^2$. 
If \( d \) is the distance from the mobile to a randomly selected beacon, then the probability density function (pdf) of \( d \) is given by \( p_d(d) = 2d/R^2 \) for \( 0 \leq d \leq R \) [49]. On the other hand, the delay spread, which constitutes the principal phenomenon that disturbs the TOA measurements, is given by [49] as \( \sigma = T_1 d^\epsilon y \) where \( T_1 \) is the mean value of \( \sigma \) at \( d = 1 \) km, \( \epsilon \) is an exponent between 0.5 and 1.0, and \( y \) is a log-normal random variable generated by \( y = 10^{Y/10} \), where \( Y \) is Gaussian with standard deviation \( \sigma_Y \) in the interval between 2 and 6 dB. The performance results shown hereafter assume that the TOA covariance matrix \( \mathbf{R} \) is diagonal, i.e. \( \mathbf{R} = \text{diag} \left( \left[ \sigma_1^2, \sigma_2^2, \ldots, \sigma_N^2 \right] \right) \) with \( \sigma_n \) samples of the random variable \( \sigma \) for the \( n \)th beacon. In all the simulations presented here, \( T_1, \epsilon, \) and \( \sigma_Y \) have
been chosen for the suburban environment following [49]. That is, $T_1 = 0.4 \mu$, $\epsilon = 0.5$, and $\sigma_Y = 4$ dB. The results are shown in Figures 13.8 and 13.9.

REFERENCES

1. Revision of the commissions rules to ensure compatibility with enhanced 911 emergency calling systems RM-8143, CC Docket 94–102, FCC, Washington, DC, July 26, 1996. FCC.


26. All T1P1.5 standardization documents can be found in the T1P1.5 documents—PCS directory [Online]. Available: http://www.t1.org


31. *Functional Stage 2 Specification of Location Services in UTRAN*, Doc. 3G TR 25.305. 3GPP.


39. European Digital Telecommunications System (Phase 2): Radio Subsystem Synchronization (GSM 05.10), May 1996. ETSI TC-SMG.


14
Channel Modeling and Measurements for 4G

14.1 MACROCELLULAR ENVIRONMENTS (1.8 GHZ)

In this section we present analysis of the joint statistical properties of azimuth spread, delay spread and shadowing fading in macrocellular environments. The analysis is based on data reported from a measurement campaign in typical urban (TU), bad urban (BU), and suburban (SU) [1] areas. In the experiment, a BS equipped with an eight-element uniform linear antenna array and an MS with an omnidirectional dipole antenna are used. The MS is equipped with a differential global positioning system (GPS) and an accurate position encoder so its location is accurately known by combining the information from these two devices. MS displacements of less than one centimeter can, therefore, be detected. The system is designed for transmission from the MS to the BS. Simultaneous channel sounding is performed on all eight branches, which makes it possible to estimate the azimuth of the impinging waves at the BS. The sounding signal is a maximum length linear shift register sequence of length 127 chips, clocked at a chip rate of 4.096 Mbps. This chip rate has been initially used in WCDMA proposals in Europe. The testbed operates at a carrier frequency of 1.8 GHz. Additional information regarding the stand-alone testbed can be found in [1, 2 and 5]. A summary of macrocellular measurement environments is given in Table 14.1.

The channel’s azimuth delay spread function at the BS is modeled as

$$h(\phi, \tau) = \sum_{l=1}^{L} \alpha_l \delta(\phi - \phi_l, \tau - \tau_l)$$  \hspace{1cm} (14.1)

where the parameters $\alpha_l$, $\tau_l$, and $\phi_l$ are the complex amplitude, delay, and incidence azimuth of the $l$th impinging wave at the BS. In general, $h(\phi, \tau)$ is considered to be a time-variant function, since the constellation of the impinging waves is likely to change as the MS moves along a certain route. The local average power azimuth delay spectrum is given as

$$P(\phi, \tau) = E \left\{ \sum_{l=1}^{L} |\alpha_l|^2 \delta(\phi - \phi_l, \tau - \tau_l) \right\}$$  \hspace{1cm} (14.2)
Table 14.1 Summary of macrocellular measurement environments

<table>
<thead>
<tr>
<th>Class</th>
<th>BS antenna height</th>
<th>Description of environment</th>
</tr>
</thead>
<tbody>
<tr>
<td>TU Typical urban</td>
<td>10 m and 32 m</td>
<td>The city of Aarhus, Denmark. Uniform density of buildings ranging from 4–6 floors. Irregular street layout. Measurements carried out along six different routes with an average length of 2 km. No line-of-sight between MS and BS. MS–BS distance varies from 0.2 km to 1.1 km.</td>
</tr>
<tr>
<td>TU Typical urban</td>
<td>21 m</td>
<td>Stockholm city, Sweden (Area #1). Heavily built-up area with a uniform density of buildings, ranging from 4–6 floors. Ground is slightly rolling. No line-of-sight between MS and BS. MS–BS distance varies from 0.2 km to 1.1 km.</td>
</tr>
<tr>
<td>BU Bad urban</td>
<td>21 m</td>
<td>Stockholm city, Sweden (Area #2). Mixture of open flat areas (river) and densely built-up zones. Ground is slightly rolling. No line-of-sight between MS and BS. MS–BS distance varies from 0.9 km to 1.6 km.</td>
</tr>
<tr>
<td>SU Suburban</td>
<td>12 m</td>
<td>The city of Gistrup, Denmark. Medium-sized village with family houses of one–two floors and small gardens with trees and bushes. Typical Danish residential area. The terrain around the village is rolling with some minor hills. No line-of-sight between MS and BS. MS–BS distance varies from 0.3 km to 2.0 km.</td>
</tr>
</tbody>
</table>

From Equation (14.2), the local power azimuth spectrum (PAS) and the local power delay spectrum (PDS) are given as

\[
P_A(\phi) = \int P(\phi, \tau) \, d\tau \tag{14.3}
\]

\[
P_D(\tau) = \int P(\phi, \tau) \, d\phi \tag{14.4}
\]

The radio channel’s local azimuth spread (AS) \(\sigma_A\) and the local delay spread (DS) \(\sigma_D\) are defined as the root second central moments of the corresponding variables. The values of the local AS and DS are likely to vary as the MS moves within a certain environment. Hence, we can characterize \(\sigma_A\) and \(\sigma_D\) as being random variables, with the joint pdf \(f(\sigma_A, \sigma_D)\). Their individual pdfs are

\[
f_A(\sigma_A) = \int f(\sigma_A, \sigma_D) \, d\sigma_D \tag{14.5}
\]

\[
f_D(\sigma_D) = \int f(\sigma_A, \sigma_D) \, d\sigma_A \tag{14.6}
\]

The function \(f(\sigma_A, \sigma_D)\) can be interpreted as the global joint pdf of the local AS and DS. If the expectation in Equation (14.2) is computed over the radio channel’s fast fading component, we can, furthermore, apply the approximation

\[
\int \int P(\phi, \tau) \, d\phi \, d\tau = h_{\text{channel}} \cong h_{\text{loss}}(d)h_s \tag{14.7}
\]
where \( h_{\text{channel}} \) is the radio channel’s integral path loss, \( h_{\text{loss}}(d) \) is the deterministic long-term distance-dependent path loss, while \( h_s \) is the channel’s shadow fading component, which is typically modeled with a log-normal distributed random variable [4, 5]. The global pdf of \( h_s \) is denoted \( f_s(h_s) \). The global degree of shadow fading is described by the root second central moment of the random shadow fading component expressed in decibels, i.e.

\[
\sigma_s = \text{Std} \{ 10 \log_{10}(h_s) \} 
\]  

(14.8)

where \( \text{Std\{ \}} \) denotes standard deviation. Empirical results for cumulative distribution functions (cdfs) for \( \sigma_A \) and \( \sigma_D \) are given in Figure 14.1 and Figure 14.2 respectively. The log-normal fit for \( \sigma_A \) results is given as

\[
\sigma_A = 10^{\epsilon_A X + \mu_A} 
\]

(14.9)

where \( X \) is a zero mean Gaussian distributed random variable with unit variance, \( \mu_A = E\{\log_{10}(\sigma_A)\} \) is the global logarithmic mean of the local AS, and \( \epsilon_A = \text{Std}\{\log_{10}(\sigma_A)\} \) is the logarithmic standard deviation of the AS.

Similarly,

\[
\sigma_D = 10^{\epsilon_D Y + \mu_D} 
\]

(14.10)

where \( Y \) is a zero mean Gaussian distributed random variable with unit variance, \( \mu_D = E\{\log_{10}(\sigma_D)\} \) is the global logarithmic mean of the local DS, and \( \epsilon_D = \text{Std}\{\log_{10}(\sigma_D)\} \) is the logarithmic standard deviation of the DS. A summary of the results for these parameters is given in Table 14.2.

### 14.1.1 PDF of shadow fading

The shadow fading component is extracted from the measurement data under the assumption that the deterministic distance path loss can be expressed in decibels as

\[
10 \log_{10}(h_{\text{loss}}(d)) = A + B \log_{10}(d) 
\]

(14.11)
Figure 14.2 Examples of empirical cdfs of the DS obtained in different environments. The cdf of a log-normal distribution is fitted to the empirical results for comparison [1] © 2002, IEEE.

Table 14.2 Summary of the first and second central moments of the AS, DS and shadow fading in the different environments [1] © 2002, IEEE

<table>
<thead>
<tr>
<th>Class</th>
<th>$\sigma_s$</th>
<th>$E{\sigma_A}$</th>
<th>$\mu_A$</th>
<th>$\epsilon_A$</th>
<th>$E{\sigma_D}$</th>
<th>$\mu_D$</th>
<th>$\epsilon_D$</th>
</tr>
</thead>
<tbody>
<tr>
<td>TU-32</td>
<td>7.3 dB</td>
<td>$8^0$</td>
<td>0.74</td>
<td>0.47</td>
<td>0.8 $\mu$s</td>
<td>-6.20</td>
<td>0.31</td>
</tr>
<tr>
<td>TU-21</td>
<td>8.5 dB</td>
<td>$8^0$</td>
<td>0.77</td>
<td>0.37</td>
<td>0.9 $\mu$s</td>
<td>-6.13</td>
<td>0.28</td>
</tr>
<tr>
<td>TU-20</td>
<td>7.9 dB</td>
<td>$13^0$</td>
<td>0.95</td>
<td>0.44</td>
<td>1.2 $\mu$s</td>
<td>-6.08</td>
<td>0.35</td>
</tr>
<tr>
<td>BU</td>
<td>10.0 dB</td>
<td>$7^0$</td>
<td>0.54</td>
<td>0.60</td>
<td>1.7 $\mu$s</td>
<td>-5.99</td>
<td>0.46</td>
</tr>
<tr>
<td>SU</td>
<td>6.1 dB</td>
<td>$8^0$</td>
<td>0.84</td>
<td>0.31</td>
<td>0.5 $\mu$s</td>
<td>-6.40</td>
<td>0.22</td>
</tr>
</tbody>
</table>

where $d$ is the distance between the BS and MS expressed in kilometers. As an example, the Okumura–Hata model [6] is based on a similar structure. From Equations (14.11) and (14.7) we have

$$10 \log_{10} (h_{\text{channel}} (d)) = A + B \log_{10} (d) + 10 \log_{10} (h_s)$$  \hspace{1cm} (14.12)$$

Assuming that $E\{10 \log_{10} (h_s)\} = 0.0$ dB, the parameters $A$ and $B$ can be obtained as the least squares estimates from a large number of measurements. Subsequently, the shadow fading component can be isolated from each measurement segment by rearranging Equation (14.12).

From the measurements it is found that a log-normal distribution function provides a good match to the empirical pdf of the shadow fading component. This observation is in coherence with numerous other studies, see [4, 7 and 8] among others. The shadow fading standard deviation is found to be in the range $\sigma_s = [6–10]$ dB depending on the environment class, with the largest standard deviation observed in the BU, and the smallest in SU environments. These findings are in accordance with previous findings in the open literature as well. Hence, the random variable describing the shadow
The fading component can be expressed as

\[ h_s = 10^{\rho_A Z/10} \]  \hspace{1cm} (14.13)

where \( Z \) is a zero mean Gaussian random variable with unit variance. The experimental results and analytical approximation for the spatial autocorrelation function are given in Figures 14.3 and 14.4.

Now,

\[ \rho_A(d) = \exp(-d/d_A) \]  \hspace{1cm} (14.14)

A double exponential decaying function is matched to the empirical results

\[ \rho_A(d) = k \exp\left(-\frac{d}{d_{A,1}}\right) + (1-k) \exp\left(-\frac{d}{d_{A,2}}\right) ; \quad k \in [0, 1] \]  \hspace{1cm} (14.15)
Figure 14.5 Empirical spatial autocorrelation function of DS in the BU environment. An exponential decaying function is matched to the empirical results.

Table 14.3 Spatial decorrelation distance for AS, DS and shadow fading in different environments expressed in meters. The two numbers presented for SU, correspond to the short and long decorrelation coefficients [1]

<table>
<thead>
<tr>
<th>Class</th>
<th>$d_A$</th>
<th>$d_D$</th>
<th>$d_s$</th>
</tr>
</thead>
<tbody>
<tr>
<td>TU-32</td>
<td>50</td>
<td>40</td>
<td>45</td>
</tr>
<tr>
<td>TU-21</td>
<td>50</td>
<td>50</td>
<td>25</td>
</tr>
<tr>
<td>TU-20</td>
<td>75</td>
<td>50</td>
<td>55</td>
</tr>
<tr>
<td>BU</td>
<td>65</td>
<td>95</td>
<td>120</td>
</tr>
<tr>
<td>SU</td>
<td>25/200 ($k = 0.2$)</td>
<td>15/150 ($k = 0.3$)</td>
<td>30/200 ($k = 0.4$)</td>
</tr>
</tbody>
</table>

The same results for DS are given in Figure 14.5 and Equations (14.16) and (14.17)

BU environment: $\rho_D(d) = \exp(-d/d_D)$  \hspace{1cm} (14.16)
SU environment: $\rho_D(d) = k \exp(-d/d_{D,1}) + (1 - k) \exp(-d/d_{D,2})$  \hspace{1cm} (14.17)

Finally, the results for the decorrelation distance are summarized in Table 14.3.

The mutual interdependence between the different components of the fading is characterized by their crosscorrelation functions. In general, the crosscorrelation coefficient between $a$ and $b$ is computed according to

$$\rho(a, b) = \frac{\sum_{i=1}^{N} (a(i) - \bar{a})(b(i) - \bar{b})}{\sqrt{\sum_{i=1}^{N} (a(i) - \bar{a})^2} \sqrt{\sum_{i'=1}^{N} (b(i') - \bar{b})^2}}$$  \hspace{1cm} (14.18)

where $\bar{a}$ and $\bar{b}$ are the sample means of the sets $\{a(i)\}$ and $\{b(i)\}$ with set size $N$. The results for AS, DS and shadow fading, based on measurements, are summarized in Table 14.4.

Additional details on the topic can be found in [1–29].
Table 14.4  Empirical crosscorrelation coefficients for the different environment classes for the AS, DS and shadow fading, expressed in both linear and logarithmic forms [1] © 2002, IEEE

<table>
<thead>
<tr>
<th>Class</th>
<th>$\rho \langle \sigma_A, \sigma_D \rangle$</th>
<th>$\rho \langle \sigma_A, \sigma_D \rangle \log$</th>
<th>$\rho \langle \sigma_A, h_s \rangle$</th>
<th>$\rho \langle \sigma_A, h_s \rangle \log$</th>
<th>$\rho \langle \sigma_D, h_s \rangle$</th>
<th>$\rho \langle \sigma_D, h_s \rangle \log$</th>
</tr>
</thead>
<tbody>
<tr>
<td>TU-32</td>
<td>0.39</td>
<td>0.44</td>
<td>−0.51</td>
<td>−0.7</td>
<td>−0.38</td>
<td>−0.4</td>
</tr>
<tr>
<td>TU-21</td>
<td>0.34</td>
<td>0.36</td>
<td>−0.47</td>
<td>−0.54</td>
<td>−0.34</td>
<td>−0.5</td>
</tr>
<tr>
<td>TU-20</td>
<td>0.6</td>
<td>0.6</td>
<td>−0.65</td>
<td>−0.72</td>
<td>−0.44</td>
<td>−0.48</td>
</tr>
<tr>
<td>BU</td>
<td>0.69</td>
<td>0.67</td>
<td>−0.44</td>
<td>−0.53</td>
<td>−0.55</td>
<td>−0.69</td>
</tr>
<tr>
<td>SU</td>
<td>0.46</td>
<td>0.46</td>
<td>−0.47</td>
<td>−0.5</td>
<td>−0.38</td>
<td>−0.46</td>
</tr>
</tbody>
</table>

Figure 14.6 The measurement area with all three RX sites; TX positions of the sample plots are marked [30] © 2002, IEEE.

14.2 URBAN SPATIAL RADIO CHANNELS IN MACRO/MICROCELL (2.154 GHZ)

The discussion in this section is based on experimental results collected with a wideband channel sounder using a planar antenna array [30]. The signal center frequency was 2154 MHz and the measurement bandwidth 100 MHz. A periodic PN sequence, 255 chips long, was used. The chip rate was 30 MHz and the sampling rate 120 MHz, giving an oversampling factor 4. The correlation technique is used for the determination of the impulse response. Hence, the delay range is 255/30 MHz = 8.5 μs, with a resolution of 1/30 MHz = 33 ns. The transmit antenna at the MS was a vertically polarized omnidirectional discone antenna. The vertical 3 dB beamwidth was 87° and the transmit power 40 dBm. Approximately 80 different transmitter positions were investigated.

The receiving BS was located at one of three different sites: below, at, and above the rooftop level (RX 1–RX 3, see, Figure 14.6). A 16-element physical array with dual polarized λ/2-spaced patch antennas was combined with a synthetic aperture technique to build a virtual two-dimensional (2D)
antenna structure. The patches were linearly polarized at \(0^\circ\) (horizontal direction) and \(90^\circ\) (vertical direction). With these \(16 \times 62\) elements, the direction of arrival (DOA) of incoming waves, both in azimuth (horizontal angle), and elevation (vertical angle), could be resolved by using the super-resolution Unitary ESPRIT algorithm [31–33]. Note that the number of antenna elements limits the number of identifiable waves, but not the angular resolution of the method. Together with a delay resolution of 33 ns, the radio channel can be characterized in all three dimensions separately for the two polarizations. Array signal processing – including estimation of the DOAs and a comparison of ESPRIT with other algorithms—was discussed in Chapter 13.

One prerequisite for the applicability of the synthetic aperture technique is that the radio channel is static during the whole data collection period. To avoid problems, the whole procedure was done at night with minimum traffic conditions.

14.2.1 Description of environment

A typical urban environment is shown in Figure 14.6 [30] with three receiver locations (RX 1–3) marked by triangles pointing broadside in the direction of the array. Figure 14.6 also provides information about all the corresponding TX positions. The location RX 1 (height \(h_{RX} = 10\ m\)) is a typical microcell site below the rooftop height of the surrounding buildings, and measurements are performed with 20 different TX positions. RX 2 (height \(h_{RX} = 27\ m\)) is at the rooftop level, and 32 TX positions are investigated. RX 3 (height \(h_{RX} = 21\ m\)) is a typical macrocell BS position above rooftop height, and 27 TX positions are measured.

14.2.2 Results

The measurement results show that it is possible to identify many single (particular, different) multi-path components, impinging at the receiver from different directions. But these components are not randomly distributed in the spatial and temporal domain, they naturally group into clusters. These clusters can be associated with objects in the environment due to the high angular and temporal resolution of evaluation. Sometimes even individual waves within a cluster can be associated with scattering objects. The identification of such clusters is facilitated by inspection of the maps of the environment.

A cluster is defined as a group of waves whose delay, azimuth and elevation at the receiver are very similar, while being notably different from other waves in at least one dimension. Additionally, all waves inside a cluster must stem from the same propagation mechanism. The definition of clusters always involves a certain amount of arbitrariness. Even for mathematically ‘exact’ definitions, arbitrary parameters (e.g. thresholds or numbers of components) must be defined. Clustering by human inspection, supported by maps of the environment, seems to give the best results.

The received power is calculated within each cluster (cluster power) by means of Unitary ESPRIT and a following beamforming algorithm. The results are plotted in the azimuth–elevation, azimuth–delay, and elevation–delay planes.

According to the obvious propagation mechanism, each cluster is assigned to one of three different classes.

- **Class 1. Street-guided propagation:** Waves arrive at the receiver from the street level after traveling through street canyons.

- **Class 2. Direct propagation—over the rooftop:** The waves arrive at the BS from the rooftop level by diffraction at the edges of roofs, either directly or after reflection from buildings surrounding the MS. The azimuth mostly points to the direction of the transmitter, with some spread in azimuth and delay.

- **Class 3. Reflection from high-rise objects—over the rooftop:** The elevation angles are near the horizon, pointing at or above the rooftop. The waves undergo a reflection at an object rising above the average building height before reaching the BS. The azimuth shows the direction of the reflecting building, the delay is typically larger than for Class 1 or Class 2.
The sum of the powers of all clusters belonging to the same class is called class power. In some cases, the propagation history is a mixture of different classes, e.g. street guidance followed by diffraction at rooftops. Such clusters are allocated to the class of the final path to the BS.

Some statistical evaluations are now presented to illustrate and quantify the clusterization. First we look at the number of clusters that is required to get a specific percentage of the total power; secondly, the powers of the clusters versus the delay; thirdly, the crosspolarization discrimination (XPD) versus the delay; fourthly, the relative class powers; and finally, the distribution of the number of clusters. Such questions are important for designing algorithms for adaptive antennas, e.g. should one capture, in uplink, the power of one or more clusters; or, in downlink, how does one distribute the available transmit power, and to which directions?

The first example illustrates the percentage contribution to the total power of the strongest cluster. If we refer to the 10% level in Figures 14.7–14.8, we can see that in 90% of all cases, the power in (in Figure 14.7) and 40% (in Figure 14.8) the strongest cluster is at least 55% of the total power. The same results for clusters separated by class are given in Figures 14.9 and 14.10.

These results are summarized in Table 14.5.

For the evaluation of delays we define the vector \( \mathbf{P} \) containing the powers of the clusters and vector \( \mathbf{\tau} \) of corresponding mean delays. A particular cluster \( i \) has mean delay \( \tau_i \), and power \( P_i \). The relation between the delays \( \mathbf{\tau} \) and the powers \( \mathbf{P} \) is modeled as exponential

\[
P_n \propto P(\tau_n) = ae^{-\tau_n/b}
\]

Experimental data are fit into the model in Equation (14.19) by using the least squares (LS) estimation. The logarithmic estimation error \( v \) is defined as

\[
v = 10 \log P - 10 \log s(\theta)
\]

and its standard deviation \( \sigma_v \) as

\[
\sigma_v = \sqrt{\text{var}\{v\}}
\]

Some results are shown in Figures 14.11 and 14.12. The summary of the results for parameters \( a \) and \( b \) is shown in Tables 14.6 and 14.7.
In Equation (14.21) \( \sigma_v \) was defined as the standard deviation of the logarithmic estimation error in dB. This estimation error was found to be log-normally distributed and up to a delay of about 1 \( \mu \)s, \( \sigma_v \) is independent of the delay \( \tau \). The value of \( \sigma_v \) is 9.0 dB and 10.0 dB (co- and crosspolarization), respectively, averaged over the first microsecond.

The XPD is defined as the ratio of the received power of the copolarized component to the power of the crosspolarized component, evaluated for each cluster. Some results are shown in Figure 14.13.
Figure 14.10 CDF of the relative power for the two strongest clusters separated by class, all TX positions. *N* is the number of samples.

Table 14.5 Average relative power in dB (relation of cluster power to total received power $P_{\text{tot}}$) of the strongest cluster of RX 1, RX 2, and RX 3, and the ratio in %

<table>
<thead>
<tr>
<th>RX</th>
<th>VP-VP dB (% of $P_{\text{tot}}$)</th>
<th>VP-HP dB (% of $P_{\text{tot}}$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>−0.9 (81%)</td>
<td>−1.2 (76%)</td>
</tr>
<tr>
<td>2</td>
<td>−0.4 (91%)</td>
<td>−0.6 (87%)</td>
</tr>
<tr>
<td>3</td>
<td>−1.3 (74%)</td>
<td>−1.8 (66%)</td>
</tr>
</tbody>
</table>

Figure 14.11 Relation of the cluster power to the total power versus delay, all TX positions and copolarization. *N* is the number of clusters [30] © 2002, IEEE.
Figure 14.12 Relation of the cluster power to the total power versus delay, all TX positions and cross polarization. $N$ is the number of clusters [30] © 2002, IEEE.

### Table 14.6
The model parameters $a$ and $b$ for both received polarizations (VP and HP) averaged over all available clusters. The transmitter was VP

<table>
<thead>
<tr>
<th></th>
<th>VP-VP</th>
<th>VP-HP</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a$</td>
<td>$-3.9$ dB</td>
<td>$-3.6$ dB</td>
</tr>
<tr>
<td>$b$</td>
<td>$8.9$ dB/µs</td>
<td>$11.8$ dB/µs</td>
</tr>
</tbody>
</table>

$P_i = ae^{-\tau_i/b}$

### Table 14.7
Average delay of the strongest cluster of RX 1, RX 2 and RX 3

<table>
<thead>
<tr>
<th>RX</th>
<th>Average delay VP-VP (µs)</th>
<th>Average delay VP-HP (µs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.068</td>
<td>0.071</td>
</tr>
<tr>
<td>2</td>
<td>0.38</td>
<td>0.28</td>
</tr>
<tr>
<td>3</td>
<td>0.11</td>
<td>0.048</td>
</tr>
</tbody>
</table>
Figure 14.13 (a) XPD versus delay, all TX positions; (b) CDF of the XPD separately for the three classes. $N$ is the number of clusters [30] © 2002, IEEE.
The average powers for different classes of cluster are shown in Table 14.8. Additional data on the topic can be found in [30–41].

### 14.3 MIMO CHANNELS IN MICROCELL AND PICOCELL ENVIRONMENTS (1.71/2.05 GHZ)

The model presented in this section is based upon data collected in both picocell and microcell environments [42]. The stochastic model has also been used to investigate the capacity of MIMO radio channels, considering two different power allocation strategies, water filling and uniform, and two different antenna topologies, $4 \times 4$ and $2 \times 4$. It will be demonstrated that the space diversity used at both ends of the MIMO radio link is an efficient technique in picocell environments, achieving capacities within 14 b/s/Hz and 16 b/s/Hz in 80% of the cases for a $4 \times 4$ antenna configuration implementing water filling at an SNR of 20 dB.

The basic parameters of the measurements setup are shown in Figure 14.14. The following notation is used subsequently: $d_{MS-BS}$ stands for distance between MS and BS, $h_{BS}$ for the height of BS above the ground floor, and AS for azimuth spread [42].

The vector of received signals at BS can be represented as $y(t) = [y_1(t), y_2(t), \ldots, y_M(t)]^T$, where $y_m(t)$ is the signal at the $m$th antenna port and $[\cdot]^T$ denotes transposition. Similarly, the signals at the MS are $s(t) = [s_1(t), s_2(t), \ldots, s_N(t)]^T$. The NB MIMO radio channel $H \in \mathbb{C}^{M \times N}$ which describes the connection between the MS and the BS can be expressed as

$$
H = \begin{bmatrix}
\alpha_{11} & \alpha_{12} & \cdots & \alpha_{1N} \\
\alpha_{21} & \alpha_{22} & \cdots & \alpha_{2N} \\
\vdots & \vdots & \ddots & \vdots \\
\alpha_{M1} & \alpha_{M2} & \cdots & \alpha_{MN}
\end{bmatrix}
$$

(14.22)

where $\alpha_{mn}$ is the complex transmission coefficient from the antenna at the MS to the antenna at the BS. For simplicity, it is assumed that $\alpha_{mn}$ is complex Gaussian distributed with identical average power. However, this latest assumption can be easily relaxed. Thus, the relation between the vectors $y(t)$ and $s(t)$ can be expressed as

$$
y(t) = H(t)s(t)
$$

(14.23)
MIMO CHANNELS IN MICROCELL AND PICOCELL ENVIRONMENTS

Figure 14.14 Functional sketch of the MIMO model [42] © 2002, IEEE.

Subsequently, we will use the following correlations:

\[
\rho_{m_1m_2}^{BS} = \left\langle \alpha_{m_1n}, \alpha_{m_2n} \right\rangle
\]  
(14.24a)

\[
\rho_{m_1m_2}^{MS} = \left\langle \alpha_{mn_1}, \alpha_{mn_2} \right\rangle
\]  
(14.24b)

\[
R_{BS} = \begin{bmatrix}
\rho_{11}^{BS} & \rho_{12}^{BS} & \cdots & \rho_{1M}^{BS} \\
\rho_{21}^{BS} & \rho_{22}^{BS} & \cdots & \rho_{2M}^{BS} \\
\vdots & \vdots & \ddots & \vdots \\
\rho_{M1}^{BS} & \rho_{M2}^{BS} & \cdots & \rho_{MM}^{BS}
\end{bmatrix}_{M \times M}
\]  
(14.24c)

\[
R_{MS} = \begin{bmatrix}
\rho_{11}^{MS} & \rho_{12}^{MS} & \cdots & \rho_{1N}^{MS} \\
\rho_{21}^{MS} & \rho_{22}^{MS} & \cdots & \rho_{2N}^{MS} \\
\vdots & \vdots & \ddots & \vdots \\
\rho_{N1}^{MS} & \rho_{N2}^{MS} & \cdots & \rho_{NN}^{MS}
\end{bmatrix}_{N \times N}
\]  
(14.24d)

The correlation coefficient between two arbitrary transmission coefficients connecting two different sets of antennas is expressed as

\[
\rho_{n_1m_1}^{n_2m_2} = \left\langle \alpha_{m_1n_1}, \alpha_{m_2n_2} \right\rangle
\]  
(14.25)
which is equivalent to
\[ \rho_{n_1 m_1} = \rho_{n_2 m_2} \]  
(14.26)
provided that Equations (14.24a) and (14.24b) are independent of \( n \) and \( m \), respectively. In other words, this means that the spatial correlation matrix of the MIMO radio channel is the Kronecker product of the spatial correlation matrix at the MS and the BS and is given by
\[ R_{\text{MIMO}} = R_{\text{MS}} \otimes R_{\text{BS}} \]  
(14.27)
where \( \otimes \) represents the Kronecker product. This has also been confirmed in [43].

### 14.3.1 Simulation of channel coefficients

Correlated channel coefficients, \( \alpha_{mn} \), are generated from zero mean complex independent identically distributed (i.i.d.) random variables \( a_{mn} \) shaped by the desired Doppler spectrum such that
\[ A = Ca \]  
(14.28)
where \( A_{MN \times 1} = [\alpha_{11}, \alpha_{21}, \ldots, \alpha_{M1}, \alpha_{12}, \ldots, \alpha_{MN}]^T \) and \( a_{MN \times 1} = [a_1, a_2, \ldots, a_{MN}]^T \). The symmetrical mapping matrix \( C \) results from the standard Cholesky factorization of the matrix \( R_{\text{MIMO}} = CC^T \), provided that \( R_{\text{MIMO}} \) is non-singular [44].

Subsequently, the generation of the simulated MIMO channel matrix \( \tilde{H} \) can be deduced from the vector \( A \). Note that the correlation matrices and the Doppler spectrum cannot be chosen independently, as they are connected through the PAS at the MS [45].

### 14.3.2 Measurement setups

The Tx is at the MS and the stationary Rx is located at the BS. The two setups from Figure 14.14 provide measurement results with different correlation properties of the MIMO channel for small antenna spacings of the order of 0.5\( \lambda \) or 1.5\( \lambda \). The BS consists of four parallel Rx channels. The sounding signal is an MSK-modulated linear shift register sequence of length 127 chips, clocked at a chip rate of 4.096 Mcps. At the Rx, the channel sounding is performed within a window of 14.6 \( \mu s \), with a sampling resolution of 122 ns (1/2 chip period) to obtain an estimate of the complex impulse response (IR). The NB information is subsequently extracted by averaging the complex delayed signal components. A more thorough description of the stand-alone testbed (i.e. Rx and Tx) is documented in [23, 46]. The description of the measurement environments is summarized in Table 14.9. A total of 107 paths are investigated within these seven environments. The first measurement setup is used to investigate 15 paths in a microcell environment, i.e. environment A in Table 14.9. The MS is positioned in different locations inside a building, while the BS is mounted on a crane and elevated above roof top level (i.e. 9 m) to provide direct LOS to the building. The antenna is located 300 m away from the building. The second setup is used to investigate 92 paths for both microcell and picocell environments, i.e. environments B and C to G, respectively, as shown in Table 14.9. The distance between the BS and the MS is 31 to 36 m for microcell B, with the BS located outside.

### 14.3.3 Validation of the stochastic MIMO channel model assumptions

The validity of the underlying assumptions has been verified for a 4 \( \times \) 4 MIMO configuration. These assumptions are that: (1) the spatial correlation at the BS, Equation (14.24a), and the MS, Equation (14.24b), is independent of \( n \) and \( m \), respectively; and (2) the spatial correlation matrix of the MIMO radio channel is the Kronecker product of the spatial correlation matrices at the BS and the MS, Equation (14.27).

To verify assumption (1), the standard deviation (std) of each measured spatial correlation coefficient \( |\rho_{m_1m_2}^{\text{BS}}| \) and \( |\rho_{n_1n_2}^{\text{MS}}| \) is computed over the \( N \) and \( M \) reference antennas, respectively, for each environment. The std at the BS is expressed as
\[ \text{std}_{\rho_{m_1m_2}^{\text{BS}}} = \text{std} \left( |\rho_{n_1n_2}^{\text{MS}}| \right), \quad \forall n \in [1 \ldots 4] \]  
(14.29)
Table 14.9 Summary and description of the different environments \cite{42} © 2002, IEEE

<table>
<thead>
<tr>
<th>Cell type</th>
<th>Environment</th>
<th>MS locations</th>
<th>Measurement setup</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microcell</td>
<td>A</td>
<td>15</td>
<td>1st</td>
<td>The indoor environment consists of small offices with windows metalically shielded – 300 m between MS and BS</td>
</tr>
<tr>
<td>Microcell</td>
<td>B</td>
<td>13</td>
<td>2nd</td>
<td>The indoor environment consists of small offices – 31 to 36 m between MS and BS</td>
</tr>
<tr>
<td>Picocell</td>
<td>C</td>
<td>21</td>
<td>2nd</td>
<td>The indoor environment is the same as in A</td>
</tr>
<tr>
<td>Picocell</td>
<td>D</td>
<td>12</td>
<td>2nd</td>
<td>Reception hall – large open area</td>
</tr>
<tr>
<td>Picocell</td>
<td>E</td>
<td>18</td>
<td>2nd</td>
<td>Modern open office with windows metalically shielded</td>
</tr>
<tr>
<td>Picocell</td>
<td>F</td>
<td>16</td>
<td>2nd</td>
<td>The indoor environment is the same as in B</td>
</tr>
<tr>
<td>Picocell</td>
<td>G</td>
<td>12</td>
<td>2nd</td>
<td>Airport – very large indoor open area</td>
</tr>
</tbody>
</table>

Figure 14.15 Example of the cdf of (a) $\text{std}_{\rho_{n_1n_2}}$ and (b) $\text{std}_{\rho_{n_1n_2}^{\text{MS}}}$. The cdf is performed over all the measured environments and for all seven correlation coefficients.

and at the MS

$$\text{std}_{\rho_{n_1n_2}^{\text{MS}} = \text{std} \left( \left\{ |\rho_{n_1m}^{n_2m}| \right\} \right)} , \quad \forall \ m \in [1 \ldots 4]$$  \hspace{1cm} (14.30)

Figure 14.15 presents the empirical cumulative distribution function (cdf) of $\text{std}_{\rho_{m_1m_2}^{\text{BS}}}$ (a) and $\text{std}_{\rho_{n_1n_2}^{\text{MS}}}$ (b) computed over the 92 paths considered with the second measurement setup for the six different correlation coefficients, i.e. the upper triangular coefficient of the correlation matrix, when a
4 × 4 MIMO configuration is used. To validate the statistical significance of the empirical results, the empirical cdf is compared with a cdf obtained from simulations performed under similar conditions. The matching of the two cdfs demonstrates that assumption (1) is fulfilled, as explained hereafter. For each of the 92 × 6 different measured correlation coefficients, two correlated, Rayleigh-distributed signals of length 1000λ are generated. These 1000λ-long vectors are truncated into 11.8λ-long runs over which the correlation coefficient is computed once again. Hence, a wider new set of correlation values is collected, exhibiting a standard deviation std_{11.8λ}. This operation is repeated 92 × 6 times. A simulated cdf of std_{11.8λ} is then obtained under similar conditions as for the measured cdf.

### 14.3.4 Input parameters to the validation of the MIMO model

The input parameters used in the validation stage are illustrated by the shaded areas of Figure 14.16. They are the average spatial complex correlation matrices $R_{BS}$ and $R_{MS}$, and the associated average Doppler spectrum.

The measured spatial complex correlation matrices are the results of an average over the reference antennas $n$ and $m$ with respect to which the matrices are computed.

The averaged measured Doppler spectrum is obtained by averaging over all the $MN$ channel coefficients. It is defined at the MS, since the BS is fixed. This limitation is due to the measurement setup implementation, but is not inherent to the model. If both MS and BS were moving, the Doppler spectrum of the channels would have been defined as the convolution of separate Doppler spectra, defined either at the MS or at the BS, considering, respectively, the BS or the MS as fixed. The corresponding complex coefficients of the vector $\mathbf{a}$ in Equation (14.28) have their amplitudes shaped by the average measured Doppler spectrum and assigned a random phase uniformly distributed over $[0, 2\pi]$ such that $MN$ independent and identically distributed variables are generated. Two examples of typical paths are presented by Equations (14.31) and (14.32) [42].

![Figure 14.16 Illustration of the two measurement setups](Image)
In both Examples 1 and 2, $|\mathbf{R}_{\text{MS}}|$ is decorrelated. This is expected, since the MS is surrounded by scatterers. On the other hand, $\mathbf{R}_{\text{BS}}$ presents two different behaviors. In Example 1, the spatial correlation coefficients remain low, as expected in the case of an indoor termination. On the other hand, the spatial correlation coefficients at the BS are highly correlated in Example 2, with a mean absolute value of the coefficient of 0.96. The high correlation is explained by the fact that regarding this specific example, the BS is identified to be located above any surrounding scatterer. Therefore, it experiences a low azimuth spread (AS), which causes its antenna array elements to be highly correlated. An illustration of the averaged measured Doppler spectrum of Example 1 is presented in Figure 14.17. The spectrum is normalized in frequency to its maximum Doppler shift $f_m$ and in power to its maximum value.

### 14.3.5 The eigenanalysis method

The eigenvalue decomposition (EVD) of the instantaneous correlation matrix $\mathbf{R} = \mathbf{H}\mathbf{H}^H$ (not to be confused with $\mathbf{R}_{\text{MIMO}}$), where $[\cdot]^H$ represents Hermitian transposition, can serve as a benchmark of the validation process. The channel matrix $\mathbf{H}$ may offer $K$ parallel subchannels with different mean gains, with $K = \text{Rank}(\mathbf{R}) \leq \min(M, N)$, where the functions $\text{Rank}(\cdot)$ and $\min(\cdot)$ return the rank of the matrix and the minimum value of the arguments, respectively, [27]. The $k$th eigenvalue can be interpreted as the power gain of the $k$th subchannel [27]. In the following, $\lambda_k$ represents the eigenvalues. In order to assess the qualitative accuracy of the model, the comparison between...
measured and simulated eigenvalues is made for an antenna configuration where the largest number of eigenvalues is achievable within the limitation of the measurement setup antenna topology. This is the case for a $4 \times 4$ scenario, since at most four eigenvalues can be expected. In the following, the eigenvalues are normalized to the mean power of the single Tx and a single Rx channel coefficient $(1/MN) \sum_{m=1}^{M} \sum_{n=1}^{N} |\alpha_{mn}|^2$.

### 14.3.5.1 Validation procedure

For each of the 107 paths, the input parameters are fed into the proposed stochastic MIMO model and a Monte Carlo simulation consisting of 100 iterations is performed to generate the elements of the simulated matrix $\tilde{H}$. $\tilde{H}$ is a three-dimensional (3D) matrix $(M \times N \times L)$, where $L$ is the number of samples equivalent to the time domain definition in Equation (14.23). For each iteration, the seed of the random generator which defines the phase of the complex coefficient of the vector $a$ is different.

At iteration $q$, $\tilde{H}_{4\times4\times L}$ counts as many samples $L$ as in the measured $H_{4\times4\times L}$ collected during one antenna array run, that is to say $20\lambda$ or $11.8\lambda$, depending on the measurement setup used. The EVD of $\tilde{H}_{4\times4\times L}\tilde{H}_{4\times4\times L}^H$ is then performed for each sample $l$ in order to identify the corresponding simulated eigenvalues denoted by the vector,

$$\lambda_{\text{sim},kq} = [\lambda_{\text{sim},kq,1}, \ldots, \lambda_{\text{sim},kq,l}]^T, \quad k = 1, \ldots, K.$$  

From these eigenvalues, $k$ vectors $[\lambda_{\text{sim},k}]_{1\times QL}$ containing the 100 iterations of the simulated eigenvalues $\lambda_{\text{sim},kq}$ are deduced, so that $\lambda_{\text{sim},k} = (\lambda_{\text{sim},kq})$, where $\{\}$ represents a set of variables. For the measured data, the eigenvalues were deduced so that $\lambda_{\text{meas},k} = [\lambda_{\text{meas},k,1}, \ldots, \lambda_{\text{meas},k,l}]^T$. Some results are presented in Figure 14.18.
Figure 14.18 (a) Local validation. Cdf of $\lambda_{\text{meas},k}$ and $\lambda_{\text{sim},k}$ from Example 1 (picocell decorrelated); (b) local validation. Cdf of $\lambda_{\text{meas},k}$ and $\lambda_{\text{sim},k}$ for each of the 107 paths from Example 2 (microcell correlated); (c) global validation. Cdf of $|\Delta_{\text{error},k}|$ over the 107 paths.
14.3.5.2 Definition of the power allocation schemes

In the situation where the channel is known at both Tx and Rx, and is used to compute the optimum weight, the power gain in the \( k \)th subchannel is given by the \( k \)th eigenvalue, i.e. the signal to noise ratio (SNR) for the \( k \)th subchannel equals

\[
\gamma_k = \frac{P_k}{\sigma_N^2}
\]  

(14.33)

where \( P_k \) is the power assigned to the \( k \)th subchannel, \( \lambda_k \) is the \( k \)th eigenvalue and \( \sigma_N^2 \) is the noise power. For simplicity, it is assumed that \( \sigma_N^2 = 1 \). According to Shannon, the maximum capacity normalized with respect to the bandwidth (given in terms of b/s/Hz spectral efficiency) of parallel subchannels equals [47]

\[
C = \sum_{k=1}^{K} \log_2(1 + \gamma_k)
\]  

(14.34)

\[
= \sum_{k=1}^{K} \log_2 \left(1 + \frac{\lambda_k P_k}{\sigma_N^2}\right)
\]  

(14.35)

where the mean SNR is defined as

\[
\text{SNR} = \frac{E[P_{\text{Rx}}]}{\sigma_N^2} = \frac{E[P_{\text{Tx}}]}{\sigma_N^2}
\]  

(14.36)

Given the set of eigenvalues \( \{\lambda_k\} \), the power \( P_k \) allocated to each subchannel \( k \) is determined to maximize the capacity by using Gallagher’s water filling theorem [27] such that each subchannel is filled up to a common level \( D \), i.e.

\[
\frac{1}{\lambda_1} + P_1 = \cdots = \frac{1}{\lambda_K} + P_k = \cdots D
\]  

(14.37)

with a constraint on the total Tx power such that

\[
\sum_{k=1}^{K} P_k = P_{\text{Tx}}
\]  

(14.38)

where \( P_{\text{Tx}} \) is the total transmitted power. This means that the subchannel with the highest gain is allocated the largest amount of power. In the case where \( 1/\lambda_k > D \), then \( P_k = 0 \).

When the uniform power allocation scheme is employed, the power \( P_k \) is adjusted according to

\[
P_1 = \cdots = P_K
\]  

(14.39)

Thus, in the situation where the channel is unknown, the uniform distribution of the power is applicable over the antennas [27] so that the power should be equally distributed between the \( N \) elements of the array at the Tx, i.e.

\[
P_n = \frac{P_{\text{Tx}}}{N}, \quad \forall \ n = 1 \ldots N
\]  

(14.40)

Some results are given in Figure 14.19. In Figure 14.19(a), \( C_k \) is the capacity of the \( k \)th subchannel of Figure 14.18(a).

14.4 OUTDOOR MOBILE CHANNEL (5.3 GHz)

In this section we discuss the mobile channel at 5.3 GHz. The discussion is based on measurement results collected at six different sites [48]. Site A is an example of a dense urban environment, the
Figure 14.19 (a) Cdf of the capacity per subchannel $C_i$ and its total results $\sum C_i$ for Example 1 (picocell decorrelated). A $4 \times 4$ antenna topology is presented here [42]; (b) capacity (10% level) versus SNR for Example 1 (picocell decorrelated) and Example 2 (microcell correlated). The water filling power allocation scheme; (c) cdf over all the 79 picocell paths of the total capacity deduced from (b) at SNR = 20 dB for $4 \times 4$ and $2 \times 4$ antenna configurations [42]; (d) cdf over all the 28 microcell paths of the total capacity deduced from (b) at SNR = 20 dB for $4 \times 4$ and $2 \times 4$ antenna configurations [42] © 2002, IEEE.
Figure 14.19 (Cont.).
transmitting antenna is about 45 m above ground level, representing a case with the BS antenna over the rooftops. Site B is a dense urban residential environment. Here, the transmitting antenna is placed on a mast with a height of 4 m, which is a typical case with the BS antenna lower than the rooftops. The measurement routes for this site are shown in Figure 14.20(a). The receiving antenna mobile station is at a height of 2.5 m on top of a car for both of the sites mentioned above. Site C

Figure 14.20 (a) Measurement routes for Site B with Tx height of 4m; (b) Rotation measurements in an urban environment, Site C [48] © 2002, IEEE.
Table 14.10  System configuration for mobile measurements in urban (U), suburban (S) and rural (R) areas

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Receiver</td>
<td>Direct sampling/5.3 GHz</td>
</tr>
<tr>
<td>Transmitter power</td>
<td>30 dBm</td>
</tr>
<tr>
<td>Chip frequency</td>
<td>30 MHz</td>
</tr>
<tr>
<td>Delay range</td>
<td>4.233 µs</td>
</tr>
<tr>
<td>Doppler range</td>
<td>124 Hz (U), 62 Hz (S, R)</td>
</tr>
<tr>
<td>Measurement rate</td>
<td>248 sets/s (U), 124 sets/s (S, R)</td>
</tr>
<tr>
<td>Sampling frequency</td>
<td>120 Ms/s</td>
</tr>
<tr>
<td>IRs/wavelength</td>
<td>5 (U), 4.2 (S, R)</td>
</tr>
<tr>
<td>Receiver velocity</td>
<td>2.80 m/s (U), 1.67 m/s (S, R)</td>
</tr>
<tr>
<td>Antennas and polarization</td>
<td>Omnidirectional antenna with 1 dBi gain; vertical polarization</td>
</tr>
</tbody>
</table>

Table 14.11  Path loss models for urban environments

<table>
<thead>
<tr>
<th>Urban models</th>
<th>Tx height: 4 m</th>
<th>Tx height: 12 m</th>
<th>Tx height: 45 m</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>n</td>
<td>b (dB)</td>
<td>std (dB)</td>
</tr>
<tr>
<td>LOS</td>
<td>1.4</td>
<td>58.6</td>
<td>3.7</td>
</tr>
<tr>
<td>NLOS</td>
<td>2.8</td>
<td>50.6</td>
<td>4.4</td>
</tr>
</tbody>
</table>

is located in a typical city center. The goal was to place the transmitter at some elevation relative to the ground, but still keep it below the rooftops. The transmitting antenna is placed at a height of 12 m, and the receiving antenna is on top of a trolley with a height of 2 m above ground level. In Site C, the rotation measurements are taken using a directive horn antenna. The 3 dB beamwidth of the horn antenna is 30° in the H-plane and 37° in the E-plane, and the peak sidelobe level was 26 dB. The specific environment for rotation measurements is shown in Figure 14.20(b). Site D represents a semiurban/semirural residential area. The three-story buildings are the tallest ones around, and the transmitting antenna is placed over the rooftops at a height of 12 m from ground level. Site E was selected to represent the rural case. The transmitting antenna is placed on top of a 5 m mast at the hilltop so that the antenna is about 55 m above the surroundings. Site F represents a typical semiurban/urban case. The transmitting antenna is placed on top of a 5 m mast. The receiving antenna is always on top of a car at a height of 2.5 m. The routes are measured using the wideband channel sounder.

System parameters are summarized in Table 14.10.

14.4.1 Path loss models

Similarly to Equation (14.11), we have

$$PL (dB) = b + 10n \log_{10} d$$  \hspace{1cm} (14.11a)

where $d_0 = 1$ m, $n$ is the attenuation exponent, $b$ is the intercept point in the semilog coordinate, and $d_m$ is the distance from the receiver to the transmitter. The measurement distances are about 30–300 m in this case. Some results are given in Tables 14.11 and 14.12 and Figure 14.21.

The measurement results for delays are summarized in Table 14.13.
Table 14.12  Path loss models for suburban and rural environments

<table>
<thead>
<tr>
<th>Models</th>
<th>n</th>
<th>b (dB)</th>
<th>std (dB)</th>
<th>Models</th>
<th>n</th>
<th>b (dB)</th>
<th>std (dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOS</td>
<td>3.3</td>
<td>21.8</td>
<td>3.7</td>
<td>LOS</td>
<td>3.3</td>
<td>21.8</td>
<td>3.7</td>
</tr>
<tr>
<td>NLOS</td>
<td>5.9</td>
<td>-27.8</td>
<td>1.9</td>
<td>NLOS</td>
<td>2.5</td>
<td>38.0</td>
<td>4.9</td>
</tr>
</tbody>
</table>

Figure 14.21  (a) Path loss for urban LOS with Tx height of 4 m; (b) mobile terminal turning around a corner with Tx height of 12 m in an urban environment [48] © 2002, IEEE.
### Table 14.13  Measured values for mean excess delay and rms delay spread

<table>
<thead>
<tr>
<th>( ) Tx height in meters</th>
<th>Urban</th>
<th>Suburban</th>
<th>Rural</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOS Mean excess delay (ns)</td>
<td>38 (4)</td>
<td>36 (5)</td>
<td>29 (55)</td>
</tr>
<tr>
<td>NLOS Mean excess delay (ns)</td>
<td>70 (4)</td>
<td>68 (12)</td>
<td></td>
</tr>
<tr>
<td>LOS Mean delay spread (ns)</td>
<td>43 (12)</td>
<td>25 (5)</td>
<td>22 (55)</td>
</tr>
<tr>
<td>NLOS Mean delay spread (ns)</td>
<td>44 (4)</td>
<td>66 (12)</td>
<td>25 (4)</td>
</tr>
<tr>
<td>LOS Median delay spread (ns)</td>
<td>31 (12)</td>
<td>13 (5)</td>
<td>15 (55)</td>
</tr>
<tr>
<td>NLOS Median delay spread (ns)</td>
<td>86 (45)</td>
<td>63 (12)</td>
<td>93 (4)</td>
</tr>
<tr>
<td>LOS CDF &lt;90%</td>
<td>64 (12)</td>
<td>57 (5)</td>
<td>44 (55)</td>
</tr>
<tr>
<td>NLOS CDF &lt;90%</td>
<td>63 (4)</td>
<td>105 (12)</td>
<td></td>
</tr>
</tbody>
</table>

#### 14.4.2 Window length for averaging fast fading components at 5 GHz

Multipath propagation causes fast fading in mobile communications. Thus, an important consideration in experimental data processing is how to average out the fast fading components and still preserve the slow fading characteristics. In [49], it was suggested that a suitable window length for data taken from macrocells is $4\lambda$. However, examination of data taken from microcells showed that the local mean could suffer quite large variations over short distances, and in [50], $5\lambda$ (about 1.7 m) was considered a more appropriate window length for microcells from the experimental data at 900 MHz. In this experiment, the least squares method with wide and narrowband received power is used to give the linear regression curves. Let us take the regression curves as the reference values, and then change the window length to 5, 10, 20, and $40\lambda$ for averaging the fading signals. Figure 14.22(a) shows the wideband received power for urban LOS with a transmitter height of 12 m. If we now take the linear regression values as the average received power, the standard deviations (std) are 2.47, 2.25, 1.93, and 1.62 dB, corresponding to the window lengths of 5, 10, 20 and $40\lambda$, respectively. It is seen that the fast fading components are averaged out if the window length is in the range from $20\lambda$ to $40\lambda$, namely, 1–2 m. The same conclusion can also be obtained for averaging narrowband fast fading components, and the corresponding result can be found in Figure 14.22(b). So, based on [1203] and the experience of processing much measured data at 5 GHz, it seems that the practical window length for averaging out fast fading components is 1–2 m in micro- and picocells at 900 MHz–5 GHz frequency bands.

#### 14.4.3 Spatial and frequency correlations

Spatial and frequency correlation study is useful for the design of antenna diversity to reduce the multipath fading. Because the correlation behavior is a small-scale effect, a wide-sense stationary uncorrelated scattering (WSSUS) situation should be assumed. To meet this condition, about $40\lambda$ is used as the window length to give the average correlation function. The formulas for calculating spatial and frequency correlation functions can be found in [51, 52]. In this section, envelope correlation is considered for narrowband signals. However, recent research [53] has shown that spatial correlation characteristics do not largely depend on frequency bandwidth up to approximately 20% of the
carrier frequency \( (B/f_c = 0.2, \text{ where } B \text{ is the bandwidth of a transmitted signal and } f_c \text{ is the carrier frequency}) \). Therefore, the narrowband model is sufficient for computing the spatial correlation characteristics within \( B \leq 0.2 \, f_c \).

Figure 14.23 shows the spatial and frequency envelope correlation functions for LOS outdoor environments at different transmitter heights. It is seen that the correlation distances are strongly dependent on the transmitter heights. The correlation distances with the envelope correlation coefficient
Figure 14.23  (a) Spatial correlations in LOS outdoor environments for urban cases with three transmitter heights. (b) Spatial correlations in LOS outdoor environments for rural and suburban cases with two transmitter heights. (c) Frequency correlations in LOS outdoor environments for urban cases with three transmitter heights. (d) Frequency correlations in LOS outdoor environments for rural and suburban cases with two transmitter heights.
of 0.7 are between 1 and 11λ (about 0.06–0.62 m). The respective correlation bandwidths are between 1.2 and 11.5 MHz. In LOS cases, due to the direct wave superimposed by only weak scattered waves, the coherence is high and the correlation length is large.

### 14.4.4 Path number distribution

The multipath number distribution was regarded as Poisson’s and modified Poisson’s in [54], and modified Poisson’s distribution has been shown to have good agreement with the experimental results in some cases. However, the modified Poisson’s distribution does not have an explicit expression, just a process. Therefore, it is not convenient for practical use. In [48], another simple and useful
path number distribution was suggested, by considering the path number variation of radio waves in land mobile communications a Markov process at finite state space, and it was shown to have good agreement with the experimental results. The path number distributions given by Poisson and Gao can be expressed as

\[
P(N) = \frac{\eta^{N_T-N}}{(N_T-N)!} e^{-\eta} \quad (14.41)
\]

\[
P(N) = C_N^{N_T} \frac{\eta^{N_T-N}}{(1 + \eta)^{N_T}} \quad (14.42)
\]

where \( N \) is variable and \( C \) means combination. \( N_T \) is the maximum number of paths that the mobile can receive. The parameters \( \eta \) and \( N_T \) can be fitted by the experimental data. For Poisson’s probability density function (PDF), the mean path number is \( \langle N \rangle = \eta \). For Gao’s PDF, the mean value is \( \langle N \rangle = N_T/(1 + \eta) \).

The empirical path number distributions for the outdoor measurements are fitted by using Equations (14.41) and (14.42), respectively. The path numbers are obtained from measured data by counting the peaks of the power delay profiles. The best fit is obtained by minimizing the following standard deviation

\[
\text{std} = \sqrt{\frac{1}{N_T} \sum_{i=1}^{N_T} \left( p_i - p_i^e \right)^2} \quad (14.43)
\]

where \( p_i^e \) is the experimental probability corresponding to path number \( i \), and \( p_i \) is the fitted probability using Equations (14.41) and (14.42). The fitted parameters are available in Table 14.14. If the dynamic range is cut at different levels, for example, \(-25\), \(-20\) and \(-15\) dB, the fitting parameters in Equations (14.41) and (14.42) will be changed, but the path number distributions still follow Poisson’s and Gao’s distributions.

### 14.4.5 Rotation measurements in an urban environment

The rotation measurements at points \( P_1 \) and \( P_2 \) were performed at Site C and are shown in Figure 14.20(b). The transmitter height was 12 m and the receiver was on a rotating stand at a height of 1.6 m, close to the receiver is a large open square. In the experiments, large excess delays up to 1.2 \( \mu s \) and rms delay spread of about 0.42 \( \mu s \) are found (Figure 14.24).

The power angular profiles (PAPs) \( P_\tau(\phi) \) of the measurements were calculated by using the maximal ratio combining algorithm in the delay domain [52]

\[
P_\tau(\phi) = \alpha_{\text{cal}} \int_{t_{\text{min}}}^{t_{\text{max}}} |h(\tau, \phi)|^2 \, d\tau \quad (14.44)
\]
where $\alpha_{\text{cal}}$ is a factor which is obtained from the calibration measurement with a cable and an attenuator, $\tau_{\text{min}}$ and $\tau_{\text{max}}$ are the delays of the first and last detectable IR components, and $\phi$ is the angle of arrival of the waves in the azimuth plane. In the rotation measurements, the dynamic range is cut at $-26$ dB relative to the strongest path. Some results for angular profiles of relative received power are shown in Figure 14.25.

The number of paths as a function of azimuth angle is shown in Figure 14.26.
14.5 MICROCELL CHANNEL (8.45 GHz)

In this section spatio-temporal channel characterization in a suburban non line-of-sight microcellular environment is discussed. Figure 14.27 shows a map of the environment under consideration [55]. This is a residential area with predominantly wooden houses of 8 m average height and is considered to be a typical suburban microcellular environment of size 600 × 600 m². The traffic was very light and the environment was considered to be static throughout the experiments. A non line-of-sight (NLOS) transmitter and receiver shown in Figure 14.27 are considered. The distance between the transmitter and the receiver is 219 m. As a transmitter antenna, simulating the mobile station (MS), a vertically polarized omnidirectional half wave sleeve dipole is set at a height of 2.7 m. At the receiving
Table 14.15  Electrical parameters used in the ray-tracing simulation

<table>
<thead>
<tr>
<th>Material</th>
<th>$\epsilon_r$</th>
<th>$\sigma$ [s/m]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Concrete [1158]</td>
<td>5.5</td>
<td>0.023</td>
</tr>
<tr>
<td>Foliage [1169]</td>
<td>1.2</td>
<td>0.0003</td>
</tr>
<tr>
<td>Ground [1171]</td>
<td>15.0</td>
<td>1.3</td>
</tr>
<tr>
<td>Metal</td>
<td>—</td>
<td>$\infty$</td>
</tr>
</tbody>
</table>

Figure 14.28  Azimuth profile. Solid line: experiment; dotted line: simulation [55] © 2002, IEEE.

Azimuth profiles are obtained by summing up the power of azimuth delay profiles with respect to the delay time. Figure 14.28 shows the azimuth profiles obtained from the experiment as the solid line, and from the simulation as the dotted line. From Figure 14.28, the forward arrival waves within the range from $-40^\circ$ to $44^\circ$ are in agreement for both data sets with respect to the level and the shape of the profile. The experimental profile has the floor level of about 30 dB below the peak, but this level is low enough so that its effect on the transmission property is negligibly small.
14.5.2 Delay profile for the forward arrival waves

For the forward arrival waves (−40° to −44°), the delay profiles are obtained by summing up the azimuth delay profiles with respect to the azimuth. The experimental result is shown in Figure 14.29(a), and the simulation result is shown in Figure 14.29(b). The experimental result exhibits an exponential decay. The results of least squares fitting are shown as the dashed line in Figure 14.29(a). The function is expressed as

\[ P(\tau) = -0.038\tau - 28.6 \]  

(14.45)

where \( P \) is the path gain in dB, and \( \tau \) is the delay time in ns.

The ray-tracing simulation can predict accurately the first two peaks in the delay profile. However, the exponential decay of the profile cannot be accurately predicted. The problem here seems to be due to incomplete modeling of the effect of random scattering in the ray-tracing simulation. If,

![Figure 14.29 Delay profile of forward arrival waves: (a) experiment; (b) simulation.](image)
Figure 14.30  (a) CDF of the fluctuation component of the experimental delay profile from its exponential fit for forward arrival waves: experiment; (b) autocorrelation of the fluctuation component of the experimental delay profile from its exponential fit for forward arrival waves: experiment.

however, the gradient of this exponential function can be determined by some independent means, the ray-tracing results can be extrapolated to predict accurately the delay profile.

The cumulative distribution function (CDF) of the fluctuation component of the experimental data from its exponential fit is shown in Figure 14.30(a). This fluctuation component can be very accurately approximated by a log-normal distribution with a standard deviation of 5.3 dB. In the ray-tracing simulation, waves with a long delay time cannot be predicted. Instead, their statistical properties are used for the extrapolation.

The autocorrelation of this fluctuation component is presented in Figure 14.30(b). The correlation decreases monotonously, with a correlation distance and a time at a correlation coefficient of 0.5 of
Figure 14.31 Short-term AS for forward arrival waves.

about 3 m and 10 ns, respectively. Considering the resolution of the delay profile, which is equal to
the chip duration 20 ns, the fluctuation is modeled as uncorrelated, just as in the wide-sense stationary
uncorrelated scattering (WSSUS) model.

14.5.3 Short-term azimuth spread (AS) for forward arrival waves

Figure 14.28 showed the azimuth profile averaged over the delay time. Here, we focus on the short-
term azimuth profiles, which are obtained every 10 ns. The short-term AS for the forward arrival
waves, $\sigma_\psi$, is defined as

$$\sigma_\psi(\tau) = \sqrt{\langle \psi^2(\tau) \rangle - \langle \psi(\tau) \rangle^2}$$

(14.46)

where $\langle \cdot \rangle$ is the average of the $\psi$ weighted by the power for a fixed delay time, $\tau$. At each delay time,
the threshold level is set to be 30 dB below the peak in the profile in order to calculate the short-term
AS in Equation (14.46).

Figure 14.31 shows the resultant short-term angular spread. The experimental result and the result
of the simulation agree well within the range 740–860 ns in which the ray-tracing simulation is known
to predict the delay profile accurately (as shown in Figure 14.29(b)). The experimental results are
observed to exhibit the same properties for a larger delay time. Therefore, this behavior can be used
for an extrapolation of the angular profile beyond the range in which the simulation can predict the
delay profile.

Figure 14.31 indicates that the variation of the short-term AS can be modeled as a stationary
process. To characterize this stationary process, the CDF of the short-term AS is shown in Figure 14.32.
The solid line indicates the experimental result and the dotted line indicates the simulation result. It
is noted that the simulation result is obtained within a range of delay times from 700 ns to 880 ns.
Although the distribution functions of the experiment and the simulation look slightly different, their
average and their standard deviation are both in agreement. A Gaussian distribution has been used
as an approximation in the figure, although the most appropriate distribution function to use is still
under consideration.

Figure 14.33 shows the autocorrelation function of the short-term AS. The solid line indicates
the experimental result and the dotted line indicates the simulation result. In a similar way to
Figure 14.30(b), the correlation decreases monotonously. The correlation distance at a correlation
coefficient of 0.5 is about 2 m for the experimental data and about 5 m for the simulation. The
correlation distance is smaller using the experimental data, since random scattering is not taken
into account in the simulation. As in Figure 14.30(b), considering a chip duration of 20 ns, which
Figure 14.32 CDF of the short-term AS for forward arrival waves. Solid line: experiment; dotted line: simulation [55] © 2002, IEEE.

Figure 14.33 Autocorrelation of the short-term AS for forward arrival waves. Solid line: experiment; dotted line: simulation.

corresponds to a distance of 6 m, the short-term angular spread is also modeled as uncorrelated. It is noted that the correlation distances for Figures 14.30(b) and 14.33 are comparable and that, therefore, these two fluctuations seem to be related to each other.

Since the fluctuation component from the exponential function of the delay profile and the short-term azimuth profile for the forward arrival waves have similar correlation lengths, this may suggest some relationship between them.

A Nakagami–Rice fading model for the short-term azimuth profile, as shown in Figure 14.34 is used. This is composed of a stable strong signal component plus a weak scattered signal component. The scattered signal component is assumed to be stationary. Under these assumptions, the power increases and the AS decreases when the stable signal component is large, i.e. there is a negative correlation between these parameters.
To evaluate this model, the crosscorrelation is calculated between the fluctuation component of the experimental delay profile from its exponential fit and the short-term azimuth profile for the forward arrival waves, which is defined as

\[
\rho_{P\Delta \varphi} = \frac{\langle (P(\tau) - \langle P(\tau) \rangle)(\Delta \varphi(\tau) - \langle \Delta \varphi(\tau) \rangle) \rangle}{\sqrt{\langle (P(\tau) - \langle P(\tau) \rangle)^2 \rangle \langle (\Delta \varphi(\tau) - \langle \Delta \varphi(\tau) \rangle)^2 \rangle}}
\]  

(14.47)

where \( \langle \cdot \rangle \) is the sample average within the given time window, \( P(\tau) \) is the fluctuation component from the exponential function at a delay time, \( \tau \), in dB, and \( \Delta \varphi(\tau) \) is the short-term angular spread at a delay time, \( \tau \), in degrees. Figure 14.35 shows the crosscorrelation with a time window of 200 ns for the experimental data. It is clear from Figure 14.35 that the fluctuation component of the delay profile and the short-term AS are negatively correlated. This result suggests that the proposed Nakagami–Rice fading model can be used successfully to model the short-term azimuth profile.

### 14.6 WIRELESS MIMO LAN ENVIRONMENTS (5.2 GHZ)

The presentation in this section is based on the results of a measurement campaign [57] in two courtyards in the 5.2 GHz band assigned for wireless LANs (e.g. HYPERLAN (see www.etsi.org), or IEEE 802.11a). These standards specify wireless communication between computers, which is a
compelling application for MIMO systems. For measurement, a channel sounder with a bandwidth of 120 MHz connected via a fast RF switch to a uniform linear receiver antenna array is used. This array consists of \(N_R = 8\) antenna elements (± 60° element beam width), plus two dummy elements at each end of the array. All these components together constitute a single-directional channel sounder. A virtual array at the transmitter consists of a monopole antenna mounted on an X–Y-positioning device with stepping motors.

The experiment starts by positioning the transmit antenna at a certain position. At the receiver, the RF switch is connected to the first antenna element of the array, so that the transfer function (measured at 192 frequency samples) from the first transmit to the first receive element of the array is measured. Then, the switch is connected to the next receive antenna element, and the next transfer function is measured. The measurement of all the transfer functions is repeated 256 times, in order to assess the time variance of the channel (see below). Then, the transmit antenna is moved to the next position, and the procedure is repeated. \(N_T = 16\) transmit antenna positions are situated on a cross (i.e. 8 positions on each axis of the cross) and bursts of complex channel transfer functions are recorded. Any virtual array requires that the channel remains static during the measurement period. One complete measurement run (\(2 \times 8\) antenna positions at TX times eight spatial samples at RX times 192 frequency samples and 256 temporal samples gives \(16 \times 8 \times 192 \times 256 = 6291456\) complex samples) takes about five minutes.

### 14.6.1 Data evaluation

Starting from the four-dimensional transfer function (time, frequency, position of RX antenna, position of TX antenna), the Doppler-variant transfer function is computed first, by Fourier transforming the 256 temporal samples (with Hanning windowing). Next, all components that do not exhibit zero Doppler shift (Doppler filtering) are eliminated. Those components correspond, for example, to MPCs scattered by leaves moving in the wind. The eliminated components carry on the order of 1% of the total energy. The three-dimensional (static) transfer function obtained in that way is then evaluated by Unitary ESPRIT [33] to estimate the delays \(\tau_i\). Unitary ESPRIT is an improved version of the classical ESPRIT algorithm discussed in Chapter 13. They both estimate the signal subspace for extraction of the parameters of (spatial or frequency) harmonics in additive noise. One important step in ESPRIT is the estimation of the model order. Different methods have been proposed in the literature for that task. The relative power decrease between neighboring eigenvalues with additional correction by visual inspection of the scree graph showing the eigenvalues is an option used for generating the results presented subsequently.

After estimation of the parameters \(\tau_i\), we can determine the corresponding ‘steering’ matrix \(A_\tau\). Subsequent beamforming with its Moore–Penrose pseudoinverse [33, 58–61] \(A_\tau^+\) gives the vector of delay weights for all \(x_R, x_T\)

\[
h_\tau(x_T, x_R) = A_\tau^+ T_f(x_T, x_R)
\]  

(14.48)

where \(T_f\) is the vector of transfer coefficients at the 192 frequency subbands sounded. This gives us now the transfer coefficients from all positions \(x_T\) to all positions \(x_R\) separately for each delay \(\tau_i\). Thus, one dimension, namely the frequency, has been replaced by the parameterized version of its dual, the delays.

For the estimation of the directions of arrival (DOA) in each of the two-dimensional transfer functions, ESPRIT estimation and beamforming by the pseudoinverse are used

\[
h_{x_R}(\tau_i, x_T) = A_{x_R}^+ h_{x_T}(\tau_i, x_T)
\]  

(14.49)

Finally, for the direction of departure (DOD) we have

\[
h_{x_T}(\tau_i, \psi_{R,i,j}) = A_{x_T}^+ h_{x_R}(\tau_i, \psi_{R,i,j})
\]  

(14.50)

Figure 14.36 illustrates these steps.

The procedure gives us the number and parameters of the MPCs, i.e. the number and values of delays, which DOA can be observed at these delays, and which DOD corresponds to each DOA at a specific delay. Furthermore, we also obtain the powers of the MPCs. One important point in
Figure 14.36 Sequential estimation of the parametric channel response in the different domains: alternating estimation and beamforming [57] © 2002, IEEE.

the application of the sequential estimation procedure is the sequence in which the evaluation is performed. Roughly speaking, the number of MPCs that can be estimated is the number of samples we have at our disposal.

14.6.2 Capacity computation

In a fading channel, the capacity is a random variable, depending on the local (or instantaneous) channel realization. In order to determine the cumulative distribution function (cdf) of the capacity, and thus the outage capacity, we would have to perform a large number of measurements, either with slightly displaced arrays, or with a temporally varying scatterer arrangement. Since each single measurement requires a huge effort, such a procedure is highly undesirable.

To improve this situation, an evaluation technique that requires only a single measurement of the channel is used. This technique relies on the fact that we can generate different realizations of the transfer function by changing the phases of the multipath components. It is a well established fact in mobile radio that the phases are uniformly distributed random variables, whose different realizations occur as either transmitter, receiver or scatterers move [27]. We can thus generate different realizations of the transfer function from the $m$th transmit to the $k$th receive antenna as

$$ h_{k,m}(f) = \sum_i a_i \exp \left( -j \frac{2\pi}{\lambda} d \left[ k \sin (\phi_{R,i}) + m \sin (\phi_{T,i}) \right] \right) \times \exp (-j 2\pi f \tau_i) \exp (j \alpha_i) $$

where $\alpha_i$ is a uniformly distributed random phase, which can take on different values for the different MPCs numbered $i$. Note, however, that $\alpha_i$ stays unchanged as we consider different antenna elements $k$ and $m$. To simplify discussion, we, for now, consider only the flat fading case, i.e. $\tau_i = 0$. We can thus generate different realizations of the transfer function from the $m$th transmit to the $k$th receive antenna as

$$ h_{k,m}(f) = \sum_i a_i \exp \left( -j \frac{2\pi}{\lambda} d \left[ k \sin (\phi_{R,i}) + m \sin (\phi_{T,i}) \right] \right) \times \exp (-j 2\pi f \tau_i) \exp (j \alpha_i) $$

(14.51)

where $\alpha_i$ is a uniformly distributed random phase, which can take on different values for the different MPCs numbered $i$. Note, however, that $\alpha_i$ stays unchanged as we consider different antenna elements $k$ and $m$. To simplify discussion, we, for now, consider only the flat fading case, i.e. $\tau_i = 0$. We can thus generate different realizations of the channel matrix $\mathbf{H}$

$$ \mathbf{H} = \begin{bmatrix} h_{11} & h_{12} & \cdots & h_{1N_T} \\ h_{21} & h_{22} & \cdots & h_{2N_T} \\ \vdots & \vdots & \ddots & \vdots \\ h_{N_R 1} & h_{N_R 2} & \cdots & h_{N_R N_T} \end{bmatrix} $$

(14.52)

by the following two steps.
1. From a single measurement, i.e. a single snapshot of the channel matrix, determine the DOAs and DODs of the MPCs as described earlier in the section.

2. Compute synthetically the impulse responses at the positions of the antenna elements, and at different frequencies. Create different realizations of one ensemble by adding random phase factors (uniformly distributed between 0 and $2\pi$) to each MPC. For each channel realization, we can compute the capacity (from Section 4.12)

$$ C = \log_2 \det \left( I + \frac{\rho}{N_T} \mathbf{H}^H \mathbf{H} \right) $$

(14.53)

where $\rho$ denotes the SNR, $I$ is the identity matrix and superscript $H$ means Hermitian transposition. For the frequency selective case, we have to evaluate the capacity by integrating over all frequencies

$$ C = \int \log_2 \det \left( I + \frac{\rho}{N_T} \mathbf{H}^H (f) \mathbf{H} (f) \right) df $$

(14.54)

Here, $\mathbf{H} (f)$ is the frequency-dependent transfer matrix. The integration range is the bandwidth of interest.

### 14.6.3 Measurement environments

As an example, the following scenarios are evaluated with the procedure described above [57]:

- **Scenario I**: A courtyard with dimensions $26 \, \text{m} \times 27 \, \text{m}$, open on one side. The RX array broadside points into the center of the yard, the transmitter is located on the positioning device 8m away in LOS. The power delay profile (PDP) in this scenario is given in Figure 14.37 (a).

- **Scenario II**: Closed backyard of size $34 \, \text{m} \times 40 \, \text{m}$ with inclined rectangular extension. The RX array is situated in one rectangular corner with the array broadside of the linear array pointing under $45^\circ$ inclination directly to the middle of the yard. The LOS connection between TX and RX measures 28 m. Many metallic objects are distributed irregularly along the building walls (power transformers, air conditioning fans, etc.). This environment looks very much like the backyard of a factory (Figure 14.38).

![Figure 14.37](image_url)

Figure 14.37 Power delay profiles (lines) in (a) the LOS Scenario I and (b) in the obstructed LOS Scenario IV. Superimposed circles are the identified MPCs that are further used to compute the simulated capacities.
Figure 14.38 Geometry of the environment of Scenarios II to IV (backyard) in plan view. Superimposed are the extracted DOAs and DODs for Scenario III [57] © 2002, IEEE.

- **Scenario III**: *Same closed backyard as in II but with artificially obstructed LOS path*. It is expected that the metallic objects generate serious multipath and higher order scattering that can only be observed within the dynamic range of the device if the LOS path is obstructed.

- **Scenario IV**: *Same as Scenario III but with different TX position and LOS obstructed*. The TX is situated nearer to the walls. Figure 14.37 (b) gives the measured power delay profile. The PDPs in Scenarios II and III look similar, besides the LOS component that occurs dominantly for Scenario II. More details about the scenarios can be found in [62].

Some of the measurement results for these scenarios are presented in Figure 14.39.

### 14.7 INDOOR WLAN CHANNEL (17 GHz)

In this section we discuss the indoor radio propagation channel at 17 GHz. The presentation is based on results reported in [63]. Wideband parameters, such as coherence bandwidth or rms delay spread, and coverage are analyzed for the design of an OFDM-based broadband WLAN. The method used to obtain the channel parameters is based on using a simulator described in [63]. This simulator is a site-specific propagation model based on three-dimensional (3D) ray-tracing techniques, which has been specifically developed for simulating radio coverage and channel performance in enclosed spaces such as buildings, and for urban microcell and picocell calculations. The simulator requires the input of the geometric structure and the electromagnetic properties of the propagation environment, and is based on a full 3D implementation of geometric optics and the uniform theory of diffraction (GO/UTD). Examples of the measurement environments are given in Figure 14.40.

The results for coherence bandwidth $B_c = 1/\alpha \tau_{\text{rms}}$ are given in Table 14.16 and Figure 14.41.
Figure 14.39 (a) Azimuth power spectra at the transmitter (upper plot) and receiver (lower plot) for Scenario III (obstructed LOS). Spectra computed with MVM (minimum variance method, Capon’s beamformer). Angles refer to array broadside, so that (due to the array position) +8 and −53 degrees correspond [57]; (b) cdfs of the MIMO channel capacity encountered in Scenarios I–IV, and the cdf for an ideal channel. The SNR is 20 dB, and 4 × 4 antenna elements were used; (c) outage capacity at the 10% level in Scenarios I (LOS) and IV (LOS) over the number of antenna element pairs; (d) capacity of a 4 × 4 antenna arrangement in Scenario I at different bandwidths and SNR = 20 dB (capacity is normalized to unit bandwidth); (e) capacity distribution for the narrowband case (dashed) and 100 MHz bandwidth (solid) and 10 dB SNR in Scenario I for array sizes $N_T = N_R = 1, 2, 4, 8$. © 2002, IEEE.
Figure 14.39 (Cont.).
Figure 14.40 (a) Hall (49 m × 26 m) [63]; (b) Floors −2 and −3 below top T (34 m × 20 m) [63]; (c) office building (72 m × 38 m), 3D representation [63] © 2002, IEEE.
A further requirement related to the correct and efficient channel estimation process by the receiver is the selection of a number of subcarriers in OFDM satisfying the condition of being separated between approximately $B_c/5$ and $B_c/10$. Results for delay spread are shown in Figure 14.42 and Tables 14.17–14.20.

Results for the path loss exponent are given in Figure 14.43 and Tables 14.21 and 14.22.

For channel modeling purposes, the mean power of the received signal will be represented as

$$P_{RX|dB} = P_{TX|dB} + G_{TX|dB} + G_{RX|dB} - L_{fs|dB} + 10 \cdot \log \left( \int_0^\infty PDP(t) \cdot dt \right)$$

(14.55)

where $P_{TX}$ is the mean power at the transmitting antenna input, $G_{TX}$ is the transmitting antenna gain, while $G_{RX}$ is the receiving antenna gain. $L_{fs}$ is free space propagation loss, given by

$$L_{fs|dB} = 32.45 \text{ dB} + 20 \cdot \log_{10}(d_{Km} + f_{MHz})$$
Figure 14.42 (a) RMS delay spread CDF ($B_c = 1/\alpha \tau_{\text{rms}}$); (b) maximum delay CDF, 30 dB criterion; (c) maximum delay CDF, 20 dB criterion; (d) alpha CDF.
Figure 14.42 (Cont.).

Table 14.17 RMS delay spread CDF for Figure 14.42(a) [63]

<table>
<thead>
<tr>
<th>CDF value</th>
<th>RDS value (ns)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.2</td>
<td>12.1</td>
</tr>
<tr>
<td>0.4</td>
<td>14.3</td>
</tr>
<tr>
<td>0.6</td>
<td>17.5</td>
</tr>
<tr>
<td>0.8</td>
<td>34.3</td>
</tr>
<tr>
<td>1</td>
<td>58.3</td>
</tr>
</tbody>
</table>

Table 14.18 Maximum delay CDF, 30 dB criterion (Figure 14.42(b)) [63]

<table>
<thead>
<tr>
<th>CDF value</th>
<th>$T_{\text{max}}$ value (ns)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.2</td>
<td>62</td>
</tr>
<tr>
<td>0.4</td>
<td>76</td>
</tr>
<tr>
<td>0.6</td>
<td>101</td>
</tr>
<tr>
<td>0.8</td>
<td>122</td>
</tr>
<tr>
<td>1</td>
<td>197</td>
</tr>
</tbody>
</table>

Table 14.19 Maximum delay CDF, 20 dB criterion (Figure 14.42(c)) [63]

<table>
<thead>
<tr>
<th>CDF value</th>
<th>$T_{\text{max}}$ value (ns)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.2</td>
<td>51</td>
</tr>
<tr>
<td>0.4</td>
<td>56</td>
</tr>
<tr>
<td>0.6</td>
<td>69</td>
</tr>
<tr>
<td>0.8</td>
<td>94</td>
</tr>
<tr>
<td>1</td>
<td>156</td>
</tr>
</tbody>
</table>
Table 14.20  Alpha CDF, $B_C = 1/\alpha_{\text{rms}}$ (Figure 14.42(d))

<table>
<thead>
<tr>
<th>CDF value</th>
<th>Alpha value</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.2</td>
<td>2.17</td>
</tr>
<tr>
<td>0.4</td>
<td>2.67</td>
</tr>
<tr>
<td>0.6</td>
<td>3.75</td>
</tr>
<tr>
<td>0.8</td>
<td>4.44</td>
</tr>
<tr>
<td>1</td>
<td>5.78</td>
</tr>
</tbody>
</table>

Figure 14.43  CDF of path loss exponent ‘n’.

Table 14.21  Mean values of ‘n’

<table>
<thead>
<tr>
<th>Type of path</th>
<th>LOS</th>
<th>OLOS</th>
<th>NLOS</th>
</tr>
</thead>
<tbody>
<tr>
<td>‘n’ mean value</td>
<td>1.68</td>
<td>2.14</td>
<td>2.61</td>
</tr>
</tbody>
</table>

Table 14.22  Fading statistic over distance, LOS case

<table>
<thead>
<tr>
<th>Radius (m)</th>
<th>$K$ factor</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>17</td>
</tr>
<tr>
<td>5</td>
<td>10</td>
</tr>
<tr>
<td>6</td>
<td>9</td>
</tr>
<tr>
<td>7</td>
<td>8</td>
</tr>
<tr>
<td>8</td>
<td>6</td>
</tr>
<tr>
<td>9</td>
<td>5</td>
</tr>
<tr>
<td>10</td>
<td>1</td>
</tr>
</tbody>
</table>
and PDP(t) is the modeled power delay profile. Once the PDF is modeled, to obtain the discrete channel impulse response, \( h_i \), we only have to add a random phase to the square root of each delay bin amplitude, as follows:

\[
h_i = \sqrt{p_i} e^{j\phi_i} \quad \text{r.v.} \quad \text{unif}[0, 2\pi]
\]  

(14.56)

where \( h_i \) is the \( i \)th bin of the modeled channel impulse response and \( p_i \), the module of the \( i \)th bin of the modeled power delay profile.

It can be assumed that phases of different components of the same channel impulse response are uncorrelated at the frequency of interest (17 GHz), because their relative range is higher than a wavelength, even for high resolution models [64].

As the total bandwidth assigned to the communication is 50 MHz, a selection of 10 ns for the bin size must be made. Using 99% of the total power criterion for the maximum duration of the PDF, the former bin size selection leads to a total of nine taps for the LOS case and seventeen for the NLOS case.

The statistical variability of the bin amplitudes has been modeled following different probability density functions. Taking into account the fact that the area of service of future applications (SOHO–small office home office) has small ranges, the variability has been analyzed considering a medium scale, that is, the environment is divided in the LOS area and the NLOS one. In the LOS case, a Frechet PDF [1218] is chosen for the first bin and exponential PDFs for the rest.

A continuous random variable \( X \) has a Frechet distribution if its PDF has the form

\[
f(x; \sigma; \lambda) = \frac{\lambda}{\sigma} \left( \frac{x}{\sigma} \right)^{\lambda+1} \exp\left\{ -\left( \frac{x}{\sigma} \right)^{\lambda} \right\} \quad x \geq 0; \quad \sigma, \lambda > 0
\]  

(14.57)

A Frechet variable \( X \) has the CDF

\[
F(x; \sigma; \lambda) = \exp\left\{ -\left( \frac{x}{\sigma} \right)^{\lambda} \right\}
\]  

(14.58)

This model has a scale structure, with \( \sigma \) a scale parameter and \( \lambda \) a shape parameter. A continuous random variable \( X \) has an exponential distribution if its PDF has the form

\[
f(x; \mu) = \frac{1}{\sigma} \exp\left\{ -\left( \frac{x-\mu}{\sigma} \right) \right\} \quad x \geq 0; \quad \mu, \sigma > 0
\]  

(14.59)

This PDF has a location–scale structure, with a location parameter, \( \mu \), and a scale, \( \sigma \). The CDF of the exponential variable \( X \) is

\[
F(x; \mu) = 1 - \exp\left\{ -\left( \frac{x-\mu}{\sigma} \right) \right\}
\]  

(14.60)

These PDFs were considered the most suitable after a fitting process.

The NLOS case needs a combination of exponential and Weibull PDFs for the first bin and exponential PDFs for the others. A continuous random variable \( X \) has a Weibull distribution if its PDF has the form

\[
f(x; \sigma; \lambda) = \frac{\lambda}{\sigma} \left( \frac{x}{\sigma} \right)^{\lambda-1} \exp\left\{ -\left( \frac{x}{\sigma} \right)^{\lambda} \right\} \quad x \geq 0; \quad \sigma, \lambda > 0
\]  

(14.61)

While the CDF is

\[
F(x; \sigma; \lambda) = 1 - \exp\left\{ -\left( \frac{x}{\sigma} \right)^{\lambda} \right\}
\]  

(14.62)

This model has a scale structure, that is, \( \sigma \) is a scale parameter, while \( \lambda \) is a shape parameter. Tables 14.23 and 14.24 show the probability density functions employed for LOS and NLOS channel models [63]. For both tables, the units of parameter \( \sigma \) are Hz (s\(^{-1}\)), while \( \lambda \) has no units. These
units have no physical correlation but make the last term of Equation (14.55) non-dimensional, as it represents a factor scale between the free space behavior and the real one. The mean value of the probability density functions is so high due to the ulterior integral over the time (in seconds) required, and the PDF duration (tens of nanoseconds). As expected, the mean value of the first bin is the highest, since it includes the direct ray (LOS case).

Additional details on the topic can be found in [64–76].

14.8 INDOOR WLAN CHANNEL (60 GHZ)

Based on the results reported in [77], in this section we present spatial and temporal characteristics of 60 GHz indoor channels. In the experiment, a mechanically steered directional antenna is used to resolve multipath components. An automated system is used to position the receiver antenna precisely along a linear track and then rotate the antenna in the azimuthal direction as illustrated in Figure 14.44. Precision of the track and spin positions is less than 1 mm and 1°, respectively. When a highly directional antenna is used, the system provides high spatial resolution to resolve multipath components with different angles of arrival (AOAs). The sliding correlator technique was used to further resolve multipath components with the same AOA by their times of arrival (TOAs).
The spread spectrum signal has an RF bandwidth of 200 MHz, which provides a time resolution of approximately 10 ns.

For this measurement campaign, an open-ended waveguide with 6.7 dB gain is used as the transmitter antenna, and a horn antenna with 29 dB gain is used as the receiver antenna. These antennas are chosen to emulate typical antenna systems that have been proposed for millimeter-wave indoor applications. In these applications, a sector antenna is used at the transmitter and a highly directional antenna is used at the receiver. Both antennas are vertically polarized and mounted on adjustable tripods about 1.6 m above the ground. The theoretical half power beamwidths (HPBW) are 90° in azimuth and 125° in elevation for the open-ended waveguide, and 7° in azimuth and 5.6° in elevation for the horn antenna. Some measurement results for specific environments and locations are shown in Figure 14.45(a)–(c).

14.8.1 Definition of the statistical parameters

14.8.1.1 Path loss and received signal power

The free space path loss at a reference distance of \(d_0\) is given by

\[
\overline{PL}_{fs}(d_0) = 20 \log \left( \frac{4\pi d_0}{\lambda} \right)
\]  
(14.63)

where \(\lambda\) is the wavelength. Path loss over distance \(d\) can be described by the path loss exponent model as follows:

\[
\overline{PL}(d) [\text{dB}] = PL_{fs}(d_0) [\text{dB}] + 10n \log_{10} \left( \frac{d}{d_0} \right)
\]  
(14.64)

where \(\overline{PL}(d)\) is the average path loss value at a TR separation of \(d\), and \(n\) is the path loss exponent that characterizes how fast the path loss increases with the increase of TR separation. The path loss values represent the signal power loss from the transmitter antenna to the receiver antenna. These path loss values do not depend on the antenna gains or the transmitted power levels. For any given transmitted power, the received signal power can be calculated as

\[
P_r(\text{dBm}) = P_t(\text{dBm}) + G_t(\text{dB}) + G_r(\text{dB}) - \overline{PL}(d) [\text{dB}]
\]  
(14.65)
Figure 14.45 (a) AOA measurements for propagation within a room (location 4), relative power levels are shown on the polar plots, and peak multipath power \( P \) is given in the text boxes. Rays are shown only for locations 4.2 and 4.4 in the figure, although a similar procedure can be performed for all the locations; (b) AOA measurements for propagation along a hallway (location 2), relative power levels are given in the polar plots, and peak multipath power \( P \) is given in the text at the bottom of the figure; (c) AOA measurements for propagation into rooms (locations 5 and 6), relative power levels are given in the polar plots, and peak multipath power \( P \) is given in the text boxes [77] © 2002, IEEE.
where $G_t$ and $G_r$ are transmitter and receiver gains, respectively. In this measurement campaign, the transmitted power level was 25 dBm, the transmitter antenna gain was 6.7 dB, and the receiver antenna gain was 29 dB.

### 14.8.1.2 TOA parameters

TOA parameters characterize the time dispersion of a multipath channel. The calculated TOA parameters include mean excess delay ($\bar{\tau}$), rms delay spread ($\sigma_\tau$), and also timing jitter ($\delta(x)$) and standard deviation ($\Delta(x)$), in a small local area. Parameters $\bar{\tau}$ and $\sigma_\tau$ are given as [78]:

\[
\bar{\tau} = \frac{\sum_{i=1}^{N} P_i \tau_i}{\sum_{i=1}^{N} P_i} \quad (14.66)
\]

\[
\sigma_\tau = \sqrt{\bar{\tau}^2 - (\bar{\tau})^2} \quad (14.67)
\]

\[
\bar{\tau}^2 = \frac{\sum_{i=1}^{N} P_i \tau_i^2}{\sum_{i=1}^{N} P_i} \quad (14.68)
\]
where $P_i$ and $\tau_i$ are the power and delay of the $i$th multipath component of a PDF, respectively, and $N$ is the total number of multipath components. Timing jitter is calculated as the difference between the maximum and minimum measured values in a local area. Timing jitter $\delta(x)$ and standard deviation $\Delta(x)$ are defined as

$$\delta(x) = \max_{i=1}^{M} \{x_i\} - \min_{i=1}^{M} \{x_i\}$$

(14.69)

$$\Delta(x) = \sqrt{x^2 - (\bar{x})^2}$$

(14.70)

$$\bar{x} = \frac{1}{M} \sum_{i=1}^{M} x_i$$

(14.71)

$$x^2 = \frac{1}{M} \sum_{i=1}^{M} x_i^2$$

(14.72)

where $x_i$ is the measured value for parameter $x$ (either $\bar{\tau}$ or $\sigma_\tau$) in the $i$th measurement position of the spatial sampling, and $M$ is the total number of spatial samples in the local area. For example, for the track measurements, $M$ was chosen to be 80.

Mean excess delay and rms delay spread are the statistical measures of the time dispersion of the channel. Timing jitter and standard deviation of $\bar{\tau}$ and $\sigma_\tau$ show the variation of these parameters over the small local area.

These TOA parameters directly affect the performance of high-speed wireless systems. For instance, the mean excess delay can be used to estimate the search range of RAKE receivers and the rms delay spread can be used to determine the maximum transmission data rate in the channel without equalization. The timing jitter and standard deviation parameters can be used to determine the update rate for a RAKE receiver or an equalizer.

### 14.8.1.3 AOA parameters

AOA parameters characterize the directional distribution of multipath power. The recorded AOA parameters include angular spread $\Lambda$, angular constriction $\gamma$, maximum fading angle $\theta_{\max}$, and maximum AOA direction. Angular parameters $\Lambda$, $\gamma$ and $\theta_{\max}$ are defined based on the Fourier transform of the angular distribution of multipath power, $p(\theta)$ [79]:

$$\Lambda = \sqrt{1 - \frac{||F_1||^2}{||F_2||^2}}$$

(14.73)

$$\gamma = \frac{||F_0F_2 - F_1^2||}{||F_0||^2 - ||F_1||^2}$$

(14.74)

$$\theta_{\max} = \frac{1}{2} \operatorname{phase} \left\{ F_0F_2 - F_1^2 \right\}$$

(14.75)

where

$$F_n = \int_{0}^{2\pi} p(\theta) \exp(jn\theta) \, d\theta$$

(14.76)

$F_n$ is the $n$th Fourier transform of $p(\theta)$. As shown in [79], angular spread, angular constriction and maximum fading angle are three key parameters to characterize the small-scale fading behavior of the channel. These new parameters can be used for diversity techniques, fading rate estimation, and other space–time techniques. The maximum AOA provides the direction of the multipath component with the maximum power. It can be used in system installation to minimize the path loss. The results of measurements for the parameters defined by Equations (14.63–14.76) are given in Tables 14.25–14.27 and Figure 14.46.

More details on the topic can be found in [79–90].
14.9 UWB CHANNEL MODEL

UWB channel parameters were discussed to some extent in Chapter 8. In this section we will present some additional results with the focus on channel modeling, mainly based on [91].

The measurements environment is presented in Figure 14.47 and the signal format used in these experiments in Figure 14.48. The repetition rate of the pulses is $2 \times 10^6$ pulses per second, implying that multipath spreads up to 500 ns could have been observed unambiguously. Multipath profiles with a duration of 300 ns were measured. Sample results are shown in Figure 14.49. Multipath profiles were measured at various locations in 14 rooms and hallways on one floor of the building presented in Figure 14.47. Each of the rooms is labeled alphanumerically. Walls around offices are framed with metal studs and covered with plasterboard. The wall around the laboratory is made from acoustically silenced heavy cement block. There are steel core support pillars throughout the building, notably along the outside wall and two within the laboratory itself. The shield room’s walls and door are metallic. The transmitter is kept stationary in the central location of the building near a computer server in a laboratory denoted by F. The transmit antenna is located 165 cm from the floor and 105 cm from the ceiling.

**Table 14.25** Spin measurements: transmitter–receiver separations (TR) in m, time dispersion parameters ($\bar{\tau}$ and $\sigma_\tau$) in ns, angular dispersion parameters ($\Lambda$ and $\gamma$) are dimensionless, maximum fading angle ($\theta_{\text{max}}$) and AOA of maximum multipath (max AOA) in degrees, ratio of maximum multipath power to average power (Peak/avg) in dB and maximum multipath power ($P_{\text{max}}$) in dBm [77]

<table>
<thead>
<tr>
<th>Site information</th>
<th>#</th>
<th>TR</th>
<th>$\bar{\tau}$</th>
<th>$\sigma_\tau$</th>
<th>$\Lambda$</th>
<th>$\gamma$</th>
<th>$\theta_{\text{max}}$ (max AOA)</th>
<th>Peak/avg</th>
<th>$P_{\text{max}}$</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOS, hallway</td>
<td>1.1</td>
<td>5</td>
<td>80.0</td>
<td>14.7</td>
<td>0.46</td>
<td>0.83</td>
<td>$-80.7$</td>
<td>$-4.0$</td>
<td>12.3</td>
<td>$-14.9$</td>
</tr>
<tr>
<td></td>
<td>1.2</td>
<td>10</td>
<td>52.0</td>
<td>18.8</td>
<td>0.44</td>
<td>0.74</td>
<td>$-86.6$</td>
<td>4.0</td>
<td>12.0</td>
<td>$-18.2$</td>
</tr>
<tr>
<td>Durham Hall</td>
<td>1.3</td>
<td>20</td>
<td>85.9</td>
<td>40.1</td>
<td>0.56</td>
<td>0.28</td>
<td>$-61.9$</td>
<td>8.0</td>
<td>14.5</td>
<td>$-28.8$</td>
</tr>
<tr>
<td></td>
<td>1.4</td>
<td>30</td>
<td>116.6</td>
<td>38.7</td>
<td>0.42</td>
<td>0.22</td>
<td>$-66.4$</td>
<td>5.0</td>
<td>14.7</td>
<td>$-28.3$</td>
</tr>
<tr>
<td></td>
<td>1.5</td>
<td>40</td>
<td>84.9</td>
<td>60.0</td>
<td>0.25</td>
<td>0.69</td>
<td>4.3</td>
<td>5.0</td>
<td>13.9</td>
<td>$-38.2$</td>
</tr>
<tr>
<td></td>
<td>1.6</td>
<td>50</td>
<td>52.1</td>
<td>26.1</td>
<td>0.66</td>
<td>0.26</td>
<td>8.2</td>
<td>10.0</td>
<td>13.3</td>
<td>$-38.2$</td>
</tr>
<tr>
<td></td>
<td>1.7</td>
<td>60</td>
<td>53.2</td>
<td>30.3</td>
<td>0.78</td>
<td>0.36</td>
<td>4.0</td>
<td>2.0</td>
<td>13.2</td>
<td>$-40.8$</td>
</tr>
<tr>
<td></td>
<td>2.1</td>
<td>5</td>
<td>51.0</td>
<td>20.7</td>
<td>0.48</td>
<td>0.88</td>
<td>$-73.5$</td>
<td>5.0</td>
<td>12.5</td>
<td>$-13$</td>
</tr>
<tr>
<td></td>
<td>2.2</td>
<td>10</td>
<td>62.1</td>
<td>29.4</td>
<td>0.66</td>
<td>0.79</td>
<td>$-72.3$</td>
<td>21.0</td>
<td>11.4</td>
<td>$-21.7$</td>
</tr>
<tr>
<td></td>
<td>2.3</td>
<td>20</td>
<td>90.7</td>
<td>14.6</td>
<td>0.36</td>
<td>0.43</td>
<td>$-73.8$</td>
<td>4.0</td>
<td>12.9</td>
<td>$-29.8$</td>
</tr>
<tr>
<td></td>
<td>2.4</td>
<td>30</td>
<td>41.2</td>
<td>12.3</td>
<td>0.41</td>
<td>0.15</td>
<td>$-64.8$</td>
<td>10.0</td>
<td>13.8</td>
<td>$-31.7$</td>
</tr>
<tr>
<td></td>
<td>2.5</td>
<td>40</td>
<td>83.7</td>
<td>53.8</td>
<td>0.72</td>
<td>0.19</td>
<td>5.0</td>
<td>1.0</td>
<td>13.2</td>
<td>$-36.0$</td>
</tr>
<tr>
<td>LOS, room</td>
<td>3.1</td>
<td>4</td>
<td>42.6</td>
<td>16.2</td>
<td>0.86</td>
<td>0.64</td>
<td>$-79.2$</td>
<td>0.0</td>
<td>12.5</td>
<td>$-11.8$</td>
</tr>
<tr>
<td>Durham Hall</td>
<td>3.2</td>
<td>3</td>
<td>47.7</td>
<td>17.5</td>
<td>0.81</td>
<td>0.70</td>
<td>$-79.1$</td>
<td>5.0</td>
<td>13.1</td>
<td>$-12.1$</td>
</tr>
<tr>
<td></td>
<td>4.1</td>
<td>7</td>
<td>46.6</td>
<td>13.0</td>
<td>0.84</td>
<td>0.55</td>
<td>$-88.0$</td>
<td>$-60.0$</td>
<td>12.3</td>
<td>$-26.8$</td>
</tr>
<tr>
<td>LOS, room</td>
<td>4.2</td>
<td>3</td>
<td>64.3</td>
<td>13.3</td>
<td>0.62</td>
<td>0.74</td>
<td>$-89.6$</td>
<td>$-1.0$</td>
<td>13.1</td>
<td>$-25.6$</td>
</tr>
<tr>
<td>Whittemore</td>
<td>4.3</td>
<td>5</td>
<td>66.3</td>
<td>17.7</td>
<td>0.73</td>
<td>0.84</td>
<td>$-35.2$</td>
<td>49.0</td>
<td>14.0</td>
<td>$-30.4$</td>
</tr>
<tr>
<td></td>
<td>4.4</td>
<td>2</td>
<td>77.8</td>
<td>13.3</td>
<td>0.78</td>
<td>0.72</td>
<td>$-38.2$</td>
<td>$-49.0$</td>
<td>14.2</td>
<td>$-28.6$</td>
</tr>
<tr>
<td></td>
<td>5.1</td>
<td>2</td>
<td>49.1</td>
<td>21.4</td>
<td>0.81</td>
<td>0.13</td>
<td>$-76.3$</td>
<td>0.0</td>
<td>12.0</td>
<td>$-6.0$</td>
</tr>
<tr>
<td></td>
<td>5.2</td>
<td>2</td>
<td>41.6</td>
<td>18.1</td>
<td>0.74</td>
<td>0.44</td>
<td>$-89.6$</td>
<td>5.0</td>
<td>10.3</td>
<td>$-14.1$</td>
</tr>
<tr>
<td></td>
<td>5.3</td>
<td>2</td>
<td>95.8</td>
<td>14.6</td>
<td>0.63</td>
<td>0.40</td>
<td>$-88.1$</td>
<td>0.0</td>
<td>12.1</td>
<td>$-5.6$</td>
</tr>
<tr>
<td></td>
<td>5.4</td>
<td>2</td>
<td>80.3</td>
<td>16.0</td>
<td>0.68</td>
<td>0.27</td>
<td>72.3</td>
<td>5.0</td>
<td>11.9</td>
<td>$-8.9$</td>
</tr>
<tr>
<td>Hallway to room</td>
<td>6.1</td>
<td>3</td>
<td>42.7</td>
<td>16.6</td>
<td>0.80</td>
<td>0.40</td>
<td>$-25.3$</td>
<td>52.0</td>
<td>11.5</td>
<td>$-36.4$</td>
</tr>
<tr>
<td></td>
<td>7.1</td>
<td>1</td>
<td>41.3</td>
<td>17.4</td>
<td>0.12</td>
<td>0.97</td>
<td>$-81.2$</td>
<td>2.0</td>
<td>13.9</td>
<td>$-15.0$</td>
</tr>
<tr>
<td>LOS, outdoor</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>parking lot</td>
<td>7.2</td>
<td>1</td>
<td>56.6</td>
<td>16.1</td>
<td>0.49</td>
<td>0.94</td>
<td>$-66.7$</td>
<td>20.0</td>
<td>8.5</td>
<td>$-29.9$</td>
</tr>
<tr>
<td>LOS, outdoor</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>8.1</td>
<td>2</td>
<td>24.4</td>
<td>7.7</td>
<td>0.26</td>
<td>0.76</td>
<td>$-66.3$</td>
<td>3.0</td>
<td>13.9</td>
<td>$-10.1$</td>
</tr>
</tbody>
</table>
Table 14.26  Track measurement results: TR separations (TR) in m, time dispersion parameters ($\bar{\tau}$ and $\sigma_{\tau}$) in ns, variations of time dispersion parameters ($\delta \bar{\tau}$, $\Delta \bar{\tau}$, $\delta \sigma_{\tau}$ and $\Delta \sigma_{\tau}$) in ns and average received power ($P_{rx}$) in dBm [77] © 2002, IEEE

<table>
<thead>
<tr>
<th>Site information</th>
<th>LOC #</th>
<th>TR</th>
<th>$\bar{\tau}$</th>
<th>$\sigma_{\tau}$</th>
<th>$\delta \bar{\tau}$</th>
<th>$\Delta \bar{\tau}$</th>
<th>$\delta \sigma_{\tau}$</th>
<th>$\Delta \sigma_{\tau}$</th>
<th>$P_{rx}$</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOS, hallway</td>
<td>1.1</td>
<td>5</td>
<td>1.20</td>
<td>6.95</td>
<td>6.33</td>
<td>1.91</td>
<td>1.20</td>
<td>0.29</td>
<td>−13.7</td>
<td></td>
</tr>
<tr>
<td>Durham Hall</td>
<td>1.2</td>
<td>10</td>
<td>6.16</td>
<td>5.88</td>
<td>5.06</td>
<td>1.20</td>
<td>6.16</td>
<td>1.73</td>
<td>−20.3</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1.3</td>
<td>20</td>
<td>32.61</td>
<td>47.25</td>
<td>32.89</td>
<td>8.43</td>
<td>32.61</td>
<td>9.02</td>
<td>−36.6</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1.4</td>
<td>30</td>
<td>15.50</td>
<td>31.15</td>
<td>10.16</td>
<td>3.43</td>
<td>15.50</td>
<td>5.69</td>
<td>−31.2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1.5</td>
<td>40</td>
<td>27.60</td>
<td>37.04</td>
<td>25.89</td>
<td>8.81</td>
<td>27.60</td>
<td>9.76</td>
<td>−40.5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1.6</td>
<td>50</td>
<td>46.42</td>
<td>28.17</td>
<td>36.70</td>
<td>8.10</td>
<td>46.42</td>
<td>10.73</td>
<td>−42.8</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1.7</td>
<td>60</td>
<td>6.38</td>
<td>22.57</td>
<td>5.99</td>
<td>1.82</td>
<td>6.38</td>
<td>1.57</td>
<td>−41.5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.1</td>
<td>5</td>
<td>2.22</td>
<td>6.24</td>
<td>7.52</td>
<td>2.38</td>
<td>2.22</td>
<td>0.73</td>
<td>−16.7</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.2</td>
<td>10</td>
<td>2.78</td>
<td>6.48</td>
<td>8.24</td>
<td>2.61</td>
<td>2.78</td>
<td>0.82</td>
<td>−24.4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.3</td>
<td>20</td>
<td>2.3</td>
<td>4.56</td>
<td>7.81</td>
<td>2.55</td>
<td>2.30</td>
<td>0.55</td>
<td>−32.86</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.4</td>
<td>30</td>
<td>22.02</td>
<td>33.87</td>
<td>13.17</td>
<td>4.60</td>
<td>22.02</td>
<td>6.30</td>
<td>−34.7</td>
<td></td>
</tr>
<tr>
<td>LOS, room</td>
<td>2.5</td>
<td>40</td>
<td>77.3</td>
<td>45.07</td>
<td>105.04</td>
<td>34.41</td>
<td>77.30</td>
<td>25.86</td>
<td>−36.3</td>
<td></td>
</tr>
<tr>
<td>Durham Hall</td>
<td>3.1</td>
<td>4.2</td>
<td>0.74</td>
<td>4.85</td>
<td>6.20</td>
<td>1.88</td>
<td>0.74</td>
<td>0.20</td>
<td>−12.1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3.2</td>
<td>4.2</td>
<td>0.92</td>
<td>4.95</td>
<td>5.97</td>
<td>1.87</td>
<td>0.92</td>
<td>0.23</td>
<td>−12.9</td>
<td></td>
</tr>
<tr>
<td>Whittemore</td>
<td>4.1</td>
<td>7.1</td>
<td>2.74</td>
<td>4.72</td>
<td>11.16</td>
<td>3.08</td>
<td>2.47</td>
<td>0.36</td>
<td>−29.7</td>
<td></td>
</tr>
<tr>
<td></td>
<td>4.2</td>
<td>3.8</td>
<td>2.4</td>
<td>4.98</td>
<td>11.11</td>
<td>3.17</td>
<td>2.40</td>
<td>0.47</td>
<td>−24.2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>4.3</td>
<td>5.2</td>
<td>12.88</td>
<td>31.10</td>
<td>26.36</td>
<td>6.86</td>
<td>12.88</td>
<td>2.95</td>
<td>−56.2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>4.4</td>
<td>4.2</td>
<td>21.3</td>
<td>33.94</td>
<td>31.5</td>
<td>7.4</td>
<td>21.3</td>
<td>5.43</td>
<td>−57.9</td>
<td></td>
</tr>
<tr>
<td>Whittemore</td>
<td>5.1</td>
<td>2.4</td>
<td>0.83</td>
<td>5.50</td>
<td>2.41</td>
<td>0.69</td>
<td>0.83</td>
<td>0.32</td>
<td>−5.5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>5.2</td>
<td>2.4</td>
<td>2.46</td>
<td>7.41</td>
<td>2.61</td>
<td>0.84</td>
<td>2.46</td>
<td>0.94</td>
<td>−14.3</td>
<td></td>
</tr>
<tr>
<td></td>
<td>5.3</td>
<td>2.4</td>
<td>0.71</td>
<td>5.36</td>
<td>1.30</td>
<td>0.41</td>
<td>0.71</td>
<td>0.25</td>
<td>−6.7</td>
<td></td>
</tr>
<tr>
<td></td>
<td>5.4</td>
<td>2.4</td>
<td>1.16</td>
<td>5.19</td>
<td>1.85</td>
<td>0.61</td>
<td>1.16</td>
<td>0.36</td>
<td>−9.1</td>
<td></td>
</tr>
<tr>
<td>Hallway to room</td>
<td>6.1</td>
<td>3</td>
<td>10.67</td>
<td>14.72</td>
<td>23.07</td>
<td>6.62</td>
<td>10.67</td>
<td>1.30</td>
<td>−12.8</td>
<td></td>
</tr>
<tr>
<td></td>
<td>6.2</td>
<td>3</td>
<td>14.82</td>
<td>21.78</td>
<td>34.30</td>
<td>8.57</td>
<td>14.82</td>
<td>3.37</td>
<td>−48.3</td>
<td></td>
</tr>
<tr>
<td>Room to room</td>
<td>8.1</td>
<td>2</td>
<td>7.63</td>
<td>24.59</td>
<td>10.24</td>
<td>2.66</td>
<td>7.63</td>
<td>1.75</td>
<td>−2.4</td>
<td></td>
</tr>
<tr>
<td>LOS, outdoor</td>
<td>8.1</td>
<td>2</td>
<td>7.63</td>
<td>24.59</td>
<td>10.24</td>
<td>2.66</td>
<td>7.63</td>
<td>1.75</td>
<td>−2.4</td>
<td></td>
</tr>
</tbody>
</table>

Table 14.27  Measured penetration losses and results from literature

<table>
<thead>
<tr>
<th>Material</th>
<th>Penetration loss (dB)</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Composite wall with studs not in the path</td>
<td>8.8</td>
<td>[1230]</td>
</tr>
<tr>
<td>Composite wall with studs in the path</td>
<td>35.5</td>
<td>[1230]</td>
</tr>
<tr>
<td>Glass</td>
<td>2.5</td>
<td>[1230]</td>
</tr>
<tr>
<td>Concrete wall one week after concreting</td>
<td>73.6</td>
<td>[1233]</td>
</tr>
<tr>
<td>Concrete wall two weeks after concreting</td>
<td>68.4</td>
<td>[1233]</td>
</tr>
<tr>
<td>Concrete wall five weeks after concreting</td>
<td>46.5</td>
<td>[1233]</td>
</tr>
<tr>
<td>Concrete wall 14 months after concreting</td>
<td>28.1</td>
<td>[1233]</td>
</tr>
<tr>
<td>Plasterboard wall</td>
<td>5.4 to 8.1</td>
<td>[1234]</td>
</tr>
<tr>
<td>Partition of glass wool with plywood surfaces</td>
<td>9.2 to 10.1</td>
<td>[1234]</td>
</tr>
<tr>
<td>Partition of cloth-covered plywood</td>
<td>3.9 to 8.7</td>
<td>[1234]</td>
</tr>
<tr>
<td>Granite with width of 3 cm</td>
<td>&gt;30</td>
<td>[1235]</td>
</tr>
<tr>
<td>Glass</td>
<td>1.7 to 4.5</td>
<td>[1235]</td>
</tr>
<tr>
<td>Metalized glass</td>
<td>&gt;30</td>
<td>[1235]</td>
</tr>
<tr>
<td>Wooden panels</td>
<td>6.2 to 8.6</td>
<td>[1235]</td>
</tr>
<tr>
<td>Brick with width of 11 cm</td>
<td>17</td>
<td>[1235]</td>
</tr>
<tr>
<td>Limestone with width of 3 cm</td>
<td>&gt;30</td>
<td>[1235]</td>
</tr>
<tr>
<td>Concrete</td>
<td>&gt;30</td>
<td>[1235]</td>
</tr>
</tbody>
</table>
Figure 14.46 Scatter plot of the measured path loss values.

Figure 14.47 The floor plan of a typical modern office building where the propagation measurement experiment was performed. The concentric circles are centered on the transmit antenna and are spaced at 1 m intervals [91] © 2002, IEEE.
Figure 14.48 The transmitted pulse measured by the receiving antenna located 1 m away from the transmitting antenna with the same height.

In each receiver location, impulse response measurements were made at 49 measurement points, arranged in a fixed-height, $7 \times 7$ square grid with 15 cm spacing, covering 90 cm $\times$ 90 cm. A total of 741 different impulse responses were recorded. One side of the grid is always parallel to the north wall of the room. The receiving antenna is located 120 cm from the floor and 150 cm from the ceiling. Profiles measured in offices U, W and M are shown in Figure 14.49. The approximate distances between the transmitter and the locations of these measurements are 10, 8.5 and 13.5 m, respectively.
Table 14.28 Symbols and parameters [91] © 2002, IEEE

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \tau )</td>
<td>Excess delay</td>
</tr>
<tr>
<td>( \tau_{\text{Ref}} )</td>
<td>Absolute propagation delay</td>
</tr>
<tr>
<td>( \tau_k = (k - 1)\Delta \tau )</td>
<td>( k )th Delay Bin</td>
</tr>
<tr>
<td>( \Delta \tau = 2 \text{ ns} )</td>
<td>Bin width</td>
</tr>
<tr>
<td>( N_{\text{bins}} )</td>
<td>Number of bins</td>
</tr>
</tbody>
</table>

Energy gains

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \bar{G}_{\text{tot}} )</td>
<td>Total average energy gain</td>
</tr>
<tr>
<td>( \bar{G}_k )</td>
<td>Average energy gain of the ( k )th delay bin</td>
</tr>
<tr>
<td>( G_k )</td>
<td>Energy gain of the ( k )th delay bin</td>
</tr>
<tr>
<td>( \bar{g}(\tau) )</td>
<td>Exponential received energy at excess delay ( \tau )</td>
</tr>
<tr>
<td>( \varepsilon )</td>
<td>Decay constant</td>
</tr>
<tr>
<td>( r = G_2/G_1 )</td>
<td>Power ratio</td>
</tr>
</tbody>
</table>

Figure 14.49 also shows that the response to the first probing pulse has decayed almost completely in roughly 200 ns, and has disappeared before the response to the next pulse arrives at the antenna. The multipath profiles recorded in the offices W and M have a substantially lower noise floor than those recorded in office U. This can be explained, with the help of Figure 14.47, by observing that office U is situated at the edge of the building with a large glass window, and is subject to more external interference (e.g. from radio stations, television stations, cellular and paging towers), while offices W and M are situated roughly in the middle of the building. In general, an increased noise floor was observed for all the measurements made in offices located at the edges of the building with large glass windows.

The large-scale fading characterizes the changes in the received signal when the receiver position varies over a significant fraction of the transmitter–receiver (T–R) distance and/or the environment around the receiver changes. This situation typically occurs when the receiver is moved from one room to another room in a building. The small-scale effects, on the other hand, are manifested in the changes of the PDP caused by small changes of the receiver position, while the environment around the receiver does not change significantly. This occurs, for instance, when the receiver is moved over the measurement grid within a room in a building.

In the following, we refer to the PDP measured at one of the 14(rooms) \( \times \) 49 locations as local PDP, while we denote the PDP averaged over the 49 locations within one room as the small-scale averaged PDP (SSA-PDP). This spatial averaging (mostly) removes the effect of small-scale fading. The small-scale statistics are derived by considering the deviations of the 49 local PDPs from the respective SSA-PDP. The large-scale fading may be investigated by considering the variation of the SSA-PDPs over the different rooms. We also make a distinction between the ‘local’ parameters, which refer to the small-scale effects, and the ‘global’ parameters, extracted from the SSA-PDPs. For clarity, all the symbols and parameters are listed in Table 14.28.

14.9.1 The large-scale statistics

All SSA-PDPs exhibit an exponential decay as a function of the excess delay. Since we perform a delay axis translation, the direct path always falls in the first bin in all the PDPs. It also turns out that the direct path is always the strongest path in the 14 SSA-PDPs, even if the LOS is obstructed. The energy of the subsequent MPCs decays exponentially with delay starting from the second bin. This is illustrated by the fit (linearly on a decibel scale) in Figure 14.50 using the SSA-PDP of a typical high signal to noise ratio (SNR) room. Let \( \bar{G}_k = A_{\text{Spa}} \{G_k\} \) be the locally averaged energy gain, where the \( A_{\text{Spa}} \{\cdot\} \) denotes the spatial average over the 49 locations of the measurement grid. The average energy of the second MPC may be expressed as a fraction \( r \) of the average energy of the
direct path, i.e. $r = \bar{G}_2/\bar{G}_1$. We refer to $r$ as the power ratio. As we will show later, the SSA-PDP is completely characterized by $\bar{G}_1$, the power ratio $r$, and the decay constant $\varepsilon$ (or equivalently, by the total average received energy $\bar{G}_{\text{tot}}$, $r$, and $\varepsilon$). The number of resolved MPCs is given by the number of the MPCs that exceed a threshold and thus, given the threshold, it depends on the shape of the SSA-PDP, characterized by the parameters $\bar{G}_1$, $r$, and $\varepsilon$. Best fit procedures are used to extract the $\varepsilon$s and the $r$s from the SSA-PDP of each room.

The power ratio $r$ and the decay constant $\varepsilon$ vary from location to location, and should be treated as stochastic variables. As only 14 values for $\varepsilon$ and $r$ were available, it was not possible to extract the shape of their distribution from the measurement data. Instead, a model was assumed a priori and the parameters of this distribution were fitted. Previous narrowband studies showed that the decay constants are well modeled as log-normal variables [6]. It was found that the log-normal distribution, denoted by $\varepsilon \sim \mathcal{N}(\mu_{\varepsilon_{\text{dB}}}, \sigma_{\varepsilon_{\text{dB}}})$, with $\mu_{\varepsilon_{\text{dB}}} = 16.1$ and $\sigma_{\varepsilon_{\text{dB}}} = 1.27$ gives the best agreement with the empirical distribution. The histograms of the experimental decay constants and the theoretically fitted distribution are shown in Figure 14.51. Applying the same procedure to characterize the power ratios $r$,
it was found that they are also log-normally distributed, i.e. \( r \sim L_N(\mu_{\text{dB}}, \sigma_{\text{dB}}) \), with \( \mu_{\text{dB}} = -4 \) and \( \sigma_{\text{dB}} = 3 \), respectively.

The possible correlation of the decay constant with the T–R separation was also investigated, by applying a linear regression to the \( \varepsilon_s \) versus the distance. As Figure 14.52 shows, the regression fit of the decay constants \( \varepsilon \) decreases with the increasing distance so slightly that we can conclude that it is de facto independent.

By integrating the SSA-PDP of each room over all delay bins, the total average energy \( \tilde{G}_{\text{tot}} \) within each room is obtained. Then its dependence on the T–R separation is analyzed. As suggested by the scatter plot of Figure 14.53, a breakpoint model, commonly referred to as a dual slope model, can be adopted for path loss PL as a function of the distance. The regression lines are shown in Figure 14.53.
and the parameters extracted by performing a best fit of the empirical attenuation are

$$\text{PL} = \begin{cases} 20.4 \log_{10}(d/d_0), & d \leq 11 \text{ m} \\ -56 + 74 \log_{10}(d/d_0), & d > 11 \text{ m} \end{cases}$$

(14.77)

where PL is expressed in decibels, $d_0 = 1$ m is the reference distance, and $d$ is the T–R separation distance in meters. Because of the shadowing phenomenon, the $\bar{G}_{\text{tot}}$ varies statistically around the value given by Equation (14.77). A common model for shadowing is the log-normal distribution [92, 93]. By assuming such a model, it was found that $\bar{G}_{\text{tot}}$ is log-normally distributed about Equation (14.77) with a standard deviation of the associated normal random variable equal to 4.3.

### 14.9.2 The small-scale statistics

The differences between the PDPs at the different points of the measurement grid are caused by small-scale fading. In ‘narrowband’ models, it is usually assumed that the magnitude of the first (quasi-LOS) multipath component follows Rician or Nakagami statistics and the later components are assumed to have Rayleigh statistics [94]. However, in UWB propagation, each resolved MPC is due to a small number of scatterers, and the amplitude distribution in each delay bin differs markedly from the Rayleigh distribution. In fact, the presented analysis showed that the best fit distribution of the small-scale magnitude statistics is the Nakagami distribution [95], corresponding to a Gamma distribution of the energy gains. This distribution has been used to model the magnitude statistics in mobile radio when the conditions of the central limit theorem are not fulfilled [96].

The small-scale statistics are characterized by fitting the received normalized energies $\{G_k^0\}$ in each bin at the 49 locations of the measurement grid to a distribution. The variations over the measurement grid are treated as stochastic. The result shows that the statistics of the energy gain vary with delays. Let us denote by $\Gamma(\Omega; m)$ the Gamma distribution with parameters $\Omega$ and $m$. The $\Gamma(\Omega; m)$ gives a good fit of the empirical distribution of the energy gains. The accuracy of the fit has been quantified in terms of the relative mean squared error, which varies between 0.0105 (for the highest SNR) to 0.1137 (for the lowest SNR). A comparison between experimental and theoretical histograms for one exemplary bin in a typical high SNR is shown in Figure 14.54.

![Experimental histogram](image1)

**Figure 14.54** Histogram of the received energy in the 34th bin of a typical high SNR room, compared with the theoretical Gamma distribution, whose mean $\Omega_{34}$ and $m_{34}$ were extracted from the experimental PDF. The energies on the horizontal axes are expressed in arbitrary units [91] © 2002, IEEE.
Figure 14.55 Scatter plot of the $m$-Nakagami of the best fit distribution versus excess delay for all the bins except the LOS components. Different markers correspond to measurements in different rooms [91] © 2002, IEEE.

The parameters of the Gamma distribution vary from bin to bin: $\Gamma(\Omega; m)$ denotes the Gamma distribution that fits the energy gains of the local PDPs in the $k$th bin within each room. The $\Omega_k$ are given as $\Omega_k = \bar{G}_k$, i.e. the magnitude of the SSA-PDP in the $k$th bin. The $m_k$ are related to the variance of the energy gain of the $k$th bin. Figure 14.55 shows the scatter plot of the $m_k$, as a function of excess delay for all the bins (except the LOS components). It can be seen from Figure 14.55 that the $m_k$ values range between 1 and 6 (rarely 0.5), decreasing with the increasing excess delay. This implies that MPCs arriving with large excess delays are more diffused than the first arriving components, which agrees with intuition.

The $m_k$ parameters of the Gamma distributions themselves are random variables distributed according to a truncated Gaussian distribution, denoted by $m \sim T_N(\mu_m; \sigma_m^2)$, i.e. their distribution looks like a Gaussian for $m \geq 0.5$ and zero elsewhere

$$f_m(x) = \begin{cases} K_m e^{-(x-\mu_m)^2/2\sigma_m^2}, & \text{if } x \geq 0.5 \\ 0, & \text{otherwise} \end{cases}$$

where the normalization constant $K_m$ is chosen so that the integral over the $f_m(x)$ is unity. Figure 14.56 shows the mean and variance of such Gaussian distributions that fit the $m_k$ as a function of the excess delay, along with the respective regression lines. The regression lines are given by

$$\mu_m(\tau_k) = 3.5 - \frac{\tau_k}{73}$$

$$\sigma_m^2(\tau_k) = 1.84 - \frac{\tau_k}{160}$$

where $\tau_k$ is in nanoseconds.
Figure 14.56 Scatter plot of the mean values (dots) and the variance (circles) of the Gaussian distributions that fit the experimental distribution of $m$ values at each excess delay. The solid lines represent the linear regression for these parameters, respectively [91] © 2002, IEEE.

### 14.9.3 Correlation of MPCs among different delay bins

We next evaluate the correlation between the energy gain of the MPCs arriving in the same room at different excess delays as

$$
\rho_{k,k+m} = \frac{A_{Spa}\{(G_k - \overline{G_k})(G_{k+m} - \overline{G_{k+m}})\}}{\sqrt{A_{Spa}\{(G_k - \overline{G_k})^2\} A_{Spa}\{(G_{k+m} - \overline{G_{k+m}})^2\}}} \tag{14.81}
$$

The analysis shows that the correlation coefficients remain below 0.2 for almost all rooms and delay bins and are, thus, negligible for all practical purposes.

### 14.9.4 The statistical model

The received signal is a sum of the replicas (echoes) of the transmitted signal, being related to the reflecting, scattering and/or deflecting objects via which the signal propagates. Each of the echoes is related to a single such object. In a narrowband system, the echoes at the receiver are only attenuated, phase-shifted and delayed, but undistorted, so that the received signal may be modeled as a linear combination of $N_{\text{path}}$ delayed basic waveforms $w(t)$

$$
r(t) = \sum_{i=1}^{N_{\text{path}}} c_i w(t - \tau_i) + n(t) \tag{14.82}
$$

where $n(t)$ is the observation noise. In UWB systems, the frequency selectivity of the reflection, scattering and/or diffraction coefficients of the objects via which the signal propagates, can lead to a distortion of the transmitted pulses. Furthermore, the distortion and, thus, the shape of the arriving
echoes, varies from echo to echo. The received signal is given as
\[
r(t) = \sum_{i=1}^{N_{\text{path}}} c_i \tilde{w}_i(t - \tau_i) + n(t)
\] (14.83)

If the pulse distortion was greater than the width of the delay bins (2 ns), one would observe a significant correlation between adjacent bins. The fact that the correlation coefficient remains very low for all analyzed sets of the data implies that the distortion of a pulse due to a single echo is not significant, so that in the following, Equation (14.82) can be used. The SSA-PDP of the channel may be expressed as
\[
\bar{g}(\tau) = \sum_{k=1}^{N_{\text{bins}}} \bar{G}_k \delta(\tau - t_k)
\] (14.84)

where the function \(\bar{g}(\tau)\) can be interpreted as the average energy received at a certain receiver position and a delay \(\tau\), normalized to the total energy received at one meter distance, and \(N_{\text{bins}}\) is the total number of bins in the observation window. Assuming an exponential decay starting from the second bin, we have
\[
\bar{g}(\tau) = \bar{G}_1 \delta(\tau - \tau_1) + \sum_{k=2}^{N_{\text{bins}}} \bar{G}_2 \exp \left[-(\tau_k - \tau_2)/\varepsilon\right] \delta(\tau - t_k)
\] (14.85)

where \(\varepsilon\) is the decay constant of the SSA-PDP. The total average energy received over the observation interval \(T\) is:
\[
\bar{G}_{\text{tot}} = \int_0^T \bar{g}(\tau) \, d\tau = \bar{G}_1 + \sum_{k=2}^{N_{\text{bins}}} \bar{G}_2 \exp \left[-(\tau_k - \tau_2)/\varepsilon\right]
\] (14.86)

Summing the geometric series, gives
\[
\bar{G}_{\text{tot}} = \bar{G}_1 [1 + r F(\varepsilon)]
\] (14.87)

where \(r = \bar{G}_2/\bar{G}_1\) is the power ratio, and
\[
F(\varepsilon) = \frac{1 - \exp \left[-(N_{\text{bins}} - 1)\Delta \tau/\varepsilon\right]}{1 - \exp (-\Delta \tau/\varepsilon)} \approx \frac{1}{1 - \exp (-\Delta \tau/\varepsilon)}
\] (14.88)

The total normalized average energy is log-normally distributed, due to the shadowing, around the mean value given from the path loss model in Equation (14.77):
\[
G_{\text{tot}} \sim \mathcal{L}_N(-PL; 4.3)
\] (14.89)

From Equation (14.87), we have, for the average energy gains,
\[
\bar{G}_k = \begin{cases} 
\frac{\bar{G}_{\text{tot}}}{1 + r F(\varepsilon)}, & \text{for } k = 1 \\
\frac{\bar{G}_{\text{tot}}}{1 + r F(\varepsilon)} r e^{-\left(\tau_k - \tau_2\right)/\varepsilon}, & \text{for } k = 2, \ldots, N_{\text{bins}} 
\end{cases}
\] (14.90)

and Equation (14.84) becomes
\[
\bar{g}(\tau) = \frac{\bar{G}_{\text{tot}}}{1 + r F(\varepsilon)} \left\{ \delta(\tau - \tau_1) + \sum_{k=2}^{N_{\text{bins}}} \left[ r e^{-\left(\tau_2 - \tau_2\right)/\varepsilon} \right] \delta(\tau - t_k) \right\}
\] (14.91)

### 14.9.5 Simulation steps

In the model, the local PDF is fully characterized by the pairs \(\{G_k, \tau_k\}\), where \(\tau_k = (k - 1)\Delta \tau\), with \(\Delta \tau = 2\) ns. The \(G_k\) are generated by a superposition of large- and small-scale statistics.
The process starts by generating the total mean energy $\bar{G}_{\text{tot}}$ at a certain distance according to Expression (14.89). Next, the decay constant $\varepsilon$ and the power ratio $r$ are generated as log-normal distributed random numbers

$$\varepsilon \sim \mathcal{LN}(16.1; 1.27) \quad (14.92)$$

$$r \sim \mathcal{LN}(-4; 3) \quad (14.93)$$
Table 14.29  Statistical models and parameters

<table>
<thead>
<tr>
<th>Global parameters ⇒ $\bar{G}_{\text{tot}}$ and $\bar{G}_k$</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Path loss</strong> $\text{PL} =$ $\begin{cases} 20.4 \log_{10}(d/d_0), &amp; d \leq 11 \text{ m} \ -56 + 74 \log_{10}(d/d_0), &amp; d &gt; 11 \text{ m} \end{cases}$</td>
</tr>
<tr>
<td><strong>Shadowing</strong> $\bar{G}_{\text{tot}} \sim \mathcal{L}_N(-\text{PL}; 4.3)$</td>
</tr>
<tr>
<td><strong>Decay constant</strong> $\varepsilon \sim \mathcal{L}_N(16.1; 1.27)$</td>
</tr>
<tr>
<td><strong>Power ratio</strong> $r \sim \mathcal{L}_N(-4; 3)$</td>
</tr>
<tr>
<td><strong>Energy gains</strong> $G_k \sim \Gamma(\bar{G}_k; m_k)$</td>
</tr>
<tr>
<td><strong>m values</strong> $m_k \sim \mathcal{T}_N(\mu_m(\tau_k); \sigma_m^2(\tau_k))$</td>
</tr>
</tbody>
</table>

$\mu_m(\tau_k) = 3.5 - \frac{\tau_k}{73}$

$\sigma_m^2(\tau_k) = 1.84 - \frac{\tau_k}{160}$

Figure 14.58  The measured 49 local PDPs for an exemplary room [91] © 2002, IEEE.

The width of the observation window is set to be $T = 5\varepsilon$. Thus, the SSA-PDP is completely specified according to Equation (14.91). Finally, the local PDPs are generated by computing the normalized energy gains $G_k^{(i)}$ of every bin $k$ and every location $i$ as Gamma distributed independent variables. The Gamma distributions have averages given by Equation (14.90), and the $m_k$’s are generated as independent truncated Gaussian random variables

$$m_k \sim \mathcal{T}_N(\mu_m(\tau_k); \sigma_m^2(\tau_k))$$

with $\mu_m(\tau_k)$ and $\sigma_m^2(\tau_k)$ given by Equations (14.79) and (14.80). These steps are summarized in Figure 14.57 and Table 14.29.

Some results are shown in Figures 14.58–60.
REFERENCES


REFERENCES


REFERENCES


15
Adaptive 4G Networks

15.1 ADAPTIVE MAC LAYER

In this section we discuss a centralized quasi-asynchronous DS-CDMA packet radio network (PRN) with adaptive bit rate and optimal packet size in low mobility environments. The channel load sensing protocol (CLSP) is used to control the packet access on the uplink so that contention is avoided and throughput is maximized. Due to the high uncertainty of radio channels and imperfect power control (PC), the performance of CLSP/DS-CDMA may suffer from notable degradations. The PC problem is essential to the capacity and coverage of interference-limited DS-CDMA systems, in particular on the uplink. It often requires a fast closed loop PC along with an open loop PC for initial power setting. In a PRN, the closed loop PC appears to be impractical due to the connectionless nature of datagram transmissions. So, the near–far effect represents a major factor affecting the design and performance of PC. This section introduces a system with bit rate adaptation, based on a location dependent rate and power (LDRP) resource allocation. The protocol compensates for PC inaccuracy and improves spectral efficiency of the system. The optimal packet size and data rates are derived to enhance robustness of radio transmissions over correlated fading to reduce the inter-cell interference and improve energy efficiency over the protocol overhead. The modeling includes the impacts of radio propagation attenuation, correlated fading, spatial user distribution, user mobility, and traffic load. Based on such modeling, optimal adaptive mechanisms are developed and performance characteristics are derived.

An ALOHA/DS-CDMA PRN [1–6] is a simple and practical system for providing a wide range of communication services for wireless data users in low mobility environments. It is used for both military and commercial applications, such as tactical networks, satellite communications, ad hoc networks, wireless LANs, packet-access domains of cellular networks, etc. The pure ALOHA systems, however, provide a poor throughput-access delay performance. Throughout the years, a significant amount of research effort has been invested in studying the media access control (MAC) for improving those characteristics, resulting in numerous MAC schemes. The centralized quasi-asynchronous (unslotted or spread-slotted) CLSP/DS-CDMA, described in [1–4], is a simple and effective system that outperforms the conventional ALOHA/DS-CDMA. It also overcomes the hidden-terminal problem of the distributed carrier sense multiple access, CSMA/DS-CDMA. In a CLSP/DS-CDMA system, a hub is responsible for sensing the channel load, which represents the received multiple access interference (MAI). If overload is about to occur, it starts using interference cancellation techniques, described in Chapter 5, to remove the interference or to force users to refrain from transmission with feedback control. Thus, in ideal conditions, the contention of the multiaccess DS-CDMA channel can be
avoided, resulting in optimal resource utilization and throughput. The performance of non-adaptive
CLSP systems, under the perfect PC assumption, has been investigated extensively [1–4]. It was
found that, due to the high uncertainty of radio channels and imperfect PC, the system suffers from
notable performance degradations.

The PC problem is essential to the capacity, coverage and performance of interference-limited
DS-CDMA systems, in particular on the uplink [5–9]. It is desired to have all signals received at
the hub with exact targets of signal to interference ratio (SIR) for the required radio performance.
To solve the PC problem properly, it often requires a fast closed loop PC along with an open loop
PC for initial power setting. Such solutions are used, for example, in mobile cellular systems [5–
9]. In a PRN, the closed loop PC over a short transmission time interval (TTI) of a radio packet
appears to be impractical. The accuracy of open loop PC, on the other hand, is inversely proportional
to the dynamic range of the signal variation. This variation is often large and uncertain in radio
networks [5–6]. Therefore, in order to guarantee the performance requirements, the design of the
non-adaptive PRN tends to be based on worst-case scenarios of power consumption and resource
utilization. Recent research results have demonstrated that adaptive mechanisms can be used at all
layers of protocol stacks to accommodate the dynamics of the wireless channel [9, 10], leading to
crosslayer optimization. This section discusses the possibility of bit rate adaptation and packet-size
optimization to solve the PC problem and to enhance the performance of the CLSP/DS-CDMA PRN
in low mobility environments.

The concept of trading off transport formats, including bit rate, packet length in bits and TTI
in milliseconds, with radio performance is discussed in [5–9], [10–18]. There are numerous papers
investigating adaptive rate and packet size in different contexts, such as adaptive modulation and
coding, adaptive spreading factor, adaptive multicodes, etc. [9–12]. In this section we focus explicitly
on PC problems and inter-cell interference reduction in energy-efficient design of MAC protocols.
The system keeps the transmit power of mobile terminals at a constant level (within a small range
of a preset target) throughout the cell. In the meantime, it adapts the user bit rate to the distance
between the terminal and the hub using a specified LDRP RAP. The abbreviation stands for location
dependent rate and power resource allocation protocol. The closer the user is to the hub, the higher
the bit rate required to transmit with, and vice versa. As we will see later, this bit rate adaptation
substantially reduces the PC headroom and stabilizes the transmit power of mobile terminals. It
reduces the peak transmit power (and power consumption) of the mobile unit. It increases throughput
of datagram packet transmissions, due to reduced inter-cell interference, and enhances system capacity
and coverage.

To discuss the packet length adaptation, we should be aware that the open loop PC, in general,
is not efficient against the impact of fading, because using an additional power rise to eliminate
the fading impact means a significant increase in the interference. The diversity techniques, using a
RAKE receiver operating in frequency selective fading channels, for example, can reduce the impact
of deep fades [19–20]. In low mobility environments of the PRN, the Doppler bandwidth is small.
Therefore, the fading process is highly correlated causing burst error states of the channel. The burst
error states expand over tens or hundreds of milliseconds. So, in this case, the forward error correcting
(FEC) channel coding is not efficient [7, 14–16]. It is also well known that the smaller the packet
size (length in bits or TTI in milliseconds), the higher the average probability of successful packet
transmission over fading channels. At the same time, the shorter the packet, the heavier the protocol
overhead is. Thus, there is an optimal packet size for radio transmission over fading channels so that
the actual goodput, defined as the effective data rate successfully transmitted, excluding the protocol
overhead, is maximized. The goodput is used to measure the energy efficiency of mobile terminals as
well as the spectral efficiency of the PRN. This section presents such an optimal packet size for the
PRN to improve robustness of packet transmissions over correlated fading channels, while keeping
the variations of transmit power of mobile terminals low. The correlation between the fade/interfade
duration statistics and the packet size is also discussed.

So, location dependent bit rate adaptation and the packet size optimization together, not only
help to resolve PC problems, but also improve efficiency of resource utilization for the PRN. They
stabilize the transmit power of mobile terminals throughout the cell allowing the terminals to operate
with much lower maximum power requirements, reduce inter-cell interference, and therefore give the
possibility for introducing new systems with minimal environmental and health risks. Furthermore, under certain conditions, such adaptive schemes improve the performance of the PRN without the need for excessive and power consuming signal processing. This section also provides a comprehensive analytical tool for investigating the simultaneous impacts of radio propagation attenuation, correlated fading, spatial user distribution, user mobility, and MAI statistics on the PRN performance.

15.1.1 Signal variations and the power control problem

In a centralized PRN, the received signal power varies depending on the distance, the shadowing and the multipath fading between the mobile terminal and the hub. From Chapter 14, it can be expressed as

\[ P_{rx} = P_{tx}d^{-\alpha}10^{(\zeta/10)}\phi \]  \hspace{1cm} (15.1)

where \( P_{tx} \) and \( P_{rx} \) are the transmitted and the received power of the signal at the hub respectively, \( d \) is the distance between the terminal and the hub; \( \alpha \) is the path loss exponent (in the range of 2 to 5); \( \zeta \) is the attenuation in dB due to shadowing (assumed to be a Gaussian random variable with zero mean and standard deviation 8dB); and \( \phi \) represents the impact of correlated fading in low mobility environments.

The received energy per bit per interference density, for the given bit rate \( R \) is

\[ (E_b/I_0)_R = (P_{tx}/I)(W/R) \]  \hspace{1cm} (15.2)

where \( I \) is the received wideband interference power including the background noise, and \( W \) is the signal chip rate. The ratio \( (P_{tx}/I) \) is the SIR, and \( (W/R) \) is the processing gain. From Equations (15.1) and (15.2) we have

\[ (E_b/I_0)_R = (P_{tx}/I)W(d^{-\alpha}/R)10^{(\zeta/10)}\phi \]  \hspace{1cm} (15.3)

Due the randomness of signal variations, \( (E_b/I_0)_R \) is a random variable as well. The PC is used for keeping \( (E_b/I_0)_R \) above a specified target, \( \gamma_R \), for the required BER/PER performance, yet as close to \( \gamma_R \) as possible to minimize MAI for capacity enhancement. The probability that the link quality requirement is maintained is \( P\{ (E_b/I_0)_R \geq \gamma_R \} \).

15.1.2 Spectral efficiency and effective load factor of the multirate DS-CDMA PRN

In the case of \( N \) simultaneous packet transmissions with different bit rates, \( (E_b/I_0) \) of user \( j \) with bit rate \( R_j \) is given by

\[ y_j = (E_b/I_0)_j = \frac{P_{rx,j}}{I_{total} - P_{rx,j}/R_j} \]  \hspace{1cm} (15.4)

where \( P_{rx,j} \) is the received signal power from user \( j \), \( I_{total} \) is the total received wideband power at the hub. Parameter \( I_{total} \) may include own-cell interference, \( I_{own} \), other-cell interference, \( I_{other} \), and background noise, \( N_0 \). Solving Equation (15.52) for \( P_{rx,j} \) gives

\[ P_{rx,j} = u_jI_{total}, \quad u_j = \left[ 1 + \frac{W}{(E_b/I_0)_jR_j} \right]^{-1} \]  \hspace{1cm} (15.5)

The variable \( u_j \) in Equation (15.5) represents the load factor of user \( J \)’s transmission. Thus, the normalized load of the multiaccess channel is

\[ \sum_j u_j = \sum_j P_{rx,j}/I_{total} = I_{own}/I_{total} \]  \hspace{1cm} (15.6)
In the equilibrium condition with the maximum tolerable cell interference, $I_{\text{max}}$, we have

$$\lim_{I_{\text{total}} \to I_{\text{max}}} E \left[ \frac{\sum_{j \in N} u_j}{|N|} \right] = \lim_{I_{\text{total}} \to I_{\text{max}}} E \left[ \frac{I_{\text{own}}}{I_{\text{total}}} \right]$$

(15.7)

Providing that $I_{\text{total}} = I_{\text{own}} + I_{\text{other}} + N_0$, $E[N_0/I_{\text{total}}] \to I_{\text{max}} = \eta$, $E[I_{\text{other}}/I_{\text{own}}] = \iota$, and the link quality requirement is maintained, by using Equation (15.4–15.7) the upper limit on Equation (15.6), which is an effective load representing the spectral efficiency, can be approximated with the ratio $(1 - \eta) / (1 + \iota)$. From Equation (15.4–15.7) the average cell capacity for a class of users with the same bit rate $R$ and target $E_b/I_0$ ratio $\gamma_R$, defined as the maximum number of tolerable simultaneous transmissions, $C_R$, can be approximated by

$$C_R = \left\lfloor \frac{1 - \eta}{(1 + \iota) \omega_R} \right\rfloor$$

(15.8)

where $\lfloor x \rfloor$ denotes the largest integer that does not exceed the real number $x$, $\omega_R$ is the effective load factor of a packet transmission within the user class, which can be calculated with the formula of $u_j$ in Equation (15.5) given that $R_j = R$ and $(E_b/I_0)_j = \gamma_R$. The contention or the system outage state occurs in the DS-CDMA channel if

$$\sum_{j \in N} u_j > (1 - \eta) / (1 + \iota)$$

(15.9)

The objective of MAC schemes in PRN is to prevent the contention and to optimize the system performance in terms of energy and spectral efficiency.

### 15.1.3 CLSP/DS-CDMA packet access and traffic model

In the quasi-asynchronous CLSP/DS-CDMA, the users communicate via the hub using different code sequences for packet transmissions with the same link quality requirement. The user data is coded and segmented into transport blocks. Then, a header that contains system-specific addresses, transmission control information and error control fields is added to each transport block to form a radio packet. The packet header is assumed to have a fixed length of $H$ bits. The transmission of radio packets over the air toward the hub is controlled by the CLSP, which attempts to prevent the condition in Inequality (15.9). For instance, in the non-adaptive system, it keeps the number of simultaneous transmissions under the average system capacity given by Equation (15.8). The hub is responsible for sensing the channel load or the number of simultaneous transmissions. It broadcasts control information periodically in a downlink control channel. The control signal is either a soft ‘inhibit’ signal when the channel load is reaching a certain threshold, or a ‘transmission free’ signal otherwise. The terminal that has a packet to send will listen to the control channel and decide whether to transmit or to refrain from transmission in a non-persistent fashion. The above discussion assumes identical cell structures, zero propagation delays, perfect sensing of the number of ongoing transmissions, and perfect radio reception on the downlink.

The traffic model is based on the following assumptions. The system population is infinite. The scheduling of packet transmissions at the mobile terminal, including the retransmission of unsuccessful packets, is randomized sufficiently so that the offered traffic of each user is the same and the overall number of packet arrivals at the hub is generated according to the Poisson process with rate $\lambda$.

### 15.1.4 Bit rate adaptation

As discussed above, in PRNs the PC often relies on an initial power setting at the beginning of each packet transmission. The transmit power is then kept unchanged during the short packet interval. The power setting in the open loop PC is based on an estimate of the path loss at the mobile terminal by using a downlink beacon signal. The dynamic range of signal variations largely affects the accuracy
of the PC, and therefore the performance of interference-limited DS-CDMA systems. The bit rate adaptation compensates the dominant component of the path loss, the near–far effect. It adapts the bit rate $R$ to the distance $d$, by keeping factor $(d^{-\alpha}/R)$ in Equation (15.3) approximately constant throughout the cell, resulting in approximately constant $P_{tx}$.

In particular, we assume that the system supports $(M + 1)$ different bit rates, denoted by $R_m$ with $m \in M, M = \{0, 1, \ldots, M\}$. Let $R_0$ be the basic bit rate, $R_0 = 32$ kbps, specified for the cell coverage taking into account the power restriction of the mobile terminal on the uplink, and $R_m = 2^m R_0$. Let $d_0$ be the normalized radius of the cell coverage, $d_0 \equiv 1$. The distance $d$ between the terminal and the hub therefore takes a value in the $(0, 1]$ interval. In the near–far resistant bit rate adaptation, the mobile location resolution is specified with a division of the cell area into $(M + 1)$ rings centered to the hub. The normalized radius of the ring boundary, denoted by $d_m$, is given by $d_m = 2^{-m/\alpha}$ for all $m \in M$. Thus, $d_m^{-\alpha} / R_m \equiv 1 / R_0$ is constant regardless of the distance. The bit rate adaptation mechanism can be formally expressed as

$$\text{If } d_{m+1} < d \leq d_m \text{ then } R = R_m \text{ for all } m \in M, \text{ and } d_{M+1} = 0.$$  \hspace{1cm} (15.10)

The packet transmission of mobile terminals from an outer ring with a smaller index $m$ will use a lower bit rate $R_m$. The location awareness of mobile terminals is feasible, especially in low mobility environments. The fineness of the rate-location-power resolution increases with larger $M$ and smaller $\alpha$. This is illustrated in Figure 15.1.

The near–far resistant bit rate adaptation, stabilizes the transmit power of mobile terminals, thus improving the PC accuracy, reducing the inter-cell interference, and enhancing the spectral efficiency and system performance. As an illustration, let us consider the following simplified scenario. The system consists of identical hexagonal cells using omnidirectional antennas. The cell of interest is in the middle. The transmit power of mobile terminals, $P_{tx}$, is kept constant. It is assumed that the active users located outside the typical $5d_0$ radius circle area around the hub do not cause any interference to the cell. Let the spatial user distribution (SUD) density function be $f(d, \theta)$. The mean of $I_{\text{other}}/I_{\text{own}}$
has an upper bound that can be represented as
\[
E \left[ \frac{I_{\text{other}}}{I_{\text{own}}} \right] \leq \frac{\int_0^{2\pi} \int_0^{5d_0} P_{\text{tx}} d^{-\alpha} f(d, \theta) \, dd \, d\theta}{\int_0^{2\pi} \int_0^{d_0} P_{\text{rx}} d^{-\alpha} f(d, \theta) \, dd \, d\theta}
\]

In the case where the SUD is a two-dimensional uniform distribution and \( \alpha = 3 \), the right-hand side of the above expression gives the numerical result of about 5%. This is remarkably small compared to the 40%–65% of the same factor in the non-adaptive system with perfect PC as shown in [5]. In general, the near–far resistant bit rate adaptation allows the transmit power of mobile terminals to be kept within a small range of a preset target throughout the cell. The impact of fading and the fade-margin setting, as well as the correlation between fade/interfade duration and packet size are investigated in the next section.

15.1.5 The correlated fading model and optimal packet size

The application of a PRN is primarily targeted for low mobility environments. The fading process is therefore highly correlated. Let us consider two important fade statistics: the level crossing rate (LCR) and the average fade duration (AFD), which characterize the rate of occurrence and the average length of burst errors in fading channels, respectively [9, 18, 19–24]. These are functions of the ratio \( \rho = R_s/R_{\text{rms}} \), where \( R_s \) is the specified signal level and \( R_{\text{rms}} \) is the local root mean square amplitude of the fading envelope, and the maximal Doppler frequency \( f_d = f_c/c \), where \( f_c \) is the carrier frequency, \( v \) is the velocity of the mobile user, and \( c \) is the speed of light [8]. The fade margin \( F \), in decibels, is defined as \( F = -20 \log(\rho) \) that represents the power rise in the transmit power target in order to compensate the impact of fading. Thus, studying LCR and AFD allows us to relate the rate of change of the received signal to the target signal level and the user velocity. This is necessary for the design of optimal packet radio transmission over burst error correlated fading channels.

The statistics of LCR and AFD in correlated fading channels with diversity have been studied, for example, in [19, 20]. Table 15.1, based on [19], illustrates the behavior of LCR and AFD for different fade margins and diversity techniques with an assumption of perfect two-branch combining in multipath Rayleigh fading channels.

It includes results for non-diversity, selection combining (SC), equal-gain combining (EGC), and maximal ratio combining (MRC) cases. There is a significant gain in the fade margin, and thus in the power efficiency as well, with the diversity cases for the same range of fade statistics or similar fading conditions. It is therefore natural to use diversity techniques to combat deep fades in interference-limited DS-CDMA systems. In the following, we will assume that the MRC is used in the system. It is desirable to have as small a fade margin as possible for the required PER performance. The probability of correct packet transmission over fading channels, in general, depends on the number of fades occurring during TTI, duration of burst errors, and capability of FEC coding. Let us define the

<table>
<thead>
<tr>
<th>( F ) (dB)</th>
<th>Non-diversity</th>
<th>SC</th>
<th>EGC</th>
<th>MRC</th>
</tr>
</thead>
<tbody>
<tr>
<td>LCR/( f_d )</td>
<td>AFD*( f_d )</td>
<td>LCR/( f_d )</td>
<td>AFD*( f_d )</td>
<td>LCR/( f_d )</td>
</tr>
<tr>
<td>20</td>
<td>0.2482</td>
<td>0.0401</td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>0.4319</td>
<td>0.0721</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>0.7172</td>
<td>0.1327</td>
<td>0.1365</td>
<td>0.0663</td>
</tr>
<tr>
<td>5</td>
<td>0.5571</td>
<td>0.1319</td>
<td>0.4076</td>
<td>0.1277</td>
</tr>
<tr>
<td>0</td>
<td>1.1658</td>
<td>0.3427</td>
<td>1.0279</td>
<td>0.3066</td>
</tr>
</tbody>
</table>

Table 15.1 Fade statistics of fading channels with diversity (based on the analytical results reported in [19]) © 1988, IEEE
The condition identically distributed and independent random variables having exponential PDF \([21, 22, 25–27]\).

Equation (15.13) becomes

Figure 15.2 shows the equilibrium \( \text{tfia} \) having PDF \( s \) of \( T \), in milliseconds. The probability of successful packet transmission over fading \( P_d \) as a function of \( T \), \( \rho \), and \( f_d \), can be expressed by [13]:

\[
P_d(T, \rho, f_d) = \int_0^\infty \int_0^\infty (t_f - T)(t_f + t_f)^{-1}h(t_f)g(t_f) \, dt_f \, df_f
\] (15.11)

The closed-form expressions for the PDF of the fade/interfade duration are not available for a general case, but are, rather, assessed on an individual basis. However, in the related literature so far, the Markov channel model is applied extensively [5–16, 18, 21, 24–27]. In particular, in the case of correlated fading channels with \( \text{tfavrg}/\text{tifavrg} \ll 1 \), it is reasonably accurate to assume that \( t_{fd} \) and \( t_f \) are identically distributed and independent random variables having exponential PDF [21, 22, 25–27]. The condition \( \text{tfavrg}/\text{tifavrg} \ll 1 \) is required for reliable data communications over burst error fading channels. In the system model based on the numerical results for the MRC case in Table 15.1, the ratio \( \text{tfavrg}/\text{tifavrg} \) ranges from 0.5% to 36% for the fade margin of 10 dB and 0 dB, respectively. It is 4% if the fade margin is 5 dB, which is reasonable for the above condition as well as for the transmit power rise needed. It is below a 6 dB transmit power rise corresponding to \( \rho = 0.5 \), or 3 dB for the average SIR thereof. In the following, we set the fade margin to 5 dB.

It is obvious that the smaller the \( T \), the higher the \( P_d \), and at the same time, the heavier the protocol overhead. For given constraints of the required PER, the user velocity \( v \), the bit rate \( R \), the fade margin \( F \), and the length of the packet header \( H \), one can find an optimal packet duration \( T \) so that the normalized goodput \( G_{sf} \)

\[
G_{sf}(T, \rho, f_d) = \frac{P_d(T, \rho, f_d)(TR - H)}{TR}
\] (15.12)

is maximized. The goodput is defined above as the effective data rate successfully transmitted excluding the protocol overhead. The product \( TR \) in Equation (15.12) gives the value of the packet length \( L \) in bits, \( L = TR \). One can further exploit the potential of adaptive packet size based on Equations (15.11) and (15.12). For example, \( T \) and/or \( L \) can be adapted to \( \rho \) and/or \( f_d \) along with adaptive \( R \). These details are not included due to limited space. In the mobility equilibrium, we have

\[
P_d(T) = \int_{f_d} P_d(T, \rho, f_d) \, df_d
\] (15.13)

where \( p(f_d) \) is the PDF of \( f_d \) resulting from the distribution of the user velocity in the system equilibrium. Let us assume that \( f_d \) takes a discrete value in \( \{1, 2, \ldots, f_{d\text{-max}}\} \). Then the right-hand side of Equation (15.13) becomes

\[
P_{sf}(T) = \sum_{f_d} P_{sf}(T, f_d)P(f_d)
\] (15.14)

Figure 15.2 shows the equilibrium \( P_{sf} \) and \( G_{sf} \) characteristics versus \( T \) with \( R = R_0 = 32 \) kbps, \( H = 160 \) bits, \( \rho = 10^{-5/20} \), and \( f_{d\text{-max}} = 30 \) Hz for three different mobility scenarios. The most dynamic one is characterized with a uniform \( p(f_d) \) over the limit range of \( f_d \), for which the user
velocity \( v \) is distributed evenly within 0–4.5 m/s. The least dynamic one is characterized with an exponential \( p(f_d) \) with a mean of \( f_d \) about 7 Hz, for which the user velocity is distributed exponentially around a mean of 1 m/s. For the one in between, we use a generic gamma PDF for \( p(f_d) \) because of its flexibility and richness in modeling [28]:

\[
p(f_d) = \frac{1}{b^a \Gamma(a)} f_d^{a-1} e^{-f_d/b}
\]

where \( a \) is the shape parameter, \( b \) is the scale parameter, and \( \Gamma(x) \) is the gamma function. Thus, we can adjust the values of parameters \( a \) and \( b \) to obtain suitable PDFs for a number of mobility scenarios and expansions of the Doppler frequency range. This PDF family also takes the exponential PDF form as its special member when the shape parameter \( a \) is set to 1. For the moderate mobility scenario of Figure 15.2, we choose a gamma PDF with \( a = 2 \) and \( b = 6 \) for instance. The optimal packet size is obtained at \( T_0 \) around 30 ms, which is also the maximum TTI if PER is not to exceed 15%. Based on that, we set the optimal packet size for the most dynamic scenario as \( T_0 = 20 \) ms and \( L_0 = T_0 R_0 = 640 \) bits; for the least dynamic one, \( T_0 = 40 \) ms and \( L_0 = T_0 R_0 = 1280 \) bits; and for the moderate one, \( T_0 = 32 \) ms and \( L_0 = T_0 R_0 = 1024 \) bits.

### 15.1.6 Performance

Let \( T_0 \) be the normalized unit of time, \( T_0 \equiv 1 \). Let us define the following notation. \( \Lambda \) is the offered system traffic, i.e. the average number of packet arrivals per normalized unit of time. Given that the packets arrive at the hub according to the Poisson process with rate \( \lambda \) and \( T_0 \equiv 1 \), we have \( \Lambda = \lambda T_0 = \lambda \). The offered system traffic in terms of data rate is kept the same for both the non-adaptive and the adaptive systems, and takes the value \( A L_0 \). \( S \) is the system throughput (the average number of successful packet transmissions per normalized unit of time for a given offered traffic \( \Lambda \)). \( D \) is the average packet delay (the average time interval from the instant a given packet is generated to the instant the packet is transmitted successfully). \( G \) is the system goodput (the effective average data rate successfully transmitted excluding protocol overhead). For instance, in the system with a fixed packet length of \( L \) bits, including a packet header of \( H \) bits, the system goodput is \( G = S(L - H) \).
Performance characteristics of the non-adaptive system

In this case we have $R = R_0$, $T = T_0$ and $L = L_0$. The system capacity, $C = C_0$, is obtained from Equation (15.8). The formal notation of the queuing system model for this system is: exponentially distributed inter-arrival time; deterministic service time as the packet duration $T$; finite number of servers equal to the system capacity $C$; no waiting room and infinite population. Let us define $n$ as the system state (the number of ongoing packet transmissions in the system) and $p_n$ as the steady-state probability of the system given by:

$$p_n = \frac{\beta^n}{n!} \left( \sum_{i=0}^{C} \frac{\beta^i}{i!} \right)^{-1}, \quad 0 \leq n \leq C$$ (15.15)

where $\beta = \lambda T$ [29]. The equilibrium probability of successful packet transmission, $P_{\text{succ}}$, depends on the following three factors. The first one is the equilibrium probability that the packet is not blocked by the CLSP, which is given by

$$P_{\text{nb}} = 1 - p_C$$ (15.16)

The second one is the equilibrium probability that the packet is not corrupted by the system outage, which can be expressed as $P_{\text{nso}} = E[\text{the number of transmissions not hit by the system outage}] / E[\text{the number of transmissions in the system}]$. This probability can be expressed as

$$P_{\text{nso}} = \frac{\sum_{n=0}^{C} np_n P_{\text{ok}}(n)}{\sum_{n=0}^{C} np_n}$$ (15.17)

where $P_{\text{ok}}(n)$ is the probability that $n$ simultaneous transmissions are not corrupted by the system outage, i.e. the condition in Inequality (15.9) does not occur. The impact of imperfect PC, in the long run, is often characterized with a log-normal error of the average $E_b/I_0$ ratio with a standard deviation of $\sigma$ in decibels [5, 6]. Therefore, $P_{\text{ok}}(n)$ can be expressed as

$$P_{\text{ok}}(n) = 1 - Q\left( \frac{C - E[Z|n]}{\sqrt{\text{Var}[Z|n]}} \right)$$ (15.18)

where $Q(x)$ is the standard Gaussian integral, $Z$ is the normalized MAI caused by $n$ users, $E[Z|n] = n \exp((\epsilon \sigma)^2/2)$; $\text{Var}[Z|n] = n \exp(2(\epsilon \sigma)^2)$; $\epsilon = \ln(10)/10$ [5]. The third factor is the equilibrium probability that the packet is not faded, which is given by Equation (15.14) with $T = T_0$. Therefore, we have

$$P_{\text{succ}} = P_{\text{nb}} P_{\text{nso}} P_{\text{sf}}$$ (15.19)

The average system throughput for the offered system traffic $\Lambda$ is given by:

$$S = \Lambda P_{\text{succ}}$$ (15.20)

The ratio $S/\Lambda \equiv P_{\text{succ}}$ is called the normalized throughput.

The normalized average packet delay is decomposed into two components. The first one is the normalized average waiting time for accessing the channel, $D_w$. This can be calculated by the normalized average delay time, which follows Little’s result at the arrival-side [29], minus the normalized serving time $T$

$$D_w = (\lambda T/E[n] - 1)T/T_0$$ (15.21)

where $E[n]$ is the average number of packets being served in the system. The second one is the normalized average resident time from the instant the packet enters the system to the instant the packet leaves the system successfully, $D_t$. This is obtained by using Little’s result at the departure-side:

$$D_t = E[n]/S$$ (15.22)
where \( S \) is the average system throughput. The normalized average packet delay is therefore obtained as

\[
D = D_w + D_t \quad (15.23)
\]

The system goodput is given by

\[
G = S(L - H) \quad (15.24)
\]

The normalized goodput is given by the ratio \( G/(AL_0) \equiv P_{\text{succ}}(L - H)/L. \)

### 15.1.6.2 Performance characteristics of the adaptive system

This system uses bit rate adaptation while having the same cell coverage and offered system traffic in terms of data rate as those of the non-adaptive system. Either the packet length \( L \), or the packet duration \( T \), can be kept the same as in the non-adaptive system, i.e. \( L_0 \) or \( T_0 \), which is optimized. In the first option, with \( L = RT = L_0 \), the packet transmission with higher bit rate will have shorter TTI. That compensates the increase of MAI on a time scale within the \( T_0 \) interval, resulting in the same normalized power consumption per packet for all packet transmissions. In the latter option, with \( T = L/R = T_0 \), the packet transmission with higher bit rate will have longer packet length and smaller offered traffic intensity. That compensates the increase of MAI on a time scale over the \( T_0 \) interval, resulting in less protocol overhead but longer packet delay compared to the first option. Let us denote \( T_m \) as the packet duration corresponding to bit rate \( R_m \). Thus, we have \( T_m = 2^{-m}T_0 \) for the first option, and \( T_m = T_0 \) for the latter one.

The queuing system model for the adaptive system is the stochastic model of multirate loss networks [30]. The offered system traffic is composed of portions corresponding to different bit rates, which are generated by active users from different ring areas of the cell due to the adaptation mechanism given by Expression (15.10). Let us define \( \lambda_m \) as the rate of packet arrivals from ring \( m \) using bit rate \( R_m \). The value of \( \lambda_m \) depends on SUD, and since the offered system traffic in terms of data rate is \( AL_0 = \lambda R_0 T_0 \), the same as that of the non-adaptive system, \( \lambda_m \) is given by

\[
\lambda_m = \frac{\lambda R_0 T_0}{R_m T_m} \int_0^{2\pi} \int_{d_{m+1}}^{d_m} f(d, \theta) \, dd \theta \quad (15.25)
\]

where \( f(d, \theta) \) is the PDF of SUD.

Let us define \( \mathbf{n} \) as the variable vector of the system states, \( \mathbf{n} = [n_m, m \in \mathbf{M}] \), where \( n_m \) is the number of ongoing packet transmissions using bit rate \( R_m \); and \( \omega \) as the constraint vector of the effective load factors, \( \omega = [\omega_m, m \in \mathbf{M}] \), where \( \omega_m \) is defined as \( \omega_R \) in Equation (15.8) with \( R = R_m \).

The CLSP controls the packet access on the uplink so that \( \mathbf{n} \) is kept in the set of allowed system states defined as \( \Omega = \{ \mathbf{n}, \mathbf{n} \omega \leq (1 - \eta)/(1 + \iota) \} \) based on Inequality (15.9). Let us define \( p(\mathbf{n}) \) as the steady-state probability of the system, \( \mathbf{n} \in \Omega \), with the product form solution [30]:

\[
p(\mathbf{n}) = \frac{1}{G_0} \prod_m \frac{\beta_m^{n_m}}{n_m!}, \quad G_0 = \sum_{\mathbf{n} \in \Omega} \prod_m \beta_m^{n_m} / n_m! \quad (15.26)
\]

where \( \beta_m = \lambda_m T_m \). For a large set of \( \Omega \), the computational complexity with the above formulas is prohibitively high.

Let us define \( s \) as the effective load state of \( \mathbf{n} \) packet transmissions, \( \mathbf{n} \in \Omega \), given by the product \( s = \mathbf{n} \omega \). It forms a set \( \Psi \) of possible effective load states corresponding to the set \( \Omega \) of system states. The steady-state probability of the system, in terms of effective load states, can be calculated based on the so-called stochastic knapsack-packing approximation [30] as follows.

\[
p(s) = \frac{q(s)}{\sum_{\mathbf{s} \in \Psi} q(s)} \quad (15.27)
\]
with \( q(s) \) given in recursive form as:
\[
q(s) = \frac{1}{s} \sum_m \sigma_m \beta_m q(s - \omega_m), \, s \in \Psi^+, \, q(0) = 1, \, q(-) = 0
\]

Based on the results of the non-adaptive system above, the equilibrium probability of successful packet transmission with bit rate \( R_m \), denoted by \( P_{\text{succ},m} \), depends on the following three factors. The first one is the equilibrium probability that the packet is not blocked by the CLSP. This is given by
\[
P_{\text{nb},m} = 1 - \sum_{s \in \Psi, s > 1 - \eta - \sigma_m} p(s) \tag{15.28}
\]
The second is the equilibrium probability that the packet is not corrupted by the system outage. This can be expressed as
\[
P_{\text{nso}} = \frac{\sum n \cdot p(n) \cdot P_{\text{ok}}(n)}{\sum n \cdot p(n)} \tag{15.29}
\]
where \( P_{\text{ok}}(n) \) is the probability that \( n \) simultaneous transmissions are not corrupted by the system outage, i.e. the condition in Inequality (15.9) does not occur. Let us denote the standard deviation of the log-normal error of the average \( E_b/I_0 \) ratio corresponding to bit rate \( R_m \), with \( \sigma_m \) in decibels. It can be expected that \( \sigma_m \) of the adaptive system is smaller than \( \sigma \) of the non-adaptive counterpart because the PC in the adaptive system is more stable and accurate as discussed above. To simplify the computation, we assume that \( \sigma_m = \varphi \) for all \( m \). Based on Equation (15.18), \( P_{\text{ok}}(n) \) can be expressed by
\[
P_{\text{ok}}(n) = 1 - Q \left( \frac{1 - \eta}{1 + \epsilon} - \frac{E[Z|n]}{\sqrt{\text{Var}[Z|n]}} \right) \tag{15.30}
\]
where \( Z \) is the normalized MAI caused by \( n \) users,
\[
E[Z|n] = \sum_m n_m \sigma_m \exp((\epsilon \sigma_m)^2/2) = \exp((\varphi)^2/2) \sum_m n_m \sigma_m = s \exp((\varphi)^2/2)
\]
\[
\text{Var}[Z|n] = \sum_m n_m \sigma_m \exp(2(\epsilon \sigma_m)^2) = s \exp(2(\varphi)^2)
\]
Thus, \( P_{\text{ok}}(n) \) can now be replaced with \( P_{\text{ok}}(s) \) and Equation (15.29) is equivalent to
\[
P_{\text{nso}} = \sum_{s \in \Psi} sp(s)P_{\text{ok}}(s)/\sum_{s \in \Psi} sp(s) \tag{15.31}
\]
The third factor is the equilibrium probability that the packet is not faded. This is given by Equation (15.14) with \( T = T_m \), i.e. \( P_{\text{sf}}(T_m) \). Therefore, we have
\[
P_{\text{succ},m} = P_{\text{nb},m} P_{\text{nso}} P_{\text{sf}}(T_m) \tag{15.32}
\]
The average system throughput is given by:
\[
S = \sum_m \lambda_m T_0 P_{\text{succ},m} \tag{15.33}
\]
The normalized system throughput is given by the ratio \( S/\sum_m \lambda_m T_0 \). The normalized average packet delay is given by Equation (15.23) with the components
\[
D_w = \left( \sum_m \lambda_m T_m \sigma_m / \sum_{s \in \Psi} sp(s) - 1 \right) \left( \sum_m \lambda_m T_m / \sum_m \lambda_m T_0 \right) \tag{15.34}
\]
\[
D_r = \sum_{s \in \Psi} sp(s) / \sum_m \lambda_m T_0 P_{\text{succ},m} \sigma_m
\]
The system goodput is given by

\[ G = \sum_m \lambda_m T_0 P_{\text{suc}}(R_m T_m - H) \]  

(15.35)

and the normalized goodput is \( G/(\Delta L_0) \).

### 15.1.6.3 Performance examples

Table 15.2 summarizes the system parameters. The adaptive system supports four different bit rates: 32 kbps, 64 kbps, 128 kbps and 256 kbps. The target \( E_b/I_0 \) is set to 5 dB, which includes an increase of 2.5 dB due to the fade margin of 5 dB.

The value of the above analytical method is in the modeling of the system dynamics due to effects of numerous simultaneous factors. These factors include imperfect PC, propagation attenuations, SUD, user mobility, and offered traffic intensity over burst error correlated fading channels. The optimal packet size, obtained from the optimization presented in the above section is used for both the non-adaptive and the adaptive systems. That is, \( T_0 = 20 \text{ ms and } L_0 = 640 \text{ bits} \) for the dynamic mobility scenario characterized by a uniform \( p(f_d); \) or \( T_0 = 40 \text{ ms and } L_0 = 1280 \text{ bits} \) for the less dynamic one with an exponential \( p(f_d) \). To investigate the effects of SUD, two simple functions for \( f(d, \theta) \) are used, the one-dimensional uniform PDF and the two-dimensional uniform PDF. The first one is often used to model indoor office environments, where the users are located along the corridors. The latter one is for more open areas. The offered system traffic \( \Lambda \) varies and takes values in \{4, 8, 16, \ldots, 60\} packet arrivals per normalized unit of time \( T_0 \). The simulation results are based

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>( f_c )</td>
<td>The carrier frequency</td>
<td>2 GHz</td>
</tr>
<tr>
<td>( W )</td>
<td>The DS-CDMA chip rate</td>
<td>4.096 Mcps</td>
</tr>
<tr>
<td>( \eta )</td>
<td>The ratio of the background noise power normalized to the maximum tolerable received interference</td>
<td>–10 dB</td>
</tr>
<tr>
<td>( \alpha )</td>
<td>The path loss law exponent</td>
<td>3 (4)</td>
</tr>
<tr>
<td>( F )</td>
<td>The fade margin</td>
<td>5 dB</td>
</tr>
<tr>
<td>( R_0 )</td>
<td>The primary bit rate</td>
<td>32 kbps</td>
</tr>
<tr>
<td>( T_0 )</td>
<td>The optimal packet duration</td>
<td>20 ms (40 ms)</td>
</tr>
<tr>
<td>( L_0 )</td>
<td>The optimal packet length</td>
<td>640 bits (1280 bits)</td>
</tr>
<tr>
<td>( M + 1 )</td>
<td>The number of supported bit rates</td>
<td>4</td>
</tr>
<tr>
<td>( \gamma_m )</td>
<td>The target ( E_b/I_0 ) ratio of packet transmission with bit rate ( R_m )</td>
<td>5 dB</td>
</tr>
<tr>
<td>( \sigma )</td>
<td>The standard deviation of log-normal error of the average ( E_b/I_0 ) for the non-adaptive system</td>
<td>3 dB</td>
</tr>
<tr>
<td>( \varphi )</td>
<td>The standard deviation of log-normal error of the average ( E_b/I_0 ) for the adaptive system</td>
<td>2 dB</td>
</tr>
<tr>
<td>( f_{d_{\text{max}}} )</td>
<td>The upper bound of maximal Doppler frequency</td>
<td>30 Hz</td>
</tr>
<tr>
<td>( H )</td>
<td>The length of the packet header plus trailer</td>
<td>160 bits</td>
</tr>
</tbody>
</table>
on Omnet ++ [28], a multipurpose discrete-event simulator. The channel sampling period is set to 0.1 ms. The simulation is run over 100 000 packet arrivals for each rate of the offered system traffic.

Figures 15.3–15.7 present the performance of both the non-adaptive and the adaptive systems in different system scenarios under the effects of different system parameters. Each figure consists of two subfigures: (a) the normalized system goodput and (b) the throughput-delay tradeoffs. The adaptive system outperforms the non-adaptive counterpart in all cases.

Figure 15.3 Single-cell system performance in dynamic mobility scenario. (a) Normalized system goodput; (b) throughput – delay tradeoffs [31] © 2002, IEEE.
Figure 15.4 Single-cell system performance in less dynamic mobility scenario. (a) Normalized system goodput; (b) throughput – delay tradeoffs [31] © 2002, IEEE.
Figure 15.5 Impact of SUD on single-cell system performance. (a) Normalized system goodput; (b) throughput – delay tradeoffs [31] © 2002, IEEE.
Figure 15.6 Impact of path loss law exponent on single-cell system performance. (a) Normalized system goodput; (b) throughput – delay tradeoffs.

Figures 15.3–15.4 show the impact of user mobility on the performance and how to choose a practical option of the optimal packet size for the adaptive system. For these figures, we use a single-cell scenario with the two-dimensional uniform $f(d, \theta)$ and the path loss law exponent $\alpha = 3$. The optimal packet size, as mentioned above, includes the optimal packet length in bits and the optimal packet duration in milliseconds. The less dynamic mobility scenario in Figure 15.4 has larger optimal packet size and better goodput performance than the more dynamic one in Figure 15.3, due to fading. This is also shown in Figure 15.2. There are two options for the optimal packet size, with the optimal packet length in bits, or the optimal TTI in milliseconds for the adaptive system. In the first option,
keeping the packet length $L$ constant at the optimized $L_0$ results in a constant $(L - H)/L$ ratio. Thus, there is no further gain in the goodput, but instead, a significant gain in the packet delay, due to the shorter TTI of packet transmissions with higher bit rates. However, if the optimal packet length is not large enough, the packet transmission with a certain high bit rate may require an impractically short TTI. In the latter one, keeping the packet duration $T$ constant at the optimized $T_0$ results in a better
(RT - H)/RT ratio for larger R. Thus, there is a gain in the goodput, but not in the packet delay. If
the optimal packet duration is large enough, the gain in the goodput is not significant and the packet
length in bits may become impractically large for a certain high bit rate. Based on this consideration
and the results shown in Figures 15.3–15.4, T = T₀ = 20 ms for the dynamic mobility scenario, and
L = L₀ = 1280 bits for the less dynamic one, are used.

Figures 15.5–15.6 present the impacts of SUD and path loss exponent on the performance char-
acteristics. For these figures, the less dynamic mobility scenario is used with L = L₀ = 1280 bits
and a single cell. The adaptive system is sensitive to the SUD and the path loss exponent, which
affect the patterns of the user location resolution and the offered system traffic as formulated in
Expression (15.10) and Equation (15.25). The more users put to inner rings with one-dimensional
uniform f(d, θ) in Figure 15.5, and/or α = 4 in Figure 15.6, the more the bit rate is boosted and the
greater the reduction in TTI of packet transmissions, resulting in better goodput and throughput delay
performance for the adaptive system. The differences are not of significance though. One should also
keep in mind that the larger path loss law exponent and the non-uniform SUD cause larger headroom
of the transmit power, especially in the non-adaptive system.

The increase in goodput, i.e. energy efficiency, offered by the proposed bit rate adaptation and
packet length optimization, can well be over 100 %, as shown in Figure 15.7 for a multicell system
scenario with identical cells and two-dimensional uniform SUD. The inter-cell interference in the
adaptive system (with the factor ι = E[Iother/Iown] set to the maximum of 5 %) is much smaller than
in the non-adaptive counterpart (with ι set to the minimum of 40 %) as presented above. This, added
to the advantages of the adaptive system as presented above with a single-cell scenario, significantly
enhances system performance and cell deployment.

15.2 MINIMUM ENERGY PEER-TO-PEER MOBILE WIRELESS NETWORKS

This section describes a distributed network protocol optimized for achieving the minimum consump-
tion energy for randomly deployed ad hoc networks. A position-based algorithm, based on [32], is
presented, which is supposed to set up and maintain a minimum energy network between users that
are randomly deployed over an area and are allowed to move with random velocities. These mobile
users are referred to as ‘nodes’ over the two-dimensional plane. The network protocol reconfigures
the links dynamically as nodes move around, and its operation does not depend on the number of nodes
in the system. Each mobile node is assumed to have a portable set with transmission, reception and
processing capabilities. In addition, each has a low-power global positioning system (GPS) receiver
on board, which provides position information within at least 5 m of accuracy. The principles of
GPS system operation are discussed in Chapter 14. The recent low-power implementation of a GPS
receiver [33] makes its presence a viable option in minimum energy network design.

15.2.1 Network layer requirements

In peer-to-peer communications, each node is both an information source and an information sink.
This means that each node wishes to both send messages to, and receive messages from, any other
node. An important requirement of such communications is strong connectivity of the network. A
network graph is said to be ‘strongly connected’ if there exists a path from any node to any other
node in the graph [34]. A peer-to-peer communications protocol must guarantee strong connectivity.

For mobile networks, since the position of each node changes over time, the protocol must be
able to update its links dynamically in order to maintain strong connectivity. A network protocol
that achieves this is said to be ‘self-reconfiguring.’ A major focus of this section is the design of a
self-reconfiguring network protocol that consumes the least amount of energy possible.

In order to simplify the discussion, we take one of the nodes to be the information sink for all
nodes in the network. We call this node the ‘master site.’ The master site can be thought of as the
headquarters located at the edge of the digital battlefield, the supervisory station in a multisensor
network, or the base station in a cellular phone system. All of these scenarios are special cases of peer-to-peer communications networks.

Each node knows its own instantaneous position via GPS, but not the position of any other node in the network, and its aim is to send its messages to the master site whenever necessary.

A protocol that solves the minimum energy problem with a single master site simultaneously solves the general peer-to-peer communications problem, because each node can independently be taken as a master site, and the optimal topologies can be superimposed. We take advantage of this simplification and concentrate on the problem with a single master site without loss of generality.

15.2.2 The power consumption model

In the network configuration algorithm, this section concentrates only on path loss that is distance-dependent. The algorithm does not depend on the particular value of the path loss exponent $\alpha (\alpha > 2$ for outdoor propagation models; see Chapter 14) and thus offers the flexibility to be applied in various propagation environments. So, since the transmit power falls as $1/d^\alpha$, $\alpha \geq 2$, as given by the path loss model, relaying information between nodes may result in lower power transmission than communicating over large distances.

For illustration, consider three nodes A, B and C on a line, as in Figure 15.8. Assume that all three nodes use identical transmitters and receivers. Node A wants to send a message to C. Let $t$ denote the predetection threshold (in mW) at each receiver. In other words, the minimum power that a transmitter must radiate in order to allow detection at distance $d$ meters away is $td^n$, where $n$ is the exponent in the path loss model. Assuming that node A knows the positions of B and C, it has two options: it can transmit the signal directly to C, which requires a power consumption of $td_{AC}^n$ at node A, or it can relay the message through node B and have it retransmit it with the minimum power needed for B to reach C. In this second case, the total transmit power consumption is $td_{AB}^n + td_{BC}^n$. In the case of three collinear nodes, it is easily seen that relaying the message through the middle node always comes at a lower total transmit power consumption than transmitting directly.

When the three nodes are allowed to lie on a two-dimensional plane, which is denoted by $\mathbb{R}^2$, the option that costs less total power becomes a function of where the receive node is positioned. In the next section, we find the positions for the receive node, where relaying will always consume less total power than transmitting directly.

There is another source of power consumption that must be considered in addition to path loss. In the previous example, when node A relays through B, node B has to devote part of its receiver to receive and store node A’s message. This additional power will be referred to as the receiver power at the relay node, and will be denoted by $c$. Each relay induces an additional receiver power to be consumed at the relay node. For the previous example, the total power consumption, including transmit and receiver power consumption in the transmission, is thus $td_{AB}^n + td_{BC}^n + c$ when node B is used as a relay.

The main goal of the section is to arrive at an algorithm that requires only local computation for updates, and requires as little global information as possible. A protocol requiring only local information is extremely advantageous for networks with mobile nodes, since delays associated with disseminating global information would be intolerable. From the perspective of power consumption, a distributed protocol running almost exclusively on local information requires transmission only over small distances. This in turn conserves the total power required for transmitting that information. A third advantage of the use of only local information is that it reduces the interference levels...
dramatically, since a user’s communication with only nodes in its immediate surroundings causes little interference to nodes further away.

15.2.3 Minimum power networks

In order to investigate the implications of local information on power-efficient transmission, we consider three nodes in $\mathbb{R}^2$, denoted by $i$, $r$, and $j$. Node $i$ is a node that wishes to transmit information to node $j$. Node $i$ is referred to as the ‘transmit node’ and node $j$ the ‘receive node.’ Node $i$ considers the third node, $r$, to be used as a relay for transmission from $i$ to $j$. Node $r$ is called the ‘relay node.’ The aim is to transmit information from $i$ to $j$ with minimum total power incurred by $i$, $j$, and $r$. By varying the position of $j$, we investigate under which conditions it consumes less power to relay through $r$. Below, the position of $j$ is denoted by $(x, y)$.

The following definitions will be used:

Relay region: The relay region $R_{i\rightarrow r}$ of the transmit–relay node pair $(i, r)$ is defined to be

$$R_{i\rightarrow r} = \{(x, y)|P_{i\rightarrow r\rightarrow (x, y)} < P_{i\rightarrow (x, y)}\}$$  \hspace{1cm} (15.36)

where $P_{i\rightarrow r\rightarrow (x, y)}$ denotes the power required to transmit information from node $i$ to $(x, y)$ through the relay node $R$, whereas $P_{i\rightarrow (x, y)}$ denotes the power required to transmit information from $i$ to $(x, y)$ directly.

Deployment region: Any bounded set in $\mathbb{R}^2$ that has the position of the nodes in $\mathbb{N}$ as a subset is said to be a deployment region for the node set $\mathbb{N}$. The deployment region is introduced because, in practice, there is a finite area beyond which no nodes should be looking for neighbors with which to communicate. The boundaries of deployment regions can also be taken as known and impenetrable obstacles to communication. Then, the nodes near the edge can use this fact not to search unnecessarily beyond the deployment region.

Enclosure and Neighbor: The enclosure of a transmit node $i$ is defined as the non-empty solution $\varepsilon_i$ to the set of equations defined as

$$\varepsilon_i = \bigcap_{k\in N(i)} R_{i\rightarrow k}^c \cap D$$  \hspace{1cm} (15.37)

and

$$N(i) = \{n \in \mathbb{N} |(x_n, y_n) \in \varepsilon_i, n \neq i\}$$  \hspace{1cm} (15.38)

In Equations (15.37–15.38) $R^c$ represents the complemnt of set $R$, and $D$ denotes the deployment region for the node set $\mathbb{N}$. Each element of $N(i)$ is said to be a ‘neighbor’ of $i$ and $N(i)$ is called the ‘neighbor set’ of $i$. A node $i$ is said to be enclosed if it has communication links to each of its neighbors and no other node.

Enclosure graph: The enclosure graph of a set of nodes $\mathbb{N}$ is the graph whose vertex set is $\mathbb{N}$ and whose edge set is

$$\bigcup_{i\in \mathbb{N}} \bigcup_{k\in N(i)} l_{i\rightarrow k}$$  \hspace{1cm} (15.39)

where $l_{i\rightarrow k}$ is the directed communications link from $i$ to $k$.

Minimum power topology: A graph on the stationary node set $\mathbb{N}$ is said to be a minimum power topology on $\mathbb{N}$ if:

1. Every node has a directed path to the master site;
2. The graph consumes the least total power over all possible graphs on $\mathbb{N}$ for which 1) holds.
15.2.4 Distributed network routing protocol

The main idea in this protocol is that a node does not need to consider all the nodes in the network to find the global minimum power path to the master site. By using a very localized search, it can eliminate any nodes in its relay region from consideration and pick only those few links in its immediate neighborhood to be the only potential candidates. The protocol effectively operates in two phases: first, a local search executed by each node to find the enclosure graph, and second, a minimum cost search from the master site to every node. The cost metric is the total power required for a node to reach the master site along a directed path.

15.2.4.1 Search for enclosure (Phase 1)

In order for a protocol to find the enclosure graph, each node must find its enclosure and its neighbor set. Since computing enclosure requires knowledge of the positions of nearby nodes, each node broadcasts its position to its search region. The search region is defined as the region where a node’s transmitted signal (and hence its position) can be correctly detected by any node in the region. When searching for neighbors, a node must keep track of whether a node found is in the relay region of previous nodes found in the search. The relay graph defined below is, in effect, a data structure which stores this information.

Relay graph of a node

Let \( A \) denote the set of all nodes that transmit node \( i \) has found thus far in its search. Let \( j \) and \( k \) be two nodes in \( A \). Whenever \( k \in R(j) \), we form a directional edge from \( j \) to \( k \) and denote it by \( e_{j\rightarrow k} \). The relay graph of a transmit node \( i \) is defined to be the directed graph whose vertex set is \( A \) and whose edge set is

\[
\bigcup_{j\in A} \bigcup_{k\in R(j)} e_{j\rightarrow k}
\]  

(15.40)

The relay graph of \( i \) is denoted by \( G(i) \). \( e_{j\rightarrow k} \) represents a relation between \( j \) and \( k \) based on their positions. It indicates that \( k \) lies in the relay region \( R_{i\rightarrow j} \). It does not represent a communication link between \( j \) and \( k \).

To find \( N(i) \), namely the neighbor set of \( i \), each node in the algorithm starts a search by sending out a beacon search signal that includes the position information of that node. Since every node executes the same algorithm, we will focus on a particular node, referred to as the transmit node. The transmit node also listens for signals from nearby nodes. When it receives and decodes these signals, it finds out the position of the nearby nodes and calculates the relay region for them. As we described in the discussion preceding the definitions of enclosure and the relay graph, the transmit node must keep only those nodes that do not lie in the relay regions of previously found nodes. Therefore, each time new nodes are found, the transmit node must update its relay graph.

The nodes that have been found thus far in the neighbor search fall into two categories: if a node found (call it node \( k \)) falls in the relay region of some other found node (call it \( j \)), then we mark \( k \) as ‘dead’, otherwise the node is marked ‘alive’. The set of alive nodes when the search is over constitutes the set of neighbors for transmit node \( i \). When the search terminates, the transmit node is enclosed, and the nodes that enclose the transmit node are not in the relay region of any node found, as required by the definition of a neighbor.

15.2.4.2 Cost distribution (Phase 2)

After the enclosure graph has been found in Phase 1, the distributed Bellman–Ford minimum cost algorithm \([13]\) on the enclosure graph is applied, using power consumption as the cost metric. In Phase 2, each node broadcasts its cost to its neighbors. The cost of a node \( i \) is defined as the minimum power necessary for \( i \) to establish a path to the master site. Each node calculates the minimum cost it
can attain given the cost of its neighbors. Let \( n \in N(i) \). When \( i \) receives the information \( \text{Cost}(n) \), it computes
\[
C_{i,n} = \text{Cost}(n) + P_{\text{transmit}}(i, n) + P_{\text{receiver}}(n)
\]
where \( P_{\text{transmit}}(i, n) \) is the power required to transmit from \( i \) to \( n \), and \( P_{\text{receiver}}(n) \) is the additional receiver power that \( i \)'s connection to \( n \) would induce at \( n \). \( P_{\text{receiver}}(n) \) is either known to \( i \), if for instance every user carries an identical receiver, or can be transmitted to \( i \) as a separate piece of information along with \( \text{Cost}(n) \). Then, node \( i \) computes
\[
\text{Cost}(i) = \min_{n \in N(i)} C_{i,n}
\]
and picks the link corresponding to the minimum cost neighbor. This computation is repeated, and the minimum cost neighbor is updated each time. The data transmission from \( i \) to the master site can start on the minimum cost neighbor link, which is the global minimum power link.

### 15.2.4.3 Computation of the relay region

In the following example, we illustrate the relay region of a single node, assuming the two-ray propagation model for terrestrial communications, which implies a \( 1/d^4 \) transmit power rolloff. The close-in reference distance is taken as 1 m. The carrier frequency is 1 GHz, and the transmission bandwidth 10 kHz. We assume omnidirectional antennas with 0 dB gain, \(-160\) dBm/Hz thermal noise, 10 dB noise figure in the receiver, and a predetection signal to noise ratio (SNR) of 10 dB. Using the Friis free space formula gives \(-67.5\) dBm as the minimum transmit power required for detection at 1 m. We take this to be roughly \(-70\) dBm for our simulations. This can be treated as an effective predetection threshold to be used with the \( 1/d^4 \) rolloff formula to compute the minimum required transmit power for any distance.

We assume the following model for receiver power at any relay node: a fixed receiver power of 80 mW is consumed at each node, with a 20 mW increase for each additional node from which transmission is received. This model can be easily modified according to the actual receiver design [35, 36], see also Chapters 10 and 11.

With the previous assumptions, the relay region is obtained by solving the following two equations simultaneously:
\[
d_{ij}^4 \geq d_{ir}^4 + d_{rj}^4 + c/t
\]
and
\[
d_{ij}^2 = d_{ir}^2 + d_{rj}^2 - 2d_{ir}d_{rj} \cos \theta
\]
where \( \theta \) is the angle between position vectors \( \mathbf{r}_{r \rightarrow i} \) and \( \mathbf{r}_{r \rightarrow j} \). In Inequality (15.43), \( c \) denotes the additional receiver power cost of 20 mW for relays, and \( t \) the predetection threshold of \( 10^{-7} \) mW. Figure 15.9 displays the relay region in the case where the relay node is at (0,0), and the transmit node is at (80,0). The relay region has been shaded. The units are meters.

### 15.2.4.4 Stationary network simulation

We now present the simulation results for a stationary network with nodes deployed over a square region of 1 km on each side. The \((x, y)\) coordinates of the nodes are generated as independent, identically distributed (i.i.d.) uniform random variables over this region. Since the nodes are stationary, once each node is enclosed and obtains a valid cost, the network remains in the minimum power topology.

The transmit and receiver powers for providing point-to-point connections are as described above. In this simulation, we investigate how the total power consumption of the minimum power topology varies with the number of nodes. Figure 15.10 illustrates this relationship. As the number of nodes
grows larger, the average power decreases toward its asymptote of 300 mW receiver power/node. The plot has been normalized to the receiver power.

### 15.2.5 Distributed mobile networks

The protocol presented so far has been for stationary networks. However, due to the localized nature of its search algorithm, it proves to be an effective energy-conserving protocol for the mobile case as well.
Synchronization in a mobile network can be achieved by use of the absolute time information provided by GPS up to 100 ns resolution [37]. The network synchronization is discussed in Section 14.3. In a synchronous network, each node wakes up regularly to ‘listen’ for change and goes back to the sleep mode to conserve power. The time between successive wakeups is referred to as the cycle period of the network. If the cycle period is too long, the power costs to the master site can change significantly from one wakeup to the next. In this case, the network cannot track the correct costs. If the cycle period is too short, then the network consumes unnecessary energy to compute costs that change only slowly. The choice of the cycle period for energy-efficient operation of a wireless network must address this tradeoff. In our simulation, we assume that the cycle period has been chosen to meet these two constraints.

After wakeup, each node executes Phase 1 of the protocol, as described in the previous section. When a node completes Phase 2, it either starts data transmission on the optimal link, or it goes to the sleep mode to conserve power.

The protocol is self-reconfiguring since strong connectivity is ensured within each cycle period, and the minimum power links are dynamically updated. It can be seen that this protocol is also fault tolerant. A network protocol is ‘fault tolerant’ if it is self-reconfiguring when nodes leave or new nodes join the network. Under such a scenario, each node employing the protocol would compute its new enclosure and find the minimum power topology.

15.2.5.1 Mobile network simulation

In this case, the initial positions of 100 nodes are generated as i.i.d. uniform random variables over a square field, 1 km on each side. The velocity in each coordinate direction is uniformly distributed on the interval $(-v_{max}, v_{max})$. The velocity is the vector sum of the velocities in each coordinate direction. Parameter $v_{max}$ is varied to observe how the energy consumption changes.

The choice of the SetSearchRegion function in the search algorithm, which is optimized to perform the minimum energy neighbor search, is a separate topic. In this simulation, omnidirectional antennas and an heuristic strategy for the choice of the search radius are assumed. The results indicate that even with an heuristic strategy, the energy consumption is very low.

Let $T$ be the cycle period of the network. Assume that node $i$ is enclosed in the $n$th iteration, and let $e_n$ be the distance of $i$ to its furthest neighbor in the $n$th iteration. In the next iteration, if $i$ sets its search radius to

$$r_{n+1} = e_n + 2\sqrt{2}v_{max}T$$

then its neighbors in the $n$th iteration must fall within this radius. Because the cycle period is small enough to allow positions to vary only slightly from one iteration to the next, in most cases the node will have its previous neighbors in its new enclosure as well. Nodes employing this strategy are enclosed within one iteration of the search algorithm presented earlier in this section.

From a system perspective, the measure of mobility is not the velocities, but rather the displacements of nodes in a cycle period of the network. The maximum displacement of a node in a cycle period is $\sqrt{2}v_{max}T$ from the previous analysis.

Figure 15.11 displays the search period power level per node averaged over 10 000 iterations and averaged over all the nodes. The horizontal axis on this graph is the maximum displacement in meters. Since the average distance between nodes is about 100 m in this particular simulation, it was estimated that the network cannot track correct costs for maximum displacements greater than 8 m, so that power consumption over only this range was graphed.

Figure 15.12 displays the search period power consumption per meter of maximum displacement. The graph indicates that the power consumption per node scales better than linearly with maximum displacement for the range of displacements for which the network can track the correct costs.
Figure 15.11 Power consumption per node during search period.

Figure 15.12 Power consumption per node per meter of maximum displacement during a search.
15.3 LEAST RESISTANCE ROUTING IN WIRELESS NETWORKS

In the previous section we have already seen that packet radio networks can provide wireless communication and data distribution among mobile terminals, but adaptive protocols are required in order for them to do so. Two important characteristics of a communication link in a mobile packet radio network are its unreliability and its variability. The links in such a network are unreliable because of fading, interference, noise, and perhaps failure of the transmitting or receiving radios. They are variable because of the mobility of the radios, the dynamic nature of the propagation medium and the interference. The interference can be generated externally (e.g. another system operating in the same frequency band or intentional jamming), or it can be generated within the network (e.g. multiple access interference).

If a network is to provide reliable service over communication links with these characteristics, the link and network protocols must adapt to changes in the network. In order to adapt effectively, these protocols must be provided with information about the current status of certain elements of the network, such as the communication channels and radio buffers. The mobile networks considered in this section are multiple hop networks with distributed control, including distributed forwarding and routing protocols. The presentation is based on [38]. Since there is no central controller or other entity that can provide status information for all of the network elements, the required information must be derived by the radios themselves. As in the previous section, the exchange of information is among neighboring radios (i.e. local exchange rather than global).

15.3.1 Least resistance routing (LRR)

An important feature of LRR is its use of link quality information as a basis for route selection. The link resistance, which is determined for each link in the network, provides a measure of the noise and interference in the communication channel and the congestion in the receiving radio. The link resistance is a quantitative measure of the receiving radio’s ability to demodulate, decode, store and forward a packet that is transmitted to it on that link.

The metric in LRR specifies how a radio is to calculate the resistance for each of its incoming links. The probability that a transmission from radio A to radio B is successful depends on both the condition of the communication channel from A to B and the ability of B to store and forward the packet. In this section we restrict attention to metrics of the form

\[ LR(A, B) = \alpha I(A, B) + \beta W(B) \]  

(15.46)

where LR(A, B) is the resistance of the link from radio A to radio B. The term I(A, B) represents the resistance of the communication channel from A to B and the term W(B) represents the resistance of radio B. The coefficients \( \alpha \) and \( \beta \) are selected to give the desired relative weightings to these two components of the link resistance.

The resistance of the communication channel accounts for fading, propagation loss and other features that are unique to transmissions from A to B. The resistance of radio B characterizes the conditions at radio B that equally affect each of the transmissions to radio B. Included in these conditions are the number of packets in the radio’s buffer, the amount of traffic near the radio, interference that equally affects each transmission to radio B, and the expected delay in forwarding packets. If all packets transmitted to radio B are received at approximately the same power level, any radio frequency (RF) interference has approximately the same effect on each transmission to radio B, in which case the interference could be accounted for in the term W(B) only. In practice, however, the packets may be transmitted at different power levels, and the propagation losses for the different communication channels to radio B usually differ greatly. Because a given source of RF interference may have substantially different effects on transmissions to radio B from different radios, it is better to include the effects of interference in the resistance of the communication channel rather than in the resistance of the receiving radio.
15.3.2 Multimedia least resistance routing (MLRR)

In MLRR, each link in the network is assigned a link resistance for each of the types of traffic that it may need to handle. The link resistance can be thought of as a vector with a component for each different type of traffic. A different metric can be employed for each component in order to provide a mechanism for accounting for different service requirements for different message types.

In order to simplify the presentation this section will focus on two types of multimedia packet, type-D packets and type-V packets. The intent is that type-D packets have the characteristics associated with data traffic (e.g. computer file transfers), while type-V packets have the characteristics associated with voice or video traffic. Type-D packets must be delivered to their destinations with no errors or erasures; however, a moderate delay is permitted. On the other hand, type-V packets are required to be delivered with much less delay than type-D packets, but the type-V packets may have a small number of frame erasures and still be considered acceptable to the destination. In describing the components of the link resistance, the subscripts $d$ and $v$ are used to distinguish between the link resistance for the two types of packet. The two components of the link resistance for MLRR are

\[
LR_d(A, B) = \alpha_d I_d(A, B) + \beta_d W_d(B) \tag{15.47}
\]

and

\[
LR_v(A, B) = \alpha_v I_v(A, B) + \beta_v W_v(B) \tag{15.48}
\]

In the example used to illustrate MLRR in this section, the importance of the channel resistance relative to the receiver resistance is determined for each type of traffic from the following specifications. A type-D packet is accepted by its destination if all the received words in the packet are decoded correctly. A type-V packet is accepted by its destination if $\lambda$ or fewer of the received words in the packet do not decode correctly.

The selection of the coefficients to provide the proper weighting of the channel resistance and the radio resistance will be aided by the simulation results, in which the general trends will be observed. Because type-D packets are more sensitive to errors than to delay, $\alpha_d$ is set to a relatively large value compared to $\beta_d$. Type-V packets are more sensitive to delay than to errors, however, so $\beta_v$ is set to a relatively large value compared to $\alpha_v$.

If a received word in a type-V does not decode correctly, that word is erased. If the receiving radio is not the final destination for the packet, the packet is forwarded with the erasure inserted in place of the missing word. Because of the requirements on type-V packets, any radio along the route discards any type-V packet that has accumulated more than $\lambda$ word erasures. The choice for the value of $\lambda$, the maximum number of word erasures that are permitted, depends primarily on the speech or video compression method.

The packet rejection probability is defined as the probability that a packet with no previous word erasures is discarded as a result of a single transmission. For frequency-hopping (FH) systems this probability depends on the code, the modulation and demodulation, the number of other simultaneous FH transmissions, the number of words per packet and the signal to noise ratio on the channel.

The curves in Figure 15.13 illustrate the sensitivity of the packet reception probability to the value of $\lambda$. In this graph, the packet rejection probability is shown as a function of $E_b/N_0$ for six different values of $\lambda$, where $E_b$ is the energy per information bit and $N_0$ is the one-sided spectral density for the thermal noise. For convenience, we refer to $E_b/N_0$ as the signal to noise ratio in the text of this section. For the results given in Figure 15.13, binary orthogonal modulation and optimum non-coherent demodulation are employed, there are three interfering FH transmissions, a $(32, 22)$ extended Reed–Solomon code is used, and there are 15 code words per packet.

In general, as discussed in Chapters 2 and 3, one may want to use different modulation techniques or different combinations of modulation and error control coding for different types of multimedia traffic. If the modulation or coding is different for different types of packet, the metrics for the channel resistance may differ also. For the numerical examples presented in this section, the coding and modulation are the same for both types of packet, and $I_v$ and $I_d$, the metrics used for the channel resistance, are identical. The metric for each type of message is the errors and erasures (EE) metric,
Figure 15.13 Packet rejection probability for six different values of $\lambda$.

and the coefficients $\alpha_d$ and $\alpha_v$ are adjusted to account for the differences in the requirements for the two types of packet.

The resistance components for the receiver (i.e. $W_d$ and $W_v$) are used to represent the delay that a packet is expected to experience before it is forwarded by a radio that has decoded the packet successfully. The number of packets waiting in the receiving radio’s buffer is a simple metric for estimating the expected forwarding delay. The number of type-D packets at a radio is denoted by $N_d$ and the number of type-V packets is $N_v$. The receiver resistance components are then defined as

$$W_d(B) = N_d + \omega_d N_v$$
$$W_v(B) = N_v + \omega_v N_d.$$  

15.3.3 Network performance examples: LRR versus MLRR

For the numerical examples included in this section, the (32, 22) extended Reed–Solomon code is employed with errors-and-erasures decoding. There are 15 code words per packet, and the code words are fully interleaved. Ten test symbols are included in each dwell interval. A type-V packet is accepted if no more than three words have been erased (i.e. $\lambda = 3$).

For both the MLRR and LRR routing protocols, the forwarding protocol gives type-V packets a higher priority than type-D packets. A radio that is preparing to transmit a packet first checks for a type-V packet in its buffer, and transmits such a packet if possible. If there are no type-V packets in the buffer, or if none of the type-V packets can be forwarded because of the retransmission policy, then the radio checks for a type-D packet to transmit.

The service requirements of a packet also affect the number of times a radio may retransmit that packet. For the results presented here, type-V packets are permitted up to two forwarding attempts at a given radio, while type-D packets are permitted six attempts at most. To prevent outdated packets from congesting the network, packets expire and are discarded after a certain number of packet intervals since they were generated. For the performance curves that follow, this number is 500 for type-D packets and 50 for type-V packets.
Various combinations of parameters for the MLRR resistance metrics have been examined, and it was found that the following two metrics perform well over a wide range of network conditions. For the results presented in this section, the two metrics employed for MLRR are $LR_d = 2t + e$ and $LR_e = t + 0.5e + N_e + 0.7N_d$. The metric employed for LRR is the EE metric, $LR = 2t + e$. A word that has no more than $e$ erasures and $t$ errors is decoded correctly if $2t + e$ does not exceed $n - k$.

### 15.3.3.1 Example 1: Nine-node network

An example of a network with nine radios is illustrated in Figure 15.14 [38]. The signal to noise ratios for the channels represented by solid lines is 15 dB, which results in a very low error probability on the channel. The signal to noise ratios for the channels shown with dashed lines are indicated by the labels on the channels. Both type-D and type-V packets are generated at radio 1 and routed to radio 9. In a given packet interval, the probability that a marked type-D packet is generated is denoted by $q_d$, and the probability that a marked type-V packet is generated is denoted by $q_d$. A radio can generate no more than one packet of each type within a packet interval. Additionally, the interfering type-D packets are generated at radios 5 and 6 and sent to radios 9 and 1, respectively, and their generation probability is fixed at 0.05. The inclusion of these interfering packet results in increased congestion in the upper three routes of the network shown in Figure 15.14.

This simple network topology is selected to illustrate the advantages of using multimedia routing. The two types of packet generated at radio 1 must be routed to radio 9, but there is not a single route that is preferred for both types of packet. The upper route has large delays because of the congestion caused by the extra packets generated at radios 5 and 6. On the other hand, the lower route has poorer quality channels that give a bit error rate in the range of 1–5%. Although type-V packets can be accepted with some word erasures, type-D packets cannot. For a given channel, this difference in service requirements leads to a larger expected number of retransmissions for type-D packets than for type-V packets.

Two scenarios are presented that illustrate the value of selecting a route for each packet type rather than using a single ‘best’ route for all packet types. For the first scenario, the generation probability for type-D packets is fixed at $q_d = 0.04$, and the network performance is evaluated as the generation probability for type-V packets is increased. This scenario is of particular interest, because it has been reported that some existing radio networks suffer from severely degraded service in the delivery of data traffic when voice traffic increases significantly in the network. The results of the simulation for this scenario are shown in Figure 15.15. The results for MLRR are shown as dashed curves, and the results for LRR are shown as solid curves.

---

Figure 15.14 Nine-node network.
In the second scenario the situation is reversed: the generation probability for type-V packets is held constant at \( q_v = 0.04 \), and the generation probability for type-D packets is increased. The results in Figure 15.16 for this scenario are similar to those discussed above for the first scenario. In particular, the MLRR protocol maintains a high level of throughput and a large end-to-end success probability for type-V packets, even as the generation probability for type-D packets is increased.

![Figure 15.15 (a) Throughput for \( q_d = 0.04 \); (b) word success probability for \( q_d = 0.04 \); (c) Delay of type-V packets, \( q_d = 0.04 \).](image)
However, for the LRR protocol, the throughput and success probability drop as the type-D traffic increases in the network. The MLRR protocol gives better performance than the LRR protocol for all situations depicted in Figure 15.16.

### 15.3.3.2 Example 2: Twelve-node network

In this case, the MLRR and LRR protocols are used in the 12-node network illustrated in Figure 15.17 [38], in which the number of possible routes and the amount of traffic are larger than in the nine-node network of Example 1. The 12 radios are connected by communication channels for which the signal to noise ratios are as labeled in Figure 15.17. Packets of both types have the origin-destination pairs (5, 6), (6, 5), (7, 8) and (8, 7). Marked type-D packets have generation probability $q_d = 0.04$, and the generation probability for marked type-V packets is denoted by $q_v$. Radios 1–4 each generate additional interfering type-D packets with origin-destination pairs (1, 4), (4, 1), (2, 3) and (3, 2). The generation probability for these interfering packets is fixed at 0.05.

The throughput for each packet type is shown in Figure 15.18(a), and the end-to-end word success probability is shown in Figure 15.18(b). As $q_v$ increases, the throughput for type-D packets declines rapidly for the LRR protocol, but a higher throughput is maintained by MLRR. Furthermore, the word success probability for type-D packets is uniformly larger for MLRR than for LRR. For type-V packets, the MLRR protocol gives higher throughput, as shown in Figure 15.18(a), and larger word success probabilities, as shown in Figure 15.18(b).

### 15.3.4 Sensitivity to the number of allowable word erasures

Because the performance of MLRR depends on $\lambda$, the number of word erasures that are permitted in a type-V packet, it is important to examine the sensitivity of the throughput and word success probability to the value of this parameter. The LRR protocol does not attempt to distinguish between type-D and type-V packets, so the performance of LRR does not depend significantly on the value of...
To illustrate the dependence of MLRR on the choice of $\lambda$, results are presented for the nine-node network of Example 1.

The generation probability for type-D packets is fixed at $q_d = 0.04$, and the network performance is evaluated as a function of the generation probability for type-V packets. The results for the nine-node network of Example 1 are shown in Figure 15.19. We see that the performance of MLRR for type-D

![Figure 15.19](image_url)

Figure 15.19 (a) Throughput for $q_v = 0.04$; (b) word success probability for $q_v = 0.04$; (c) delay of type-V packets, $q_v = 0.04$. 
packets is approximately independent of the value of $\lambda$, but the performance of MLRR for type-V packets improves as the number of allowable word erasures is increased. If $\lambda = 0$, the performance for type-V packets is still significantly better for MLRR than for LRR, but the performance advantage obtained from MLRR is even greater if $\lambda = 1$ (i.e. a type-V packet is accepted if no more than one of its words is erased). A small performance increase is obtained as the number of allowable word erasures is increased from $\lambda = 1$ to $\lambda = 2$, but additional increases in $\lambda$ produce no significant benefit for this network. If the number of relays required for the type-V packets is much larger than in the networks we consider in this section, there may be a significant difference in performance if a larger number of word erasures can be permitted.

More information on the topic can be found in [39–42].
Figure 15.18  (a) Throughput for $q_d = 0.04$; (b) word success probability for $q_d = 0.04$.

15.4 POWER OPTIMAL ROUTING IN WIRELESS NETWORKS FOR GUARANTEED TCP LAYER QoS

15.4.1 Constant end-to-end error rate

In this section we consider the case when a packet is transmitted from a source to its destination along multiple hops, where there is some probability of error per hop that depends on the distance between the hops and the transmit power. In Section 15.2, the assumption was that if the received power (signal to noise ratio)$P_{\text{recv}} \geq \gamma$, where $\gamma$ is some constant, then the packet is successfully received; otherwise the packet is lost. The symbol error rate (SER) is a monotonically decreasing function of
\( P_{\text{recv}} \), therefore, for each hop, \( P_{\text{recv}} \) can be made large enough that the SER for the hop satisfies

\[
\text{SER} \leq \text{SER}(\gamma) \tag{15.49}
\]

Since any transmission will use the minimum amount of power required to meet the necessary error rate, this means that \( \text{SER} = \text{SER}(\gamma) \). Assuming that errors per hop are independent, this implies that

![Figure 15.19](image-url)

Figure 15.19 (a) Throughput for MLRR with \( q_d = 0.04 \); (b) end-to-end word success probability for MLRR with \( q_d = 0.04 \); (c) end-to-end delay for type-V packets with \( q_d = 0.04 \).
Figure 15.19 (Cont.).

the error seen by the transport control protocol layer (end-to-end error rate) SER_{e2e} is given by

$$\text{SER}_{e2e} = 1 - (1 - \text{SER}(\gamma))^N$$  \hspace{1cm} (15.50)

where $N$ is the number of hops along the path. SER_{e2e} is monotonically increasing with $N$.

Instead of using a routing scheme and power cost metric that allows the end-to-end error rate to increase with hop count, we examine the effect of constraining the end-to-end error rate to be a constant. This criterion will be referred to as TCP layer power optimal routing.

### 15.4.2 Optimization problem

The presentation in this section is based on [43]. Let $X = (X_0, X_1, \ldots, X_N)$ be an $N$-hop path from node $X_0$ to $X_N$ that passes through nodes $X_1, X_2, \ldots, X_{N-1}$ in that order. Let $P_i$ be the power allocated for the hop $(X_{i-1}, X_i)$, and let $\text{SER}_i$ be the corresponding symbol error rate for the hop $(i = 1, 2, \ldots, N)$. Assuming independent errors per hop, the TCP layer QoS (the end-to-end SER) is given by

$$\text{SER}_{e2e} = 1 - \prod_{i=1}^{N} (1 - \text{SER}_i)$$  \hspace{1cm} (15.51)

We define the power cost function as follows:

$$\mathcal{PC}^*(\varepsilon; X) = \min_{P_1, \ldots, P_N} \sum_{i=1}^{N} P_i \quad \text{given} \quad \text{SER}_{e2e} \leq \varepsilon$$  \hspace{1cm} (15.52)

We will assume that we are interested in the case when the quantity $\varepsilon$ is much smaller than one, i.e. when there is a very low probability of error.

A quick analysis shows that the optimization problem in Equation (15.52) leads to equations that do not have simple closed form solutions in terms of the power per hop and the power cost, even for the simplest models for the SER. We formulate our approximate power cost metric by the following
equations:

$$\mathcal{PC}(\varepsilon; X) = \min_{P_1, \ldots, P_N} \sum_{i=1}^{N} P_i \quad \text{given} \quad \sum_{i=1}^{N} \text{SER}_i \leq \varepsilon$$

We can justify this approximation by the following result: assuming $\varepsilon \geq 0$ and $\varepsilon + 4\varepsilon^2 < 1/\sqrt{2}$, and given the power cost metrics $\mathcal{PC}^*$ and $\mathcal{PC}$ defined by Equations (15.52) and (15.53) respectively, we have:

$$\mathcal{PC}(\varepsilon + 4\varepsilon^2; X) \leq \mathcal{PC}^*(\varepsilon; X) \leq \mathcal{PC}(\varepsilon; X) \quad (15.54)$$

The proof of Inequality (15.54) only relies on the power cost metric being a minimization problem where the error rate constraint is met with equality at the solution. In particular, it does not depend on the expression for $\text{SER}_i$, or the fact that the metric being minimized is the sum of the power per hop.

For the rest of this section, we use $\mathcal{PC}$ as the power cost metric because it will lead to analytical expressions for the power cost of a path. While this is an approximation, Inequality (15.54) provides a way of bounding the error $E(\varepsilon; X) = \mathcal{PC}(\varepsilon; X) - \mathcal{PC}^*(\varepsilon; X)$ since

$$E(\varepsilon; X) \leq \mathcal{PC}(\varepsilon; X) - \mathcal{PC}(\varepsilon + 4\varepsilon^2; X) \quad (15.55)$$

When we compute the power cost with specific models for $\text{SER}$, we will use this expression to show that the approximation error is $O(\varepsilon)$, and thus small compared to the power cost of the path.

Error correction mechanisms (both ARQ and FEC) can be easily included in this framework with minor changes, as discussed in [44].

### 15.4.3 Error rate models

#### 15.4.3.1 Time-invariant attenuation

The first model is a deterministic power model where we assume that the receive power of a link is attenuated by a time-invariant quantity. This attenuation coefficient can be given a physical interpretation by setting it to $d^{\alpha}/a$, where $d$ is the distance between the transmitter and receiver, and $\alpha > 2$ and $a$ are constants. We assume that the expression for the $\text{SER}$ of a link is given by:

$$\text{SER}_j = be^{-P_j/a_j} \quad (15.56)$$

where $P_j$ is the transmit power and $a_j$ is the time-invariant attenuation coefficient of the link. Equation (15.56) is sufficiently parameterized to be able to provide a bound for the probability of error for most digital modulations that are detected optimally in the presence of additive Gaussian noise [45].

Under the assumptions of Inequality (15.54) and the link $\text{SER}$ assumption given by Equation (15.56), the optimal power cost $\mathcal{PC}(\varepsilon; X)$ for path $X = (X_0, X_1, \ldots, X_N)$ is obtained when

1. $\text{SER}_j = e^{\frac{a_j}{\sum_{i=1}^{N} a_i}}$

2. $P_j = a_j \left( \log(b/\varepsilon) + \log \left( \sum_{i=1}^{N} \frac{a_i}{a_j} \right) \right) \quad (15.57)$

3. $E(\varepsilon; X) \leq \log(1 + 4\varepsilon) \sum_{j=1}^{N} a_j$

where $a_j$, $\text{SER}_j$, and $P_j$ are the attenuation coefficient, link error rate, and link power allocation, respectively, for link $(X_{j-1}, X_j)$. 
15.4.3.2 Large- and small-scale fading

In a wireless mobile scenario the received power is affected by many more factors than the mere
distance, and it is common to represent the received power as a doubly stochastic random variable,
with long-term and short-term variations (see Chapter 14). The transmit power is attenuated by two
factors: \( G_L(t) \) caused by large-scale fading, and \( G_S(t) \) caused by small-scale fading. To account
for the effects of large- and small-scale fading, the time-varying SER of a link is given by the random
process

\[
SER_i = e^{-a \Omega_i}
\]  

(15.58)

where \( a \) and \( b \) are constants, and \( \Omega_i \) is the received power, whose statistics are functions of position
and time. As is standard, we assume a log-normal distribution for the large-scale fading coefficient.
For the small-scale fading, several distributions have been introduced [26] and using Equation (15.58),
the corresponding average SER for a given large-scale fading parameter is the characteristic function
of the small-scale fading density. For example, for the Nakagami \( m \)-distribution, the expected SER
for a link is given by [26]:

\[
SER_i = b(1 + P_i/a_i)^{-m}
\]  

(15.59)

where \( a_i \) is the contribution from the slow-varying large-scale fading coefficient and \( P_i \) is the average
transmit power.

The Nakagami \( m \)-distribution captures the intermediate ground between strong line-of-sight and
non-line-of-sight systems. Note, in fact, that Rayleigh fading is a special case of Nakagami fading
when \( m = 1 \), while the deterministic case is obtained as \( m \to \infty \).

One can generalize the previous power cost expressions and power optimal routing to these
scenarios.

Under the assumptions of Inequality (15.54) and the link SER assumption given by Equation (15.59),
the optimal power cost \( PC(\varepsilon; X) \) for path \( X = (X_0, X_1, \ldots, X_N) \) is obtained when

1. \( SER_j = \varepsilon \frac{a_j^{m/(m+1)}}{\sum_{i=1}^{N} a_i^{m/(m+1)}} \)

2. \( P_j = a_j \left( \frac{b}{\varepsilon} \right)^{1/m} \left( \sum_{i=1}^{N} a_i^{m/(m+1)} \right)^{1/m} - 1 \)  

(15.60)

3. \( E(\varepsilon; X) \leq \frac{4\varepsilon}{m} \left( \frac{b}{\varepsilon} \right)^{1/m} \left( \sum_{i=1}^{N} a_i^{m/(m+1)} \right)^{(m+1)/m} \)

where \( a_j \), \( SER_j \), and \( P_j \) are the large-scale attenuation coefficient, link error rate, and link power
allocation, respectively, for link \((X_{j-1}, X_j)\).

15.4.4 Properties of power optimal paths

In this section we discuss some of the consequences of adopting a TSP layer QoS (end-to-end SER)
constraint for power optimization. In particular, we compare these results to the concepts similar to
those discussed in Section 15.2, which assume that the amount of power required for a link \((X_{i-1}, X_i)\)
is given by

\[
P_i = \log(b/\varepsilon) \frac{d_i^\alpha}{a} \]  

(15.61)

where \( a \) and \( \alpha > 2 \) are constants, and \( d_i = |X_{i-1} - X_i| \) is the distance between points \( X_{i-1} \) and \( X_i \).
This model assumes that the link SER is constant (= \( \varepsilon \)), and the attenuation coefficient \( a_i \) is given by
\( d_i^\alpha / a \).
Under this model, the power cost of using a link does not depend on the path under consideration, unlike the link power allocation in Equations (15.57) and (15.60). The power cost using this model, which we denote $KC(\varepsilon; X)$, is given by

$$KC(\varepsilon; X) = \log(b/\varepsilon) \sum_{i=1}^{N} \frac{d_{i}^{\alpha}}{a} = \log(b/\varepsilon) \sum_{i=1}^{N} \alpha_{i},$$

(15.62)

where $d_{i} = |X_{i-1} - X_{i}|$ is the distance between points $X_{i-1}$ and $X_{i}$. Note that since $\varepsilon$ only appears in the factor in front of this particular power cost metric, the best path to route a packet according to $KC$ will not depend on $\varepsilon$. [46].

We compare the properties of power optimal paths obtained by the metric introduced in Equation (15.57) with those obtained using the metric from Equation (15.62).

### 15.4.4.1 Comparing power cost metrics

If we use Equation (15.61) to determine the power for a hop, then the SER per hop can be as high as $\varepsilon$, thereby increasing the total end-to-end SER. The metric $KC$ will therefore underestimate the amount of power required to transmit a packet along a path with an error that does not exceed $\varepsilon$. Examining the expressions in Equation (15.57), we conclude that

$$PC(\varepsilon; X) = \sum_{j=1}^{N} a_{j} \left( \log(b/\varepsilon) + \log \left( \sum_{i=1}^{N} \frac{a_{i}}{a_{j}} \right) \right)$$

$$\geq KC(\varepsilon; X)$$

(15.63)

with equality holding if and only if we are considering a one-hop path. If we treat $p_{j} = a_{j}/\sum_{i=1}^{N} a_{i}$ as a probability distribution, observe that

$$PC(\varepsilon; X) = \left( \sum_{j=1}^{N} a_{j} \right) \left( \log(b/\varepsilon) + \sum_{i=1}^{N} p_{i} \log 1/p_{i} \right)$$

$$\leq KC(\varepsilon; X) \left( 1 + \frac{\log N}{\log(b/\varepsilon)} \right)$$

(15.65)

with equality holding when $p_{1} = p_{2} = \cdots = p_{N}$. When examining long paths with equidistant hops, using a simple additive cost function will underestimate the power cost by a factor that grows logarithmically with the number of hops compared to a metric that keeps the end-to-end error bounded.

More details on adaptive routing in wireless networks can be found in [47–74].

### REFERENCES


As indicated in Chapter 10, cognitive radio is able to observe the environment and, based on this, make a number of decisions on its own. A potential problem with using cognitive radios in a network is their selfish behavior. Therefore, efficient algorithms/protocols are needed to motivate such terminals to cooperate in the interest of overall social gain. In this chapter we analyse the behavior of such terminals in wireless networks by using a game theory framework. In the sequel algorithms/protocols based on cognition, used by these radios, will be referred to as cognitive algorithms/protocols and a network of cognitive radios coordinating such algorithms/protocols will be referred to as a cognitive network.

16.1 COGNITIVE POWER CONTROL

16.1.1 Noncooperative power control game

As the first step, in this section we present a power control solution for wireless data within a game theoretic framework. In this context, the equivalent quality of service (EQoS) that a wireless terminal receives is referred to as the utility, and distributed power control is a noncooperative power control game where users maximize their utility. In addition to the conventional QoS parameters, like BER, the EQoS incorporates requirements that a given BER = $P_e$ is obtained with minimum power, which preserves the energy and minimizes the interference inflicted on other users. The outcome of the game results in a Nash equilibrium that is inefficient. By introducing pricing of transmit powers, Pareto improvement of the noncooperative power control game is obtained.

Assume a single-cell system where each user transmits $L$ information bits in frames (packets) of $M > L$ bits at a rate $R$ b/s using $p$ W of power. We assume fixed rate $R$ for all terminals. The utility function suggested in [1–3] can be presented as:

\[ u = \frac{L R f(\gamma)}{M p} \text{ bits joule} \]

\[ f(\gamma) = (1 - 2P_e)^M \]  

(16.1)
where $P_e$ is the BER. Parameter $f(\gamma)$ is derived from the probability of correct frame reception with no coding, $P_e = (1 - P_e)^n$, by modification $P_e \rightarrow 2P_e$ so that $f(\gamma)$ has maximum value 1 for $P_e = 0$ and minimum value 0 for $P_e = 0.5$. Assuming perfect error detection and no error correction, $P_e$ is given as $Q(\sqrt{2\gamma})$ for BPSK, $0.5 \exp(-\gamma)$ for DPSK, $Q(\sqrt{\gamma})$ for CFSK and $0.5 \exp(-\gamma/2)$ for NCFSK in the case of an additive white Gaussian noise (AWGN) channel. In all cases, the BER decreases monotonically with SIR $\gamma$. Therefore, $P_e$ can be expressed as a function of $\gamma$ and substituted in Equation (16.1) to obtain the utility function for a specific system.

With this in mind, let $G = [N, \{P_j\}, \{u_j(\cdot)\}]$ denote the noncooperative power control game (NPG) where $N = \{1, 2, \ldots, N\}$ is the index set for the mobile users currently in the cell, $P_j$ is the strategy set, and $u_j(\cdot)$ is the payoff function of user $j$. Each user selects a power level $p_j \in P_j$. Let the power vector $p = (p_1, \ldots, p_N) \in P$ denote the outcome of the game in terms of the selected power by all the users, where $P$ is the set of all power vectors. The resulting utility level for the $j$th user is $u_j(p)$. In the sequel we will also use notation $u_j(p_j, p_{-j})$ where $p_{-j}$ denotes the vector consisting of elements of $p$ other than the $j$th element. The latter notation emphasizes that the $j$th user has control over its own power, $p_j$, only. The strategy space of all the users excluding the $j$th user is denoted by $P_{-j}$. Now, what the utility user obtains by expending $p_j$ can be expressed more formally as:

$$ u_j(p_j, p_{-j}) = \frac{LR}{M p_j} f(\gamma_j) \text{ bits joule} $$

(16.2)

where $\gamma_j$ is the SIR of user $j$. In CDMA uplink this is defined as:

$$ \gamma_j = \frac{W}{R} \frac{h_j p_j}{\sum_{j \neq i} h_j p_j + \sigma^2} $$

(16.3)

where $W$ is the available spread-spectrum bandwidth, $\sigma^2$ is the AWGN power at the receiver, and $\{h_j\}$ is the set of path gains from the mobile to the base station. We assume that the strategy space $P_j$ of each user is a compact, convex set with minimum and maximum power constraints denoted by $\bar{p}_j$, respectively, which results in the strategy space $P_j = [0, \bar{p}_j]$. The utility function takes the generic form given in Figure 16.1 for fixed interference.

![Figure 16.1 Utility versus the user transmit power for fixed interference.](image)
where $u_j$ is given in Equation (16.2) and $P_j = [0, \bar{p}_j]$ is the strategy space of the user. The transmit power that optimizes individual utility depends on the transmit powers of all the other terminals in the system. It is necessary to characterize a set of powers where the users are satisfied with the utility they receive given the power selections of other users. Such an operating point is called an equilibrium.

16.1.2 Nash equilibrium

The solution that is most widely used for game theoretic problems is the Nash equilibrium.

A power vector $\mathbf{p} = (p_1, \ldots, p_N)$ is a Nash equilibrium of the NPG $G = [\mathbf{N}, P_j, u_j(\cdot)]$ if, for every $j \in \mathbf{N}$, $u_j(p_j, \mathbf{p}_{-j}) \geq u_j(p'_j, \mathbf{p}_{-j})$ for all $p'_j \in P_j$.

In other words, at a Nash equilibrium, given the power levels of other players, no user can improve its utility level by making individual changes in its power. The power level chosen by a rational self-optimizing user constitutes a best response to the powers actually chosen by other players. Formally, a terminal's best response $r_j : P_{-j} \rightarrow P_j$ is the correspondence that assigns to each $\mathbf{p}_{-j} \in P_{-j}$ the set:

$$r_j(\mathbf{p}_{-j}) = \{ p_j \in P_j : u_j(p_j, \mathbf{p}_{-j}) \geq u_j(p'_j, \mathbf{p}_{-j}) \text{ for all } p'_j \in P_j \}$$

The Nash equilibrium can be restated as: $\mathbf{p} = (p_1, \ldots, p_N)$ is a Nash equilibrium of the NPG $G = [\mathbf{N}, \{P_j\}, \{u_j(\cdot)\}]$ if and only if $p_j \in r_j(\mathbf{p}_{-j})$ for all $j \in \mathbf{N}$. The Nash equilibrium concept offers a predictable, stable outcome for a game where multiple agents with conflicting interests compete through self-optimization, and reach a point where no player wishes to deviate. However, such a point does not necessarily exist. A Nash equilibrium exists in the NPG, $G = [\mathbf{N}, \{P_j\}, \{u_j(\cdot)\}]$.

The rigorous proof can be found in [1–6]. Here we outline only the main logic of the proof.

Nash equilibrium exists in the game $G = [\mathbf{N}, \{P_j\}, \{u_j(\cdot)\}]$ if, for all $j = 1, \ldots, N$, then

(i) $P_j$ is a nonempty, convex, and compact subset of some Euclidean space $\mathbb{R}^N$.

(ii) $u_j(\mathbf{p})$ is continuous in $\mathbf{p}$ and quasi-concave in $p_j$.

The proof is completed by showing that the conditions (i) and (ii) are met in NPG.

Each user has a strategy space that is defined by the power values between a minimum and a maximum power. Thus, the first condition is satisfied. It remains to be shown that the utility function $u_j(\mathbf{p})$ is quasi-concave in $p_j$ for all $j$ in NPG. This has been already indicated by the example in Figure 16.1. Exact proof can be found in [1–3].

For a differentiable function, the first-order necessary optimality condition is given as $(\partial u_j(p_j, \mathbf{p}_{-j})/\partial p_j) = 0$. From Equation (16.2), the partial derivative of $u_j(\cdot)$ with respect to $p_j$ is

$$\frac{\partial u_j(p_j, \mathbf{p}_{-j})}{\partial p_j} = \frac{LR}{MP_j} \left( f'(y_j) \gamma_j - f(y_j) \right)$$

where $f'(y_j) = df(y_j)/d\gamma_j$. For noncoherent FSK, Equation (16.6) gives $(M/2)\gamma_j e^{-\gamma_j/2} - (1 - e^{-\gamma_j/2}) = 0$ or

$$\frac{M}{2} \gamma_j + 1 = e^{\gamma_j/2}.$$  

resulting in an optimal value $\gamma_j = \bar{\gamma}$ that can be derived numerically as a solution of Equation (16.7). It has the same value for all users, assuming each user operates with the same efficiency function. The second order partial derivative of the utility with respect to the power reveals that this point is a local
where of utility in Equation (16.8) with respect to there exists an NPG-dominant power vector. For the terminals in the set \( j \) For terminal \( \gamma \mu \) and for all \( j \in H \), dissipate a power lower than equilibrium powers. If this decrease in hurting any other terminal. A formal definition is as follows:

A power vector \( \hat{p} \) Pareto dominates another vector \( p \) if, for all \( j \in N \), \( u_j(\hat{p}) \geq u_j(p) \) and for some \( j \in N \), \( u_j(\hat{p}) > u_j(p) \). In addition, a power vector \( p^* \) is Pareto optimal (efficient) if there exists no other power vector \( p \) such that \( u_j(p) \geq u_j(p^*) \) for all \( j \in N \) and \( u_j(p) > u_j(p^*) \) for some \( j \in N \).

To show that the NPG equilibrium is inefficient recall that, at the equilibrium of the NPG, there are two types of terminal: those that achieve \( \bar{\gamma} \) and those that transmit at maximum power \( \bar{p} \) while attaining less than \( \bar{\gamma} \). Let \( H \) denote the index set of terminals that are able to reach \( \bar{\gamma} \) and \( \bar{H} \) denote index set for the rest of the terminals.

If at NPG, all \( j \in H \) reduce their powers by a factor of \( \mu \) (\( 0 < \mu \leq 1 \)), while \( j \in \bar{H} \) all keep their powers at \( \bar{p} \), the utility of user \( j \in H \) with these reduced powers is:

\[
\begin{align*}
    u_j(\mu) &= \frac{LR}{MP_j} f(\gamma_j^\mu) \\
    \gamma_j^\mu &= \frac{W}{R} \frac{\mu h_j p_j}{\sum_{k \in H, k \neq j} \mu h_k p_k + \sum_{k \in \bar{H}} h_k \bar{p} + \sigma^2}, \text{ for all } j \in H \\
    u_j(\mu) &= \frac{LR}{M \bar{p}} f(\gamma_j^\mu) \\
    \gamma_j^\mu &= \frac{W}{R} \frac{h_j \bar{p}}{\sum_{k \in \bar{H}, k \neq j} h_k \bar{p} + \sum_{k \in \bar{H}} \mu h_k p_k + \sigma^2}, \text{ for all } j \in \bar{H}
\end{align*}
\]

For the change of \( \mu \) from 1 to 0, the terminals in the set \( H \), dissipate a power lower than equilibrium powers. If this decrease in \( \mu \) results in nondecreasing utilities for all terminals, we have a proof that there exists an NPG-dominant power vector. For the terminals in the set \( H \), the first-order derivative of utility in Equation (16.8) with respect to \( \mu \) at the point \( \mu = 1 \), is:

\[
\frac{\partial u_j(\mu)}{\partial \mu} \bigg|_{\mu=1} = \frac{LR}{MP_j} \times f'(\gamma_j) \left( \frac{\sum_{k \in \bar{H}} h_k \bar{p} + \sigma^2}{\sum_{k \in \bar{H}} h_k \bar{p} + \sigma^2} - f(\gamma_j) \right)
\]

For terminal \( j \in H \), \( f'(\gamma) \gamma = f(\gamma) \) is satisfied, giving

\[
\frac{\partial u_j(\mu)}{\partial \mu} \bigg|_{\mu=1} = \frac{LR}{MP_j} f(\gamma_j) \times \left( \frac{\sum_{k \in \bar{H}} h_k \bar{p} + \sigma^2}{\sum_{k \in \bar{H}} h_k \bar{p} + \sigma^2} - 1 \right)
\]
Notice that the above expression has a negative value, i.e. \((\partial u_j(\mu)/\partial \mu)|_{\mu=1} < 0\). Therefore, as \(\mu\) tends from unity, utilities of the terminals in set \(H\) have a tendency to increase. To show that the terminals in set \(\bar{H}\) also received increased utilities as a result of scaling of powers by \(\mu\) by the users in the set, we go back to Equations (16.10) and (16.11).

From these equations we can see that, when terminals in \(H\) reduce their powers by \(\mu\), the denominator term in Equation (16.11) decreases. Since the numerator of this term remains the same (the terminals in \(\bar{H}\) do not change their equilibrium power of \(\bar{p}\)), the SIR increases for a terminal in \(\bar{H}\). With an increased SIR, the utility of terminal \(j \in \bar{H}\), given in Equation (16.10), increases since the efficiency function is a monotonic increasing function of the SIR and the denominator remains the same. Thus, we conclude that there exists a \(\mu < 1\) where utilities of all terminals increase. Since at \(\mu < 1\), the utilities of all the users increase, by definition the Nash equilibrium of the NPG is not a Pareto optimum.

### 16.1.4 Supermodular games and social optimality

Roughly speaking, social welfare in a cognitive network is defined as the sum of the utilities of individual users. In the NPG, each terminal aims to maximize its own utility by adjusting its own power, but it ignores the effects it imposes on other terminals by the interference it generates. The Pareto optimality defines the best performance that a terminal can achieve without hurting other users in the network. The remaining problem is how to motivate the terminals not to behave selfishly and try to benefit further from changing parameters, even if it could hurt other participants in the network.

In this chapter we will discuss this problem on different layers in the network including the network management plane, especially the problem of radio resource management. In this section we will address the problem within the NPG game.

The self-optimizing behavior of an individual terminal is said to create an *externality* when it degrades the quality for every other terminal in the system. Among the many ways to deal with externalities, *pricing* (or taxation) has been used as an effective tool both in economy and in communication networks. In general, pricing is motivated by two different objectives: (i) it generates revenue for the network, and (ii) it motivates terminals to use system resources more efficiently. In this section, pricing does not refer to monetary incentives, but rather refers to a control signal in order to motivate users to adopt a *social* behavior. An efficient pricing mechanism makes decentralized decisions compatible with overall system efficiency by motivating efficient sharing of resources rather than the aggressive competition of the purely noncooperative game. A pricing policy is called *incentive compatible* if pricing enforces a Nash equilibrium that improves social welfare.

Various pricing policies are used, such as flat-rate, access-based, usage-based, priority-based, etc. In communications networks, the service provider determines both the pricing policy and the specific prices for the use of resources based on the system, the kind of resources it offers, and the type of demand for these services. An efficient price will accurately reflect the costs of usage of a resource and must take into account the nature of the demand for the offered service. Usage-based pricing is an approach commonly encountered in the literature. In usage-based pricing, the price a terminal pays for using the resources is proportional to the amount of resources consumed by the user. In order to improve the equilibrium utilities of NPG in the Pareto sense, in this section we resort to usage-based pricing schemes. Through pricing, the system performance is increased by implicitly inducing cooperation and yet the noncooperative nature of the resulting power control solution is maintained.

An efficient pricing scheme should be tailored for this problem. Within the context of a resource allocation problem for a wireless system, the resource being shared is the radio environment and the resource usage is determined by terminal’s transmit power. It was shown in the previous section that the decentralized power control game has an equilibrium that is inefficient. We will see that efficiency in power control can be improved by a usage-based pricing strategy where each user pays a penalty proportional to its transmit power. With the above guidelines for a pricing strategy in mind, we now discuss a noncooperative game with pricing.
If \( G = [N, \{P_j\}, \{u^j(\cdot)\}] \) denotes an \( N \)-player noncooperative power control game with pricing (NPGP), its utilities are defined as:

\[
    u^j(p) = u_j(p) - c_j(p_j, p_{-j})
\]

(16.14)

where \( c_j : \mathbb{R} \rightarrow \mathbb{R}_+ \) is the pricing function for terminal \( j \in N \). The multi-objective optimization problem that NPGP solves is defined as:

\[
    \text{(NPGP)} \quad \max_{p_j \in P_j} u^j(p_j, p_{-j}) = u_j(p) - c_j(p_j, p_{-j}), \quad \text{for all } j \in N
\]

(16.15)

For this application we use linear pricing schemes of the form

\[
    c_j(p_j, p_{-j}) = c\alpha_j p_j
\]

(16.16)

where \( c, \{\alpha_j\} \) are positive scalars. The NPGP with linear pricing is:

\[
    \text{(NPGP)} \quad \max_{p_j \in P_j} u_j(p) - c\alpha_j p_j, \quad \text{for all } j \in N
\]

(16.17)

Nash equilibrium in the NPGP is analysed by tools based on the theory of supermodular games. In a supermodular power control game, each player’s desire to increase its power results in an increase in other players’ powers, i.e. the best response of a terminal is a monotone increase in the interferers’ strategy. A formal definition of a supermodular game can be found in [9, 10]. For the special case of single dimensional user strategy sets, which are of interest in this section, the definition simplifies to the following.

**Definition 1:** Consider a generic game \( G = [N, \{P_j\}, \{u_j(\cdot)\}] \) with strategy spaces \( P_j \subset \mathbb{R} \) for all \( j \). \( G \) is supermodular if, for each \( j \), \( u_j(p_j, p_{-j}) \) has nondecreasing differences (NDD) in \( (p_j, p_{-j}) \).

If the utility of a user has NDD in \( (p_j, p_{-j}) \), then user \( j \)’s marginal utility is nondecreasing in the transmit powers of interferers, i.e. in response to an increase in the power level of another user, a terminal increases its transmit power level in order to increase its utility. The NDD property is formally defined as follows:

**Definition 2:** \( u_j(p_j, p_{-j}) \) has NDD in \( (p_j, p_{-j}) \) if for all \( p_{-j} \geq p_{-j}' \) the quantity \( u_j(p_j, p_{-j}) - u_j(p_j, p_{-j}') \) is nondecreasing in \( p_j \). Equivalently, for continuous and twice differentiable utilities, \( u_j(p_j, p_{-j}) \) has NDD in \( (p_j, p_{-j}) \) if and only if \( (\partial^2 u_j(p) / \partial p_j \partial p_{-j}) \geq 0 \) for all \( j \neq i \).

The significance of this property is the fact that such utilities lead to a system that can reach a Nash equilibrium:

**Theorem 1:** The set of Nash equilibria \( E \) of a supermodular game is nonempty [11].

In addition, \( E \) has a largest element \( p_L \) and a smallest element \( p_S \). The equilibria \( p \in E \) are located such that \( p_S \leq p \leq p_L \); however, it does not mean that all points in that interval are equilibrium points.

If the utilities of the game under consideration are such that there is a parameter that none of the users have control over (e.g., price \( c \)), we call that parameter an *exogenous one*. Consider in general a game with an exogenous parameter, \( \epsilon, G_\epsilon = [N, \{P_j\}, \{u(\cdot)(\epsilon)\}] \), with utilities \( u_j(p_j, p_{-j}, \epsilon) \). The supermodularity definition for a generic game \( G \) given earlier (corresponding to \( \epsilon = 0 \)) can be readily extended to the game \( G_\epsilon \) with an exogenous parameter \( \epsilon \) by imposing an additional NDD condition regarding the parameter.
Definition 3: A game $G_c$ with an exogenous parameter $\varepsilon$ is said to be supermodular, or it is a parameterized game with complementarities if $u_j(p_j, p_{-j}, \varepsilon)$ has NDD in $(p_j, p_{-j})$ and in $(p_j, \varepsilon)$ for all $j$.

Theorem 2: In a parameterized supermodular game, both $p_\varepsilon(\varepsilon)$ and $p_L(\varepsilon)$ are nondecreasing in $\varepsilon$ [11].

For the pricing factor $c$, NPGP $G_c = [\mathbb{N}, \{P_j\}, \{u'_i(\cdot)\}]$ is not a supermodular game by definition. However, by a proper modification of the strategy spaces of users, we can make the resulting game supermodular. The modified strategy space for user $j$ denoted by $\hat{P}_j$ is a compact set defined by $\hat{P}_j = [p_j, \bar{p}_j]$, where the smallest power in the strategy set $\partial_p$ is derived from $\gamma_j \geq 2 \ln M$, and $\gamma = 2 \ln M$ corresponds to the point of maximum rate of change of the efficiency with increasing SIR, i.e. $(\partial^2 f(\gamma)/\partial \gamma^2) = 0$. This SIR requirement is based on the condition in Definition 2 that $(\partial^2 u_i(p)/\partial p_i \partial p_j) \geq 0$ for all $i \neq j$. This means that, in the modified strategy space of NPGP described above, the power levels that yield $\gamma_j \leq 2 \ln M$ are no longer available to the terminal. The largest power $\bar{p}_j$ is the maximum power constraint of the system. In the sequel we assume that the modified strategy space $\hat{P}$ is nonempty, i.e. there exists a $p_j$ such that $0 < p_j \leq \bar{p}_j$ for all $j$. The existence of Nash equilibria in the pricing game is established by the Theorem 3:

Theorem 3: Modified NPGP $\hat{G}_c = [\mathbb{N}, \{\hat{P}_j\}, \{u'_i(\cdot)\}]$ with exogenous parameter is a supermodular game.

To prove it, we test whether the conditions in Definition 3 are satisfied. $\hat{G}_c$ has NDD in $(p_j, p_{-j})$ since the condition given in Definition 2 yields the same expression for $\hat{G}_c$ as $G$. We need only check whether the utility $u_j(p_j, p_{-j}, c)$ has NDD in $(p_j, c)$. First, perform a change of variables from $c$ to $\varepsilon$ where $\varepsilon = -c$. When we take the partial derivative with respect to both $p_j$ and $\varepsilon$, we get $(\partial u_j/\partial p_j \partial \varepsilon) = \alpha_j \geq 0$ for all $j$. Thus, $\hat{G}_c$ is supermodular.

In the sequel, we illustrate an asynchronous algorithm that generates a sequence of powers that converges to the smallest Nash equilibrium, $p_\varepsilon(c)$. Suppose that terminal $j$ updates its power at time instances given by the set $T_j = \{t_{j1}, t_{j2}, t_{j3}, \ldots\}$ where $t_{jk} < t_{jk+1}$ and $t_{j0} = 0$ for all $j$. Define $T = \{t_1, t_2, \ldots\}$ as the set of update instances $T_1 \cup T_2 \cup \cdots \cup T_N$ sorted in increasing order. Assume that no two time instances in set $T$ are exactly the same. Let $\underline{p}$ and $\bar{p}$ be the smallest and the largest vectors in modified strategy space $\hat{P}$, respectively.

Algorithm $T$ (Terminal): Consider the noncooperative power control game with pricing (NPGP) as given in Equation (16.17). Generate a sequence of powers as follows:

(i) Set the initial power vector at time $t = 0$: $\underline{p}(0) = \underline{p}$. Also let $k = 1$.

(ii) For all $k$ such that $\tau_k \in T$

(a) For all terminals $j \in N$ such that $\tau_k \in T_j$

(b) Given $\underline{p}(\tau_k-1)$, compute $r_j (\tau_k) = \arg \max_{p_j \in \underline{p}} u_j(p_j, \underline{p}_{-j}(\tau_k-1))$.

(c) Assign the transmit power as $p_j (\tau_k) = \min(r_j (\tau_k))$.

We refer to $r_j (\tau_k)$ as the set of best transmit powers for terminal $j$ at time instance $k$ in response to the interference vector $\underline{p}_{-j}(\tau_k-1)$. The terminal $j$ optimizes the net utility over the modified strategy space of the NPGP, $\hat{P}_j$, which is bounded by $\gamma_j \leq 2 \ln M$. Implementation of this lower bound in the algorithm assumes that the instantaneous SIR at the base station is known by the terminal. The terminal then uses this information to derive the lower bound on its transmit power. In the game with pricing, more than one transmit power might constitute a best response to a given interference vector.
Table 16.1 Single-cell CDMA system parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$M$, total number of bits per frame</td>
<td>80</td>
</tr>
<tr>
<td>$L$, number of information bits per frame</td>
<td>64</td>
</tr>
<tr>
<td>$W$, spread spectrum bandwidth</td>
<td>$10^6$ Hz</td>
</tr>
<tr>
<td>$R$, bit rate</td>
<td>$10^4$ bits/second</td>
</tr>
<tr>
<td>$\sigma^2$, AWGN power at receiver</td>
<td>$5 \times 10^{-15}$ watts</td>
</tr>
<tr>
<td>Modulation technique</td>
<td>non-coherent FSK</td>
</tr>
<tr>
<td>$\bar{p}$, maximum power constraint</td>
<td>2 watts</td>
</tr>
</tbody>
</table>

Figure 16.2 Sum of equilibrium utilities in a game with nine terminals versus the pricing factor $c$. In this case, the algorithm determines the transmit power of a terminal by selecting the smallest power among all possibilities as dictated by the algorithm.

*Theorem 4a: Algorithm $T$ converges to a Nash equilibrium of NPGP. Furthermore, it is the smallest equilibrium, $p_S(c)$, in the set of Nash equilibria.*

The proof can be found in [12].

For the set of parameters listed in Table 16.1 and nine terminals that are located at $d = [310, 460, 570, 660, 740, 810, 880, 940, 1000]$ m from the base station, numerical results are shown in Figure 16.2 and 16.3. Path gains are obtained using the simple path loss model $h_j = K/d_j^4$, where $K = 0.097$ is a constant. The same set of parameters is used in [12].

The following algorithm is used to find optimum $c$.

*Algorithm $N$ (Network):*

(i) Set $c = 0$ and announce $c$ to all terminals.

(ii) Get $u_j$ for all $j \in N$ at equilibrium, increment $c := c + \Delta c$ and announce to all terminals.

(iii) If $u_j^{i+\Delta c} \leq u_j$ for all $j \in N$ then go to step 2, else stop and declare $c_{BEST} = c$. 
Figure 16.3 Utilities and powers at equilibrium of NPG and NPGP with $c_{\text{BEST}} = c$.

In the previous example we chose the value of $c = c_{\text{BEST}}$ that brings maximal Pareto improvement to the solution from NPG. The connection between a social optimum and a general pricing function is defined by the following theorems. For the formal proof see [12].

**Theorem 4b:** A power vector $p^{\ast}(\beta)$ that solves the social problem $(S_\beta)$ is Pareto optimal where $(S_\beta)$ is defined as:

$$(S_\beta) \max_p \beta \cdot u = \max_{p} \sum_{i=1}^{n} \beta_i u_i$$

with $\beta$ a vector of positive scalars.

**Theorem 5:** Let $p^{\ast}(\beta)$ solve the social problem $(S_\beta)$. $p^{\ast}$ is also a Nash equilibrium for the NPGP given in Equation (16.15) with pricing function

$$c_i (p) = -\frac{1}{\beta_i} \sum_{j=1, j \neq i}^{n} \beta_j u_j (p).$$

**16.2 POWER CONTROL GAME WITH QOS GUARANTEE**

The objective of the power control in practical wireless systems is to ensure that no mobile’s SIR $\gamma_i$ falls below its target value (threshold) $\gamma_i^{\text{tar}}$, chosen to ensure adequate QoS, i.e. to maintain:

$$\gamma_i = \gamma_i^{\text{tar}}, \quad \forall i$$

where the subscript $i$ indexes the set of mobiles. For the purpose of this section we modify Equation (16.3) as

$$\gamma_i = \frac{h_i p_i}{\sum_{j \neq i} h_j p_j c_{ij} + \eta_i}$$

where $c_{ij}$ is the square of the code correlation coefficient. The attenuation is calculated from the distance $r_i$ between the mobile and base station to be $h_i = A/r_i^\alpha$ in the absence of shadow and fast
fading. $A$ is a constant gain and $\alpha$ is usually between 3 and 6 (see Chapter 14). This can be further represented as:

$$\gamma_i = \frac{g_{ii}p_i}{I_i(p_{-i})} = \frac{g_{ii}p_i}{\sum_{j \neq i} g_{ii}p_j + \eta_i}$$ (16.22)

$I_i(p_{-i}) := \sum_{j \neq i} g_{ii}p_j + \eta_i$

$g_{ij} := \begin{cases} h_j, & \text{where } j = i \\ h_j (s_j^Ts_i)^2, & \text{otherwise} \end{cases}$

where $s_i$ is the code signature of user $i$ and $c_{ij} = (s_j^Ts_i)^2$. The power control game is based on minimization of a cost function that incorporates two conflicting requirements. On one hand, the higher the SIR, the better the service. On the other hand, higher SIR is achieved at the cost of increased energy consumption and higher interference to signals of other mobiles. So, we define a cost function for each user depending on power and SIR. Since some nonzero SIR level is necessary for accurate communication, we consider the cost of the difference between the actual SIR and the target SIR that is chosen, based on the estimated FER. To ensure positivity and convexity of the cost function, we square the SIR error term. So the cost function is defined as:

$$J_i(p_i, \gamma_i) = b_i p_i + c_i (\gamma_i^{\text{tar}} - \gamma_i)^2$$ (16.23)

Choosing $b_i/c_i > 1$ places more emphasis on power usage whereas $b_i/c_i < 1$ places more emphasis on SIR error. Applying the necessary conditions for a Nash equilibrium, we have

$$\frac{\partial J_i}{\partial p_i} = 0 = b_i - 2c_i (\gamma_i^{\text{tar}} - \gamma_i) \frac{\partial \gamma_i}{\partial p_i} = b_i - 2c_i (\gamma_i^{\text{tar}} - \gamma_i) \sum_{j \neq i} g_{ij}p_j + \eta_i.$$ (16.24)

Recalling that $I_i(p_{-i}) := \sum_{j \neq i} g_{ij}p_j + \eta_i$ and rearranging terms, yields:

$$\gamma_i = \gamma_i^{\text{tar}} - \frac{b_i I_i(p_{-i})}{2c_i g_{ii}}$$ (16.25)

So, as $b_i \to 0$ (power expenditure ceases to be important) $\gamma_i \to \gamma_i^{\text{tar}}$ and as $c_i \to 0$ (only power usage matters and SIR value of negligible importance), $\gamma_i$ no longer converges to $\gamma_i^{\text{tar}}$.

By using $\gamma_i$ from Equation (16.22) we can express the required power in terms of given and measured quantities as:

$$p_i = \frac{\gamma_i^{\text{tar}}}{g_{ii}} I_i(p_{-i}) - \frac{b_i I_i^2(p_{-i})}{2c_i g_{ii}^2}$$ (16.26)

By using the interference from Equation (16.22) in (16.25) and evaluating at the Nash equilibrium, we have:

$$\gamma_i^* = \begin{cases} \gamma_i^{\text{tar}} - \frac{b_i}{2c_i g_{ii}} \left( \frac{g_{ii}p_i^{\text{tar}}}{\gamma_i^{\text{tar}}} \right), & \text{if this quantity is nonnegative} \\ 0, & \text{otherwise} \end{cases}$$ (16.27)

This yields an expression for the Nash equilibrium power $p_i^*$ in terms of the cost weighting coefficients, the target SIR, and the Nash equilibrium SIR $\gamma_i^*$, namely:

$$p_i^* = \frac{2c_i}{b_i} \gamma_i^* (\gamma_i^{\text{tar}} - \gamma_i^*)$$ (16.28)

As expected, the Nash (noncooperative) equilibrium has SIR $\gamma_i^*$ less than $\gamma_i^{\text{tar}}$. When mobiles cooperate, and the power balancing algorithm ($b_i = 0$) is used, the target SIR, if feasible, will be reached by all mobiles. As a quadratic equation in $I_i(p_{-i})$, for given values of $g_{ii}$, $\gamma_i^{\text{tar}}$ and $p_i$, Equation (16.26)
has a real solution \( I_i(\mathbf{p}_i) \), if and only if:
\[
p_i \leq \frac{c_i (y_{i}^{\text{tar}})^2}{2b_i}.
\]  
(16.29)

So, if we wish to use the entire range of powers \( 0 \leq p_i \leq p_i^{\text{max}} \), we must choose \( b_i \) and \( c_i \) to satisfy
\[
\frac{b_i}{c_i} \leq \frac{(y_{i}^{\text{tar}})^2}{2p_i^{\text{max}}}
\]  
(16.30)

An iterative numerical algorithm for solving Equation (16.26) is given as:
\[
p_i^{(k+1)} = \begin{cases} \frac{y_i^{\text{tar}}}{\gamma_{ii}^{(k)}} - \frac{b_i}{2c_i} \left( \frac{p_i^{(k)}}{\gamma_{ii}^{(k)}} \right)^2, & \text{if positive} \\
0, & \text{otherwise} 
\end{cases}
\]  
(16.31)

where \( I_i^{(k)} \) is the measured interference experienced by the \( i \)th mobile at the \( k \)th step of the algorithm.

In order to analyze its convergence, we rewrite the power update algorithm by using \( I_i^{(k)} \) from:
\[
y_i^{(k)} = g_{ii} p_i^{(k)} / I_i^{(k)}
\]
resulting in
\[
f_i(p_i) := p_i^{(k+1)} = \begin{cases} \frac{y_i^{\text{tar}}}{\gamma_{ii}^{(k)}} - \frac{b_i}{2c_i} \left( \frac{p_i^{(k)}}{\gamma_{ii}^{(k)}} \right)^2, & \text{if defined, positive} \\
0, & \text{otherwise} 
\end{cases}
\]  
(16.32)

Yates [13] showed that, if a fixed point of the algorithm \( p_i^{(k+1)} = f(p_i^{(k)}) \) exists and if the function \( f \) satisfies three properties (positivity \( f(p) > 0 \), monotonicity \( p > p' \Rightarrow f(p) > f(p') \), and scalability \( f(\alpha p) < \alpha f(p) \forall \alpha > 1 \)), then the algorithm converges to the fixed point, which is unique.

Reference [14] shows that positivity and monotonicity of \( f \) impose constraints on acceptable values of \( I_i \), but scalability restricts the allowable receiver noise power level and generates a limit weaker than that required for monotonicity. By dropping the index \( (k) \) for simplicity from Equation (16.31), we see that, in terms of the observed interference, positivity requires
\[
I_i < \frac{2c_i g_{ii} y_{i}^{\text{tar}}}{b_i}, \forall i \in \{1, 2, \ldots, N\}
\]  
(16.33)

By using notation \( q_i := \sum_{j \neq i} g_{ij} p_j; q_i' := \sum_{j \neq i} g_{ij} p_j' \), and Equation (16.22), we have:
\[
f_i(p) - f_i(p') = \frac{y_i^{\text{tar}}}{g_{ii}} (q_i - q_i') - \frac{b_i}{2c_i g_{ii}^{2}} \bigg[ q_i^2 - (q_i')^2 + 2\eta_i (q_i - q_i') \bigg]
\]  
(16.34)

Accordingly, we need:
\[
\frac{c_i y_{i}^{\text{tar}} g_{ii}}{b_i} \geq \eta_i + \frac{1}{2} \sum_{j \neq i} g_{ij} (p_j + p_j')
\]  
(16.35)

or, noting that \( p_j \geq p_j', \forall j \Rightarrow I_i(p) \geq I_i(p') \), we see that a sufficient condition for monotonicity is:
\[
I_i(p) \leq \frac{c_i y_{i}^{\text{tar}} g_{ii}}{b_i}, \quad \forall i \in \{1, 2, \ldots, N\}
\]  
(16.36)

To discuss scalability, we start again from Equation (16.31), obtaining:
\[
\alpha f_i(p) - f_i(\alpha p) = \frac{y_i^{\text{tar}}}{g_{ii}} (\alpha - 1) \eta_i - \frac{b_i}{2c_i g_{ii}^{2}} \bigg[ (\alpha - \alpha^2) q_i^2 + (\alpha - 1) \eta_i^2 \bigg]
\]  
(16.37)
From Equation (16.22) we have:

$$\left( \sum_{j \neq i} g_{ij} p_j \right)^2 = (I_i - \eta_i)^2 = I_i^2 - 2\eta_i \left( \sum_{j \neq i} g_{ij} p_j \right) - \eta_i^2$$  \hspace{1cm} (16.38)$$

and using this in Equation (16.37) we have:

$$\alpha f_i(p) - f_i(\alpha p) = \left( \frac{\alpha - 1}{c_i g_{ii}} \right) b_i \times \left[ \frac{c_i \gamma_{i\text{tar}} \eta_i}{b_i} - \frac{\eta_i^2}{2g_{ii}} + \frac{\alpha}{2} \left( I_i^2 - 2\eta_i q_i - \eta_i^2 \right) \right]$$  \hspace{1cm} (16.39)$$

Since the first factor on the right-hand side is always positive for \( \alpha > 1 \), we need only consider the second factor for all \( \alpha > 1 \). The condition for scalability now reduces to:

$$\frac{2c_i g_{ii} \gamma_{i\text{tar}} \eta_i}{b_i} + I_i^2 \geq 2\eta_i \left( \sum_{j \neq i} g_{ij} p_j \right) + 2\eta_i^2 = 2\eta_i I_i$$  \hspace{1cm} (16.40)$$

which is equivalent to \((I_i - \eta_i)^2 + 2\frac{c_i g_{ii} \gamma_{i\text{tar}} \eta_i}{b_i} - \eta_i^2 > 0 \) or, since \( \eta_i > 1 \),

$$\eta_i < \frac{2c_i g_{ii} \gamma_{i\text{tar}}}{b_i}, \forall i \in \{1, 2, \ldots, N\}$$  \hspace{1cm} (16.41)$$

is a sufficient condition for scalability. Since the scalability condition does not restrict the allowable interference but rather the allowable noise power, and since \( \eta_i < I_i \) by definition, this sufficient condition is weaker than the earlier condition & Equation (16.33) for positivity. Also, since \( \eta_i \) is necessarily at most \( I_i \), we see that the condition in Equation (16.41) is weaker than the condition in Equation (16.36), previously derived for monotonicity. Thus, the algorithm defined by Equations (16.31) or (16.32), converges to the unique fixed point defined by Equation (16.26), if it exists, under the conditions defined by Equations (16.36) and (16.29).

The condition in Equation (16.29) can be written in an equivalent form (since SIR must be non-negative) as \( \gamma_{i\text{tar}} \geq \sqrt{2b_i p_i/c_i} \). Assuming \( b_i = c_i = 1 \), we have the very simple limiting conditions, \( I_i < g_{ii} \gamma_{i\text{tar}}, p_i < 0.5 \gamma_{i\text{tar}} \) and \( \gamma_{i\text{tar}} \geq \sqrt{2p_i} \), that must be satisfied to assure convergence to the unique Nash equilibrium.

The existence of a Nash equilibrium is established by using the implicit function theorem (see, for example [15, p. 128]). Using Equation (16.22) in (16.26), the considered system of algebraic equations is given by:

$$0 = -p_i + \frac{\gamma_{i\text{tar}}}{g_{ii}} \left( \sum_{j \neq i} g_{ij} p_j + \eta_i \right) - \frac{b_i}{2c_i g_{ii}^2} \left( \sum_{j \neq i} g_{ij} p_j + \eta_i \right)^2$$  \hspace{1cm} (16.42)$$

According to the implicit function theorem, the Jacobian matrix (the matrix of partial derivatives \( \partial F_i/\partial p_j \)) must be nonsingular at the point of existence. In the case of power balancing, the corresponding algebraic equations are represented by the first two terms on the right-hand side of the above formula, so that the corresponding Jacobian matrix has \(-1\) on the main diagonal and \( \gamma_{i\text{tar}} g_{ii} / \gamma_i \) outside the main diagonal. When this Jacobian matrix is nonsingular, then the power balancing solution exists. (It is customary in power control literature to say that the solution is feasible, that is, existence and feasibility have the same meaning.)

By referring to the above algorithm as a Nash algorithm, Figure 16.4 compares its performance with a power balancing (PB) algorithm. The traces represent the average SIR and power values over the set of mobiles considered. It is apparent that, with a slight sacrifice of achieved SIR, a significant decrease in power is achieved. However, with a large number of users, the achievable power balancing SIR and the Nash equilibrium SIR are decreased significantly. Accordingly, dropping some mobiles
whose SIR fell below the minimum acceptable SIR value would be necessary to achieve QoS targets in practice.

### 16.3 Power Control Game and Multiuser Detection

In this section, we discuss how the concept of a power control game as described in Section 16.1 works in advanced wireless CDMA networks as using multiuser detectors. A CDMA signal is based on using BPSK modulation so that Equation (16.1) gives $f_x(\gamma_k) = (1 - Q(\sqrt{2\gamma_k}))^M$, where $Q(\cdot)$ is the complementary cumulative distribution function of a standard normal random variable. As in Chapter 5, the signal received by the uplink receiver (after chip-matched filtering) sampled at the chip rate over one symbol duration can be expressed as:

$$r = \sum_{k=1}^{K} \sqrt{p_k} h_k b_k s_k + w$$

(16.43)

where $p_k$, $h_k$, $b_k$, and $s_k$ are the transmit power, channel coefficient, transmitted bit, and spreading signature sequence of the $k$th user, respectively, and $w$ is the noise vector, which is assumed to be Gaussian with mean 0 and covariance $\sigma^2 I$. We will represent the linear uplink receiver of the $k$th user by a coefficient vector, $c_k$. The output of this receiver can be written as:

$$y_k = c_k^T r = \sqrt{p_k} h_k b_k c_k^T s_k + \sum_{j \neq k} \sqrt{p_j} h_j b_j c_k^T s_j + c_k^T w.$$ (16.44)

and the SINR of the $k$th user at the output of its receiver is:

$$\gamma_k = \frac{p_k h_k^2 (c_k^T s_k)^2}{\sigma^2 c_k^T c_k + \sum_{j \neq k} p_j h_j^2 (c_j^T s_j)^2}.$$ (16.45)

Here, we discuss a noncooperative game in which each user seeks to maximize its own utility by choosing its transmit power and the receive filter coefficients. By picking a particular receiver, the power-control game reduces to maximizing Equation (16.1), which we reproduce here for convenience as:

$$\max_{p_k} = \frac{f(\gamma_k)}{p_k} \text{ for } k = 1, \ldots, K.$$ (16.46)

The relationship between the achieved SINR and transmit power depends on the particular choice of the receiver. A necessary condition for Nash equilibrium $\frac{\partial u_k}{\partial p_k} = 0$, defined by Equation (16.6) is also reproduced here, i.e.:

$$p_k \frac{\partial \gamma_k}{\partial p_k} f'(\gamma_k) - f'(\gamma_k) = 0$$ (16.47)
We now examine this condition for the three detectors. For the conventional MF, we have $c_k = s_k$ and hence

$$\gamma_k^{MF} = \frac{p_k h_k^2}{\sigma^2 + \sum_{j \neq k} p_j h_j^2 (s_k^T s_j)^2}. \quad (16.48)$$

For the decorrelator (DE), we have $C = [c_1 \cdots c_K] = S (S^T S)^{-1}$ (for $K \leq N$), where $S = [s_1 \cdots s_K]$. Hence:

$$\gamma_k^{DE} = \frac{p_k h_k^2}{\sigma^2 c_k^T c_k} \quad (16.49)$$

The filter coefficients for the MMSE receiver are given by:

$$c_k = \sqrt{p_k h_k^2 + \sum_{j \neq k} p_j h_j^2} = \frac{1}{\sigma^2} A_k^{-1} s_k$$

(16.50)

As in Section 16.1, for all three receivers, we have

$$\frac{\partial \gamma_k}{\partial p_k} = \frac{\gamma_k}{p_k}. \quad (16.51)$$

and maximizing the utility function for each user is equivalent to finding $\gamma^*$, that is the solution to

$$f(\gamma) = \gamma f'(\gamma) \quad (16.52)$$

In the asymptotic case where $K, N \to \infty$ and $K/N \to \alpha < \infty$, SINR expressions are independent of the spreading sequences of the users and are approximately given by [17]:

$$\gamma_k^{MF} = \frac{p_k h_k^2}{\sigma^2 + \frac{1}{N} \sum_{j \neq k} p_j h_j^2} \quad (16.53)$$

$$\gamma_k^{DE} = \frac{p_k h_k^2 (1 - \alpha)}{\sigma^2} \quad \text{for } \alpha < 1 \quad (16.54)$$

$$\gamma_k^{MMSE} = \frac{p_k h_k^2}{\sigma^2 + \frac{1}{N} \sum_{j \neq k} (p_j h_j^2, p_k h_k^2, \gamma_k^{MMSE})} \quad (16.55)$$

where $I(a, b, c) = \frac{ab}{b+ac}$. It is clear that both $\gamma_k^{MF}$ and $\gamma_k^{DE}$ satisfy $\partial \gamma_k / \partial p_k = \gamma_k / p_k$. It can also be verified that any $\gamma_k$ which satisfies $\partial \gamma_k / \partial p_k = \gamma_k / p_k$ is a solution to Equation (16.55). As a result, finding the solution to power control in large systems is equivalent to finding the solution of $f(\gamma) = \gamma f'(\gamma)$, independent of the type of the receiver used. The minimum power solution for achieving $\gamma^*$ by all users is given by:

$$p_k^{MF} = \frac{1}{h_k^2} \frac{\gamma^* \sigma^2}{1 - \alpha \gamma^*} \quad \text{for } \alpha < \frac{1}{\gamma^*} \quad (16.56)$$

$$p_k^{DE} = \frac{1}{h_k^2} \frac{\gamma^* \sigma^2}{1 - \alpha} \quad \text{for } \alpha < 1 \quad (16.56)$$

$$p_k^{MMSE} = \frac{1}{h_k^2} \frac{\gamma^* \sigma^2}{1 - \alpha \gamma^*} \quad \text{for } \alpha < 1 + \frac{1}{\gamma^*} \quad (16.56)$$

Combining (16.56) with $u_k = \frac{L}{M} R \frac{f(\gamma^*)}{p_k}$, we obtain:

$$u_k = \frac{LR f(\gamma^*) h_k^2}{M \gamma^* \sigma^2} \quad \Gamma \quad (16.57)$$
where $\Gamma$ is given as

\[ \Gamma^{\text{MF}} = 1 - \alpha \gamma^* \quad \text{for} \quad \alpha < \frac{1}{\gamma^*} \]

\[ \Gamma^{\text{DE}} = 1 - \alpha \quad \text{for} \quad \alpha < 1 \]

\[ \Gamma^{\text{MMSE}} = 1 - \alpha \frac{\gamma^*}{1 + \gamma^*} \quad \text{for} \quad \alpha < 1 + \frac{1}{\gamma^*} \]  

(16.58)

It can be seen that $u_k^{\text{MMSE}} \geq u_k^{\text{DE}}$ and $u_k^{\text{MMSE}} \geq u_k^{\text{MF}}$, which confirms our expectation that the MMSE receiver achieves the maximum utility among all linear receivers. The utility achieved by the DE is higher than that of the MF except when $\gamma^* < 1 (=0 \text{ dB})$.

For the social optimum defined by Equation (16.8), we consider the case of equal output SINRs among all users (i.e., SINR balancing). This ensures fairness among users in terms of throughput and delay. We also assume that $\beta_1 = \cdots = \beta_k = 1$, which means that we are interested in maximizing the sum of users’ utilities. Therefore, the maximization in Equation (16.8) can be written as:

\[ \max_{p_1, \ldots, p_K} f(\gamma) \sum_{k=1}^{K} \frac{1}{p_k} \]  

(16.59)

Equal output SINRs among users is achieved with minimum power consumption when the received powers are the same for all users, i.e. $p_1 h_1^2 = p_2 h_2^2 = \cdots = p_K h_K^2 = q$, where:

\[ q^{\text{MF}} (\gamma) = \frac{\gamma \sigma^2}{1 - \alpha \gamma} \quad \text{for} \quad \alpha < \frac{1}{\gamma} \]

\[ q^{\text{DE}} (\gamma) = \frac{\gamma \sigma^2}{1 - \alpha} \quad \text{for} \quad \alpha < 1 \]

\[ q^{\text{MMSE}} (\gamma) = \frac{\gamma \sigma^2}{1 - \alpha \frac{\gamma}{1 + \gamma}} \quad \text{for} \quad \alpha < 1 + \frac{1}{\gamma} \]  

(16.60)

So, the maximization in (16.59) can equivalently be expressed as:

\[ \max_{\gamma} \frac{f(\gamma)}{q(\gamma)} \sum_{k=1}^{K} h_k^2 \]  

(16.61)

The solution to (16.61) must satisfy $\frac{\partial}{\partial \gamma} \left( \frac{f(\gamma)}{q(\gamma)} \right) = 0$. Using this fact in Equation (16.60) gives us the equations that must be satisfied by the solution to the maximization problem in (16.61) for the three linear receivers

MF: $f(\gamma) = \gamma (1 - \alpha \gamma) f'(\gamma)$

DE: $f(\gamma) = \gamma f'(\gamma)$

MMSE: $f(\gamma) = \gamma \left[ 1 - \frac{\alpha \gamma}{(1 + \gamma^2 - \alpha \gamma^2)} \right] f'(\gamma)$  

(16.62)

### 16.4 Power Control Game in MIMO Systems

For the system where each mobile user terminal has one transmit antenna, and there are $m$ receive antennas at the uplink receiver (BS), the received signal (after chip-matched filtering and chip-rate sampling) can be represented as an $N \times m$ matrix $R$, where the $l$th column represents the $N$ chips received at the $l$th antenna, i.e.

\[ R = \sum_{k=1}^{K} \sqrt{p_k} b_k s_k^T + W \]  

(16.63)
where $p_k$, $b_k$, and $s_k$ are the transmit power, transmitted bit, and spreading sequence of the $k$th user, respectively. $h_k = [h_{k1}, \ldots, h_{km}]^T$ represents the gain vector in which $h_{k1}, \ldots, h_{km}$ are the channel gains from the transmitter of the $k$th user to the $m$ receive antennas, and are assumed to be i.i.d.

In Equation (16.63), $W$ is the noise matrix. The noise is assumed Gaussian and both spatially and temporally white. Let $C_k$ be the $N \times m$ coefficient matrix for the spatial–temporal filter of the $k$th user at the base station that performs linear spatial and temporal processing on the received signal. The output of this receiver can be written as:

$$y_k = \text{tr} \left( C_k^T R \right)$$  \hspace{1cm} (16.64) 

where $\text{tr}(A)$ is the trace of $A$. Now, the resulting noncooperative game, in which users in the network are allowed to choose their uplink linear spatial–temporal receivers, as well as their transmit powers, can be expressed as the following maximization problem:

$$\max_{p_k, C_k} u_k(p_k, C_k) \text{ for } k = 1, \ldots, K$$  \hspace{1cm} (16.65) 

This maximization can equivalently be expressed as:

$$\max_{p_k, \tilde{c}_k} u_k(p_k, \tilde{c}_k) \text{ for } k = 1, \ldots, K$$  \hspace{1cm} (16.66) 

where $\tilde{c}_k$ is a vector with $mN$ elements. It is obtained by stacking the columns of $C_k$ on top of each other. To see this, notice that $y_k$ in Equation (16.64) can be alternatively written as $y_k = \tilde{c}_k^T \tilde{r}$ where $\tilde{r}$ is a vector obtained by placing columns of $R$ on top of each other, i.e.

$$\tilde{r} = \sum_{k=1}^{K} \sqrt{p_k b_k} \tilde{s}_k + \tilde{w}$$  \hspace{1cm} (16.67) 

where $\tilde{s}_k = [h_{k1}s_{k1}^T \ldots h_{km}s_{km}^T]^T$ is the effective signature and $\tilde{w}$ is the noise vector consisting of columns of $W$ stacked on top of each other.

By using the analogy with the previous derivation in this section we have as a main results:

- The MMSE receiver, whose coefficients are given by

$$\tilde{c}_k = \frac{\sqrt{p_k}}{1 + p_k (\tilde{s}_k^T \tilde{A}_k^{-1} \tilde{s}_k)} \tilde{A}_k^{-1} \tilde{s}_k$$  \hspace{1cm} (16.68) 

achieves the maximum utility among all linear receivers.

where, $\tilde{A}_k = \sum_{j \neq k} p_j \tilde{s}_j \tilde{s}_j^T + \sigma^2 I$.

- Given the MMSE receiver coefficients, maximizing the utility function for each user is again equivalent to finding the solution $\gamma^*$ to $f(\gamma) = \gamma f'(\gamma)$.

- Nash equilibrium is reached when all user terminals use the MMSE detector for their uplink receivers, and transmit at a power level that results in a SINR equal to $\gamma^*$ (SINR-balancing). This equilibrium is unique.

To compare the three different types of receivers we assume that:

(i) The MF has perfect knowledge of the channel gains of the desired user, but know only the statistics of the fading levels of the interferers. It basically performs dispersing at each receive antenna, and then applies maximal ratio combining (MRC).

(ii) The decorrelating detector is assumed to have perfect knowledge of the channel gains for the desired user, but no knowledge about the interferers (except for their spreading sequences). It applies a DE at each receive antenna, and then performs MRC.

(iii) The MMSE detector is assumed to have perfect knowledge of the channel gains of all users. The filter coefficients for the MMSE receiver are given by Equation (16.68).
We can show that for all three receivers, we have again:

\[
\frac{\partial \gamma_k}{\partial p_k} = \frac{\gamma_k}{p_k}.
\] (16.69)

Therefore, maximizing the utility function for each user is again equivalent to finding \( \gamma^* \) that is the solution to \( f'(\gamma) = \gamma f''(\gamma) \).

Using a large-system analysis similar to the one presented in the previous section, the achieved utility for user can be expressed as:

\[
 u_k = \frac{LRf(\gamma^*) \bar{h}_k^2}{M \gamma^* \sigma^2} \Gamma
\] (16.70)

where \( \Gamma \) for different receivers is given as:

\[
\Gamma_{MF} = 1 - \bar{\alpha} \gamma^* \text{ for } \bar{\alpha} < \frac{1}{\gamma^*}
\]

\[
\Gamma_{DE} = 1 - \alpha \text{ for } \alpha < 1
\]

\[
\Gamma_{MMSE} = 1 - \bar{\alpha} \frac{\gamma^*}{1 + \gamma^*} \text{ for } \bar{\alpha} < 1
\] plus

\[
\Gamma_{DE} = 1 - \alpha
\]

(16.71)

for \( \bar{\alpha} = \alpha/m \) and \( \bar{h}_k^2 = \sum_{l=1}^{m} h_{kl}^2 \). One can see that for the case of the MF and the MMSE detector, using more antennas at the receiver provides both power pooling (through \( \bar{h}_k \)) and interference reduction (through \( \bar{\alpha} \)). This means that the system behaves like a single-antenna system with processing gain \( mN \) and received power equal to the sum of the received powers at the individual antennas. The DE, on the other hand, benefits only from power pooling, and there is no pooling of the degrees of freedom (DOFs). This is because the decorrelating detector has no knowledge about the channel gains for the interferers. Therefore, each interferer effectively occupies \( m \) DOFs.

We now pose admission control as a maximization problem in which the load in the network (i.e., \( \alpha \)) is chosen such that the total utility in the network (per DOF) is maximized:

\[
\alpha^* = \arg \max \alpha \frac{1}{N} \sum_{k=1}^{K} u_k
\] (16.72)

Given (16.57), as \( K, N \to \infty \) we can use the law of large numbers to write:

\[
\alpha^* = \arg \max \alpha \frac{LRf(\gamma^*) \Gamma}{M \gamma^* \sigma^2} E\{h^2\}
\] (16.73)

which is equivalent to:

\[
\alpha^* = \arg \max \alpha \Gamma.
\] (16.74)

To find \( \alpha^* \), we set \( \partial(\partial \alpha)/\partial \alpha \Gamma = 0 \) and solve for given (16.58), we have

\[
\alpha^*_{MF} = \frac{1}{2\gamma^*}; \quad \alpha^*_{DE} = \frac{1}{2}; \quad \alpha^*_{MMSE} = \frac{1}{2} + \frac{1}{2\gamma^*}.
\] (16.75)

16.5 GAME THEORY BASED MAC FOR AD HOC NETWORKS

Designing a MAC protocol can be modelled as a bandwidth allocation problem at the link layer. When considering link layer flows, contention relations between links in a wireless ad-hoc network can be represented by a link conflict graph. In such a graph, vertices represent link flows and edges between vertices denote contention between links, which is the situation where there is interference between either the sender or the receiver of one link and either the sender or the receiver of the other link. A fully connected subgraph in a conflict graph is referred to as a clique, and a maximal clique is a clique not contained in any larger clique. Therefore, a maximal clique represents a ‘channel resource’, which has a given fixed capacity. The basic requirement for feasibility of a schedule or bandwidth assignment is that the total flowrate in each clique does not exceed the clique’s capacity, subject to the conflict
constraints. In addition, the bandwidth allocation should satisfy some performance requirement such as fairness. Assuming all nodes of the ad-hoc network use omnidirectional antennas with the same power level to transmit packets in the same shared wireless channel, a link conflict graph can be used to describe the contention relations between link flows, and each maximal clique is treated as a 'channel resource' with a given fixed capacity. The capacity of a clique depends on the topology of the network, and the fairness principle under consideration.

Given conflict graph $G$, let the flow rate of a specific link $i$ be $x_i$. Define $f_i(x_i)$ to be a strictly concave utility function for link $i$ when its flow rate is $x_i$, let $Q(i)$ be the set of cliques that include link $i$, and $S(j)$ be the set of links that form clique $j$. The fair bandwidth allocation problem can be formulated as a constrained maximization problem $p$ as follows:

$$
p : \max_{x_i} \sum_i w_i f_i(x_i), \quad x_i \geq 0, \; i = 1, \ldots, N$$

subject to:

$$\sum_{i \in S(j)} x_i \leq c_j, \quad j = 1, \ldots, M$$

(16.76)

where $c_j$ denotes the capacity of clique $j$, $N$ is the number of links, and $M$ is the number of maximal cliques in the link conflict graph. Although the objective function is separable in $x_i$, solving this problem requires coordination of, possibly, all links in the network, which is not practical in ad-hoc networks.

By considering the dual of problem $p$, a distributed algorithm based on the cooperative game framework (CGF) can be derived. As discussed in Section 16.2, the CGF algorithm is a price based algorithm. That is, in clique $j$, when the total flow rate is $\sum_{i \in S(j)} x_i$, the price rate is $\lambda_j$. From the viewpoint of link $i$, for a fixed value of $\lambda_j$, the optimal flow rate can be computed by solving the following problem:

$$\max_{x_i} \left( w_i f_i(x_i) - x_i \sum_{j \in Q(i)} \lambda_j \right)$$

(16.77)

Since $f_i(x_i)$ is strictly concave, the unique maximizer for problem exists and is given by:

$$x_i^* = f_i^{-1} \left( \frac{\sum_{j \in Q(i)} \lambda_j}{w_i} \right)$$

(16.78)

The CGF algorithm can be summarized as:

**CGF based MAC algorithm**

The algorithm is executed by each link $i$ (the sender node in a link) round by round:

(i) initially, choose a feasible flow rate $x_i(0)$;

(ii) collect local conflict information and construct local set of cliques $Q(i)$;

(iii) set initial price $\lambda_j(0)$ (global parameter) for each clique $j$ in $Q(i)$;

(iv) in round $k$, calculate a new link flowrate $x_i(k)$ according to (3);

(v) disseminate (cooperation) the new flow rate information to all links in one hop;

(vi) in round $k + 1$, calculate a new price $\lambda_j(k + 1)$ for clique $j$ as

$$\lambda_j(k + 1) = \max \left( 0, \lambda_j(k) + \gamma \left( \sum_{i \in S(j)} x_i^* - c_j \right) \right)$$

(16.79)

where $\gamma$ is a step size, $c_j$ is the capacity of clique $j$, $\sum_{i \in S(j)} x_i^*$ is the total flow rate in clique $j$ in the previous round $k$;

(vii) if the local conflict graph has changed (e.g., due to mobility), go back to (ii), otherwise go back to step (iv).
Intuitively, Equation (16.79) implies the basic requirement of supply and demand: if the total offered flowrate is less (or more) than the capacity of the clique, the price decreases (or increases respectively).

Define $\bar{Q} = \max_{i \in \mathcal{Q}} |Q(i)|$ as the largest number of cliques that contain the same link. Denote $\bar{S} = \max_{j \in \mathcal{S}} |S(j)|$ as the maximal size of a clique. Let $\bar{\alpha}$ be the upper bound of function $-f''(x)$, then the range of the step size can be defined as in [18]:

$$0 < \gamma < \frac{2}{\bar{\alpha} \bar{Q} \bar{S}}$$  \hspace{1cm} (16.80)

The CGF algorithm depends on the availability of local link contention relations and their flowrate information. Therefore, collecting such information in a dynamic environment is the key issue to the design and implementation of the algorithm.

### 16.6 TIT-FOR-TAT (TFT) GAME THEORY BASED PACKET FORWARDING STRATEGIES IN AD HOC NETWORKS

#### 16.6.1 Strategy models

In self-organizing ad-hoc networks, all the networking functions rely on the cooperation between the participants. As a basic example, nodes have to forward packets for each other in order to enable multihop communication. So, the operation of ad-hoc networks relies on the contribution of nodes. In general, there are two approaches to motivate nodes to cooperate: (i) by denying service to misbehaving nodes by means of a reputation mechanism, or (ii) by remunerating (awarding) honest nodes, using for example a micropayment scheme.

Most of the studies that consider cooperation of entities use the iterated prisoner’s dilemma (IPD) game as their underlying model [20–22]. The simplicity of the IPD makes it an attractive model. The continuous valued prisoner’s dilemma (CPD) game was studied by Wahl and Nowak [23]. In such a game, the nodes can choose a degree of cooperation between full cooperation and full defection. In [20], Axelrod identifies tit-for-tat (TFT) as a robust strategy that performs surprisingly well (in terms of maximizing the player’s payoff) in the prisoner’s dilemma games. TFT begins with cooperation in the first round and then repeats the previous move of the other player. We will see that cooperation based on TFT exists also in the ad-hoc networking context. The above games will be elaborated in detail later in this chapter for different applications. The classical prisoner’s dilemma game is not appropriate for modelling packet forwarding because it involves only two players having symmetric roles. Hence, in this section, we define a multiplayer, asymmetric game that is inspired by the classical prisoner’s dilemma game, which better suits our purposes [19].

Let us consider an ad-hoc network with set of all nodes $N$ of size $n$. Two nodes are said to be neighbors if they reside within the power range of each other. The neighbor relationship between the nodes is presented with an undirected connectivity graph. Each vertex of the connectivity graph corresponds to a node in the network, and two vertices are connected with an edge if the corresponding neighbors are neighbors.

Communication between two nonneighboring nodes is based on multihop relaying. This means that packets from the source to the destination are forwarded by intermediate nodes. For a given source and destination, the intermediate nodes are those that form the shortest path between the source and the destination in the connectivity graph. Such a chain of nodes (including the source and the destination) is referred to as a route, and the operation of the network will be referred to as a forwarding game.

In each time slot $t$, each node $i$ chooses a cooperation level $p_i(t) \in [0, 1]$, where 0 and 1 represent full defection and full cooperation, respectively. Defection means that the node does not forward traffic for the benefit of other nodes, whereas cooperation means that it does. Thus, $p_i(t)$ represents the fraction of the traffic routed through $i$ in $t$ that $i$ actually forwards. In this model $i$ has a same cooperation level $p_i(t)$ for every route in which it is involved as a forwarder. Having a different cooperation level for different routes would require identifying the source–destination pairs that would increase the computation at the nodes significantly.
We will use Equation (16.82) later as an input of the strategy function of an intermediate node.

The total payoff \( \pi_i \) of node \( i \) in time slot \( t \) is then computed as:

\[
\pi_i(t) = \sum_{q \in S_i(t)} \xi_i(q, t) + \sum_{r \in F_i(t)} \eta_i(r, t)
\]

where \( S_i(t) \) is the set of routes in \( t \) where \( i \) is the source, and \( F_i(t) \) is the set of routes in \( t \) where \( i \) is an intermediate node.

In every time slot, each node \( i \) updates its cooperation level using a strategy function \( \sigma_i \) by choosing its cooperation level \( p_i(t) \) in time slot \( t \) based on the normalized throughput it experienced in time slot \( t-1 \) on the routes where it was a source:

\[
p_i(t) = \sigma_i([\hat{\tau}(r, t-1)]_{r \in S_i(t-1)})
\]

where \([\hat{\tau}(r, t-1)]_{r \in S_i(t-1)}\) represents the normalized throughput vector for node \( i \) in time slot \( t-1 \), each element of which is the normalized throughput experienced by \( i \) on a route where it was source in \( t-1 \). The strategy of a node \( i \) is then defined by its strategy function \( \sigma_i \) and its initial cooperation level \( p_i(0) \). There is a number of possible strategies. Some options are illustrated below [19]:

(a) Always defect (AllD): a node playing this strategy defects in the first time slot, and then uses the strategy function \( \sigma_i(in) = 0 \).

(b) Always cooperate (AllC): a node playing this strategy starts with cooperation, and then uses the strategy function \( \sigma_i(in) = 1 \).
(c) Tit-for-tat (TFT): a node playing this strategy starts with cooperation, and then mimics the behavior of its opponent in the previous time slot. The strategy function that corresponds to the TFT strategy is $\sigma_i(in) = in$.

(d) Suspicious tit-for-tat (S-TFT): a node playing this strategy defects in the first time slot, and then applies the strategy function $\sigma_i(in) = in$.

(e) Anti tit-for-tat (anti-TFT): a node playing this strategy does exactly the opposite of what its opponent does. In other words, after cooperating in the first time slot, it applies the strategy function $\sigma_i(in) = 1 - in$.

If the output of the strategy function is independent of its input, then the strategy is called a nonreactive strategy (e.g., AllD or AllC) otherwise the strategy is reactive (e.g., TFT or anti-TFT). The model requires that each source be able to observe the throughput in a given time slot on each of its routes. We assume that this is made possible with high enough precision by using some higher level control protocol above the network layer.

### 16.6.2 Network nodes dependency graph and system metamodel

The payoff received by the source on $r$ depends on the cooperation levels of the intermediate nodes on $r$. We represent this dependency relationship between the nodes with a directed graph, which will be referred to as the *dependency graph*. Each vertex of the dependency graph corresponds to a network node. There is a directed edge from vertex $i$ to vertex $j$, denoted by the ordered pair $(i, j)$, if there exists a route where $i$ is an intermediate node and $j$ is the source. Intuitively, an edge $(i, j)$ means that the behavior (cooperation level) of $i$ has an effect on $j$. The concept of dependency graph is illustrated in Figure 16.5.

Reference [19] uses an automation $\Theta$ to represent the forwarding game by the metamodel. The mapping is built on the dependency graph. A mapper (machine) $M_i$ is assigned to every vertex $i$ of the dependency graph, and the edges of the dependency graph are interpreted as links that connect the mappers assigned to the vertices. Each mapper $M_i$, thus, has some input and some (possibly 0) output links. The internal structure of the mapper is illustrated in Figure 16.6. Each mapper $M_i$ consists of a ‘multiplication gate’ $\Pi$ followed by a gate that implements the strategy function $\sigma_i$ of node $i$. The multiplication gate $\Pi$ takes the values on the input links and passes their product to the strategy function gate. Finally, the output of the strategy function gate is passed to each output link of $M_i$. The assignment $\Theta$ works in discrete steps. As an example, Figure 16.7 shows the assignment that corresponds to the dependency graph of Figure 16.5.

Initially, in step 0, each mapper $M_i$ outputs some initial value $x_i(0)$. Then, in step $t > 0$, each mapper computes its output $x_i(t)$ by taking the values that appear on its input links in step $t - 1$. In order to study the interaction of node $i$ with the rest of the network, we extract the gate that implements the strategy function $\sigma_i$ from the assignment $\Theta$. What remains in $\Theta$ without $\sigma_i$ we denote by $\Theta_{-i}$ as illustrated in Figure 16.8.

![Figure 16.5 Representation of a network: (a) a graph showing five routes, and (b) the corresponding dependency graph.](image-url)
Figure 16.6 Internal structure of mapper $M_i$.

Figure 16.7 The assignment for the dependency graph of Figure 16.5.

Figure 16.8 Model of interaction between node $i$ and the rest of the network represented by the automaton $\Theta_{-i}$. 
The input of $\Theta_{\rightarrow i}$ is the sequence $\bar{x}_i = x_i(0), x_i(1), \ldots$ of the cooperation levels of $i$, and its output is the sequence $\bar{y}_i = y_i(0), y_i(1), \ldots$ of the normalized throughput values for $i$.

By using the system of equations that describe the operation of $\Theta$, one can easily express any element $y_i(t)$ of sequence $\bar{y}_i$ as some function of the preceding elements $x_i(t-1), x_i(t-2), \ldots, x_i(0)$ of sequence $\bar{x}_i$, and the initial values $x_j(0)$ ($j \neq i$) of the mappers within $\Theta_{\rightarrow j}$. For the automaton in Figure 16.7, and extracting, for instance, $\sigma_A$, we can determine the first few i/o formulae of $\Theta_{\rightarrow A}$ as follows:

$$
y_A(0) = x_C(0) \cdot x_E(0)
$$

$$
y_A(1) = \sigma_C(x_E(0)) \cdot \sigma_E(x_A(0))
$$

$$
y_A(2) = \sigma_C(\sigma_E(x_A(0))) \cdot \sigma_E(x_A(1))
$$

$$
y_A(3) = \sigma_C(\sigma_E(x_A(1))) \cdot \sigma_E(x_A(2))
$$

A dependency loop $L$ of node $i$ is a sequence $(i, v_1), (v_1, v_2), \ldots, (v_{l-1}, v_l), (v_l, i)$ of edges in the dependency graph. The length of a dependency loop $L$, denoted by $|L|$, is defined as the number of edges in $L$. If node $i$ has no dependency loops, then the cooperation level chosen by $i$ in a given time slot has no effect on the normalized throughput experienced by $i$ in future time slots. In the example, nodes $B$ and $D$ have no dependency loops. If $L$ is a dependency loop of $i$ and all other nodes $j \neq i$ in $L$ play reactive strategies, then $L$ is said to be a reactive dependency loop of $i$. If, on the other hand, there exists at least one node $j \neq i$ in $L$ that plays a nonreactive strategy, then $L$ is called a nonreactive dependency loop of $i$.

### 16.6.3 The payoff of iterative game

The goal of the nodes is to maximize the payoff that they accumulate over time. However, the end of the game is unpredictable, so the standard technique used in the theory of iterative games [20] can be applied. The finite forwarding game with an unpredictable end is modelled as an infinite game where future payoffs are discounted. The cumulative payoff $\bar{\pi}_i$ of a node $i$ is computed as the weighted sum of the payoffs $\pi_i(t)$ that $i$ obtains in each time slot $t$:

$$
\bar{\pi}_i = \sum_{t=0}^{\infty} [\pi_i(t) \cdot \omega^t],
$$

where $0 < \omega < 1$ and, hence, the weights exponentially decrease with $t$. The discounting factor $\omega$ represents the degree to which the payoff of each time slot is discounted relative to the previous time slot.

For any route $r \in F_i$, let the set of intermediate nodes on $r$ upstream from node $i$ (including node $i$) be $\Phi(r, i)$. In addition let $\Phi(r)$ be the set of all forwarder nodes on route $r$, $\text{scr}(r)$ be the source of route $r$ and the set of nodes that are forwarders on at least one route be $\Phi$ (i.e., $\Phi = \{i \in N : F_i \neq \emptyset\}$).

If a node $i$ is in $\Phi$ and it has no dependency loops, then its best strategy is AllD (i.e., to choose cooperation level 0 in every time slot).

This results from the fact that $i$ wants to maximize its cumulative payoff $\bar{\pi}_i$ defined in Equation (16.87) where

$$
\pi_i(t) = x_i(0, t) + \sum_{r \in F_i} \eta_i(r, t) = u_i(T_i \cdot y_i(t)) - \sum_{r \in \text{scr}(r)} T_{\text{scr}(r)} \cdot c \cdot \prod_{k \in \Phi(r, i)} x_k(t)
$$

(16.88)

Given that $i$ has no dependency loops, $y_i(t)$ is independent of all the previous cooperation levels $x_i(t')$ ($t' < t$) of node $i$. Thus, $\bar{\pi}_i$ is maximized if $x_i(t') = 0$ for all $t' \geq 0$.

If a node $i$ is in $\Phi$ and it has only nonreactive dependency loops, its normalized throughput $\bar{y}_i$ is again independent of its own behavior $\bar{x}_i$, and its best strategy is full defection (AllD). If every node $j (j \neq i)$ plays AllD, then the best response of $i$ to this is AllD. Hence, every node playing AllD is a
Nash equilibrium. Except for the above special conditions, we cannot determine the best strategy of a node \( i \) in general because it depends on the particular dependency graph and the strategies played by the other nodes. Still, we can show that, under certain conditions, cooperative equilibria do exist in the network.

Let us consider a route \( r \in F_i \) and assume that node \( i \) is in \( \Phi_i \), and that there exists a dependency loop \( L \) of \( i \) that contains the edge \((i, \text{src}(r))\). If all nodes in \( L \) (other than \( i \)) play the TFT strategy, then the following holds:

\[ y_i (t + \delta) \leq \prod_{k \in \Phi(r,i)} x_k(t), \]  

(16.89)

where \( \delta = |L| - 1 \). To prove it let \( L \) be the following sequence of edges in the dependency graph: \((v_0, v_1), (v_1, v_2), \ldots, (v_\delta, v_{\delta+1})\), where \( v_{\delta+1} = v_0 = i \) and \( v_1 = \text{src}(r) \). An example is given in Figure 16.9.

We know that each node is the source of a single route; let us denote by \( r_{v_j}(0 < j \leq \delta + 1) \) the route on which \( v_j \) is the source. It follows that \( r_{v_1} = r \). In addition, we know that the existence of edge \((v_j, v_{j+1})(0 \leq j \leq \delta)\) in the dependency graph means that \( v_j \) is a forwarder on \( r_{v_{j+1}} \). Because \( x_k(t) \leq 1 \), for every node \( v_j(0 \leq j \leq \delta) \), then

\[ x_{v_j}(t) \geq \prod_{k \in \Phi(r_{v_j+1}, v_j)} x_k(t) \geq \prod_{k \in \Phi(r_{v_j+1})} x_k(t) = y_{v_{j+1}}(t) \]  

(16.90)

In addition, since every node except for \( v_0 = v_{\delta+1} = i \) plays TFT, we have \( x_{v_j}(t + 1) = y_{v_j}(t) \) for every \( 0 < j \leq \delta \); Using this and (16.90) in an alternating order, we get the following:

\[ x_{v_0}(t) \geq \prod_{k \in \Phi(r_{v_1}, v_0)} x_k(t) \geq y_{v_1}(t + 1) \geq y_{v_2}(t + 1) \]

\[ = x_{v_2}(t + 2) \geq \cdots \geq y_{v_{\delta+1}}(t + \delta) \]

(16.91)

By substituting \( i \) for \( v_0 \) and \( v_{\delta+1}, \) and \( r \) for \( r_{v_1}, \) we get

\[ x_i(t) \geq \prod_{k \in \Phi(r,i)} x_k(t) \geq \cdots \geq y_i(t + \delta) \]

(16.92)

Intuitively, this means that if node \( i \) does not cooperate, then this defection ‘propagates back’ to it on the dependency loop. The delay of this effect is given by the length of the dependency loop.

Figure 16.9 Example to illustrate the propagation of behavior.
Assuming that node $i$ is in $\Phi$, the best strategy for $i$, is full cooperation in each time slot, if the following set of conditions holds:

(i) For every $r \in F_i$, there exists a dependency loop $L_{i,scr(r)}$ that contains the edge $(i, scr(r))$.

(ii) For every $r \in F_i$,

$$\frac{u_i'(T_i) \cdot T_i \cdot \omega_{i,scr(r)}}{|F_i|} > T_{scr(r)} \cdot c,$$

where $u_i'(T_i)$ is the value of the derivative of $u_i(\tau)$ at $\tau = T_i$, and $\delta_{i,scr(r)} = |L_{i,scr(r)}| - 1$.

(iii) Every node in $\Phi$ (other than $i$) plays the TFT strategy.

To prove this, we first find the maximum possible value of the total payoff for node $i$ in general. Then we show that the maximum is in the case in which node $i$ fully cooperates. In the sequel we will use the linear function $f(\tau) = u_i'(T_i) \cdot \tau + u_i(T_i) - u_i'(T_i) \cdot T_i$ which is the tangent of function $u_i$ at $\tau = T_i$. Since $u_i$ is nondecreasing and concave, $f(\tau) \geq u_i(\tau)$ for all $\tau$ and $f(T_i) = u_i(T_i)$.

By definition, the total payoff $\bar{\pi}_i$ of node $i$ is:

$$\bar{\pi}_i = \sum_{t=0}^{\infty} \left[ \sum_{r \in F_i} \xi_i(r, t) + \sum_{r \in F_i} \eta_i(r, t) \right] \omega^t = \sum_{t=0}^{\infty} \left[ \sum_{r \in F_i} u_i(T_i \cdot y_i(t)) - \sum_{r \in F_i} T_{scr(r)} \cdot c \cdot \prod_{k \in \Phi(r,i)} x_k(t) \right] \omega^t$$

Because of Condition 1 and Condition 3, we can use (16.98) to obtain the following inequality for every $r \in F_i$:

$$\prod_{k \in \Phi(r,i)} x_k(t) \geq y_i(t + \delta_{i,scr(r)})$$

which leads to

$$\bar{\pi}_i \leq \sum_{t=0}^{\infty} \left[ \sum_{r \in F_i} u_i(T_i \cdot y_i(t)) - \sum_{r \in F_i} T_{scr(r)} \cdot c \cdot y_i(t + \delta_{i,scr(r)}) \right] \omega^t$$

Since $u_i(T_i \cdot y_i(t))$, is independent of $r$, we use:

$$u_i(T_i \cdot y_i(t)) = \sum_{r \in F_i} \frac{u_i(T_i \cdot y_i(t))}{|F_i|}$$

When we use this in into (16.96), we get:

$$\bar{\pi}_i \leq \sum_{t=0}^{\infty} \left[ \sum_{r \in F_i} \frac{u_i(T_i \cdot y_i(t))}{|F_i|} - \sum_{r \in F_i} T_{scr(r)} \cdot c \cdot y_i(t + \delta_{i,scr(r)}) \right] \omega^t$$

$$= \sum_{r \in F_i} \left[ \sum_{t=0}^{\infty} \frac{u_i(T_i \cdot y_i(t))}{|F_i|} - \sum_{t=0}^{\infty} T_{scr(r)} \cdot c \cdot y_i(t + \delta_{i,scr(r)}) \cdot \omega^t \right]$$

(16.98)

We now split up the first summation into two terms, one from $t = 0$ to $\delta_{i,scr(r)} - 1$, and the other from $t = \delta_{i,scr(r)}$ to $\infty$. Then, we shift the index in the second sum in such a way that the summation goes again from $t = 0$ to $\infty$ which results in:

$$\sum_{t=0}^{\delta_{i,scr(r)}-1} \frac{u_i(T_i \cdot y_i(t))}{|F_i|} \cdot \omega^t = \sum_{t=0}^{\delta_{i,scr(r)}-1} \frac{u_i(T_i \cdot y_i(t))}{|F_i|} \cdot \omega^t + \sum_{t=\delta_{i,scr(r)}}^{\infty} \frac{u_i(T_i \cdot y_i(t))}{|F_i|} \cdot \omega^{t+\delta_{i,scr(r)}}$$

(16.99)
By using Equation (16.99) back in Equation (16.98), we get:

\[
\bar{\pi}_i \leq \sum_{r \in F_j} \left[ \sum_{t=0}^{\delta_{t,scr(r)}-1} \frac{u_i(T_t \cdot y_i(t))}{|F_i|} \cdot \omega^t + \sum_{t=0}^{\infty} \frac{u_i(T_t \cdot y_i(t + \delta_{t,scr(r)}))}{|F_i|} \cdot \omega^{t+\delta_{t,scr(r)}} \right] \cdot \omega^j - T_{scr(r)} \cdot c \cdot y_i(t + \delta_{t,scr(r)}) \]  

(16.100)

Since \( u_i \) is nondecreasing and \( y_i(t) \leq 1 \), the first term of (16.100) satisfies

\[
\sum_{t=0}^{\delta_{t,scr(r)}-1} \frac{u_i(T_t \cdot y_i(t))}{|F_i|} \cdot \omega^t \leq \sum_{t=0}^{\delta_{t,scr(r)}-1} \frac{u_i(T_t)}{|F_i|} \cdot \omega^t = \frac{u_i(T)}{|F_i|} \cdot \frac{1 - \omega^{\delta_{t,scr(r)}}}{1 - \omega} \]  

(16.101)

By using the fact that \( f(\tau) \geq u_i(\tau) \) for all \( \tau \), the second term of (16.100) satisfies:

\[
\sum_{t=0}^{\infty} \left[ \frac{u_i(T_t \cdot y_i(t + \delta_{t,scr(r)}))}{|F_i|} \cdot \omega^{t+\delta_{t,scr(r)}} - T_{scr(r)} \cdot c \cdot y_i(t + \delta_{t,scr(r)}) \right] \cdot \omega^j 
\]

(16.102)

\[
= \sum_{t=0}^{\infty} \left[ \frac{u_i(T_t) - u'_i(T_t)}{|F_i|} \cdot T_{scr(r)} \cdot c \cdot y_i(t + \delta_{t,scr(r)}) \right] \cdot \omega^j 
\]

(16.103)

\[
\leq \frac{u_i(T) - u'_i(T)}{|F_i|} \cdot \omega^{\delta_{t,scr(r)}} \cdot \omega^j + \sum_{t=0}^{\infty} \left( \frac{u_i(T_t)}{|F_i|} \cdot \omega^{\delta_{t,scr(r)}} - T_{scr(r)} \cdot c \right) \cdot \omega^j 
\]

(16.104)

where, in coming from (16.102) to (16.103), we used condition (ii) and the fact that \( y_i(t + \delta_{t,scr(r)}) \leq 1 \). By using (16.101) and (16.104) in (16.100), we get

\[
\bar{\pi}_i \leq \frac{1}{1 - \omega} \sum_{r \in F_i} \left( \frac{u_i(T)}{|F_i|} - T_{scr(r)} \cdot c \right) = \frac{1}{1 - \omega} \left( \frac{u_i(T)}{|F_i|} - c \cdot \sum_{r \in F_i} T_{scr(r)} \right) 
\]

(16.105)
Now, if \( i \) fully cooperates in every time slot, every node will always fully cooperate, since all the other nodes play TFT, and hence, every node will experience a normalized throughput equal to 1 in each time slot. This can easily be derived from the i/o formulae describing the behavior of the nodes, which take a simple form due to the simplicity of the strategy function of the TFT. As a consequence, we have \( y_i (t) = 1 \) for every \( t \), and \( x_k (t) = 1 \) for every \( k \) and for every \( t \). In this case, Equation (16.104) becomes:

\[
\bar{\pi}_i = \sum_{l=0}^{\infty} \left[ u_i (T_i \cdot y_i (t)) - \sum_{r \in F_i} T_{scr(r)} \cdot c \cdot \prod_{k \in \Phi(r,i)} x_k (t) \right] \omega^l \\
= \frac{1}{1 - \omega} \left( u_i (T_i) - c \cdot \sum_{r \in F_i} T_{scr(r)} \right)
\]

(16.106)

This means that by fully cooperating, the payoff of node \( i \) reaches the upper bound expressed in Equation (16.105), which means that there is no better strategy for node \( i \) than full cooperation.

### 16.7 TFT GAME THEORY BASED MODELING OF NODE COOPERATION WITH ENERGY CONSTRAINT

#### 16.7.1 Acceptance rate

In this section we consider a network with a finite population of \( N \) nodes distributed among \( K \) classes (e.g., laptop, PDA, cell phone, . . .). Let \( n_i \) be the number of nodes in class \( i (i = 1, \ldots, K) \) associated with an energy constraint \( E_i \), and an expectation of lifetime \( L_i \). So, nodes in class \( i \) have an average power constraint of \( \rho_i = E_i / L_i \), where \( \rho_1 > \rho_2 > \ldots > \rho_K \). In each slot, any one of the \( N \) nodes can be chosen as a source with equal probability. \( M \) is the maximum number of relays that the source can use to reach its destination. The probability that the source requires \( l \) relays is given by \( q (l) \) with \( q (0) = 0 \), for simplicity. The \( l \) relays are chosen with equal probability from the remaining \( N - 1 \) nodes. A session is said to belong to type \( j \), if at least one of the nodes involved belongs to class \( j \) and the class of any other node is less than or equal to \( j \). We assume that energy spent in transmit mode is the dominant source of energy consumption. For simplicity, we assume that the energy required to relay a session is constant and equal to 1. For any node \( h \), we denote by \( B^j_h (k) \) the number of relay requests made by node \( h \) for a session of type \( j \) till time \( k \), and by \( A^j_h (k) \) the number of relay requests generated by node \( h \) for a session of type \( j \) which has been accepted till time \( k \). Similarly, we denote by \( D^j_h (k) \) the number of relay requests made to node \( h \) for a session of type \( j \) till time \( k \), and by \( C^j_h (k) \) the number of relay requests made to node \( h \) for a session of type \( j \) which has been accepted by node \( h \) till time \( k \).

For \( 1 \leq j \leq K \) and \( 1 \leq h \leq N \), we define: \( \phi^j_h (k) = A^j_h (k) / B^j_h (k) \), and \( \psi^j_h (k) = C^j_h (k) / D^j_h (k) \). So, \( \phi^j_h \) is the throughput experienced by \( h \) with respect to type \( j \) sessions. More specifically, we define normalized acceptance rate \( R \) or throughput \( T \) as \( R = T = \lim_{\tau \to \infty} \phi^j_h (k) \). Note that \( T \) is defined for each node and session type; however, we have suppressed the indices for the sake of simplicity.

#### 16.7.2 Pareto optimum

We assume that a node in class \( i \) in a session of type \( j \) accepts a relay request with probability \( \tau_{ij} \). Consider a node \( p \) participating in a type \( j \) session (\( 1 \leq j \leq K \)). The average energy per slot spent by the node as a source, \( e_{pj}^{(s)} \), or as a relay \( e_{pj}^{(r)} \) can be written as:

\[
e_{pj}^{(s)} = \frac{R}{N} = \frac{1}{N} \sum_{l=1}^{M} \sum_{h_1,\ldots,h_j} q (l) \Gamma (l; h_1,\ldots,h_j) \tau_{ij}^{h_1} \ldots \tau_{jj}^{h_j} e_{pj}^{(r)} \\
= \frac{1}{N} \sum_{l=1}^{M} lq (l) \sum_{h_1,\ldots,h_j} \Gamma (l-1; h_1,\ldots,h_j) \tau_{ij}^{h_1} \ldots \tau_{jj}^{h_j} \tau_{class(p)ij} \quad (16.107)
\]
where:

- $1/N$ the probability that node $p$ is the source;
- $\Gamma(l; h_1, \ldots, h_K)$ is a multivariate probability function conditioned on the fact that the session belongs to type $j$ with $l$ relays and $h_i$ refers to the number of relays of class $i$ participating in the session;
- $\tau_{ij}^{h_1} \cdots \tau_{ij}^{h_j}$ represent the probability that all the relay nodes accept the request;
- $l/N$ is the probability that node $p$ is chosen as one of the $l$ relays.

The feasible region for $\tau_{ij}$ is defined by the set of inequalities:

$$\sum_{j=1}^{K} (e_{pj} + e_{pj}'') \leq \rho_{\text{class}(p)} \quad 1 \leq p \leq N$$
$$\tau_{\text{class}(p)j} \in [0, 1] \quad 1 \leq j \leq K; \quad 1 \leq p \leq N \quad (16.108)$$

where $\text{class}(p)$ is the class to which node $p$ belongs. For a feasible set of $\tau_{ij}$, the corresponding feasible set of $R$s can be directly computed from Equation (16.107). The Pareto optimal values of the $\tau_{ij}$ can be derived by imposing the equality relation in Equation (16.108).

**Example 1:** For a simple example of the system with two nodes, belonging to the same class and with a power constraint $\rho$ and $q(1) = 1$, $M = 1$ Equation (16.108) gives for each node $p = 1, 2$

$$\frac{\tau_1}{2} + \frac{\tau_2}{2} \leq \rho \rightarrow \tau_1 \leq \tau_2 - 2\rho \quad (16.109)$$

which defines the feasible region for $\tau_{ij}$. Given that $\tau_i \rightarrow R_i$ (16.109) gives also the feasibility region for $R$s as $R_1 \leq R_2 - 2\rho$. The Pareto optimal values of the $R$s are on the line $(R_1, R_2) \rightarrow R_1 = R_2 - 2\rho$ or more precisely on the line segment joining $(0, 2\rho)$ with $(2\rho, 0)$.

While operating at any of these points, both nodes are consuming energy at the maximum allowable rate. Therefore a node cannot increase its $R$ without decreasing the other node’s $R$. So, in this example, it is straightforward to see that the only Pareto optimal operating point acceptable to both rational users is $(\rho, \rho)$.

In a system with $N$ nodes, all belonging to the same class, by rationality, each node must possess the same value of $R$. So, it is a simple matter to derive the maximal value of $\tau$ which satisfies the energy constraint as in Equation (16.108). In a system with $n_1 = 1$ nodes in class 1 and $n_2$ nodes in class 2, by rationality, the lone node in class 1 will not expend more energy than the remaining nodes in class 2. This is because the node in class 1 will not receive higher throughput if it is more generous to users in class 2 than users in class 2 are to it. Indeed, self interest dictates that the lone node behaves as though it belongs to class 2. If there are two nodes in class 1 and they are involved in type 2 sessions, they have no incentive to behave any differently than as if they were class 2 nodes. Meanwhile, when they are involved in type 1 sessions, they can utilize their excess energy to their mutual benefit. So, for a set of self-interested nodes, the rational values of $\tau_{ij}$ have the following property:

$$\tau_{ij} = \tau_{jj} \rightarrow \tau_j \quad 1 \leq i \leq j \leq K \quad (16.110)$$

Given Equation (16.110), the rational Pareto optimal values of the $\tau_{ij}$ and hence the $R$s can be determined by recursively solving the energy constraints in Equation (16.108) and by using Equation (16.107).

**Example 2:** Consider a network with $N$ nodes and $K$ classes with $n_i$ nodes in class $i$, and $q(1) = 1$, $M = 1$. In this case, the session type is the maximum of the source class and the relay class. Consider a node in class $i$. The average energy per slot spent by the node as a source is:

$$e_i^{(s)} = \frac{1}{N(N - 1)} \left[ \sum_{k=1}^{i-1} n_k \tau_i + (n_i - 1) \tau_i + \sum_{j=i+1}^{K} n_i \tau_j \right] \quad (16.111)$$
When the relay belongs to a class lower than \( i \), the session is of type \( i \), and if the relay belongs to a class higher than \( i \), the session type is the same as the class of the relay. The same expression holds for the average energy per slot, \( e_i^{(r)} \), spent by the node as a relay. The rational Pareto optimal \( \tau_i \) can be derived from the set of equations:

\[
\begin{align*}
e_i^{(s)} + e_i^{(r)} &= \rho_i, \quad 1 \leq i \leq K \\
\tau_i &\in [0, 1], \quad 1 \leq i \leq K
\end{align*}
\]

In particular, for \( K = 1 \), the rational and Pareto optimal \( \tau \) is equal to \( N \rho / 2 \), and the rational Pareto optimal \( R \) is equal to \( \tau \).

**Example 3:** Consider now a network with two classes and \( M = 2 \). The energy spent by a node in class 2 as a source, \( e_2^{(s)} \), and as a relay, \( e_2^{(r)} \), are given by:

\[
e_2^{(s)} = \frac{1}{N} \sum_{l=1}^{M} q(l) \tau_2^{(l)} \quad \text{and} \quad e_2^{(r)} = \frac{1}{N} \sum_{l=1}^{M} l q(l) \tau_2^{(l)}
\]

The optimal \( \tau_2 \) can be found by solving the equation \( e_2^{(s)} + e_2^{(r)} = \rho_2 \).

The energy spent by a node in class 1 as a source, \( e_1^{(s)} \), and as a relay, \( e_1^{(r)} \), is given by:

\[
e_1^{(s)} = \frac{1}{N} \left[ q(1) \left\{ \frac{n_2}{N-1} \tau_2 + \left(1 - \frac{n_2}{N-1}\right) \tau_1 \right\} + q(2) \left\{ \frac{(n_1 - 1)(n_1 - 2)}{(N-1)(N-2)} \tau_1^2 \right. \right. \\
+ \left. \left. \left(1 - \frac{(n_1 - 1)(n_1 - 2)}{(N-1)(N-2)} \right) \tau_2^2 \right\} \right]
\]

\[
e_1^{(r)} = \frac{1}{N} \left[ q(1) \left\{ \frac{n_2}{N-1} \tau_2 + \left(1 - \frac{n_2}{N-1}\right) \tau_1 \right\} + 2q(2) \left\{ \frac{(n_1 - 1)(n_1 - 2)}{(N-1)(N-2)} \tau_1^2 \right. \right. \\
+ \left. \left. \left(1 - \frac{(n_1 - 1)(n_1 - 2)}{(N-1)(N-2)} \right) \tau_2^2 \right\} \right].
\]

Once we find \( \tau_2 \) from Equation (16.113), we can obtain \( \tau_1 \) by solving \( e_1^{(s)} + e_1^{(r)} = \rho_1 \). The method presented in these examples can be easily extended to multiple classes and relays.

### 16.7.3 Prisoner’s dilemma and TFT game

A rational selfish node could exploit the naivete of other nodes by always denying their relay requests, thereby increasing its lifetime, while keeping its \( R \) constant. In other words, any stationary strategy is dominated by ‘always deny’ behavior. Hence, stationary strategies are not sustainable, and behavioral strategies, already discussed in this chapter, are required in order to stimulate cooperation.

The problem falls into the framework of non-cooperative game theory and at this point we use the occasion to introduce the canonical example from this field, the so-called prisoner’s dilemma. In this example, two people are accused of a crime. The prosecution promises that if exactly one confesses, the confessor goes free, while the other goes to prison for 10 years. If both confess, then they both go to prison for 5 years. If neither confesses, they both go to prison for just a year. Table 16.2 presents the punishment matrix showing the years of prison that the players get depending on the decision they make.

<table>
<thead>
<tr>
<th></th>
<th>( P_2 ) ( ) confess</th>
<th>( P_2 ) ( ) ) deny</th>
</tr>
</thead>
<tbody>
<tr>
<td>( P_1 ) ( ) ) confess</td>
<td>5</td>
<td>10</td>
</tr>
<tr>
<td>( P_1 ) ( ) ) deny</td>
<td>10</td>
<td>5</td>
</tr>
</tbody>
</table>

Clearly, the mutually beneficial strategy would be for both not to confess. Because the prisoners do not know the confession of the other party, from the perspective of the first prisoner, \( P_1 \), his punishment is minimized if he confesses, irrespective of what the other prisoner, \( P_2 \), does. Since the other prisoner argues similarly, the unique Nash equilibrium is the confess strategy for both prisoners. Nevertheless, if this game were played repeatedly (iterated prisoners dilemma), it has been shown that
Table 16.2 Punishment matrix (P1,P2) for the prisoner’s dilemma.

<table>
<thead>
<tr>
<th></th>
<th>Confess</th>
<th>Not Confess</th>
</tr>
</thead>
<tbody>
<tr>
<td>Confess</td>
<td>(5,5)</td>
<td>(0,10)</td>
</tr>
<tr>
<td>Not Confess</td>
<td>(10,0)</td>
<td>(1,1)</td>
</tr>
</tbody>
</table>

Cooperative behavior can emerge as a Nash equilibrium. By employing behavioral strategies, a user can base his decision on the outcomes of previous games. This allows the emergence of cooperative equilibrium. As discussed in Section 16.6, in generous tit-for-tat (GTFT) strategy, each player mimics the action of the other player in the previous game. Each player, however, is slightly generous (see Equation (16.92) and on occasion cooperates by not confessing even if the other player had confessed in the previous game. In this section we use the GTFT algorithm where each node maintains a record of its past experience by using the two variables $\psi_h^{(j)}$ and $\phi_h^{(j)}$, $h = 1, \ldots, N$, $j = 1, \ldots, K$, defined in Section 16.7.1. Each node therefore maintains only information per session type and does not maintain individual records of its experience with every node in the network. The decisions are always taken by the relay nodes based only on their $\psi_h^{(j)}$ and $\phi_h^{(j)}$ values. For a network with $N$ nodes, $K$ classes, $q(1) = 1$ and $M = 1$ assume that a generic node $h$ receives a relay request for a type $j$ session.

The GTFT acceptance algorithm, is defined as:

- if $\psi_h^{(j)}(k) > \tau_j$ or $\phi_h^{(j)}(k) < \psi_h^{(j)}(k) - \varepsilon$ Reject \hspace{1cm} (16.115)
- else Accept

where $\varepsilon$ is a small positive number.

In (16.115) the first term refers to the case when node $h$ has relayed more traffic for type $j$ sessions than it should have, and the second term to the case when the amount of traffic relayed by node $h$ in sessions of type $j$ is greater than the amount of traffic relayed for node $h$ by others in type $j$ sessions. Since $\varepsilon$ is positive, nodes are a little generous by agreeing to relay traffic for others even if they have not received a reciprocal amount of help. As we can see the GTFT algorithm is not a stationary strategy. As opposed to the discussion in Section 16.6, each node takes its action based solely on locally gathered information. Only $4K$ variables need to be stored at each node, independently of $N$, and this makes GTFT scalable. While for the single relay case, GTFT attempts to equalize the amount of cooperation a node provides with the amount of cooperation it receives, when multiple relays are used the amount of help rendered is always more than the amount of help received. This is because a node is a relay more often than it is a source. In this case, assuming that a relay request for a type $j$ session arrives at node $h$ belonging to class $i$ the modified acceptance algorithm, called m-GTFT becomes:

- if $\psi_h^{(j)}(k) > \tau_j$ or $\phi_h^{(j)}(k) < L_{ij}\psi_h^{(j)}(k) - \varepsilon$ Reject, \hspace{1cm} (16.116)
- else Accept

where $L_{ij}$ is the ratio of the rational Pareto optimal $R$ for type $j$ session to $\tau_j$, defined as

$$L_{ij} = \frac{\text{Prob} (h \text{ is server in a type } j \text{ session})}{\text{Prob} (h \text{ accepts to relay a type } j \text{ session})}$$ \hspace{1cm} (16.117)

Reference [24] proves that the GTFT algorithm constitutes a Nash Equilibrium and show that similar arguments can be extended to prove the convergence of the m-GTFT algorithm too.

Evolution in time, of normalized acceptance rate $R$ associated with five different session types, for a single relay network is presented in Figure 16.10(a). The network has five classes, and five nodes in each class ($N = 25$). The energy constraints are given by $\rho_1 = 0.03$, $\rho_2 = 0.025$, $\rho_3 = 0.02$,
Figure 16.10 Normalized acceptance rate $R$ versus time for a network with $N = 25$, $K = 5$, $q(1) = 1$, $M = 1$, (a) all nodes employ GTFT. NAR values converge to the optimal operating point. (b) all nodes employ GTFT, and $\epsilon < 0$ ($-0.01$). If nodes are not slightly generous ($\epsilon > 0$), GTFT fails to reach the optimal operating point. (c) one node in class 2 and one node in class 4 are parasites while all other nodes employ GTFT. Performance of nodes in type 2 and type 4 sessions degrade showing that GTFT prevents parasitic behavior in rational users.
\( \rho_4 = 0.015 \) and \( \rho_5 = 0.01 \) and \( q(1) = 1 \) and \( M = 1 \), as in [24]. All \( R_\ell \)s converge to the to the rational optimal values. These values can be obtained numerically by solving the system of equations (16.111) in Example 2. Figure 16.10(b), demonstrates that it is critically important that the parameter \( \varepsilon \) be positive. In other words, nodes should always be slightly generous for the NARs in order to achieve the rational optimal values. Figure 16.10(c) demonstrates the robustness of the GTFT algorithm in the presence of parasites. We assume that a node in class 2 and a node in class 4 are parasitic, i.e., these nodes never relay traffic. We see that the performance of type 2 and type 4 sessions degrade severely, while performance for other types of session remains unaffected.

### 16.8 PACKET FORWARDING MODEL BASED ON DYNAMIC BAYESIAN GAMES

In this section we present a packet forwarding mechanism that takes into account past actions and the time-varying nature of the available resources in the network. Let \( N \) be an arbitrary ad \( \text{hoc} \) network and \( N \subset N \) a finite set of nodes (agents) belonging to \( N \). An arbitrary node of the set \( N = \{1, \ldots, n\} \) is indexed by \( i \). We assume that nodes have topology information only about the nodes within the range of their transmitter (local topology), but not about the nodes outside this region. The nodes that are within the range of the transmitter constitute the neighborhood (\( \Gamma_i \)) of a node \( i \). We also assume that the neighborhood topologies are symmetric, i.e., \( j \in \Gamma_i \iff i \in \Gamma_j \).

The nodes are energy constrained and energy aware as they know their current energy level. We approximate the energy level with a finite set of possible values. We call these energy levels the energy classes of a node and use the variable \( \theta_i(t) \) to denote the energy class of node \( i \) at an arbitrary time \( t \). The energy class of a node will be also called the type of a player (node), which is borrowed from the terminology of microeconomics and algorithmic mechanism design. We focus on an individual node \( i \) and define for each \( i \) at time period \( t_k, k = 0, 1, \ldots \), so that a new period starts when the node generates some packets and decides whether to send them to the network or to discard them. Let the number of packets generated by node \( i \) at time period \( t_k \) be \( g_i(t_k) \), and the number of the generated packets that are actually sent to the network \( s_i(t_k) \). At each time period \( s_i(t_k) \leq g_i(t_k) \). Let the action history of a sending node \( i \) at time period \( t_k \) be a vector that contains the number of packets sent at time periods \( t_0, \ldots, t_{k-1} \), denoted as \( h_i(t_k) = (s_i(t_0), \ldots, s_i(t_{k-1})) \). We assume that each message sent by an arbitrary node \( i \) is broadcast to all nodes \( j \) in the neighborhood \( \Gamma_i \). Every node \( j \) decides individually whether to forward the packets or not. With respect to the sending node, the decision and the corresponding forwarding action by a node \( j \) take place at time period \( t_k \) and we denote by \( f^j_i(t_k) \) the number of packets that node \( j \) forwards for node \( i \) at time period \( t_k \). The sender’s decision as to how many packets to send is based on its guess about the energy classes of the neighboring nodes. If it believes that all neighboring nodes have used all of their energy or that they are noncooperative, it is not rational to send anything, as sending consumes energy. The energy classes of the neighboring nodes are not known \( a \text{ priori} \), but instead we assume that node \( i \) has a probability distribution defined over the possible values of the energy class of a node \( j \). The time varying probabilities of the energy classes at time period \( t_k \) depend on the joint history profile of the actions made by node \( i \) and node \( j \). The history profile is \( hp(i,j) \rightarrow h^j_i(t_k) = (h_i(t_0), h^j_i(t_k)) \) where \( h^j_i(t_k) \) is the number of packets that node \( j \) has forwarded for node \( i \) at time periods \( t_0, \ldots, t_{k-1} \), \( h^j_i(t_k) = (f^j_i(t_0), \ldots, f^j_i(t_{k-1})) \). The guesses \( gs(i) \) a sending node \( i \) has about the energy class of a forwarding node \( j \) can be represented as a probability distribution that is conditioned on the energy class of \( i \) and \( hp(i,j) \). The formal definition of the conditional probability distribution is \( gs(i) : p_i(\theta^j_i(t_k) | h^j_i(t_k)) \), where \( \theta^j_i(t_k) \) is the energy class of a sending node \( i \) at time period \( t_k \) and \( p(\cdot) \) is an arbitrary probability distribution. By defining a conditional probability density in this way we construct a guessing mechanism for the node \( i \). The guesses reflect the level of knowledge a node has at the beginning of a time period \( t_k \). These guesses play an important role in the definition of optimality of the model in the Bayesian sense.

Similarly, as to the decisions made by the sender depend on the sender’s guesses about the energy classes of the neighboring nodes, the decisions of the forwarding nodes depend on the guesses that
the nodes have made about the energy class of the sender. There is no motivation to forward the packet of the low energy node because it might not be able to return the favour in the future. So, the model is constructed in such a way that the probabilities depend on the number of packets sent by node $i$. The definition of the probability distribution of a forwarding node $j$ is given as

$$gs(j) \rightarrow \phi_j^i(t_k) = p(\theta_j^i, \theta_j^i, s_j^i)$$

Together the guesses of the sender and the forwarder constitute the guessing system of the nodes. We use $\mu_i^j$ to denote the joint guessing system of nodes $i$ and $j$: $gs(i, j) \rightarrow \mu_i^j(t_k) = (u_i^j, t_k^i)$. Each outcome of the game yields some utility for both of the players. The value of the utility depends on the decisions they make and on their energy classes. The exact form of the utility function depends on the application. However, it is required that the utility functions are continuous and concave in the parameters. A good utility function should consider simultaneously both the possible savings in energy consumption and the possible gain in future throughput.

Since each node must act both as a forwarder and as a sender in the network, its operation can be formulated in game-theoretic settings by defining each sender–forwarder pair as a dynamic Bayesian game [9]. Thus each node $i$ participates in $2|\Gamma_i|$ games, where $|\Gamma_i|$ is the cardinality (number of elements) in set $\Gamma_i$. We define the game-theoretic system of a node $i$ to be the collection of games in which the node participates simultaneously. The sending games of the system are $|\Gamma_i|$ games in which the node $i$ acts as a sender.

A sending game is a 5-tuple $(I, \Phi, \Theta, \Phi_i^j, \Theta_i^j)$, where $I$ is the set of two players $(i, j)$, $\Phi$ defines the action space of the game, $\Theta$ defines a utility function for both players, $\Theta_i^j$ defines a utility function for the sender, and $\Theta_j^i$ defines a utility function for the forwarder. The utility function $\Theta_i^j$ is a vector containing the utility functions for both players: $\Theta_i^j = (u_i, u_j)$ where $u_i$ is the utility function of the sender and $u_j$ is the utility function of the forwarder for the messages arriving from sender node $i$. The type space $\Theta$ is the set of possible energy class values for a node and the guess system $\mu_i^j$ was defined earlier. Similarly, we can also define a forwarding game as a game in which node $i$ acts as a forwarder for some neighboring node $j$.

In these games, agents must act optimally at each individual time period and their actions also need to be optimal, given the history of game play. A new period of a game begins when the sender decides how many packets to send to the network. The period ends when the forwarding side decides whether to forward the packets or not. If the packets are forwarded, the transmission can be detected by the sender. On the other hand, if the packets are not forwarded, this can also be observed using timers and packet numbering.

The action profiles of the players are behavior strategies, defined as probabilities of the form $p_x(a, h (t_k), \theta_i^j)$, where $p_x(\cdot)$ is a suitable probability distribution, $x$ is a sending node, $j$ is a forwarding node, and $\theta_i^j$ is either the node $i$ or the node $j$. The variable $a_i$ denotes the action of $x$ and the variable $h (t_k)$ is the history profile $hp(i, j)$. If $x$ is a sender, the action $a_i$ corresponds to the number of packets sent at period $t_k$ given the history of game play and if $x$ is a forwarder, $a_i$ refers to the number of packets forwarded for the sender at period $t_k$ given the history of game play and the action of the sender. The utility of $x$ is a function of the history, the actions and the energy classes of the nodes. The utility function of a node $x$, given sender $i$ and receiver $j$, is defined as

$$u_x = u(h_i^j(t_{k+1}), a_i, a_j, \theta_i^j, \theta_j^i)$$

For the each new period the sender’s guess about the energy class of the forwarder must be updated. The update is made using the Bayes rule, and the resulting posterior probabilities of period $t_k$ are used as prior probabilities at the beginning of stage $t_k+1$.

$$u_j^i(t_{k+1}) = p(\theta_j^i(t_k) \mid h_j^i(t_k), f_j^i(t_k))$$

$$= \frac{p(h_j^i(t_k), f_j^i(t_k) \mid \theta_j^i(t_k)) \cdot p(\theta_j^i(t_k))}{p(h_j^i(t_k), f_j^i(t_k))}$$

(16.118)

The optimality analysis of the model, is simplified if subgame perfection and, especially, its extension to perfect Bayesian equilibrium in the model is applied. A game is said to be subgame perfect, if the
restriction of strategies to a single stage (time period) constitutes a Nashequilibrium. In PBE, each stage game played at a single period must constitute a Bayes–Nash equilibrium. In other words, when the actions are restricted to a single time period they must be optimal given the guesses the players have at the beginning of that time period.

In a unicast communication model the optimal behavior policy of the sender at time period \( t_k \) is

\[
\hat{s_i} = \arg \max_{s_i} \sum_{j} \sum_{\theta_j} \sigma_j^i(s_j|\theta_j) \mu_i \theta_j,
\]

where \( \mu_i \) are \( \theta_i \) probabilities and the term \( \sigma_j^i \) is a behavior strategy \( bs(j) \) of player \( j \) in the game where node \( i \) acts as the sender. The behavior strategy indicates the probability that node \( j \) performs the action \( f_j^i \) given the action of the sender \( i \). Finally, the term \( u_i \) is the utility function.

In the broadcast model it is rational to send packets if some node in \( \Gamma_i \) is willing to forward them. So, the optimal sender strategy in the broadcast model is defined as the maximum of the optimal strategies of individual ‘unicast’ games \( \hat{s_i} = \max_j \hat{s_j}^i \). Optimal behavior policy of the forwarder at time period \( t_k \) is

\[
\hat{f_j} = \arg \max_{f} \sum_{\theta} \sigma_j^i(s_j|\theta) \varphi_i^j u_j^i
\]

Here \( \sigma_j^i(s_j|\theta) \) is the \( bs(i) \) of the sender, which gives the probability that node \( i \) sends \( s_j \) packets (at time period \( t_k \)) given its energy class \( \theta_j \). Together with the guess system \( \bar{\mu}_j^i \), the pair \( (\hat{s}, \hat{f}_j) \) constitutes the Bayes–Nash equilibrium of a stage-game.

In a practical system a node joins an ad-hoc network by discovering its neighbors and by assigning prior probabilities over the energy class values of those neighbors. To discover its neighbors, a node \( i \) could send a \( \text{join} \) message to the network. The nodes that respond form the neighborhood \( \Gamma_i \). Additionally, the node \( i \) should construct a prior probability distribution over the energy class values of the neighbors.

A possible approach [25] could be to assign uniform priors over the possible values. Another possibility is that responses to the \( \text{join} \) message include information about the energy class of the sender. If the message is trusted, a suitable family of distributions is used so that the prior probabilities can be assigned easily and efficiently. In [25] the use, of a beta-distribution is considered. If \( r \) is the percentage representing the energy level of a node, a new beta-distribution can be initialized by assigning initial parameter values so that \( \alpha = 100r \) and \( \beta = 100(1-r) \). If the message is not trusted a weighing probability distribution could be used to ‘filter’ the value. A weighing probability distribution is defined as a joint distribution \( p(\theta_j^i, \theta_j) \), where the value of \( \theta_j \) belongs to the joining node and is known. The initial probability of a particular energy class can be assigned by marginalizing the joint distribution with respect to the values of \( \theta_j \) as

\[
p(\theta_j^i) = \int p(\theta_j^i, \theta_j) d \theta_j = \int p(\theta_j^i|\theta_j) p(\theta_j) d \theta_j.
\]

### 16.9 GAME THEORETIC MODELS FOR ROUTING IN WIRELESS SENSOR NETWORKS

#### 16.9.1 Cognitive wireless sensor network model

In this case, in addition to energy constraints and path length, the utility function should include the reliability of a part and, due to data aggregation, the information value of the message [26–30]. The game-theoretic model for reliable energy-constrained routing consists of \( N \) sensors (players) denoted by the set \( S = \{s_1, \ldots, s_j, \ldots, s_n\} \). In the sequel we describe the attributes of the network using a static model. A dynamic extension of the model, as described in the previous section, would view them in terms of snapshots representing successive operational periods.

The cost includes communication costs \( c_{ij} \), proportional to the distance between sensors and participation costs \( CP \), that model the cost to a sensor of deciding to participate in a given route. Each node’s strategy is a binary vector \( l_i = (l_{i1}, l_{i2}, \ldots, l_{i-1}, l_{i+1}, \ldots, l_{in}) \), where \( l_{ij} = 1 \) \((l_{ij} = 0) \)
represents sensor \( s_i \) ’s choice of sending/not sending a data packet to sensor \( s_j \). The path reliability is modelled by using sensor failure probabilities. We assume that node \( s_i \) can fail with a probability \((1 - p_i) \in [0, 1) \). In wireless sensor networks a query is sent from the sink node \( s_q = s_n \) to the nodes
The query may match the attributes of data stored at \( s_i \) each to varying degrees. This data has to be reported back to \( s_q \) and possibly aggregated along the way. Information is routed to \( s_q \) through an optimally chosen set (via the routing game) \( S' \subseteq S \) of intermediate nodes. The model should include, in the selection process of data transfer paths; the *importance (information value)* of the data being reported. The data items representing successful query matches must be routed over more reliable paths even at higher costs, as the penalty for nondelivery is higher. The model includes this feature by attaching a value \( v_i \in \mathbb{R} \) to the data from each sensor \( s_i \), \( 1 \leq i < n \).

Since we are modelling cognitive sensors, we should account for nodes with valuable information but selfish behavior, which are saving energy by deliberately not participating in the routing. One way to motivate such nodes is via a punishment mechanism that values future information coming from a node proportional to the number of previous routes it has participated in.

The benefit \( X_i \) to any sensor \( s_i \) is defined as
\[
X_i = g_i(v_1, \ldots, v_{n-1}) R_i,
\]
where \( R_i \) denotes the path reliability from \( s_i \) onwards to \( s_q \) and \( g_i(\cdot) \) is the value expectation function. As an example, for the data-aggregation tree shown in Figure 16.11, let \( g_i = g_i(v_1, \ldots, v_{n-1}) \) denote the value of the data at node \( i \) and \( F(i) \) the set of its parents.

Based on the form of \( g_i(\cdot) \) the two simple benefit models could be defined as
\[
\begin{align*}
  g_i^{(a)} &= v_i + \sum_{j \in F(i)} v_j \\
  g_i^{(b)} &= v_i + \sum_{j \in F(i)} p_j v_j
\end{align*}
\]  

The first model captures the ‘memoryless’ property of information transfer on a path, i.e. once information has reached a particular sensor its benefits in forwarding that information are not constrained by the choices of its ancestors and depend only on the survival probabilities of sensors from \( s_i \) onwards. In the second case, we model \( s_i \) as obtaining information from its parents only if they survive with the given probabilities.

If \( l = (l_1, \ldots, l_{n-1}) \) is any valid strategy profile resulting in a routing tree \( T \) rooted at \( s_q \), the payoff at node \( s_i \) under \( T \) can be written as:
\[
\Pi_i(l) = \begin{cases} 
  X_i - (c_{ij} + C P_j), & \text{if } s_i \in T \\
  0, & \text{otherwise}
\end{cases}
\]  

A strategy \( l_i \) is said to be a best response of player \( i \) to \( l_{-i} \) if \( 0 \leq \Pi_i(l'_i, l_{-i}) \leq \Pi_i(l_i, l_{-i}) \) for all \( l'_i \in \mathcal{L}_i \). If \( BR_i(l_{-i}) \) denotes the set of player \( i \)’s best response to \( l_{-i} \), a strategy profile \( l = (l_1, \ldots, l_n) \)
is said to be an optimal RQR (reliable query routing) tree $T$ if $l_i \in BR_i (l_\ldots)$ for each $s_i$, i.e. sensors are playing a Nash equilibrium.

### 16.9.2 Optimal rout computation

It can be shown that for both benefit models, given $p_i \in (0, 1]$ and $c_{ij} + CP_i = c$ for all $ij$, the most reliable tree $RT$ is always optimal. For uniform $p_i$, the optimal RQR tree is also the one with least overall cost. Let $s_i$ and $s_{i+1}$ be subsequent nodes on the most reliable tree and $R_i$ the reliability of the most reliable path from $s_i$ to $s_q$ with $R_i$ being the reliability along any alternative path from $s_i$. For any neighbor $s_j$ not on the optimal path let $\Delta c_j = (c_{i,j+1} + CP_j) - (c_{ij} + CP_j)$, and $\Delta R_i$ is defined similarly. Given $G$ and $P(s_i) = p_i \in (0, 1]$, tree $RT$ will be optimal under payoff model (a) if $\Delta c_i / \Delta R_i < s_i$ for all $s_i$ on $RT$. Also, given $G$ and $P(s_i) = p_i \in (0, 1]$, tree $RT$ will be optimal under payoff model (b) if $(\Delta c_{i+1} / \Delta c_i) < (\Delta R_{i+1} / \Delta R_i)$ for all $s_i$ and $s_{i+1}$ on $RT$.

Energy efficiency and sensor lifetime are also affected by the length of routing paths since longer paths result in energy consumption at more sensors. The strategy space of each sensor in the geographically routed RQR game includes only those neighbors closer to the destination than itself. Routing paths under this regime are thus implicitly length constrained. For each sensor, the set of downstream neighbor nodes to a given destination can be found using protocols such as GFG [31] and greedy perimeter stateless routing (GPSR) [32].

### 16.10 PROFIT DRIVEN ROUTING IN COGNITIVE NETWORKS

#### 16.10.1 Algorithmic mechanism design

In order to introduce category differential cost or profit in to our analysis, in this section we will use additional results and even terminology from economics and mechanism design theory. Conventionally in these fields, typical scenarios in which the agents act according to their own self-interests are modelled with $n$ agents where each agent $i$, for $i \in \{1, \ldots, n\}$, has some private information $t_i'$, called its type. Applied to the routing problem, type $t_i'$ is the cost to a node’s of forwarding a packet in unicast scenario. All agents’ types $t = (t_1, t_2, \ldots, t_n)$ define a type vector, which is called the profile. Given a reported profile, there is an output specification $O$ that maps each type vector $t$ to an allowed output. For each possible output $o$, agent $i$’s preferences are given by a valuation function $w_i^o$ that assigns a real number $w_i^o (t_i', o)$, which does not depend on other agents’ types.

Given a reported profile $a = (a_1, \ldots, a_n)$, a mechanism defines an output $O(a)$ and a payment vector $p(a) = (p_1(a), \ldots, p_n(a))$, where $p_i' = p_i^o (a)$ is the money given to each participating agent $i$. Agent $i$’s utility is $u_i^t (a) = w_i^o (t_i', a) + p_i^o (a)$. A rational agent $i$ always tries to maximize its utility $u_i^t (a)$ by choosing its action $a_i$. A mechanism satisfies the incentive compatibility (IC) if each agent maximizes its utility by reporting its type $t_i'$ regardless of what other agents do. A mechanism satisfies the individual rationality (IR) (also called voluntary participation) if each agent’s utility of participating in the action is nonnegative. A mechanism is strategyproof (or called truthful) if it satisfies both IC and IR properties.

The family of Vickrey–Clarke–Groves (VCG) mechanisms suggested by Vickrey [34], Clarke [35], and Groves [36] are effectively used in mechanism design maximization problems where the objective function $g(o, t) = \sum_i w_i^o (t_i', o)$, and the set of possible outputs is assumed to be finite. This is equivalent to social optimum in game theory discussed so far in this section. This maximization mechanism design problem is often called utilitarian. A mechanism $M = (O(t), p(t))$ belongs to the family of VCG mechanisms if (a) the output $o = O(t)$ computed based on the type vector $t$ maximizes the objective function $g(o, t) = \sum_i w_i^o (t_i', o)$, and (b) the payment to an agent $i$ is of the format $p_i^o (t) = \sum_{j \neq i} w_i^o (t_j', O(t)) + h(t_i' - t_i')$. Here, $h(t_i')$ is an arbitrary function of $t_i'$. A VCG mechanism is truthful [36] under mild assumptions, VCG mechanisms are the only truthful implementations for utilitarian problems[19]. Analogous to game
theory and notation used so far, $a^{-i}$ denotes the vector of strategies of all agents except $i$, i.e. $a^{-i} = (a_1, a_2, \ldots, a_{i-1}, a_{i+1}, \ldots, a_n)$ and $a^{-i} b = (a_1, a_2, \ldots, a_{i-1}, b, a_{i+1}, \ldots, a_n)$, i.e. each agent $j \neq i$ uses strategy $a^j$, except that the agent $i$ uses strategy $b$. $V = \{v_0, v_1, \ldots, v_{n-1}\}$ is a set of $n$ wireless nodes where $v_0$ is used to represent the access point (AP) of the wireless network. $G = (V, E)$ is the communication graph defined by $V$, where $E$ is the set of links $(v_i, v_j)$ such that the node $v_i$ can communicate directly with the node $v_j$. It will be assumed that $G$ is node biconnected which means that the remaining graph, after removing any node $v_i$ and its incident links from $G$, is still connected. In addition to providing fault tolerance, in this context the biconnectivity of the graph $G$ will prevent the monopoly of nodes. We assume again that each wireless node $v_i$ has a fixed cost $c_i$ of relaying/sending a data packet to any (or all) of its outgoing neighbors. This cost $c_i$ is private information, only known to node $v_i$. All $n$ nodes together define a cost vector $c = (c_0, c_1, \ldots, c_{n-1})$, which is the profile of the network $G$. In the model used in this section, when a node $v_i$ wants to make a guaranteed profit $z_i$, its declared ‘cost’ should be $c_i + z_i$.

When a node $v_i$ sends data to the access point $v_0$, the least cost path (with minimum total relaying cost) from node $v_i$ to node $v_0$, denoted by $p(v_i, v_0, c)$, is used to route the packets. For a path $\Pi (i, 0) = v_{i_1}, v_{i_2 - 1}, \ldots, v_{i_k}, v_0$ connecting node $v_i$ and node $v_0$, i.e. $v_{i_j} = v_i$ and $v_{i_0} = v_0$, where node $v_{i_j}$ can send signals directly to node $v_{i_{j-1}}$, the cost of the path $\Pi (i, 0)$ is defined as $\sum_{j=1}^{k} c_{i_j}$, which excludes the costs of the source node and the target node.

To stimulate cooperation among all wireless nodes, node $v_i$ pays some nodes of the network to forward the data to the access point. Thus, each node $v_j$ on the network declares a cost $d_j$, which is its claimed cost to relay the packets. Note that here, $d_j$ could be different from its true cost $c_j$. Then, node $v_i$ computes the least cost path $p(v_i, v_0, d)$ to connect to the access point $v_0$ according to the declared cost vector $d = (d_0, d_1, \ldots, d_{n-1})$. For each node $v_j$, a payment $p^i_j(d)$ is computed according to the declared cost vector $d$. The utility of node $v_i$ is $u^i(d) = p^i_j(d) - x_j(i) \cdot c_j$, where $x_j(i) \in \{0, 1\}$ indicates whether $v_j$ relays the packet for $v_i$. We always assume that the wireless nodes are rational: each always tries to maximize its utility $u^i(d)$. At the beginning we assume that the wireless nodes do not collude to improve their utilities. We will relax this assumption later.

### 16.10.2 Profit driven pricing mechanism

In this protocol, the payment to a node $v_k$ on the LCP (least cost path) is $d_k$ plus the difference between the cost of the least cost path without using $v_k$ and the cost of the least cost path. This simply means that there is no cheaper alternative. For a formal definition let us assume that the node $v_k$ has to send packets to $v_0$ through the relay of some other nodes. It pays these relay nodes to compensate their costs for carrying the transit traffic incurred by $v_k$. The output $(d)$ of the mechanism is the path connecting $v_i$ and $v_0$ with the minimum cost, which is known as $P(v_i, v_0, d)$. The payment to a node $v_k$ is 0 if $v_k \notin P(v_i, v_0, d)$. Otherwise, its payment is:

$$p^k_i(d) = \|P_{-v_k}(v_i, v_0, d)\| - \|P(v_i, v_0, d)\| + d_k$$  \hspace{1cm} (16.121)

Here, $P_{-v_k}(v_i, v_0, d)$ denotes the least cost path between node $v_i$ and $v_0$ without using node $v_k$, and $\|\Pi\|$ denotes the total cost of a path $\Pi$.

This payment falls into the VCG mechanism, so it is strategyproof. In other words, if $d_k = c_k$, node $v_k$ maximizes its utility $p^k_i(d) - x_k(i) \cdot c_k$. Every node participating in the relay will have a nonnegative profit; every node that does not relay the traffic will have profit 0. In this concept we need the network to be biconnected, in order for the path $P_{-v_k}(v_i, v_0, d)$ to exist. Otherwise, node $v_k$ can charge a monopoly price since it is a critical node for connecting $v_i$ and $v_0$ with no alternative. The access point can collect all nodes’ costs, and the network structure $G$ and can compute the payment to all relay nodes in a centralized manner. Using Dijkstra’s algorithm to calculate the payment for all nodes on $P(v_i, v_0, d)$ would be rather inefficient. For a network with $n$ nodes and $m$ edges, in the worst case, there will be $O(n)$ nodes on $P(v_i, v_0, d)$, so such an algorithm would result in a time complexity $O(n^2 \log n + mn)$. A fast payment calculation algorithm for edge weighted graph (by
assuming the edges are rational agents) is given in [38]. A method for finding the most vital node of a shortest path in an edge weighted graph in time $O(m + n \log n)$ is presented in [39].

Based on ideas from [38], an $O(m + n \log n)$ time complexity algorithm for fast payment calculation in a node weighted graph is presented in [33]. To calculate (16.121) the algorithm with computing $\|P_{-v_k}(v_i, v_0, d)\|$ for a node $v_k \in P(v_i, v_0, d)$. The basic idea of the algorithm is for a pair of nodes $v_a, v_b$ such that $v_av_b \in G$, the path $P_{-v_k}(v_a, v_b, d)$ and $P_{-v_k}(v_b, v_0, d)$ are calculated separately. Then, by concatenating $P_{-v_k}(v_a, v_b, d)$, link $v_a, v_b$, and $P_{-v_k}(v_b, v_0, d)$, we obtain the path with the minimum cost from $v_i$ to $v_0$ without node $v_k$ and having $v_a, v_b$ on it. Choosing the minimal cost path for all edges $v_a, v_b \in G$, we find $\|P_{-v_k}(v_i, v_0, d)\|$, as illustrated in Figure 16.12.

In wireless ad-hoc networks, due to lack of a centralized authority, it is more desirable to compute the payment in a distributed manner. Assume that there is a fixed destination node $v_0$. The distributed algorithm presented in [33,40] computes the payment of each node $v_i$ to all its relay nodes. The distributed algorithm has two stages. First, all nodes together find the shortest path tree (SPT) rooted at node $v_0$. It is assumed that the SPT tree does not have a loop. This step can be easily implemented using Dijkstra’s algorithm. Once a shortest path tree $T$ rooted at node $v_0$ is formed, and every node knows its parent and children in tree $T$, in the second step every node $v_i$ computes its payment $p_k^i$ as follows:

(i) set $p_k^i \leftarrow \infty$, if $v_k \in P(v_i, v_0, d)$; otherwise, $p_k^i \leftarrow 0$;

(ii) broadcasts its entries $p_k^i$ to its neighbors;

(iii) while $v_i$ receives an updated price from a neighbor $v_j$

   do

(iv) if $v_j$ is the parent of $v_i$ then

   (v) $p_k^i \leftarrow \min(p_k^i, p_k^j)$ if $v_k \in P(v_i, v_0, d)$

   (vi) else if $v_i$ is the parent of $v_j$ then

Figure 16.12 Computing $v_k$-avoiding shortest path.
(vii) \( p_i^k \leftarrow \min \left( p_i^k, p_j^k + d_i + d_j \right) \) if \( v_k \in P(v_i, v_0, d) \)

(viii) else

(ix) for every \( v_k \in P(v_i, v_0, d) \), \( v_i \) updates \( p_i^k \) as follows:

(x) if \( v_k \in P(v_j, v_0, d) \) then

(xi) \( p_i^k \leftarrow \min \left( p_i^k, p_j^k + d_j + \| P(v_j, v_0, d) \| - \| P(v_i, v_0, d) \| \right) \);

(xii) else

(xiii) \( p_i^k \leftarrow \min \left( p_i^k, d_k + d_j + \| P(v_j, v_0, d) \| - \| P(v_i, v_0, d) \| \right) \)

(xiv) Broadcasts its entries \( p_i^k \) to its neighbors.

Whenever some entry \( p_i^k \) stored at node \( v_i \) changes, the entry \( p_i^k \) is sent to all neighbors of \( v_i \) by node \( v_i \). When the network is static, the price entries decrease monotonically and converge to stable values after a finite number of rounds (at most, \( n \) rounds).

16.10.3 Truthful behavior in cognitive networks

The node \( v_i \) has the incentive not to correctly calculate his payment \( p_i^k \) in the second stage, and even to lie about his shortest path in the first stage. To illustrate these incentives we use the graph in Figure 16.13 where the shortest path between \( v_0 \) and \( v_1 \) should be \( v_1v_4v_3v_2v_0 \).

For \( d_k = 1 \) it is easy to calculate that \( v_1 \)'s payments to nodes \( v_2, v_3, \) and \( v_4 \) are all exactly 2. Then, the overall payment by node \( v_1 \) is 6. If node \( v_1 \) lies that it is not a neighbor of \( v_4 \), then its shortest path becomes \( v_1v_5v_0 \). Now, it only needs to pay \( v_5 \) 5 to send a packet. Thus, node \( v_1 \) benefits by lying about its neighborhood connection information, which consequently changes the SPT. This problem rises from the fact that the least cost path is not necessarily the path that you pay the least.

By noticing that the distributed method of computing the payment relies on the selfish node \( v_i \) to calculate the payment \( p_i^k \) to node \( v_k \), which cannot prevent node \( v_i \) from manipulating the calculation in its favour, [40] suggests using the following approach: all agents are required to sign all messages that they send and to verify all messages that they receive from their neighbors. Reference [40] claims that the protocol can be modified so that all forms of cheating by agents are detectable. They did not consider the possible scenario when an agent could lie about the topology (the scenario we just discussed in previous paragraph).

![Figure 16.13 Illustration of the node’s incentive to lie about its shortest path.](image-url)
In [33] a different distributed method was presented that prevents nodes from lying about the topology, and miscalculating the payment, and it does not need to store all messages. The nodes use the following algorithm to compute the shortest distance to source:

(i) Every node $v_i$ has two variables: $D(v_i)$ stores the shortest distance to $v_0$ and $FH(v_i)$ stores its parent on SPT. Initially, if $v_0$ is $v_i$’s neighbor then set $D(v_i)$ to 0 and $FH(v_i)$ to $v_0$; else set $D(v_i)$ to $\infty$ and $FH(v_i)$ to NULL.

Node $v_i$ broadcasts its information to its neighbors.

(ii) while $v_i$ received information from its neighbor $v_j$ do

(iii) if $D(v_i) > D(v_j) + c_j$ then

(iv) $D(v_i) \leftarrow D(v_j) + c_j$, and $FH(v_i) \leftarrow v_j$;

(v) if $v_i \neq FH(v_j)$ and $D(v_i) + c_i < D(v_j)$; or $v_i = FH(v_j)$ and $D(v_i) + c_i \neq D(v_j)$ then

(vi) node $v_i$ contacts $v_j$ directly using reliable and secure connection, asking $v_j$ to update his $D(v_j)$ to $D(v_i) + c_i$ and $FH(v_j)$ to $v_i$. After the necessary updating, $v_j$ must broadcast his information.

(vii) Node $v_j$ broadcasts its information to its neighbors.

and then use the following algorithm to compute the payment to the relay nodes:

(i) (i) Set $p^k_i \leftarrow \infty$, if $v_k \in p(v_i, v_0, d)$; otherwise, $p^k_i \leftarrow 0$.

(ii) (ii) Broadcasts its entries $p^k_i$ to its neighbors.

(iii) (iii) while $v_i$ received information $p^k_j$ from its neighbor $v_j$ do

(iv) (iv) $v_i$ updates $p^k_i$ using algorithm 2 (Steps 4–15).

(v) (v) when $p^k_i$ changes, $v_i$ broadcasts the value of $p^k_i$, and the ID of the node $p_j$ that triggered this change.

(vi) (vi) if $v_i$ triggered the change for this $p^k_j$ from $v_j$, $v_i$ recalculates $p^k_j$ for $v_j$ using algorithm 2 (Steps 4–15) to verify it. If his answer and the payment sent from $v_j$ do not match, node $v_i$ then notifies $v_j$ and other nodes.

(vii) Node $v_j$ will then be punished accordingly, e.g. $v_j$ is dropped from the network by all nodes.

It is easy to verify that they are truthful and no node will lie about its neighbor information and will follow the payment calculation procedure. For the example illustrated in Figure 16.13, node $v_i$ has to use the shortest path $v_1v_4v_3v_2v_0$ to compute the payment according to the latest protocol since node $v_4$ knows the existence of this shortest path and it will detect the lie by node $v_1$ if node $v_1$ choose to use path $v_1v_5v_0$ instead. The problem remaining is how to make it more efficient.

### 16.10.4 Collusion of nodes in cognitive networks

In general, the nodes may collude with each other in the hope of gaining as a group. For example, if two nodes $v_{k_1}$ and $v_{k_2}$ know that the removal of them will disconnect some nodes from the access point, then these two nodes can collude to declare arbitrarily large costs and charge a monopoly price
together. Notice that, by declaring much higher costs together, one node’s utility may decrease, but the sum of their utilities is guaranteed to increase, which they can agree to share.

A mechanism is said to be \( k \)-agents strategy proof if, when any subset of agents of size \( k \) colludes, the overall utility of this subset is made worse off by misreporting their types. A mechanism is true group strategyproof if it is \( k \)-agents strategy proof for any \( k \).

There is no a true group strategyproof mechanism for the unicast routing problem studied above. For example if all nodes but node \( v_i \) collude and declare arbitrarily high costs, then node \( v_i \) has to pay a payment arbitrarily higher than the actual payment it needs to pay if these nodes do not collude.

In the following we focus on a simpler problem and study how to design a truthful mechanism such that it can prevent nodes from colluding with its one-hop neighbors. Notice that the VCG payment scheme (16.121) does not prevent a node from colluding with its neighbors at all. It is not difficult to construct an example such that, for a node \( v_k \in P(v_i, v_0, d) \), the path \( P_{-v_k}(v_i, v_0, d) \) uses a node \( v_l \) that is a neighbor of \( v_k \) and \( v_l \in P(v_i, v_0, d) \). Then, \( v_l \) can lie its cost up to increase the utility of node \( v_k \).

If \( v_i \) pays other nodes to relay the data to another node \( v_j \) when \( N(v_k) \) is the set of neighbors of node \( v_k \), including node \( v_l \) then to have a payment scheme that prevents collusion between any two neighboring nodes, it is necessary that the graph resulted by removing \( N(v_k) \) still has a path connecting \( v_i \) and \( v_j \). By assuming that graph \( G \setminus N(v_k) \) is connected for any node \( v_k \), [31] designs the following payment scheme \( \tilde{p} \) that avoids the collusion between any two neighboring nodes. The payment \( \tilde{p}_l^k(d) \) to a node \( v_l \):

\[
\tilde{p}_l^k(d) = \|P_{-N(v_k)}(v_l, v_0, d)\| - \|P(v_l, v_0, d)\| + x_k(i) \cdot d_k \tag{16.122}
\]

where \( P_{-N(v_k)}(v_l, v_0, d) \) is the least cost path connecting \( v_i \) and \( v_j \) in graph \( G \setminus N(v_k) \) without using any node in \( N(v_k) \), and \( x_k \) denotes whether a node \( v_k \) is on the least cost path or not. Notice that the payment to a node \( v_k \notin P(v_i, v_0, d) \) could be positive when node \( v_k \) has a neighbor on \( P(v_i, v_0, d) \). This is a main difference to the payment scheme in Equation (16.121) based on VCG. The payment scheme \( \tilde{p} \) is a strategy proof mechanism that prevents any two neighboring nodes from colluding, because each individual node will be truthful since the mechanism belongs to the family of VCG family [40–42]. For any two neighboring nodes \( v_k \) and \( v_l \), their utilities can be written as:

\[
\begin{align*}
  u^k(c) &= \sum_{i=0}^{n-1} v_i(\sigma(c), c_i) + h^{-k}(c^{-N(v_k)}) \\
  u^l(c) &= \sum_{i=0}^{n-1} v_i(\sigma(c), c_i) + h^{-l}(c^{-N(v_l)})
\end{align*}
\tag{16.123}
\]

Summing them up, we get

\[
u^l(c) + u^k(c) = 2 \sum_{i=0}^{n-1} v_i(\sigma(c), c_i) + h^{-k}(c^{-N(v_k)}) + h^{-l}(c^{-N(v_l)}) \tag{16.124}\]

Notice that \( h^{-k}(c^{-N(v_k)}) + h^{-l}(c^{-N(v_l)}) \) does not depend on \( d_i \) and \( d_k \) since \( v_k \) and \( v_l \) are neighbors of each other. In addition, \( \sum v_i(\sigma(c), c_i) \) is maximized when they reveal their true costs. Thus, \( v_k \) and \( v_l \) will maximize their total utilities by revealing their true costs.

This concept can be extended to a more general case to prevent some groups of nodes from colluding. Let \( \{Q(v_1), Q(v_2), \cdots, Q(v_k)\} \) be a set of subsets of nodes, i.e. \( Q(v_k) \subset V \). We are interested in designing a truthful mechanism such that any node \( v_k \) cannot collude with other nodes in \( Q(v_k) \) to increase their total utilities. If \( v_k \in Q(v_k) \), for \( 1 \leq k \leq n \) the mechanism is truthful if the output is the least cost path connecting the source \( v_i \) and destination \( v_0 \), and the payment \( \tilde{p}_l^k(d) \) to a node \( v_k \) is

\[
\tilde{p}_l^k(d) = \|P_{-Q(v_k)}(v_l, v_0, d)\| - \|P(v_l, v_0, d)\| + x_k(i) \cdot d_k \tag{16.125}
\]

Obviously, we need graph \( G \setminus Q(v_k) \) to be connected for any node \( v_k \).

Another possible collusion happens after the payment is calculated and during the process of actually routing the packets. Let total payment \( p_i = \sum_{l=0}^{n-1} p_l^k \), of node \( v_i \) to all relay nodes on the least cost path is \( P(v_i, v_0, c) \) and \( p_j > p_i + \max(p_j, c_j) \) for some neighbor \( v_j \) of \( v_i \). Notice
that \( \max(p_i^j, c_j) = x_j(i)p_i^j + (1 - x_j(i))c_j \) since if \( v_j \) is on LCP \( P(v_i, v_0, c) \), then \( p_i^j \geq c_j \) and \( p_i^j = 0 < c_j \) otherwise. Here, \( x_j(i) \) indicates whether node \( v_j \) is on \( P(v_i, v_0, c) \). In this situation, \( v_i \) and \( v_j \) can collude in their favor as follows: (i) \( v_j \) sends the data packets for \( v_i \) and \( v_j \) pays all relay nodes on path \( P(v_i, v_0, c) \). (iii) \( v_i \) pays \( v_j \) the cost \( p_j + \max(p_i^j, c_j) \), which covers the payment by \( v_j \). (c) \( v_i \) and \( v_j \) split the difference \( p_i - (p_j + \max(p_i^j, c_j)) \), which is the saving of node \( v_i \) from colluding with node \( v_j \). Notice that it is possible that \( p_j + \max(p_i^j, c_j) \) for some neighbor \( v_j \) of \( v_i \). Figure 16.14 illustrates an example of such collusion.

Using the payment function, it is easy to compute that \( p_8 = 20 \), \( p_4 = 6 \), and \( p_8^5 = 0 \). Notice \( c_4 = 5 \). Thus, \( v_8 \) can ask \( v_4 \) to forward the data packets using its LCP to \( v_0 \). Node \( v_8 \) pays node \( v_4 \) a price \( 6 + 5 = 11 \) to cover its payment \( p_4 \) and its cost \( c_4 \), and half of the savings, which is 4.5. Thus, the total payment of node \( v_8 \) is only 15.5 now, which is less than \( p_8 \) and node \( v_4 \) also increases its utility from 0 to 4.5.

16.11 GAME THEORETICAL MODEL OF FLEXIBLE SPECTRA SHARING IN COGNITIVE NETWORKS WITH SOCIAL AWARENESS

We assume a radio network with two type of users. The type \( a \) users use much a wider frequency band \( B_a \), say WLAN based on OFDM, and type \( b \) users use a much narrower band \( B_b \), say voice users or low rate data systems. Both types of terminal are frequency agile so that the radio resource management can continuously repack the users any time when a user leaves the channel. So for an available frequency band \( B = B_a(t) \cup B_b(t) \), in any time instant all active users are arranged so to occupy all frequencies \( f \) in the range \( f_{\min} < f < f_a(t) < f_b(t) \) in the used bandwidth \( B_a(t) \) and the rest of the bandwidth \( B \) is free for contention of new arrivals and will be referred to as \( B_r(t) \). The used bandwidth and contention bandwidth do not overlap and the they change in time. To avoid collisions listen before talk principle is used by scanning the band and if nonzero \( B_r(t) \) is detected the spectra is accessed with probability \( p \).

In an advanced society, decision-making and strategy-setting people do not (or will not in the future) behave like the self-interested ‘rational’ actor depicted in neoclassical economics and classical game theory [43]. In an HE (homo equalis) society, individuals have an inequality aversion. An HE society [43] can be modelled with a utility function of player \( i \), \( u_i \) in an \( n \)-player game as:

\[
 u_i = x_i - \frac{\alpha_i}{n-1} \sum_{j > i} (x_j - x_i) - \frac{\beta_i}{n-1} \sum_{j > i} (x_i - x_j) \tag{16.126}
\]

where \( x = (x_1, \ldots, x_n) \) are the payoffs for each player and \( 0 \leq \beta_i < \alpha_i \leq 1 \). \( \beta_i < \alpha_i \) reflects the fact that HE exhibits a weak urge to inequality when doing better than the others, and a strong urge to reduce inequality when doing worse than the others. In [43], it is shown that in this model, equilibria in public goods (social awareness) games, with fairness, can be reproduced.
Figure 16.15 Airtime achieved with different weight ratios $L_a/L_b$ and no queueing.

The inequality aversion property of the HE agents is utilized in [44] to achieve fairness in the spectrum access problem. In this scheme, each radio system learns the access probability $p_i$ by itself. Here, we define $\text{Onlinetime}_i = t_{oni}$ as the averaged cumulative ‘on’ spectrum time per radio system of type $i$. Then, we define $x_i = t_{oni}/L_i$, where $L_i = \theta_i \lambda_i$, $\theta_i$ is the priority parameter and $\lambda_i$ is the traffic load for a type $i$ radio system. The cumulative $t_{oni}$ is normalized by the radio system’s traffic load and priority, which makes this spectrum access scheme able to adapt to different traffic loads and priorities and, hence, achieve more efficiency and maintain weighted fairness defined as $t_{oni}/L_i = t_{onj}/L_j = K, \forall i, j$. With initial $p_i = 1$, each time the probability $p_i$ is updated, as follows:

$$p_i = \max \left( 0, \min \left( 1, P_i + \frac{\alpha_i}{n-1} \sum_{x_j \geq x_i} \frac{x_i - x_j}{x_j} - \frac{\beta_i}{n-1} \sum_{x_j < x_i} \frac{x_i - x_j}{x_i} \right) \right)$$

for all $j \neq i$ \hspace{1cm} (16.127)

where $n$ is the number of different radio system types. This forces each radio system to make an effort to use the idle spectrum efficiently while taking fairness into consideration. Here the only local information needed is the radio system’s own history of the $\text{onlinetime}$ and the $\text{onlinetime}$ of the other radio systems whose spectrum is within the same spectrum block. Figure 16.15 illustrates the system performance.

16.12 A GAME THEORETICAL MODELLING OF SLOTTED ALOHA PROTOCOL

The ‘listen before talk’ principle used in the previous section for spectrum access would correspond to the CSMA/CD protocol for channel access in packet networks. In this section we discuss finite-size slotted ALOHA sensor networks with CDMA channel and selfish sensors [45–51]. Each sensor wishes to maximize its individual expected reward. The system exploits decentralized channel state information (CSI) to obtain transmission policies that are optimal for each sensor. The problem is formulated as a finite player, finite action, noncooperative stochastic game, where each sensor is a
selfish but rational player. We consider a network of $K$ ($K < \infty$) sensors where the uplink signal to noise ratios (SNRs) $\gamma_i$ represent the channel states of a sensor $i$. The channel state of the networks is $\vec{\gamma} \rightarrow (\gamma_1, \gamma_2, \ldots, \gamma_K)$. We assume that at the beginning of each time slot sensor $i$ knows its $\gamma_i$. Let $F_i(\cdot)$ be the probability distribution function of $\gamma_i$. During a time slot, if sensor $i$ does not transmit, a waiting cost is $c_i^{(i)}$; if it transmits and its packet is not received successfully, a transmission cost is $c_i^{(i)}$. Finally, if sensor $i$ transmits and its packet goes through, a reward of $1 - c_i^{(i)}$ is obtained with the necessary condition $0 \leq c_i^{(i)} < c_i^{(i)} < 1$ for all sensors $i$. We consider CDMA systems with matched filter receivers where a packet from sensor $i$ is considered successfully received if:

$$\frac{P_j}{\sigma^2 + P_{l-j}} = \frac{\gamma_j}{J_{-j}} > \beta$$

$$J_{-j} = 1 + \frac{1}{N} \sum_{i \neq j} \gamma_i$$

$$P_{l-j} = \frac{1}{N} \sum_{i \neq j} P_i$$

(16.128)

where $P_j$ is the received power of sensor $i$, $P_j = \gamma_j \sigma^2$; $\sigma^2$ is the noise power, $N$ is the spreading gain and $\beta$ is required SNIR (signal to noise plus interference ratio) that guarantee a certain QoS requirement. The noncooperative stochastic game is defined as follows:

- The set of players $I$ is the set of sensors indexed by $i = 1, 2, \ldots, K$.
- For any player $i$; $i = 1, 2, \ldots, K$, the set of actions includes either wait ($W$) or transmit ($T$), denoted as $A_i = \{W, T\}$. A player can choose to transmit with some probability, i.e. randomized strategies are allowed.
- A strategy is a mapping from channel states to transmit probabilities and the strategy of sensor $i$ will be a function $p_i(\cdot) : R_+ \rightarrow [0, 1]$.

If we drop indices for simplicity, the expected reward of sensor $i$ when it plays with transmit policy $p(\cdot)$, is:

$$L(p(\cdot)) = \int_0^\infty (p(\vec{\gamma}) \Psi(\vec{\gamma}) - (1 - p(\vec{\gamma})) c_w) f(\vec{\gamma}) d\vec{\gamma}$$

$$= \int_0^\infty p(\vec{\gamma}) (\Psi(\vec{\gamma}) + c_w) f(\vec{\gamma}) d\vec{\gamma} - c_w$$

(16.129)

where $\Psi(\vec{\gamma})$ is the expected reward when the channel state of the sensor is $\vec{\gamma}$. For a CDMA ALOHA system with matched filter receivers and SINR threshold reception model, $\Psi(\vec{\gamma})$ is:

$$\Psi(\gamma) = \int_0^\infty \int_0^\infty \left( \mathbf{1} \left( \frac{\gamma}{J_{-1}} > \beta \right) - c_i \right) dF_2(\gamma_2) \ldots dF_K(\gamma_K)$$

(16.130)

where $\mathbf{1}(\cdot)$ is the indicator function. Maximizing the expected reward of the player $i$ can be formulated as:

$$\max_{p(\cdot)} L(p(\cdot))$$

(16.131)

It can be shown for the game defined above that there exists a Nash-equilibrium at which each sensor adopts a threshold transmission strategy: $p(\gamma) = \mathbf{1}(\gamma > \theta)$ for some $\theta \in [0, \infty)$.

To prove it let us rewrite Equation (16.130) for sensor $i$ in the network as:

$$\Psi(\gamma) = \sum_{\gamma \in \mathbb{R}_{\gamma_2} \ldots \mathbb{R}_{\gamma_K}} \mathbf{1}(\text{SINR} > \beta) [p_2(\cdot) \ldots p_K(\cdot)] - c_i$$

$$= \Pr(\text{SINR} > \beta) [p_2(\cdot) \ldots p_K(\cdot)] - c_i$$

$$= \Pr(\gamma > \beta(\sigma^2 + P_{l-i}) [p_2(\cdot) \ldots p_K(\cdot)] - c_i$$

(16.132)
It is obvious that $\Psi(\gamma)$ is a nondecreasing function of $\gamma$. In addition, $\Psi(0) = -c_l < -c_w$, $\Psi(\infty) = 1 - c_l > 0$. As a result, there must exist a threshold $\theta$ so that $\Psi(\gamma) + c_w \leq 0$ for all $\gamma \leq \theta$ and $\Psi(\gamma) + c_w > 0$ for all $\gamma > \theta$. The value of $\theta$ may vary among sensors. Since the objective is to maximize $L(p(.)) = \int_0^{\infty} p(\gamma) (\Psi(\gamma) + c_w) f(\gamma) d\gamma - c_w$ it is optimal to select:

$$p(\tilde{\gamma}) = \begin{cases} 1 & \text{if } \Psi(\gamma) + c_w > 0 \\ 0 & \text{otherwise} \end{cases}$$

(16.133)

In other words $p(\gamma) = I(\gamma > \theta)$ is always an optimal policy. It follows that there exists a Nash equilibrium at which all players adopt (possibly different) threshold policies. The above applies to any reception models that satisfy: (i) the expected probability that if a packet is received correctly, make sure that the received SNR is equal to $\tilde{\gamma}$, and (ii) the expected probability that a packet is received correctly.

We now restrict the discussion to the class of threshold policies: $p(\gamma) = I(\gamma > \theta)$, and since the waiting cost $c_w$ is only a constant, we reformulate the optimization problem given by Equations (16.127) and (16.131) as:

$$\max_\theta L(\theta) = \int_0^{\infty} I(\gamma > 0)(\Psi(\gamma) + c_w)f(\gamma) d\gamma$$

(16.134)

The gradient of the objective function is then:

$$\nabla_\theta L(\theta) = \nabla_\theta \int_0^{\infty} I(\gamma > 0)(\Psi(\gamma) + c_w)f(\gamma) d\gamma = -(\Psi(\theta) + c_w)f(\theta)$$

(16.135)

To utilize the gradient ascent method we need to obtain an unbiased estimate of:

$$\Psi(\gamma) = E_{F_2(\gamma),\ldots,F_K(\gamma)}[I(SINR > \beta) | p_2(\cdot), \ldots, p_K(\cdot)] - c_l$$

(16.135a)

In practice, this can be done by using a temporary power control strategy in the learning phase to make sure that the received SNR is equal to $\theta$ and counting the number of ACKs and NACKs that are sent from the base station. If these counts are $\#(\text{ACK})$ and $\#(\text{NACK})$

$$\Psi(\gamma) = E_{F_2(\gamma),\ldots,F_K(\gamma)}[I(SINR > \beta) | p_2(\cdot), \ldots, p_K(\cdot)] = \#\text{ACK}/(\#\text{ACK} + \#\text{NACK})$$

(16.136)

The Nash equilibrium is defined as

$$\theta^* = \{\theta : \nabla_\theta L(\theta) = 0, \nabla^2_\theta L(\theta) < 0\}$$

(16.137)

So, the algorithm for optimal strategy selection can be summarized as follows:

• initialization:

  $l = 1$
  $\theta^{(0)} = \theta$

  • sampling, evaluation of gradient and update loop:

    while $|\Psi(\theta^{(l)}) + c_w| > 0$ do

    estimate: $\Psi^{(l)}(\theta^{(l)})$ using Equations (16.135) and (16.136);

    Update Equation: $\theta^{(l+1)} = \theta^{(l)} - \varepsilon_l \left(\Psi^{(l)}(\theta^{(l)}) + c_w\right)$

    $l = l + 1$

end while

• Conditions: $\sum_{l=0}^{\infty} \varepsilon_l = \infty; \sum_{l=0}^{\infty} \varepsilon_l^2 < \infty$
Figure 16.16 Throughput for three different transmission control schemes [45] © IEEE 2005.

The sequence \( \theta \) generated by the algorithm converges to the threshold level corresponding to a local optimizer \( \theta^* \), defined in Equation (16.137), of (16.134) of the sensor’s individual expected reward.

For a fixed initial \( \theta \), the sequence \( \nabla_{\theta} L(l)(\theta) : l = 1,2 \ldots \) are i.i.d. Equation (16.138) is the well known Robbins Munro algorithm. The convergence of this algorithm is proved in [48] under the condition in (16.139) and uniform integrability of \( \nabla_{\theta} L(l)(\theta) \), which requires that the channel distribution has finite variance. The above algorithm can be implemented in real time and can adapt to the changes in the statistics of the network. For adaptability of the algorithm, step size depends on the speed of the channel and remaining error in the parameter estimation. The faster channel will require large step size that will result in the larger remaining steady state estimation error and vice versa. Figure 16.20 presents numerical results for slotted ALOHA CDMA networks with matched filter receivers and random signature sequences in Rayleigh fading channels with fixed transmission cost \( c_t = 0.2 \) and \( c_w = 0.02 \).

Figure 16.16 compares the system throughput obtained by using the above algorithm with two other cases: (1) without transmission control (which means a sensor always transmits with certainty), and (ii) using the algorithm in [51] for optimal transmission control without CSI. In the latter algorithm \( p = A/S \) where \( A \) refers to number of arrivals and \( S \) to the number of backlogged packets. It can be seen from figure that the system throughput obtained by the decentralized algorithm with CSI, discussed in this chapter, is superior to the other two cases.

### 16.13 GAME-THEORY-BASED MODELING OF ADMISSION IN COMPETITIVE WIRELESS NETWORKS

#### 16.13.1 System model

In this section we will discuss the scenario where there are \( M \) service providers and \( N \) users. At a given time \( t \), let the number of users subscribing to provider \( i \) be denoted as \( n(t)_i \), where \( 1 \leq i \leq M \). The users are not allowed to subscribe to multiple service providers at the same time (with the same mobile device). In other words, \( \sum_{i=1}^{M} n(t)_i = N \). Any user might become unsatisfied.
with the service it receives and decide to switch operators (a process called ‘churning’). So, the vector \([n(t)_1, n(t)_2, \ldots, n(t)_M]\) changes continuously as the competition dynamics evolve due to the user’s churning behavior. Since each user may choose any of the \(M\) service providers at any time, the possible number of states in the system is given by \(M^N\). Modelling this dynamic as a single game would make it complex. However, the assumption that each user is associated with only one service provider at a time enables us to model this one-to-one relationship between a particular user and his current service provider as a two-player game \(G_j\) (for \(1 \leq j \leq N\)) at any instant. Modeling the admission control in this scenario will be based on a noncooperative and nonzero-sum game. The game is noncooperative in nature because the service provider, on one hand, wants to maximize its revenue, which is modeled as its payoff from this game. Its attempt to maximize user satisfaction, system utilization, etc. is merely a way to achieve the ultimate goal of revenue maximization. Users, on the other hand, want to maximize their satisfaction at a minimum expense, including the option to churn to a better provider in a competitive market. Thus, the user’s overall satisfaction is modelled as his payoff from the admission control game. Since these two goals are different and often conflicting with each other, the service provider and customers do not have the apparent motivation to cooperate.

The game is also a nonzero-sum. In a zero-sum game, an increase in one player’s payoff implies a decrease in the other player’s payoff. However, this may not be true for the relationship between the payoffs of a wireless service provider and a customer. For instance, in an underutilized system, admitting a new request that does not affect the QoS of other ongoing sessions would increase the service provider’s revenue as well as the customer’s satisfaction. Thus, both the payoffs for the service provider and the user are increased, implying that the game is nonzero-sum.

We start with the sequential admission control mechanism in which the requests are processed one by one, i.e. one instance of the game is played every time a new session request is made. In the next step, we will extend the game to admit a batch of \(n\) sessions at a time, leading to a \((n + 1)\)-player game. We assume that the service provider has two strategies: \(SS_1\) (admit the request) and \(SS_2\) (reject the request). The customer seeking admission can have two strategies as well: \(CS_1\) (leave the current provider) and \(CS_2\) (stay with that provider). The payoffs of the two players are expressed by \(A = \begin{bmatrix} a_{ij} \end{bmatrix}_{2 \times 2}\) and \(B = \begin{bmatrix} b_{ij} \end{bmatrix}_{2 \times 2}\), where \(a_{ij}\) and \(b_{ij}\) denote the provider’s and customer’s payoff respectively, if the provider chooses strategy \(SS_i\) and the user chooses strategy \(CS_j\), for \(i \in \{1, 2\}\) and \(j \in \{1, 2\}\). In the sequel we will use the following notation

\[
\begin{align*}
N, & \text{ total number of users in the system;} \\
K, & \text{ number of user (customer) classes;} \\
N_i, & \text{ number of class } i \text{ users currently in the system, } 1 \leq i \leq K; \\
U, & \text{ service provider’s current revenue (utility) coming from all the on-going sessions;} \\
C_i, & \text{ service provider’s average revenue earned from each session of class } i \text{ user which depends on the class number (i.e. priority) of the session, the average length of the session, and the pricing scheme; } \\
L_i, & \text{ service provider’s average revenue loss for losing a class } i \text{ customer, where } 1 \leq i \leq K. \text{ This is estimated by the cost incurred by the provider to attract a new user into the system, assuming the provider maintains a constant customer base;} \\
P_{bi}, & \text{ packet blocking probability for class } i \text{ user, if the new session request is accepted;} \\
R_i(P_{bi}), & \text{ class } i \text{ customer’s churn rate as a function of } P_{bi}. \\
\end{align*}
\]

If a class \(k\) user is requesting a session admission, the provider’s payoff matrix \(A = [a_{ij}]_{2 \times 2}\) is defined as follows:

\[
A = \begin{bmatrix} C + C_k - F & U + C_k - F \\ U - L_k & U \end{bmatrix}
\]

(16.140)
where
\[ F = \sum_{i=1}^{K} N_i R_i (P_{b_i}) L_i \]

and will be elaborated below in more detail. The term \( a_{11} = U - L_k \) denotes the provider’s payoff if it chooses the strategy SS\(_2\) (reject) while the user chooses strategy CS\(_1\) (leave). The payoff of the service provider in this case is the current revenue coming from all ongoing sessions, minus the revenue loss due to the churning of a class \( k \) user. The term \( a_{12} \) is simply the revenue from the current ongoing sessions. The term \( a_{12} \) corresponds to the strategy pair (SS\(_1\)– admit; CS\(_2\)– remains).

In general this should yield the highest payoff for the provider. However, in a fully loaded system, admitting a new customer would either result in an infeasible power assignment (for voice), or delay of the ongoing sessions (for data). Either of these cases may incur revenue loss due to churning of the customers whose ongoing sessions are being affected. How exactly the ongoing sessions are affected depends on the rate control mechanism deployed in the system, which will be described later in this section. The term \( a_{11} \) is defined in a similar way, with one additional term of \( -L_k \), which corresponds to the revenue loss due to user churning.

Admitting a new request into a fully loaded system providing voice services is undesirable because it causes dropping of ongoing calls. However, in CDMA wireless data networks, where infeasible power assignment can be avoided by reducing the transmission rates of ongoing sessions, the pros and cons of admitting the new request should be further evaluated in order to maximize the revenue for the service provider. These aspects are captured in the payoff values \( a_{11} \) and \( a_{12} \). The term \( N_i R_i (P_{b_i}) L_i \) accounts for the total predicted revenue loss for all class \( i \) users currently having ongoing sessions. The micro level rate control scheme ensures that when a class \( k \) user is admitted and the wireless resource is inadequate, only users of the same class or lower classes will be affected. Thus, \( F = \sum_{i=k}^{K} N_i R_i (P_{b_i}) L_i \) measures the total revenue loss from these classes of user. The churn rate, \( R_i (P_{b_i}) \), is obtained by applying a sigmoid utility function that will be presented later in this section.

The user’s payoff matrix \( B = \{b_{ij}\}_{2 \times 2} \) is defined as

\[ b_{ij} = \begin{cases} w_1 U_{ij} - w_2 L_c & \text{for } j = 1 \\ U_{ij} & \text{for } j = 2 \end{cases} \]  

(16.141)

where \( U_{ij} \) is the user’s payoff (utility) from the game without considering churning and \( L_c \) is his penalty if he chooses to churn. The term \( L_c \) models the situation where the customer still under contract with the service provider, is charged a penalty for early termination.

When subscribing to a new service provider, a user may be charged a certain activation fee. Moreover, \( L_c \) may differ from user to user due to a contractual agreement with the provider. Both \( U_{ij} \) and \( L_c \) are expressed in monetary values. The weights \( w_1 \) and \( w_2 \) on the user’s payoff function reflect his preference to save money and get satisfaction, respectively.

The class \( k \) user’s payoff \( (U_{ij}) \) without churning is given by:

\[ U_{ij} = \begin{cases} O_k (P_{b_i}) q + W_a & \text{for } i = 1 \\ 0 + W_b & \text{for } i = 2 \end{cases} \]  

(16.142)

where \( O_k (P_{b_i}) \) is the user’s utility (in the range of \([0, 1]) as a function of the received rate, which will be quantified later in this section, \( q \) is a constant factor mapping this utility value to a money amount, and \( W_a \) (or \( W_b \)) is the user’s payoff when the request is admitted (or rejected).

Similarly to the power control game discussed in Section 16.1, a sigmoid-like function has been used [52] to approximate the user’s satisfaction with respect to service qualities or resource allocation. Here the sigmoid function is modified to incorporate the user’s utility with respect to \( P_{b_i} \). The utility of class \( i \) user \( O_i (P_{b_i}) \) is given by:

\[ O_i (P_{b_i}) = \frac{1}{1 + e^{-\alpha_i (\delta_i - P_{b_i})}}, \]  

(16.143)

where \( \alpha_i \) (range 20–50) and \( \beta_i \) (range 0.1–0.2) are parameters that can be tuned to customize the utility for a given class of user. Given the fact that the more the user is satisfied, the less likely he is...
going to churn, the churn rate is modelled as:

\[
R_i(Pb_i) = 1 - O_i(Pb_i) = 1 - \frac{1}{1 + e^{-\alpha_i(b_i - Pb_i)}},
\]

(16.144)

### 16.13.2 Equilibrium solutions

When the packet blocking probability \(Pb_i = 0\), \(\forall i \in \{1, \ldots, K\}\), the strategy pair \(SS_1, CS_2\) = {accept, stay} is a Nash equilibrium of the game, i.e. the service provider chooses to admit the session request while the user chooses to remain with the provider. That comes from the fact that when \(Pb_i = 0\), \(\forall i \in \{1, \ldots, K\}\), the churn rate \(R_i(Pb_i) \approx 0\) and \(a_{12} = U + C_k\), \(a_{11} = U + C_k - L_k\), and \(a_{22} = U\). In the provider’s payoff matrix \(A = [a_{ij}]_{2 \times 2}\), \(a_{12} > a_{11}, a_{12} > a_{22}\), and \(a_{12} > a_{21}\). Because we have \(R_k(Pb_k) \approx 0\), the user’s payoff matrix \(B = [b_{ij}]_{2 \times 2}\) becomes:

\[
B = \begin{cases} 
  b_{11} = w_1W_a - w_2L_c & b_{12} = w_1W_a \\
  b_{21} = w_1W_b - w_2L_c & b_{22} = w_1W_b 
\end{cases}.
\]

(16.145)

Since \(b_{12} > b_{11}\) because \(W_a > W_b\), as argued while formulating the game, \(b_{12} > b_{22}\) and \(b_{12} > b_{21}\). Given that \(a_{12}\) and \(b_{12}\) are the maximum elements in matrices \(A\) and \(B\), respectively, we conclude that strategy pair \(SS_1, CS_2=\{\text{accept, stay}\}\) constitutes the Nash equilibrium to this game, by definition. At the same time, the remaining three strategy pairs \(SS_1, CS_1\), \(SS_2, CS_1\), and \(SS_2, CS_2\) do not yield an equilibrium.

In the case where \(Pb_i \neq 0, \exists i \in \{1, \ldots, K\}\) there exists a pure strategy Nash equilibrium, but the equilibrium point depends on values of certain terms in the payoffs. A pure strategy defines a specific move or action that a player will follow in every possible attainable situation in a game. Such moves may not be random or drawn from a distribution, as in the case of mixed strategies. If every player plays a pure strategy, the payoffs to all the players are deterministic.

In this context we need the concept of dominant (or dominating) strategy in game theory. In a bimatrix game defined by two \(m \times n\) matrices \(A\) and \(B\), which are the payoffs of players \(P_1\) and \(P_2\), respectively, for player \(P_1\), we say that ‘row i’ dominates ‘row k’ if \(a_{ij} \geq a_{kj}\), for \(j = 1, \ldots, n\). Here, ‘row i’ is called a dominant strategy for player \(P_1\) and ‘row j’ is called a dominated strategy for player \(P_1\). The user \(P_1\) is equal or better off when he selects the dominating ‘row i’ compared with selecting the dominating ‘row k’. So, ‘row k’ can actually be removed from the game because \(P_1\), as a rational player, would not consider this strategy at all.

When \(Pb_i \neq 0, \exists i \in \{1, 2, \ldots, K\}\) the equilibrium strategy pair \(\{SS_i, CS_j\}\) is defined as follows:

\[
i = \begin{cases} 
  1 & \text{if } C_k \geq \sum_{i=1}^{K} N_iR_i(Pb_i)L_i \\
  2 & \text{otherwise}
\end{cases},
\]

(16.146)

\[
J = \begin{cases} 
  1 & \text{if } \{i = 1 \text{ and } b_{11} \geq b_{12}\} \text{ or } \{i = 2 \text{ and } b_{21} \geq b_{22}\} \\
  2 & \text{if } \{i = 1 \text{ and } b_{11} < b_{12}\} \text{ or } \{i = 2 \text{ and } b_{21} < b_{22}\}
\end{cases}
\]

If \(C_k \geq \sum_{i=1}^{K} N_iR_i(Pb_i)L_i, a_{11} > a_{21}\) and \(a_{12} > a_{22}\). By definition, \(SS_1\) is the dominating strategy and \(SS_2\) is the dominated strategy for the service provider. Since \(SS_2(\text{reject})\) can be eliminated, \(A\) and \(B\) degenerate into two \(1 \times 2\) matrices. Obviously, if \(b_{11} > b_{12}\), the user will choose strategy \(CS_1(\text{leave})\) and if \(b_{11} < b_{12}\), he will choose \(CS_2(\text{stay})\). When \(b_{11} = b_{12}\), the strategies \(CS_1\) and \(CS_2\) have no difference to the user. Without loss of generality, we assume the user chooses \(CS_1\). Hence, the equilibrium strategy pair is \(\{SS_1, CS_1\} = \{\text{accept, leave}\}\) if \(b_{11} \geq b_{12}\) or \(\{SS_1, CS_2\} = \{\text{accept, stay}\}\) if \(b_{11} < b_{12}\).

If \(C_k < \sum_{i=1}^{K} N_iR_i(Pb_i)L_i, a_{11} < a_{21}\) and \(a_{12} < a_{22}\). By definition, \(SS_2 (\text{reject})\) is the dominating strategy and \(SS_1 (\text{stay})\) is the dominated strategy for the service provider. Following the same logic as above, the equilibrium strategy pair is given by \(\{SS_2, CS_1\} = \{\text{reject, leave}\}\) if \(b_{21} \geq b_{22}\) or \(\{SS_2, CS_2\} = \{\text{reject, stay}\}\) if \(b_{21} < b_{22}\).
The rationale behind these decisions is that, if the revenue \( C_k \) generated from accepting the new session request belonging to a class \( k \) user is greater than the possible revenue loss \( \sum_{i=k}^{K} N_i R_i (P_{bi}) L_i \) from user churning, then admitting the session is a better strategy.

Simulation results in [52] show that the strategy presented in this section can provide up to a 50% increase in a service provider’s revenues compared with the conventional approach, where the user’s request is accepted as long as the total number of users does not exceed the total number of users the system can support. For more information on this topic see [53–60].

16.14 MODELLING ACCESS POINT PRICING AS A DYNAMIC GAME

16.14.1 The system model

The interaction between an access point and user can be modelled using a simple two-player game model. The game progresses in discrete time slots or ‘periods.’ In general, the access point offers connectivity at the beginning of time slot \( t \) at price \( p_t \). The game ends the first time the user rejects the access point’s proposal. The user’s utility function \( F(\hat{T}, \tau) \) is a function of the number \( T \) of time slots the user chooses to connect and a parameter \( \tau \) which we call the user’s intended session length. \( T \) is a decision variable; it is a function of the actions the user takes in each time slot, specifically the number of slots the client chooses to connect. In contrast \( \tau \) is a type variable that specifies the maximum time the user would be interested in connecting. The user does not choose \( \tau \) in the game, but instead it is determined for the user by outside circumstances. The user knows the value of \( \tau \) at the beginning of the game while the access point only knows its probability distribution. The user’s net payoff is \( F(\hat{T}, \tau) - \sum_{t=1}^{\hat{T}} p_t \), while the access point’s net payoff is \( \sum_{t=1}^{\hat{T}} p_t \). The underlying assumption is that the access point’s marginal cost to provide the service to the client is negligible. In the model, we use dynamic Bayesian game already discussed in Section 16.8.

In general, the strategy specification of a dynamic Bayesian game is a mapping from a player’s type, and a player’s information set to an action, or in a mixed strategy, to a distribution among possible actions. Assuming players have ‘perfect recall,’ the information set is the history of actions the player has observed. In our model, if the game reaches slot \( t \), then the history is completely specified by the previous prices charged. Thus, a pure strategy for an access point is simply a price sequence. An access point’s mixed strategy, or behaviour strategy, is a probability distribution on prices to charge in each slot \( t \) dependent upon the prices actually charged in earlier slots \( 1, \ldots, t - 1 \). For the user, a strategy is specified as a mapping from its type and information set to an acceptance decision, or to a probability of accepting.

For a user browsing the web the user’s utility is proportional to the length of time \( T \) that he gets to browse the web, but his utility saturates after the maximum intended session length \( \tau \) is reached. Thus, his utility function \( F(\hat{T}, \tau) \) is written as:

\[
F(\hat{T}, \tau) = U \cdot \min(\hat{T}, \tau)
\]  

(16.147)

The user’s type is specified by his utility per slot \( U \) and intended session length \( \tau \). The user knows the values for \( U \) and \( \tau \) while the access point just knows their distributions. For this case the following strategy profile is a perfect Bayesian equilibrium PBE.

\[
\text{Strategy 1: The user connects or remains connected in slot } t \text{ iff } t \leq \tau \text{ and } U > p_t \text{ (‘myopic strategy’)} \text{ while the access point charges a nondecreasing sequence of prices } \{p_t\}: p_t \in \arg \max_p pP(U > p).
\]

An access point should pick its prices by considering just the prior distribution of \( U \). In fact, it is a PBE for the access point to pick a single maximizing value of \( pP(U > p) \), say \( p^* \), and charge the fixed price \( p_t = p^* \) in all time slots. Whenever a myopic client accepts price \( p_t \), the access point can refine his conditional distribution of \( U \) by lower bounding it by \( p_t \). One might have expected that
access point might want to try charging a higher price than \( p_t \) after learning that the user’s utility is at least \( p_t \). This intuition is not correct.

To prove it we first, find the access point’s optimal counter strategy to a user playing the ‘myopic strategy’ [60, 61]. A pure strategy for the access point is a sequence of nonnegative prices \( \{p_t\}_{t=1}^{\infty} \in R_+^{[1,2,\ldots]} \) used in each time slot \( t \in \{1, 2, \ldots \} \) which maximizes his expected revenue

\[
J_t^a ((p_t)) = \sum_{t=1}^{\infty} p_t P \left( U > \max_{u \in \{1, \ldots, t\}} p_u \right) P (\tau \geq t) .
\]

(16.148)

where superscript ‘\( a \)’ indicates the access point while the subscript ‘\( 1 \)’ indicates that it is the objective from slot 1 onward. We can find a maximizing sequence for Equation (16.148) even if we restrict ourselves to nondecreasing sequences, because for any sequence \( \{\tilde{p}_t\} \) for which there exists a \( u \) such that \( \tilde{p}_u < \tilde{p}_{u-1} \), we can define a new nondecreasing sequence \( \{p_t\} \) with \( p_t = \max (\tilde{p}_t, \ldots, \tilde{p}_1) \). In other words, an access point that has seen a myopic user accept a price \( p_t \) knows that he can charge at least \( p_t \) in slot \( t + 1 \) without any risk of exceeding the user’s willingness to pay. If \( S^+ \) is the set of nondecreasing price sequences; that is \( \{p_t\} \in S^+ \) if \( \{p_t\}_{t=1}^{\infty} \in R_+^{[1,\ldots]} \) and \( p_{t+1} \geq p_t \forall t \in \{1, 2, \ldots \} \), the access point should choose a positive sequence of prices to maximize his expected revenue, as:

\[
\max_{\{p_t\} \in S^+} J_t^a ((p_t)) = \max_{\{p_t\} \in S^+} \left[ \sum_{t=1}^{\infty} p_t P (U > p_t) (\tau > t) \right]
\]

(16.149)

Because \( U \) and \( \tau \) are finite mean, one can substitute their bounds into Equation (16.149) to show that the access point’s expected payoff against a myopic client is bounded [62]. Each term in the summation of Equation (16.149) is a function of a different price \( p_t \), so the entire sum can be maximized by independently maximizing each term in the summation. \( y (p) = P (U > p) \) is a continuous, nonnegative function, with \( y (0) = 0 \), and \( \lim_{p \to -\infty} y (p) = 0 \), and thus must achieve a maximum on \([0, \infty)\). Thus, if the access point chooses each \( p_t \) such that \( p_t \in \arg \max_p P (U > p) \), with \( \{p_t\} \in S^+ \), then the access point maximizes his expected payoff Equation (16.149).

It is straightforward to see that the myopic strategy is a best user’s response to an access point that never lowers prices. Because the players’ strategies are best responses to each other, they constitute a (Bayesian) Nash equilibrium strategy profile.

The strategy profiles remain best responses to each other in any continuation game, beginning at an arbitrary slot \( s \), i.e. the strategy profile is a PBE. A user facing nondecreasing prices in this game will face nondecreasing prices in any continuation game starting at slot \( s \). Thus, a user that expects nondecreasing prices should stick to the myopic strategy in the continuation game beginning at slot \( s \). An access point that expects his client to be myopic should choose his prices in the continuation game to maximize

\[
J_s^a ((p_t)_{t=s}^{\infty}) = \sum_{t=s}^{\infty} p_t P \left( U > \max_{\sigma \in \{t, \ldots, s\}} p_\sigma \right) P (\tau \geq t | \tau \geq s)
\]

(16.150)

For any price sequence \( \{\tilde{p}_t\}_{t=s}^{\infty} \) which has prices that are less than \( p_{s-1} \), we see that \( J_s^a ((\max (\tilde{p}_t, p_{s-1})_{t=s}^{\infty}) \geq J_s^a ((\tilde{p}_t)_{t=s}^{\infty}) \). Thus, the access point can maximize its reward by selecting its continuation game prices to be no smaller than \( p_{s-1} \). Thus, assuming \( p_\sigma \geq p_{s-1} \) for all \( \sigma \geq s \), we may write:

\[
J_s^a ((p_t)_{t=s}^{\infty}) = \frac{1}{p(U > p_{s-1})} \times \sum_{t=s}^{\infty} p_t P \left( U > \max_{\sigma \in \{t, \ldots, s\}} p_\sigma \right) \frac{1}{P (\tau \geq s)} P (\tau \geq t)
\]

(16.151)

Note that Equation (16.151) has a structure that parallels that of Equation (16.148), with the exception of the scaling factors \( 1/P (U > p_{s-1}) \) and \( 1/P (\tau \geq s) \), which have no dependence on the prices chosen from slot \( s \) forward. Thus, the same argument that was used to show that Equation (16.148) is maximized with a nondecreasing sequence with elements in \( \arg \max_p pP(U > p) \), can be used
to show that Equation (16.151) is also maximized with a nondecreasing sequence with elements in \( \arg \max_p p P(U > p) \). This shows that the access point strategy described in Strategy 1 remains a best response to a myopic client in any continuation game.

### 16.14.2 Modelling service reselling

When the user is not within the range of the access point it begins a session by sending a request for service to the reseller (relay). In order to serve the user’s request, the intermediate node sends its own request for service to the ‘root’ access point. As a response, the root access point passes the reseller a first slot price \( c_1 \), which the reseller can either accept (and game continues) or reject (and game stops). Before deciding to accept or reject the \( c_1 \) price, the reseller sends its first slot price \( p_1 \) to the user. If the user accepts \( p_1 \), the reseller in turn accepts the \( c_1 \) price from the root access point. If the user rejects \( p_1 \), and assuming that the reseller had no other use for connectivity with the access point than to serve the user, the reseller rejects the \( c_1 \) price and the game ends. If the user and reseller accept their offers in slot \( t \), the game continues into slot \( t + 1 \) with the root access point choosing a price \( c_{t+1} \) to offer the reseller, and the reseller choosing a price \( p_{t+1} \) to offer the user. The payoffs are

\[
F(T, \tau) - \sum_{t=1}^{T} p_t \text{ for the user, } \sum_{t=1}^{T} (p_t - c_t) \text{ for the reseller, and } \sum_{t=1}^{T} c_t \text{ for the access point.}
\]

In general, a pure strategy for the reseller is a mapping from prices he has been charged \( c_1, \ldots, c_T \), as well as the prices the reseller charged in the past, \( p_1, \ldots, p_{t-1} \), to a price to charge in the current slot, \( p_t \). If we denote the reseller’s history as \( h'_t \equiv (\{c_u\}_{u=1}^t, \{p_u\}_{u=1}^{t-1}) \) then a pure strategy for the reseller is a specification of the functions \( p_t(h'_t) \) for \( t = 1, 2, \ldots \). The following strategy profile for the tree parties is a PBE.

**Strategy 2:**

(i) the user connects iff \( t \leq \tau \) and \( U > p_t \);

(ii) the reseller uses \( p^*(c) \) that satisfies the properties

\[
p^*(c) \in \arg \max_{p} (p - c) P(U > p) \text{ with } p^*(c') \geq p^*(c) \quad \forall c' > c \text{ and charges the price } p_t(h'_t) := p^*(c_t) \text{ in slot } t;
\]

(iii) the access point charges a non decreasing price sequence \( \{c_t\} \) with \( c_t \in \arg \max_c [c \cdot P(U > p^*(c))] \)

The proof that Strategy 2 is PBE is based on similar arguments to those used in Section 16.14.1. The details can be also found in [60, 61].

### 16.14.3 File transfer model

This model refers to a situation in which a user is downloading a file, and the user must remain connected for the entire duration of the file (the intended session length), to earn any utility for the file. So, the user’s utility function has the form

\[
F(T, \tau) = \begin{cases} 
0 & \text{if } T < \tau \\
U \tau & \text{if } T = \tau
\end{cases}
\]  

(16.152)

If we have a system with:

(i) the user has a file transfer utility function as in Equation (16.152), with \( U \) distributed on \([l, h]\) with \( 0 \leq l < h \), and the session length \( \tau \), distributed on \([1, \ldots, n]\);

(ii) both \( U \) and \( \tau \) have sample values known to the client, and unknown to the access point;
(iii) $U$ and $\tau$ are finite mean, and that $U$ is continuously distributed.

(iv) then the Strategy 3, defined bellow has perfect Bayesian equilibria:

\begin{quote}
Strategy 3: The user accepts slot $t < \tau$ iff $p_t = 0$. When $t = \tau$, the client connects if he had been connected in all of the previous slots and $U \tau > p_t$ (he never connects if $U \tau < p_t$ but may connect if $U \tau = p_t$).

The access point charges

$$p_t = \begin{cases} 
0 & \text{if } t < t^* \\
u^*t^* & \text{otherwise} 
\end{cases}$$

where $(u^*, t^*) \in \arg \max_{(u, t)} utP(U > u, \tau = t)$
\end{quote}

The proof uses backward induction, and iterated deletion of dominated strategies as in Section 16.13.2. The details can be found in [60, 61].

16.14.4 Bayesian model for unknown traffic

In this section, we study the case in which the access point does not know if his user is a file transferer or a web browser. Instead, the access point begins the game knowing the prior probability $x$ that the user is a file transferer. The user knows his true type as well as the value of $x$. We call this combined model simply the Bayesian model [62].

When $x$ is 0, the Bayesian model is equivalent to a ‘pure’ web browsing model, so constant price is a PBE by Strategy 1. When $x$ is 1, the Bayesian model is equivalent to a ‘pure’ file transfer model, where we know that constant price is not a PBE. Under the

Assumptions 1

(i) the user is a file transferer (FT) with probability $x$ and a web browser (WB) with probability $1 - x$;

(ii) the access point knows the value of $x$, while the user knows her true type;

(iii) users of both types have an intended session length of $\tau = 2$, and this is known to both parties;

(iv) the utility per slot $U$ is uniformly distributed on $[0, 1]$;

(v) the user knows the sample value of $U$ while the access point (AP) knows only the distribution;

(vi) the utility functions for WB and FT type clients are given by Equations (16.147) and (16.152) respectively;

the following is true for the group of strategies:

Strategies 4: The (anticipative) strategy profile (pair of player strategies), $s^* = (s^*_AP, s^*_C)$ is a PBE $\forall x \in [0, 0.516]$, where the access point strategy, $s^*_AP$, and the type dependent client (user) strategy, $s^*_C$, are defined as follows:

\begin{quote}
Strategy 4a. $s^*_AP$: Charge the price sequence $\{p^*_1, p^*_2\}$ in the two slots of the game, where the prices $p^*_1$ and $p^*_2$ are dependent on $x$ as

$$p^*_1(x) = \frac{4 - 5x}{2(1 - x)(4 - x)} \quad \text{and} \quad p^*_2(x) = \frac{4 - 3x}{2(1 - x)(4 - x)}$$

4a(i) $s^*_C$ (WB): (WB clients) connect in slot 1 iff $p_1 < U$; connect in slot 2 iff connected in slot 1 and $p_2 < U$, which is a myopic strategy;

4a(ii) $s^*_C$ (FT): (FT clients) connect in slot 1 iff $p_1 + \tilde{p}_2 < U$ where $\tilde{p}_2$, the price the client expects intuitively in slot 2, is equal to $p^*_2$; connect in slot 2 iff connected in slot 1 and $p_2 < 2U$.
\end{quote}
The (pessimistic) strategy profile, $s^p = (s^p_{ AP}, s^p_C)$ is a PBE $\forall x \in [(3 - \sqrt{5})/2, 0.382, 1]$, where the player strategies are:

Strategy 4b. $s^p_{ AP}$: charge the price sequence $\{0, (1/(2 - x))\}$;

Strategy 4b(i) $s^p_C (WB)$: the myopic strategy $s^p_C (WB) = s^*_C (WB)$;

Strategy 4b(ii) $s^p_C (FT)$: the pessimistic strategy-connect in slot 1 iff $p_1 = 0$ and connect in slot 2 iff connected in slot 1 and $2U > p_2$ (the superscript $p$ in $s^p$ means pessimistic strategy profile).

Strategy 4c. For $x > 0$ there are no PBE in which the AP charges a constant price ($p_1 = p_2$).

To prove the above statements we first consider whether or not the client’s strategy $s^*_C$ is a best response to the access point strategy $s^*_{ AP}$. The access point prices are nondecreasing in $s^*_{ AP}$, and we have seen that a web browser’s best response to nondecreasing prices is a myopic strategy, so $s^*_C (WB)$ is a best response. Similarly, when the AP plays $s^*_{ AP}$, playing $s^*_C (FT)$ gives the highest possible payoff to a FT user for all possible values of $U$. Furthermore, a FT user does not benefit by unilaterally deviating in the continuation game beginning in slot 2.

Next we consider whether $s^*_{ AP}$ is a best response to $s^*_C$. We begin by expressing the AP revenue $R (p_1, p_2)$, if users play $s^*_C$, as:

$$R (p_1, p_2) = p_1 \left[ (1 - x) G (p_1) + x G \left( \frac{p_1 + p_2^* (x)}{2} \right) \right] + p_2$$

$$\times \left[ (1 - x) G (\max [p_2, p_1]) + x G \left( \max \left[ \frac{p_1 + p_2^* (x)}{2}, \frac{p_2}{2} \right] \right) \right]$$

(16.153)

where $G(p) = P (U > p) = \max ((1 - p), 0)$. For each $x$, $R (p_1, p_2)$ is piecewise quadratic in $(p_1, p_2)$ with regions divided by the lines $p_1 = p_2, p_1 + p_2^* (x)$. For instance in the region $p_2 \geq p_1$ and $p_2 \leq p_1 + p_2^* (x)$:

$$R (p_1, p_2) = (x - 1) p_1^2 + x p_2^2 \left( 1 - \frac{x p_2^* (x)}{2} \right) (p_1 + p_2) - \frac{x}{2} p_1 p_2$$

(16.154)

The maximum of $R (p_1, p_2)$ in this region occurs at $(p_1^* (x), p_2^* (x))$, indicated in Strategies 4a, and this point is also the global maximum across all the regions, whenever $x \in [0, 0.516]$ and where 0.516 is a decimal approximation to the root of a 6th order polynomial. Because the AP’s best response prices match the prices the users anticipated, we have found a PBE. However for values of $x$ larger than this root, the global maximum occurs in the region where $p_2 \geq p_1 + p_2^* (x)$. Consequently the AP wants to charge more in slot 2 than the user anticipated, and hence the strategy profile is not a PBE for $x > 0.516$.

Next we show that $s^p$ (Strategy 4b) is a PBE. Under the prices of $s^p_{ AP}$, users following $s^p_C$ connect whenever connecting will result in a positive payoff, and do not connect otherwise. Thus, $s^p_C$ is a best response to $s^p_{ AP}$, and furthermore the user’s best response in the continuation game beginning in slot 2 is to not deviate from $s^p_C$.

With users playing $s^p_C$, an AP can charge 0 in the first slot, and keep the FT clients connected, or choose a nonzero price and earn revenue from only the WB clients. If the AP chooses the latter option, then he should maximize expected revenue from WB clients, which he can do by charging 1/2 in both slots, earning him an expected revenue of $(1 - x)/2$. If the AP chooses 0 for its slot 1 price, then both FT and WB clients would be potential customers in the second slot, and the AP’s optimal second slot price is found by maximizing:

$$p_2 (1 - x) P (U > p_2) + p_2 x P (U > \frac{p_2}{2})$$

(16.155)

which has maximum at $p_2 = 1/(2 - x)$, earning an expected revenue of $1/(4 - 2x)$. The latter option of charging $\{0, (1/(2 - x))\}$, which is the same as the $s^p_{ AP}$ strategy defined in Strategy 4b, earns more expected revenue than does the option of charging 1/2 in each slot for $x \in [(3 - \sqrt{5})/2, 1]$. Thus, $s^p_{ AP}$ is a best response to $s^p_C$ for $x \in [(3 - \sqrt{5})/2, 1]$. 


REFERENCES


## Index

### A
- Adaptation/reconfiguration efficiency, 25
- Adaptive
  - 4G Networks, 753
  - coding, 3
    - reconfigurable block coding, 21
    - reconfigurable convolutional codes, 26
    - Punctured convolutional codes/code, 31
    - Viterbi algorithm, 32
  - Systematic recursive convolutional codes, 33
  - Concatenated codes with interleavers, 36
  - The iterative decoding algorithm, 37
  - practice and prospects, 43
  - Distributed source coding, 44
  - Continuous valued source, 46
  - Scalar quantization and trellis-based coset construction, 48
  - Trellis-based quantization and memoryless, 50
    - filtering, 225, 574
    - modulation, 77
    - precombining LMMSE receivers, 225
    - reconfigurable transmit power allocation, 136
    - weight adjustment, 276
  - ADC, 5, 546, 558, 584
  - Advanced time division multiple access (ATDMA), 5
  - Aggregation, 1, 3, 44, 830, 831, 852
  - Alamouti scheme, 100, 107, 171, 600
  - ALOHA, 4, 753, 791, 839, 840, 841, 842, 853
  - Autocorrelation function, 462, 494
  - AWGN channel, 26, 73, 166, 211, 332, 343, 551, 625, 626
  - Azimuth profile, 709

### B
- Bartlett estimate, 235
- Base station, 581
- Bayesian receiver, 400
- BCJR algorithm, 53, 56, 358, 419
- Beamforming, 3, 183, 189, 460, 464, 505, 509, 513, 515, 520, 523, 525, 534, 535, 536, 682, 715, 716, 749
- UWB radio, 3
- Bessel function, 86, 149, 281, 383, 385, 473
- Bezout identity, 497
- Bhattacharyya bound, 457
- Binary shift register sequences, 191
- Bit rate adaptation, 756
- Blind
  - Bayesian multiuser receiver, 400
  - equalizers, 278
  - least squares receivers, 230
  - turbo multiuser receiver, 403
INDEX

built-in time diversity, 141, 142
Bussgang, 316, 317, 318, 324, 328

C
Capacity, 77, 163, 186, 590, 621, 697, 698, 716, 719, 720, 749, 795
Cascading fixed coefficient filters, 574
CDMA multipath channel, 214
Channel estimator, 80, 82, 106, 226, 292, 337, 363, 548
Channel model, 279, 308, 536, 588, 727, 749
Macrocellular environments (1.8 GHz), 675
PDF of shadow fading, 677
Urban spatial radio channels in macro/microcell (2.154 GHz), 681
Description of environment, 682
Results, 682
MIMO channels in microcell and picocell environments (1.71/2.05 GHz), 688
Simulation of channel coefficients, 690
Measurement setups, 690
Validation of the stochastic MIMO channel model assumptions, 690
Input parameters to the validation of the MIMO model, 692
The eigenanalysis method 693
Outdoor mobile channel (5.3 GHz), 696
Path loss models, 700
Window length for averaging fast fading components at 5 GHz, 702
Spatial and frequency correlations, 702
Path number distribution, 705
Rotation measurements in an urban environment, 706
Microcell channel (8.45 GHz), 708
Azimuth profile, 709
Delay profile for the forward arrival waves, 710
Short-term azimuth spread (AS) for forward arrival waves, 712
Wireless MIMO LAN environments (5.2 GHz), 714
Data evaluation, 715
Capacity computation, 716
Measurement environments, 717
Indoor WLAN channel (17 GHz), 718
Indoor WLAN channel (60 GHz), 727
Definition of the statistical parameters, 728
UWB channel model, 732
The large-scale statistics, 736
The small-scale statistics, 739
Correlation of MPCs among different delay bins, 741
The statistical model, 741
Chebyshev filter, 281
Chernoff bound, 165
Cholesky decomposition, 214, 282, 310
Circular array, 467
CLSP/DS-CDMA packet access and traffic model, 756
Clustering models for the indoor channel, 438
Code design, 142
Coding, 5, 21, 56, 57, 58, 59, 76, 77, 154, 185, 269, 352, 534, reconfigurable block coding, 21
reconfigurable convolutional codes, 26
Punctured convolutional codes/code, 31
Viterbi algorithm, 32
Systematic recursive convolutional codes, 33
Concatenated codes with interleavers, 36
The iterative decoding algorithm, 37
practice and prospects, 43
Distributed source coding, 44
Continuous valued source, 46
Scalar quantization and trellis-based coset construction, 48
Trellis-based quantization and memoryless, 50
Coding gain, 154
Coding gain properties, 154
Cognitive networks, 1, 4, 835, 836
Cognitive power control, 4
cognitive radio, 1, 2, 4, 537, 557, 558, 561, 595, 602, 627, 643, 797
Cognitive Radio, 537
Energy-efficient cognitive radio, 537
Frame length adaptation, 537
flat fading channels, 539
The adaptation algorithm, 542
Adaptive error control, 542
Processing gain adaptation, 545
Maximum likelihood sequence equalizer, 547
Hidden Markov channel model, 548
Link layer performance
INDEX

857

inadequate equalization, 549
adequate equalization, 551
Linear multiuser detection, 556
A unified architecture and dynamic reconfigurability, 556
The effects of quantization, 564
The effect on the ‘near–far’ resistance, 565
Reconfigurable ASIC architecture, 567
present art, 569
Alternative implementations, 570
Example architecture versus an FPGA, 570
DSP against the example architecture, 571
complex 16-point DFT – the Goertzel FFT mode, 571
Fixed coefficient filters, 573
Real FIR/correlator, 574
Real IIR/correlator, 574
Cascading fixed coefficient filters, 574
Adaptive filtering, 574
Direct digital frequency synthesis, 576
CORDIC algorithm, 577
3 Discrete Fourier transform, 578
Goertzel algorithm, 578
Cognitive Wireless Networks, 587
Collusion of nodes, 836
Complex orthogonal designs, 100
Conditional posterior distributions, 401
Constant modulus algorithm, 229, 230, 232, 339
Constellation, 341, 356
Constellation rotation, 167
Contrast function, 319
Convex optimization, 505, 506, 513, 527, 534, 535
Convex optimization theory, 504
Generalized MIMO systems, 505
Convex optimization, 506
Power optimization, 507
SINR optimization, 510
Performance example, 512
Beamforming, 513, 520
Multicarrier MIMO signal model, 514
Channel diagonalization, 516
Constraints in multicarrier systems, 526
Convolution, 533
Convolutional encoder, 26, 28
Cooperative diversity, 608, 612, 587
System modeling, 587
System capacity, 588
Probability of outage, 591
Cellular coverage, 592
Protocols, 593
System and channel models, 593
Cooperative diversity protocols, 594
Outage probabilities, 595
Performance bounds, 598
Distributed space–time coding, 600
System description, 600
BER analysis in DSTC, 603
Generalization, 605
System and channel model, 605
Cooperative diversity based on repetition, 608
space–time coding, 612
Crosscorrelation spectra, 193

D

Decision
feedback equalizer (DFE), 277
metric, 108, 135
statistics combining (DSC), 249
Decoder-assisted convergence assessment, 404
Decorrelating detector, 208
Deinteleaver, 303, 305
Delay diversity, 94, 118, 120
Delayed decision feedback sequence estimation (DDFSE), 360
Description of environment, 676, 682
DFE adaptive algorithms, 315
Differential decoding, 123, 128, 129
Differential encoding, 126
Direct digital frequency synthesis, 576, 585
Discrete Fourier transform, 569, 578
Distributed mobile networks, 775
Distributed network routing protocol, 773
Distributed Source Coding, 45, 47, 49, 51
Diversity gain, 136, 153
Diversity gain properties, 153
Diversity receiver, 335
Doppler frequency, 145, 149, 150, 151, 338, 339, 346, 348, 387, 393, 417, 552, 554, 555, 758, 760, 764
DPSK, 121, 125, 206, 338, 339, 342, 426, 427, 798

E

EDGE, 2, 5, 181, 189, 325, 356, 359, 364, 366, 367, 368, 427, 428, 672
Edgeworth expansion, 318
Effective length, 141
Eigenvalues and eigenvectors, 258
Energy-efficient adaptive error control, 542
Entropy, 45, 311, 313, 314, 315, 318
Equalization, 57, 269, 318, 322, 323, 325, 360, 368, 533, 547
Zero-forcing equalizers, 269
LMS equalizer, 275
Signal model, 275
Adaptive weight adjustment, 276
Automatic systems, 276
Iterative algorithm, 277
The LMS algorithm, 277
Decision feedback equalizer (DFE), 277
Blind equalizers, 278
Statistically known, time varying channel, 279
Signal model, 279
Channel model, 279
Description of the received sequence, 281
The ML sequence (block) estimator for a statistically known channel, 281
LMS-adaptive MLSE equalization on multipath fading channels, 284
System and channel models, 284
Adaptive channel estimator and LMS estimator model, 285
The channel prediction algorithm, 285
Adaptive channel identification, 288
System model, 288
Joint channel and data estimation, 288
fading channel, 292
The static channel environment, 293
The time varying channel environment, 296
Turbo equalization, 301
Signal format, 301
Equivalent discrete time channel model, 302
Equivalent system state representations, 302
Turbo equalization, 302
Viterbi algorithm, 303
Iterative implementation, 304
Kalman Filter, 305
Channel model, 308
The received signal, 308
Channel estimation alternatives, 308
Implementing the estimator, 309
The Kalman filter, 310
Implementation issues, 310
Equalization using higher order signal statistics, 311
Problem statement, 311
Signal model, 313
Derivation of algorithms for DFE, 313
The equalizer coefficients, 314
Stochastic gradient DFE adaptive algorithms, 315
Convergence analysis, 316
Kurtosis analysis, 318
Equivalent discrete time channel model, 302
system state representations, 302
Error rate bound, 141
Error rate models, 789
ESPRIT DOA, 3
Estimation of MIMO channel, 3
Euclidean distance, 44, 64, 65, 66, 71, 141, 142, 166, 359, 422, 447
Extrinsic information, 303
Extrinsic information ratio (EIR), 248, 252
F
F/B MMSE receiver, 251
Feedback, 72, 76, 188, 312, 316, 322, 323, 325, 326, 378, 379, 576, 622
FFT block, 7, 415
FIR channel estimator, 338
Fixed coefficient filters, 573
Fixed-point iteration, 510, 512
Flexible spectra sharing, 4
FPGA, 557, 558, 559, 560, 561, 562, 563, 566, 570, 571, 572, 581
Frame length adaptation, 537, 539
Frequency selective channel model, 356
Frequency selective fading channels, 397
Frobenius, 514
Full spatial diversity, 116
G
Game Theory, 797, 851, 853, 854
Cognitive Networks and Game Theory, 797
Cognitive power control, 797
Noncooperative power control game, 797
Nash equilibrium, 799
Pareto optimality, 800
Supermodular games and social optimality, 801
Power control game with QOS guarantee, 805
MIMO systems, 811
MAC for AD HOC networks, 813
Tit-for-Tat (TFT) packet forwarding, 815
Strategy models, 815
Network nodes dependency graph and system metamodel, 817
The payoff of iterative game, 819
TFT node cooperation with energy constraint, 823
Acceptance rate, 823
Pareto optimum, 823
Prisoner’s dilemma and TFT game, 825
Packet forwarding/dynamic Bayesian games, 828
Routing in wireless sensor networks, 830
Cognitive wireless sensor network model, 830
Optimal rout computation, 832
Profit driven routing in cognitive networks, 832
Algorithmic mechanism design, 832
Profit driven pricing mechanism, 833
Truthful behavior in cognitive networks, 835
Collusion of nodes in cognitive networks, 836
Flexible spectra sharing, 838
Slotted ALOHA protocol, 839
Admission in competitive wireless networks, 842
System model, 842
Equilibrium solutions, 845
Access point pricing as a dynamic game, 846
The system model, 846
Modelling service reselling, 848
File transfer model, 848
Bayesian model for unknown traffic, 849
Gamma distribution, 739, 740, 744
Gcd, 193, 194, 195
Generalized complex orthogonal designs, 100
real orthogonal designs, 99
Gibbs multiuser detection, 402
sampler, 400, 401, 403, 404, 405, 430
Givens rotation, 173
Goertzel algorithm, 578, 579
Golay code, 199
Gold sequences, 194, 195, 210, 387
Gray mapping, 66, 121, 122
Grimm, Fitz, and Krogmeier (GFK), 119
Group interference suppression, 134, 139
GSM, 4, 5, 181, 189, 325, 580, 625, 671, 672, 673, 746, 748
H
Hadamard sequence, 200
Hamming weight, 192
Hermitian, 134, 148, 153, 155, 158, 176, 281, 310, 506, 507, 514, 518, 658, 693, 717
Hidden Markov channel model, 548
HIPERLAN, 9, 16, 528, 536
Horizontal layered space–time (HLST) architecture, 239
Householder transformation, 173
I
IBI, 360, 362, 600, 602
IC, 246, 545, 584, 832
IFFT, 8, 329, 336, 341, 344, 351, 353, 361, 362, 368, 396, 411, 421, 422, 569
IIR, 324, 567, 569, 574
In-home channel, 441
Input parameters, 692
Interarrival times, 440
Interference cancellation, 353, 429
Interleaver, 37, 140, 303, 305, 378, 382, 419
inverse fast Fourier transform, 411, 421
Iterative algorithm, 277
Iterative implementation of turbo equalization, 304
J
Jacobian, 808
Jakes model, 150, 387, 417
JPL sequences, 197
K
Kalman filter, 3, 308, 309, 310, 327, 426
Channel model, 308
The received signal, 308
Channel estimation alternatives, 308
Implementing the estimator, 309
The Kalman filter, 310
Implementation issues, 310
Kasami sequences, 196
Kronecker product, 153, 690
sequences, 197
Kurtosis-based algorithm, 318
L
Lagrange multipliers, 180
layered space–time (LST) architecture, 236
Least resistance routing, 778
Least squares (LS) receiver, 230
Linear processing orthogonal designs, 98
Link layer
adequate equalization, 551
inadequate equalization, 549
<table>
<thead>
<tr>
<th>LMS algorithm</th>
<th>227, 230, 277, 309, 321, 572, 574, 629, 630, 632, 633, 634, 636, 638, 639</th>
</tr>
</thead>
<tbody>
<tr>
<td>equalizer</td>
<td>276, 277</td>
</tr>
<tr>
<td>estimator model</td>
<td>285</td>
</tr>
<tr>
<td>LST architectures</td>
<td>237, 239</td>
</tr>
<tr>
<td>LSTC receivers</td>
<td>247</td>
</tr>
<tr>
<td>M</td>
<td></td>
</tr>
<tr>
<td>Mapping</td>
<td>573, 576, 577</td>
</tr>
<tr>
<td>Maximum likelihood (ML)</td>
<td>32, 237, 241</td>
</tr>
<tr>
<td>Measurement setups</td>
<td>690</td>
</tr>
<tr>
<td>capacity</td>
<td>188</td>
</tr>
<tr>
<td>channels</td>
<td>4, 186, 231, 497, 533, 534, 535, 536, 748</td>
</tr>
<tr>
<td>OFDM</td>
<td>351, 353</td>
</tr>
<tr>
<td>Minimum power networks</td>
<td>772</td>
</tr>
<tr>
<td>Mismatch</td>
<td>107</td>
</tr>
<tr>
<td>ML sequence (block) estimator</td>
<td>281</td>
</tr>
<tr>
<td>MLSE</td>
<td></td>
</tr>
<tr>
<td>equalization</td>
<td>3, 321, 325, 427</td>
</tr>
<tr>
<td>channel estimation</td>
<td>335</td>
</tr>
<tr>
<td>detection</td>
<td>3, 244, 265, 387, 389, 429, 430</td>
</tr>
<tr>
<td>detector structures</td>
<td>220</td>
</tr>
<tr>
<td>receiver</td>
<td>245, 252, 265, 383, 384, 430, 512, 558, 560, 564, 626, 810, 812</td>
</tr>
<tr>
<td>SIC detector</td>
<td>244</td>
</tr>
<tr>
<td>Modulation</td>
<td>63</td>
</tr>
<tr>
<td>Coded modulation</td>
<td>63</td>
</tr>
<tr>
<td>Euclidean distance</td>
<td>64</td>
</tr>
<tr>
<td>Examples of TCM schemes</td>
<td>65</td>
</tr>
<tr>
<td>Set partitioning</td>
<td>68</td>
</tr>
<tr>
<td>Representation of TCM</td>
<td>70</td>
</tr>
<tr>
<td>TCN with multidimensional constellation</td>
<td>70</td>
</tr>
<tr>
<td>Adaptive coded modulation for fading</td>
<td>72</td>
</tr>
<tr>
<td>channels</td>
<td></td>
</tr>
<tr>
<td>Maintaining a fixed distance</td>
<td>73</td>
</tr>
<tr>
<td>Information rate</td>
<td>74</td>
</tr>
<tr>
<td>Monte Carlo simulation</td>
<td>51, 156, 694, 746</td>
</tr>
<tr>
<td>Moore-Penrose pseudoinverse</td>
<td>506, 715, 749</td>
</tr>
<tr>
<td>Multicarrier CDMA system</td>
<td>369</td>
</tr>
<tr>
<td>Data demodulation</td>
<td>370</td>
</tr>
<tr>
<td>Performance examples</td>
<td>371</td>
</tr>
<tr>
<td>broadcast systems</td>
<td></td>
</tr>
<tr>
<td>Frame by frame adaptive rate</td>
<td>375</td>
</tr>
<tr>
<td>Transmitter</td>
<td>377</td>
</tr>
<tr>
<td>Receiver</td>
<td>378</td>
</tr>
<tr>
<td>Rate-compatible punctured convolutional (RCPC) codes</td>
<td>379</td>
</tr>
<tr>
<td>Rate adaptation</td>
<td>380</td>
</tr>
<tr>
<td>Intermodulation interference suppression</td>
<td>382</td>
</tr>
<tr>
<td>Transmitter</td>
<td>382</td>
</tr>
<tr>
<td>Non-linear power amplifier model</td>
<td>383</td>
</tr>
<tr>
<td>MMSE receiver</td>
<td></td>
</tr>
<tr>
<td>Successive interference cancellation</td>
<td>386</td>
</tr>
<tr>
<td>System and channel model</td>
<td>386</td>
</tr>
<tr>
<td>MMSE detection</td>
<td>387</td>
</tr>
<tr>
<td>Tracking the fading processes</td>
<td>390</td>
</tr>
<tr>
<td>Approximation of optimum multiuser receiver</td>
<td>393</td>
</tr>
<tr>
<td>Frequency selective fading channels</td>
<td>397</td>
</tr>
<tr>
<td>Receiver signal model of STBC</td>
<td>398</td>
</tr>
<tr>
<td>Blind approach</td>
<td>399</td>
</tr>
<tr>
<td>Bayesian optimal blind receiver</td>
<td>400</td>
</tr>
<tr>
<td>Blind Bayesian Monte Carlo multiuser receiver approximation</td>
<td>400</td>
</tr>
<tr>
<td>Gibbs sampler</td>
<td>400</td>
</tr>
<tr>
<td>Prior distributions</td>
<td>401</td>
</tr>
<tr>
<td>Conditional posterior distributions</td>
<td>401</td>
</tr>
<tr>
<td>Gibbs multiuser detection</td>
<td>402</td>
</tr>
<tr>
<td>Sampling space of data</td>
<td>403</td>
</tr>
<tr>
<td>The orthogonality property</td>
<td>403</td>
</tr>
<tr>
<td>Blind turbo multiuser receiver</td>
<td>403</td>
</tr>
<tr>
<td>Decoder-assisted convergence assessment</td>
<td>404</td>
</tr>
<tr>
<td>Multidimensional constellation</td>
<td>75</td>
</tr>
<tr>
<td>Multimedia least resistance routing</td>
<td>779</td>
</tr>
<tr>
<td>(MLRR)</td>
<td></td>
</tr>
<tr>
<td>Multipath decorrelation</td>
<td>213</td>
</tr>
<tr>
<td>Multiple</td>
<td></td>
</tr>
<tr>
<td>access</td>
<td>5, 201, 457, 492, 642</td>
</tr>
<tr>
<td>receive antennas</td>
<td>130</td>
</tr>
<tr>
<td>transmit antenna differential detection from generalized orthogonal designs</td>
<td>185</td>
</tr>
<tr>
<td>Multiuser Communication</td>
<td>191</td>
</tr>
<tr>
<td>Pseudorandom sequences</td>
<td>191</td>
</tr>
<tr>
<td>Binary shift register sequences</td>
<td>191</td>
</tr>
<tr>
<td>Binary maximal length sequences</td>
<td>193</td>
</tr>
<tr>
<td>Crosscorrelation spectra</td>
<td>193</td>
</tr>
<tr>
<td>Maximal connected sets of m-sequences</td>
<td>194</td>
</tr>
<tr>
<td>Gold sequences</td>
<td>194</td>
</tr>
<tr>
<td>Gold-like and dual-BCH sequences</td>
<td>195</td>
</tr>
<tr>
<td>Kasami sequences</td>
<td>196</td>
</tr>
</tbody>
</table>
INDEX 861

JPL sequences, 197
Kronecker sequences, 197
Walsh functions, 198
Optimum PN sequences, 199
Golay code, 199
Multiuser CDMA receivers, 201
Synchronous CDMA channels, 202
The decorrelating detector, 202
The optimum linear multiuser detector, 202
Multistage detection, 203
Non-coherent detector, 205
Non-coherent detection in asynchronous channels [45], 205
Frequency non-selective Rayleigh channels, 207
Multistage detection, 203
Non-coherent detector, 205
Non-coherent detection in asynchronous channels [45], 205
Frequency non-selective Rayleigh channels, 207
frequency selective Rayleigh channels, 210
Minimum mean square error (MMSE) linear multiuser detection, 216
System model in multipath fading channels, 217
MMSE detector structures, 220
Spatial processing, 221
Single user LMMSE receivers for frequency selective fading channels, 225
Adaptive precombining LMMSE receivers, 225
Blind least squares receivers, 230
Least squares (LS) receiver, 230
Matrix inversion lemma, 231
Signal subspace-based channel estimation for CDMA systems, 232
Estimating the signal subspace, 234
Channel estimation, 235
Iterative receivers for layered space–time coding, 236
LST architectures, 237
LST receivers, 241
QR decomposition/SIC detector, 242
MMSE/SIC detector, 244
Iterative LST receivers, 246
MUSIC, 3, 236, 237

N
Nakagami-Rice, 713, 714, 748
Nash equilibrium, 797, 799, 800, 801, 802, 803, 804, 805, 806, 808, 809, 812, 820, 825, 826, 830, 832, 841, 845, 847, 851
Networks, 2, 16, 17, 18, 19, 182, 262, 263, 266, 267, 321, 643, 645, 671, 746, 792, 793, 794, 797, 851, 852
Non-coherent detector, 205
Non-linear power amplifier model, 383
Null space, 506

O
Timing and frequency offset in OFDM, 329
Robust frequency and timing synchronization for OFDM, 331
Fading channel estimation for OFDM systems, 334
Statistics of mobile radio channels, 334
Diversity receiver, 335
MMSE channel estimation, 335
FIR channel estimator, 337
System performance, 338
Reference generation, 339
64 DAPSK and 64 QAM/OFDM signals, 339
Space–time coding with OFDM signals, 344
Signal and channel parameters, 345
The wireless ATM system, 349
Space–time coded adaptive modulation for OFDM, 349
Turbo and space–time coded adaptive OFDM, 349
Layered space–time coding for MIMO OFDM, 351
System model (two times two transmit antennas), 352
Interference cancellation, 353
Four transmit antennas, 353
Space–time coded TDMA/OFDM reconfiguration efficiency, 356
Frequency selective channel model, 356
Front end prefilter, 357
Time-invariant channel, 357
Optimization problem, 358
Average channel, 358
Prefiltered M-BCJR equalizer, 358
Decision, 359
Prefiltered MLSE/DDFSE equalizer complexity, 359
Delayed decision feedback sequence estimation (DDFSE), 360
INDEX

OFDM
Equalization schemes for STBC, 360
Single-carrier frequency domain equalized ST block coding SC FDE STBC, 361
Optimum PN sequences, 199
Orthogonality, 127
Outage capacity, 138, 140

P
Pareto optimality, 800, 801
Partitioned, 256
Path loss modeling, 440
PDF of shadow fading, 677
Picocell, 691, 693
PIC-STD, 247, 250, 251, 253
Pilot symbols, 378, 379
Polynomial, 56, 197
Positioning 645
cellular networks 645
AD and RD measurements 646
The circular, hyperbolic, and mixed multilateration 646
WLS solution of the location problem 648
Accuracy measure 649
Circural multilateration 650
Hyperbolic multilateration 651
Mixed multilateration 652
Performance results for three stations 652
Performance results for N stations 654
Relative positioning in wireless sensor networks 655
Performance bounds 656
Relative location estimation 659
Average performance of geolocation 664
Signal models and performance limits 664
Performance of location techniques 666
Average performance 667
Postcombining interference suppression receiver, 220
Power
allocation, 136, 794
delay profile, 528, 717
optimal routing, 793
PPM UWB multiple access, 3
Preamble, 555
Precoding
Space–time precoders for MIMO channels, 497
ISI modelling in MIMO channels, 497
MIMO system, 499
STBC systems, 502
Convex optimization theory, 504
Generalized MIMO systems, 505
Convex optimization, 506
Power optimization, 507
SINR optimization, 510
Performance example, 512
Beamforming, 513, 520
Multicarrier MIMO signal model, 514
Channel diagonalization, 516
Constraints in multicarrier systems, 526
Precombining interference suppression receiver, 221
Prefiltered MLSE/DDFSE equalizer complexity, 359
Prior distributions, 401
Processing gain adaptation, 545
Product sum distance, 141
Profit driven pricing, 833
punctured convolutional codes, 3, 31, 43, 44, 58, 60, 429

Q
QAM constellation, 68, 70, 73, 95, 96, 102, 340, 420
Quasi-static fading, 144, 145

R
Raised cosine window, 331
RAKE receiver, 6, 221, 222, 226, 227, 229, 374, 375, 437, 496, 582, 583, 731, 746, 754
Rate adaptation, 380
Rate-compatible punctured convolutional (RCPC) codes, 379
rate-distortion, 44, 60, 622

Real
constellation set, 131
FIR/correlator, 574
IIR/correlator, 574
Receiver signal model of STBC MC CDMA systems, 398
Recursive least squares (RLS), 291
Rician distributions, 85, 86
RLS algorithm, 291, 292, 294, 309, 310
Robust frequency and timing synchronization, 331
RS decoder, 338, 339

S
Sampling space of data, 403
Sato function, 278
Schur product, 225
Schur-concave, 513, 514, 518, 519, 520, 521, 522, 523, 524, 527
Pseudorandom sequences, 191
Binary shift register sequences, 191
Binary maximal length sequences, 193
Crosscorrelation spectra, 193
Maximal connected sets of \( m \)-sequences, 194
Gold sequences, 194
Gold-like and dual-BCH sequences, 195
Kasami sequences, 196
JPL sequences, 197
Kronecker sequences, 197
Walsh functions, 198
Optimum PN sequences, 199
Golay code 199
SEDUMI, 507, 535
Self-describing packets, 555
sensor networks, 1, 3, 4, 44, 655, 656, 673, 795, 830, 839, 852
Set partition, 68, 69, 70, 76
Signal and channel parameters, 345
Signal model, 275, 279, 313, 460, 664
Simulation of channel coefficients, 690
social optimality, 801
Space–time codes, 180, 181, 185, 57, 79
Diversity gain 79
Two-branch transmit diversity scheme with one receiver 80
Two transmitters and \( M \) receivers 82
The system model 84
The case of independent fade coefficients 85
Rayleigh fading 86
Design criteria for Rayleigh ST codes 86
Code construction 87
Reconfiguration efficiency of space–time coding 91
Delay diversity 94
Space–time block codes from orthogonal designs 96
Channel model and the diversity criterion 96
Real orthogonal designs 97
Space–time encoder 97
The diversity order 97
The decoding algorithm 98
Linear processing orthogonal designs 98
Generalized real orthogonal designs 99
Encoding 99
The Alamouti scheme 100
Complex orthogonal designs 100
Generalized complex orthogonal designs 100
Special codes 101
Channel estimation imperfections 102
Channel estimator 106
Quasi-orthogonal space–time block codes 107
Decoding 108
Decision metric 108
Space–time convolutional codes 109
Algebraic space–time codes 111
Full spatial diversity 116
QPSK modulation 116
Differential space–time modulation 116
The encoding algorithm 122
Differential decoding 123
Multiple transmit antenna differential detection from generalized orthogonal designs 125
Differential encoding 126
Received signal 126
Orthogonality 127
Encoding 127
Differential decoding 128
Received signal 129
Demodulation 130
Multiple receive antennas 130
The number of transmit antennas lower than the number of symbols 130
Real constellation set 131
Layered space–time coding 133
Receiver complexity 134
Group interference suppression 134
Suppression method 134
The null space 134
Receiver 135
Space–time codes (Cont.)
- Decision metric 135
- Multilayered space–time coded modulation 135
- Diversity gain 136
- Adaptive transmit power allocation 136
- Concatenated space–time block coding 140
- System model 140
- Product sum distance 141
- Error rate bound 141
- The case of low SNR 142
- Code design 142
- Estimation of MIMO channel 145
- System model 146
- Training 148
- Performance measure 148
- Definitions 148
- Channel estimation error 148
- Error statistic 149
- ST codes for frequency selective channels 151
- Diversity gain properties 153
- Coding gain properties 154
- Space–time trellis code design 155
- Optimization of a MIMO system 157
  - The channel model 157
  - Gain optimization by singular value decomposition (SVD) 158
  - The general \((M, N)\) case 159
  - Gain optimization by iteration for a reciprocal channel 161
  - Spectral efficiency of parallel channels 162
  - Capacity of the \((M, N)\) array 163
- MIMO systems with constellation rotation 163
- System model 163
- Performance in a Rayleigh fading channel 165
- Diagonal algebraic space–time block codes 167
- System model 167
- The DAST coding algorithm 169
- The DAST decoding algorithm 170
- QR Factorization 173
- Lattice code decoder for space–time codes 175
- MIMO channel capacity 176
- Space–time trellis code design, 155
- Spatial and frequency correlation, 702
- Spatial processing, 221
- Special codes, 101
- Spectral efficiency of parallel channels, 162
- Square root filtering, 310
- Standard iterative receiver, 246
- Statistical description of the received sequence, 281

Statistics of mobile radio channels, 334
STBC, 140, 144, 145, 360, 361, 362, 364, 365, 368, 396, 397, 398, 399, 403, 404, 405, 406, 499, 502, 503, 504
Street-guided propagation, 682
StrongARM embedded processor, 543
Supermodular games, 801
Survivor, 297
Synchronous CDMA channels, 202
System model, 140, 146, 163, 167, 217, 278, 288, 352, 842
Systematic convolutional codes, 33

T
Tarokh, Seshadri, and Calderbank (TSC), 119
TCSQ, 48, 50, 54
The case of low SNR, 142
The received signal model, 436
threaded layered space–time (TLST) structure, 240
TIA (Telecommunications Industry Association), 5
Timing, 332, 731
TLST architectures, 239
Tracking the fading processes, 390
Transmit diversity, 598
Transversal filter equalizer, 272, 273, 275
Turbo, 5, 57, 58, 59, 76, 77, 183, 185, 267, 302, 306, 307, 326, 327, 344, 349, 404, 419
equalization, 302, 306, 307, 327
Signal format 301
Equivalent discrete time channel model 302
Equivalent system state representations 302
Turbo equalization 302
Viterbi algorithm 303
Iterative implementation 304

U
Ultra Wide Band Radio, 3, 433
Universal Wireless Communications (UWC), 5
Utility, 798, 853
UWB, 1, 2, 3, 4, 5, 11, 13, 15, 16, 433, 435,
436, 437, 439, 440, 441, 442, 443,
444, 445, 446, 447, 448, 449, 451,
453, 457, 458, 459, 460, 461, 463,
464, 465, 467, 469, 471, 473, 475,
477, 479, 481, 483, 485, 487, 489,
491, 492, 493, 494, 495, 496, 625,
626, 627, 628, 629, 630, 632, 633,
634, 635, 636, 638, 641, 642, 643,
732, 735, 737, 739, 741, 743
channel, 3, 4, 433, 492, 496, 732

V
Variable QoS, 557
vertical Bell Laboratories layered space–time
(VBLAST), 237
vertical layered space–time (VLST),
237
Viterbi algorithm, 3, 32, 33, 34, 43, 44, 46, 50,
56, 57, 58, 90, 211, 241, 285, 303,
308, 454

W
Walsh functions, 198
water-filling, 179, 180, 521, 522, 523, 525
Wiener solution, 285, 389, 390, 391, 392, 393,
394
Wind-flex channel model, 727

Z
Zero-forcing equalizers, 269
ZF strategy, 242