

A Framework for Low-Complexity Iterative Interference Cancellation in Communication Systems

by

Albert M. Chan

B.A.Sc., Electrical Engineering, University of Toronto (1997)
S.M., Electrical Engineering and Computer Science, MIT (1999)

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2004

© Massachusetts Institute of Technology 2004. All rights reserved.

Author
Department of Electrical Engineering and Computer Science
May 21, 2004

Certified by
Gregory W. Wornell
Professor
Thesis Supervisor

Accepted by
Arthur C. Smith
Chairman, Department Committee on Graduate Students

A Framework for Low-Complexity Iterative Interference Cancellation in Communication Systems

by

Albert M. Chan

Submitted to the Department of Electrical Engineering and Computer Science
on May 21, 2004, in partial fulfillment of the
requirements for the degree of Doctor of Philosophy in
Electrical Engineering and Computer Science

Abstract

Communication over interference channels poses challenges not present for the more traditional additive white Gaussian noise (AWGN) channels. In order to approach the information limits of an interference channel, interference mitigation techniques need to be integrated with channel coding and decoding techniques. This thesis develops such practical schemes when the transmitter has no knowledge of the channel.

The interference channel model we use is described by $\mathbf{r} = \mathbf{H}\mathbf{x} + \mathbf{w}$, where \mathbf{r} is the received vector, \mathbf{H} is an interference matrix, \mathbf{x} is the transmitted vector of data symbols chosen from a finite set, and \mathbf{w} is a noise vector. The objective at the receiver is to detect the most likely vector \mathbf{x} that was transmitted based on knowledge of \mathbf{r} , \mathbf{H} , and the statistics of \mathbf{w} . Communication contexts in which this general integer programming problem appears include the equalization of intersymbol interference (ISI) channels, the cancellation of multiple-access interference (MAI) in code-division multiple-access (CDMA) systems, and the decoding of multiple-input multiple-output (MIMO) systems in fading environments.

We begin by introducing *mode-interleaved precoding*, a transmitter precoding technique that conditions an interference channel so that the pairwise error probability of any two transmit vectors becomes asymptotically equal to the pairwise error probability of the same vectors over an AWGN channel at the same signal-to-noise ratio (SNR).

While mode-interleaved precoding dramatically increases the complexity of exact ML detection, we develop *iterated-decision detection* to mitigate this complexity problem. Iterated-decision detectors use optimized multipass algorithms to successively cancel interference from \mathbf{r} and generate symbol decisions whose reliability increases monotonically with each iteration. When used in uncoded systems with mode-interleaved precoding, iterated-decision detectors asymptotically achieve the performance of ML detection (and thus the interference-free lower bound) with considerably lower complexity. We interpret these detectors as low-complexity approximations to message-passing algorithms.

The integration of iterated-decision detectors into communication systems with coding is also developed to approach information rates close to theoretical limits. We present joint detection and decoding algorithms based on the iterated-decision detector with mode-interleaved precoding, and also develop analytic tools to predict the behavior of such systems. We discuss the use of binary codes for channels that support low information rates, and multilevel codes and lattice codes for channels that support higher information rates.

Thesis Supervisor: Gregory W. Wornell
Title: Professor

Acknowledgments

A number of people have helped me along the long path to this completed thesis.

First and foremost, I have been honored to have had the opportunity to work with my advisor, Prof. Gregory Wornell. Greg has motivated me to be a better researcher by encouraging me to develop a broader view of my research that includes more than just the technical details. Greg's interest in my professional and personal growth has made my experience in MIT's graduate program enriching and valuable in more ways than I expected.

I also thank the two other members of my thesis committee, Prof. John Tsitsiklis and Prof. Madhu Sudan, for their reading and reviewing of my work.

During my graduate studies at MIT, I was fortunate to have been a member of the Signals, Information, and Algorithms Laboratory and the Digital Signal Processing Group. I thank you, my colleagues, for not only inspiring me with your brilliance and challenging me intellectually, but also for being a bunch of superb friends to work and socialize with. I am particularly grateful for the technical advice you have provided me over the years. I thank especially Yonina Eldar—my work with her led to some of the tools for random matrix analysis that are used in this thesis. I thank Huan Yao for, among other things, helping me visualize tricky geometric problems in N -dimensional space. My thanks also go to Uri Erez, J. Nicholas Laneman, Michael Lopez, and Emin Martinian, for allowing me to drop by their windowed offices to pick their brains. The expert insight of these and the rest of my colleagues has been truly helpful, and has allowed me to make this thesis what it is. I'd also like to thank Giovanni Aliberti for his computer hacking wisdom, even if none of it ever rubbed off on me.

I gratefully acknowledge the generous financial support of Hewlett-Packard under the MIT/HP Alliance, the National Science Foundation under Grant No. CCR-9979363, the Semiconductor Research Corporation under Contract No. 2000-HJ-794, Texas Instruments through the Leadership Universities Program, and the Natural Sciences and Engineering Research Council of Canada (NSERC) Postgraduate Scholarship Program.

My experience in Boston has been made so much richer by great friendships. I thank the truly unique Alecia Batson for cultivating the artist in me and also being the #1 client of my photographic services. I also thank Ajit Dash for our afternoon coffee breaks and for being a wonderful conversationalist and faithful friend. I'm delighted that Suriya Jeyapalan has decided to stay in Boston—I thank her for being an incredibly thoughtful and spirited friend. Thanks also to Eric Yaffe who, in his sincere and charming style, has shown me the way of the true Bostonian: Go Pats, Go Sox! Of course, I can't forget Boguslaw Zdzieblo, who goes back with me almost to the beginning of my time in Boston. Bogush has been my family away from home, and I give a heartfelt thanks for that.

I extend my utmost thanks and gratitude to Richard Possemato, who has successfully distracted me during the latter part of my graduate studies. As a researcher himself, Rich has inspired me with his dedication, inquisitiveness, and love of his work. Rich has avidly supported me in all my endeavors, even the zany and idiosyncratic ones. I sincerely thank you, Rich, for making my life more fulfilling.

Finally, I thank my parents and my brother Willy for their unconditional and unflagging love, encouragement, and support over the years. It has been your values, earnest, and assiduous efforts that have shaped me into the person I am. It wasn't always easy being away from home, but I have always felt that you were with me throughout my experience here. Mom and Dad, you can tell everyone I'm finally graduating!

Contents

1	Introduction	17
1.1	Interference Channel Model	18
1.1.1	Notation	20
1.2	Information Limits of Communication Channels	22
1.2.1	AWGN Channels	23
1.2.2	Interference Channels with Water Pouring	23
1.2.3	Interference Channels without Water Pouring	25
1.2.4	A Comparison of Maximum Information Rates	26
1.3	Classical Detection Schemes	28
1.3.1	ML Detection	29
1.3.2	Linear Detection	31
1.3.3	Decision-Feedback Detection	32
1.3.4	Multistage Detection	33
1.4	Two Challenges with Interference Channels	35
1.5	Thesis Outline	36
2	Mode-Interleaved Precoding	39
2.1	Mode-Interleaved Precoding	41
2.1.1	Large \mathbf{H} Matrices	44
2.1.2	Small \mathbf{H} Matrices	50
2.1.3	Properties of the Effective Channel	52
2.2	Frequency-Interleaved Precoding	56
2.2.1	Properties of the Effective Channel	60
2.3	Detection for Mode- and Frequency-Interleaved Precoded Systems	62

3	The Iterated-Decision Detector	65
3.1	Iterated-Decision Detection	66
3.1.1	Structure	67
3.1.2	Characterizing the Slicer Input	68
3.1.3	Filter Optimization	69
3.1.4	Computing the Correlation Coefficient ρ	72
3.1.5	Asymptotic Performance with Mode-Interleaved Precoding	73
3.2	Iterated-Decision Equalization	87
3.2.1	Asymptotic Performance with Frequency-Interleaved Precoding	91
3.3	Adaptive Iterated-Decision Detector	100
4	Iterated-Decision Detection as Low-Bandwidth Message Passing	107
4.1	The Sum-Product Algorithm	108
4.1.1	Single-Marginal Sum-Product Algorithm	109
4.1.2	The Complete-Marginal Sum-Product Algorithm	113
4.2	Applying Sum-Product Algorithm to ISI Channels	114
4.2.1	Trellis Processing	114
4.2.2	Iterative Processing	116
4.3	The Max-Sum Algorithm	118
4.3.1	Max-Sum Semiring	118
4.3.2	ML Detection	120
4.4	Relating the Max-Sum Algorithm to Iterated-Decision Detection	121
4.4.1	Binary Signalling Over AWGN Channels	124
4.4.2	Low-Bandwidth Message Passing from Variable to Factor Nodes	127
4.4.3	Message Passing from Factor to Variable Nodes	129
4.4.4	Combining Messages at Variable Nodes	130
4.4.5	Nonbinary Variables	132
4.5	Summary	133
5	Iterative Detection And Decoding Algorithms	135
5.1	Binary Codes	137
5.1.1	Implementation with Iterated-Decision Detection	139
5.1.2	Binary Signalling Over AWGN Channels	141

5.1.3	Transfer Characteristic of the Detection Block	143
5.1.4	Transfer Characteristic of the Decoding Block	147
5.1.5	EXIT Charts	151
5.1.6	Comparison to Turbo Equalization	155
5.2	Multilevel Codes	156
5.2.1	Multilevel Encoding	157
5.2.2	Decoding for AWGN Channels	159
5.2.3	Iterated-Decision Detection and Multistage Decoding	164
5.2.4	A Class of Iterative Detection/Decoding Algorithms	171
5.3	Lattice Codes	175
5.3.1	Decoding for AWGN Channels	175
5.3.2	Generalized Iterated-Decision Detection	176
6	Conclusions	179
6.1	Contributions	179
6.2	Future Work	181
6.2.1	Low-Bandwidth Message Passing	181
6.2.2	Analogy to Price's Result	181
6.2.3	Iterated-Decision Transmitter Precoding	182
6.2.4	Iterated-Decision Lattice Decoding for AWGN Channels	183
6.2.5	Iterated-Decision Quantizer	183
6.2.6	Approximate Solutions to Complex Problems	184
A	Isotropically Distributed Vectors and Matrices	185
B	Proof of Theorem 3.2	187
C	Information Limits of Random ISI Channels	195
D	Rate-Normalized SNR	199
D.1	AWGN Channels	199
D.2	Interference Channels with Water Pouring	201
D.3	Interference Channels without Water Pouring	203
E	The Correlation Coefficient for Lattice Codes	205

List of Figures

1.1	The vector model.	18
1.2	Uplink scenario of a CDMA system.	21
1.3	Multiple antenna system.	21
1.4	ISI channel.	21
1.5	A communication system for an AWGN channel.	23
1.6	A communication system that pre-cancels channel interference.	25
1.7	Optimal joint detection and decoding.	26
1.8	Classic suboptimal separation of detection and decoding.	26
1.9	Information rates for the three-tap ISI channel of (1.16).	28
1.10	Maximum-likelihood (ML) detection.	30
1.11	(a) Bounded lattice representing the uncoded set of vectors \mathcal{X}^2 . (b) Corresponding decision regions for the AWGN channel.	30
1.12	(a) Bounded lattice representing all possible vectors $\mathbf{H}\mathbf{x}$ for an interference channel. (b) Corresponding decision regions.	31
1.13	Linear detection.	32
1.14	Decision-feedback detection.	33
1.15	A multistage detector.	34
1.16	Bit-error probability of various detection schemes as a function of SNR for the interference channel of (1.21). The AWGN channel bound is also pictured, corresponding to the interference-free case.	35
2.1	(a) A communication system with transmitter precoding. (b) The effective channel created by transmitter precoding.	40
2.2	A spectrum for 2×2 \mathbf{H} matrices of Frobenius norm 2. At the left end, the two eigenvalues are equal; at the right end, the two eigenvalues are as unequal as possible. Matrices located near the left end tend to be more desirable in the sense that the distance between two vectors does not change as dramatically after being transmitted over the channel. The \mathbf{H} matrix in (2.1), which has eigenvalues of 1.866 and 0.134, is located near the right end of the spectrum.	42
2.3	A two-dimensional signal constellation after a linear transformation by the precoder in cascade with a matrix \mathbf{H} with a zero mode. The black dots are constellation points after the possible \mathbf{x} vectors are rotated by the precoder and \mathbf{V} , and the white dots are the points after the two-dimensional space is scaled by the modes $\lambda_1 = 0$ and $\lambda_2 = 1$. (a) The combined rotation of the precoder and \mathbf{V} is such that all points remain distinguishable. (b) The combined rotation of the precoder and \mathbf{V} is such that some points become indistinguishable. The difference in rotation angle between the two examples is approximately 45°	43

2.4	Average pairwise error probability as a function of $\alpha^2 = \ \mathbf{\Delta}_N\ ^2/4\sigma_N^2$ when half of the eigenvalues of $\mathbf{H}^\dagger\mathbf{H}$ are 0.5 and half are 1.5, and the eigenvector matrix is an independent and isotropically distributed unitary matrix. . . .	48
2.5	ISI channel model with a cyclic prefix.	56
2.6	Frequency-interleaved precoded system.	58
3.1	Bit-error probability of various detection schemes as a function of SNR for the interference channel of (2.1). Except for the iterated-decision detector, the various detectors are not used in conjunction with mode-interleaved precoding. Since the iterated-decision detector with mode-interleaved precoding achieves the AWGN channel bound at high SNR, we can infer that ML detection with mode-interleaved precoding (not pictured) would also achieve the AWGN channel bound.	66
3.2	Iterated-decision detection.	67
3.3	Iterated-decision detector performance for i.i.d. Gaussian \mathbf{H} . The solid curves plot QPSK symbol error rate as a function of the correlation coefficient ρ for various $\beta = N/Q$ values, with an SNR per bit of 7 dB. Along each curve, \circ 's identify the theoretically predicted decreasing error rates achieved with $l = 1, 2, \dots$ decoding passes, and the intersections with the dashed line are the steady-state values ($l \rightarrow \infty$).	80
3.4	Theoretical iterated-decision detector performance for i.i.d. Gaussian \mathbf{H} , as a function of SNR per bit. The solid curves depict the QPSK bit-error rate with $\beta = N/Q = 0.77$ as a function of SNR per bit for 1, 2, 3, 5, and ∞ decoding iterations.	81
3.5	Theoretical iterated-decision detector performance for i.i.d. Gaussian \mathbf{H} , as a function of $\beta = N/Q$. The solid curves depict the QPSK bit-error rate as a function of β for various values of SNR per bit, while the corresponding dashed curves depict the AWGN channel bound.	83
3.6	High SNR performance limits of the iterated-decision detector. The solid curve plots the QPSK symbol error rate as a function of ρ for a channel with $\kappa = 1.641$ and all nonzero eigenvalues equal to κ . The SNR per bit is 12 dB, and \circ 's identify the theoretically predicted decreasing error rates. The dash-dot curves indicate the high-SNR limits of performance.	83
3.7	Theoretical ($Q \rightarrow \infty$) and experimentally observed ($Q = 128$) performance for various detectors for i.i.d. Gaussian \mathbf{H} . The solid curves depict QPSK bit-error rates with $\beta = N/Q = 1$ as a function of SNR per bit.	86
3.8	Theoretical ($Q \rightarrow \infty$) and experimentally observed ($Q = 128$) performance for various detectors, for i.i.d. Gaussian \mathbf{H} . The solid curves depict QPSK bit-error rates at an SNR per bit of 10 dB as a function of $\beta = N/Q$	86
3.9	Experimentally observed QPSK bit-error probability of various detection schemes as a function of SNR for the interference channel of (2.1). Except for the iterated-decision detector which is used both with and without mode-interleaved precoding, the various detectors are not used in conjunction with mode-interleaved precoding. Since the iterated-decision detector with mode-interleaved precoding achieves the AWGN channel bound at high SNR, we can infer that ML detection with mode-interleaved precoding (not pictured) would also achieve the AWGN channel bound.	87
3.10	Iterated-decision equalizer.	88

3.11 Iterated-decision equalizer implemented in the DFT domain.	90
3.12 Iterated-decision equalizer performance. The curves plot QPSK symbol error rate as a function of the correlation coefficient ρ for various SNRs. Along each curve, \circ 's identify the theoretically predicted decreasing error rates achieved with $l = 1, 2, \dots$ decoding passes, and the intersections with the dashed line are the steady-state values ($l \rightarrow \infty$). The associated experimentally obtained values for $L = 256$ are depicted using \times 's.	95
3.13 Theoretical iterated-decision equalizer performance as a function of SNR per bit. The successively lower solid curves depict the QPSK bit-error rate as a function of SNR per bit for 1, 2, 3, 5, and ∞ decoding iterations.	95
3.14 Theoretical ($L \rightarrow \infty$) and experimentally observed ($L = 256$) performance for various equalizers. The solid curves depict QPSK bit-error rates as a function of SNR per bit.	97
3.15 Theoretical iterated-decision equalizer performance with frequency-interleaved precoding for the three-tap channel (3.112). The successively lower curves plot the QPSK symbol error rate as a function of SNR per bit for 1, 2, 3, 5, 10, and 20 detection iterations.	98
3.16 Iterated-decision equalizer performance with frequency-interleaved precoding. The solid curves plot QPSK symbol error rate as a function of ρ for the three-tap channel (3.112) at various SNRs. Along the curve, \circ 's identify the theoretically predicted decreasing error rates achieved with $l = 1, 2, \dots$ decoding passes, and the intersections with the dashed line are the steady-state values ($l \rightarrow \infty$).	98
3.17 Experimentally observed performance for various equalizers. The solid curves depict QPSK bit-error rates for the 3-tap channel in (3.112) as a function of SNR per bit. Except for the iterated-decision detector which is used with and without frequency-interleaved precoding, the various other detectors are not used in conjunction with precoding. Since the iterated-decision detector with frequency-interleaved precoding achieves the matched filter bound at high SNR, we can infer that ML detection with precoding (not pictured) would also achieve the matched filter bound.	99
3.18 Experimentally observed QPSK bit-error rate for the adaptive iterated-decision detector as a function of the number of decoding iterations and the number of training vectors transmitted with each packet of 10000 data vectors at an SNR per bit of 7 dB. The channel is an i.i.d. Gaussian matrix with $\beta = N/Q = 1$	102
3.19 Experimentally observed ($Q = 128$) QPSK bit-error rates for the adaptive iterated-decision detector and the RLS-based adaptive linear detector (forgetting factor $\lambda = 1$), with $\beta = N/Q = 1$. Each packet consists of 10000 data vectors plus either 500, 1000, or 5000 additional vectors of training symbols.	102
3.20 Experimentally observed QPSK bit-error rate for the adaptive iterated-decision equalizer and the RLS-based adaptive DFE (with forgetting factor $\lambda = 1$) as a function of SNR per bit. Blocks of 10000 data symbols were transmitted through 128-tap channels, which were equalized using 257 feedforward taps and 256 noncausal feedback taps in the case of the iterated-decision equalizer, and using 257 feedforward taps and 128 strictly causal feedback taps in the case of the DFE.	104

4.1	An expression tree representing $(x + y) \times z$	109
4.2	A factor graph for the factorization $g_A(x_1, x_3)g_B(x_2)g_C(x_2, x_3, x_4)g_D(x_4, x_5)$	110
4.3	Local substitutions that transform a rooted cycle-free graph to an expression tree for a marginal function. On the left is the substitution at a variable node; on the right is the substitution at a factor node g with parent x	110
4.4	(a) The factor graph of Fig. 4.2 redrawn as a tree rooted at x_1 . (b) The corresponding expression tree for $g_1(x_1)$	111
4.5	(a) The factor graph of Fig. 4.2 redrawn as a tree rooted at x_2 . (b) The corresponding expression tree for $g_2(x_2)$	112
4.6	A cycle-free factor graph for the global function $p(\mathbf{x}, \mathbf{s}, \mathbf{y} \mathbf{r})$: the x_n are input variables, the s_n are state variables, the y_n are output variables, and each r_n is the noisy observation of y_n	115
4.7	A factor graph with cycles for the global function $p(\mathbf{x} \mathbf{r})$: the x_n are input variables and the r_n are ISI channel output variables. In this example, the channel taps h_n are nonzero for $n = 0, 1, 2$ and the block length is $N = 5$	116
4.8	Local substitutions that transform a rooted cycle-free graph to an expression tree for a marginal function in the max-sum semiring. On the left is the substitution at a variable node; on the right is the substitution at a factor node g with parent x	120
4.9	A cycle-free factor graph for the global function $\log p(\mathbf{x}, \mathbf{s}, \mathbf{y} \mathbf{r})$: the x_n are input variables, the s_n are state variables, the y_n are output variables, and each r_n is the noisy observation of y_n	122
4.10	A factor graph with cycles for the global function $\log p(\mathbf{x} \mathbf{r})$: the x_n are input variables and the r_n are ISI channel output variables. In this example, the channel taps h_n are nonzero for $n = 0, 1, 2$ and the block length is $N = 5$	122
4.11	A block diagram for the computation of \tilde{x}_n^l in iterated-decision equalization. In this example, $h_n \neq 0$ for $n = 0, 1, 2$ and $b_n \neq 0$ for $n = -2, -1, 0$. The coefficients next to the edges are scaling factors.	124
4.12	The expression tree for the computation of \tilde{x}_n^l during an iteration of the max-sum algorithm on the factor graph in Fig. 4.10. In this example, $h_n \neq 0$ for $n = 0, 1, 2$	124
4.13	The function $\rho(\tilde{x}) = \tanh(\tilde{x}/\sigma^2) $	127
4.14	(a) A different <i>a priori</i> probability mass function is associated with each binary variable at the beginning of each iteration. (b) The probability mass functions are approximated by impulse functions located at the associated Bayes' least squares estimates. (c) The expected absolute value of the Bayes' least squares estimate replaces the magnitude of the Bayes' least squares estimates for each binary variable.	128
4.15	The quantities $E[x \tilde{x}]$, $\rho(\tilde{x})$, and $f(\tilde{x})$ as a function of \tilde{x} , for 4-PAM where $\mathcal{X} = \{-3, -1, +1, +3\}$	134
5.1	Iterative processing using separate detection and decoding blocks.	136
5.2	A factor graph representing the global function $p(\mathbf{x} \mathbf{r})$ for an LDPC-coded system with interference.	138
5.3	A factor graph representing the global function $p(\mathbf{x} \mathbf{r})$ for an RA-coded system with interference.	139
5.4	The flow of L -values between the detector and decoder blocks.	140

5.5	Relationship between mutual information $I_{\hat{X}}$ and the probability of bit error $\Pr(\epsilon)$	143
5.6	Transfer characteristic for the iterated-decision equalizer at various SNRs, with frequency interleaving and the three-tap ISI channel of (3.112).	146
5.7	Transfer characteristic for the iterated-decision equalizer at various SNRs, with the asymptotic random ISI channel.	146
5.8	The density of the message from a variable node converges either to (a) a density with a finite probability of error or (b) a “point mass at infinity.”	149
5.9	Transfer characteristic for sum-product decoding of various regular LDPC codes. An infinite number of decoding iterations is assumed.	150
5.10	EXIT chart for the iterated-decision equalizer and an iterative decoder, for a (5, 10)-regular LDPC code and the asymptotic random ISI channel at an SNR_{norm} of 3 dB.	152
5.11	Ten simulated trajectories for the iterated-decision equalizer and an iterative decoder, for a (5, 10)-regular LDPC code and a random ISI channel of length 1024 at an SNR_{norm} of 3 dB.	153
5.12	EXIT chart for the iterated-decision equalizer and an iterative decoder, for a (5, 10)-regular LDPC code and the asymptotic random ISI channel.	153
5.13	Simulated performance of the iterated-decision equalizer and an iterative decoder, for a (5, 10)-regular LDPC code and a random ISI channel of length 1024.	154
5.14	EXIT chart for the iterated-decision equalizer an SNR of 6 dB, for an uncoded system with the asymptotic random ISI channel.	155
5.15	A multilevel encoder.	157
5.16	Different mapping schemes for 8-PAM.	158
5.17	Block partitioning for 8-PAM.	158
5.18	Ungerboeck partitioning for 8-PAM.	158
5.19	Channels for block partitioning of 8-PAM. (a) Physical channel. (b) Parallel theoretical subchannels.	160
5.20	A multistage decoder.	161
5.21	Mutual information of the physical AWGN channel and the theoretical subchannels for a three-level code (8-PAM) with block partitioning.	162
5.22	Mutual information of the physical AWGN channel and the theoretical subchannels a for three-level code (8-PAM) with Ungerboeck partitioning.	162
5.23	Mutual information of an AWGN channel with PAM constellations.	163
5.24	Achievable rate region for two-level coding.	164
5.25	Multistage decoding combined with iterated-decision detection.	165
5.26	Information rates supported by each iteration of the combined multistage decoder and iterated-decision equalizer for a three-level code (8-PAM) with block partitioning over the limiting random ISI channel.	168
5.27	Information rates supported by the combined multistage decoder and iterated-decision equalizer for block partitioning over the limiting random ISI channel.	168
5.28	Information rates supported by each iteration of the combined multistage decoder and iterated-decision equalizer for a three-level code (8-PAM) with Ungerboeck partitioning over the limiting random ISI channel.	170
5.29	Information rates supported by the combined multistage decoder and iterated-decision equalizer for Ungerboeck partitioning over the limiting random ISI channel.	170

5.30	Information rates supported by the combined multistage decoder and iterated-decision detector for block partitioning over the limiting i.i.d. Gaussian matrix channel with $\beta = N/Q = 1$	171
5.31	Iterated-decision detection and decoding for lattice codes.	176
C.1	For the purposes of computing the capacity of the random ISI channel, the magnitude squared of the channel frequency response can be derived from an exponential distribution.	196
C.2	Information rates for the asymptotic random ISI channel.	197
D.1	The noise balls for (a) 4-QAM and (b) 16-QAM when the normalized SNR is equal. Normalization makes both constellations essentially the same, since they both come from the same infinite lattice.	200
E.1	The sets $\mathcal{S}_{\mathbf{e}_k^+}$ and $\mathcal{S}_{\mathbf{e}_k^-}$ are defined as the codewords that, when transmitted and decoded, cannot produce the error events $+\mathbf{e}_k$ and $-\mathbf{e}_k$ respectively. The plane perpendicular to the minimum-distance error event \mathbf{e}_k divides the spherical bounding region of the lattice into two hemispheres; the black dots of the two hemispheres constitute $\mathcal{S}_{\mathbf{e}_k^+}$ and $\mathcal{S}_{\mathbf{e}_k^-}$	208

Chapter 1

Introduction

In a digital communication system, channel interference limits the ability of the receiver to detect the discrete symbols sent by the transmitter. The channel is a part of the communication system we typically cannot change, so techniques to detect the transmitted symbols that take into account the channel interference are important.

In this thesis, we develop practical schemes to mitigate the effects of channel interference to make digital communication more reliable. We focus on the scenario in which the transmitter has no knowledge of the channel interference as is often the case in, for example, wireless communication over radio waves. Furthermore, we assume that the receiver is aware of the nature of the channel interference via, for example, channel probing or channel tracking. These schemes apply to a wide range of communication channels, and are thus presented in a common signal space framework.

In this introductory chapter, we present some relevant background material. In Section 1.1, we describe a discrete-time baseband model for digital communication systems and define the associated detection problem. We also establish notation that is used throughout this thesis. We present a broader perspective in Section 1.2, where we explain how detection can be combined with channel coding and decoding in different communication scenarios to reliably transmit information at rates close to the theoretical maximum. Section 1.3 reviews the maximum-likelihood (ML) detector, which minimizes the probability of vector detection error, as well some classical suboptimal detectors that are low-complexity alternatives to ML detection. In Section 1.4, by means of a comparison of the performance of the various detectors, we gain some insight into the challenges posed by communication

over interference channels. We present the outline for the rest of the thesis in Section 1.5.

1.1 Interference Channel Model

In this thesis, we focus on discrete-time baseband channel models, which abstract the channel impairments and hide the specific implementational details of the digital communication system. In doing so, we can talk about different digital communication systems with different kinds of channel interference in one common signal space framework.

Let us now describe the channel model that we use in this thesis. The $N \times 1$ vector \mathbf{x} contains the data to be transported over the channel, and is chosen from a finite equiprobable set. Depending on the underlying communication system, the components of \mathbf{x} may correspond either to distinct time instants, distinct carrier frequencies, distinct physical locations, etc. The channel interference is modelled as linear interference, which is represented by multiplication of \mathbf{x} with a $Q \times N$ matrix \mathbf{H} . With channel noise being composed of the superposition of many independent actions, the central limit theorem suggests that we can model the noise as a zero-mean, complex-valued, additive white Gaussian noise (AWGN) vector \mathbf{w} with circularly symmetric components of variance \mathcal{N}_0 . The $Q \times 1$ vector \mathbf{r} that is obtained at the receiver is thus

$$\mathbf{r} = \mathbf{H}\mathbf{x} + \mathbf{w}, \quad (1.1)$$

as illustrated in Fig. 1.1.

In this thesis, we are primarily concerned with detection at the receiver of the transmit vector \mathbf{x} based on knowledge of \mathbf{r} , \mathbf{H} , and the statistics of \mathbf{w} . The parameters of \mathbf{H} can be learned at the receiver via techniques collectively known as *training*, in which \mathbf{H} is estimated by sending vectors jointly known to the transmitter and receiver across the channel. If the channel changes with time, then the estimate of \mathbf{H} can be updated using the detection

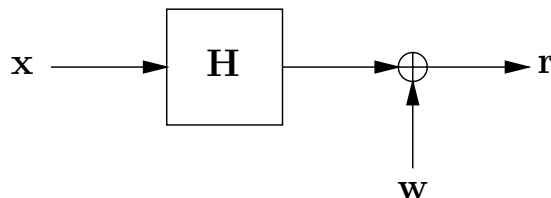


Figure 1.1: The vector model.

decisions. Sometimes it is also useful to periodically perform training in case tracking becomes unsuccessful. In any event, we assume in most of the thesis that \mathbf{H} and the statistics of \mathbf{w} are explicitly known at the receiver.

Though we focus specifically on applications of the vector detection model (1.1) to digital communication systems, the detection schemes we develop in this thesis are applicable to any scenario in which (1.1) applies. We now complete this section with a few applications in digital communication.

One example of a communication system in which the channel model (1.1) applies is the uplink scenario of a N -user discrete-time synchronous code-division multiple-access (CDMA) system, shown in Fig. 1.2. In this system, the i th user modulates a complex symbol x_i onto a signature sequence $h_i[k]$ of length Q assigned to that user. The modulated signature sequence is sent across the channel, where it encounters channel attenuation by a factor of A_i . The base station receives the superposition of all the users' signals in noise as described by (1.1), where the columns of \mathbf{H} are the users' signatures scaled by the corresponding channel attenuation factors; i.e.,

$$\underbrace{\begin{bmatrix} r[0] \\ r[1] \\ \vdots \\ r[Q-1] \end{bmatrix}}_{\mathbf{r}} = \underbrace{\begin{bmatrix} A_0 h_0[0] & A_1 h_1[0] & \cdots & A_{N-1} h_{N-1}[0] \\ A_0 h_0[1] & A_1 h_1[1] & \cdots & A_{N-1} h_{N-1}[1] \\ \vdots & \vdots & \ddots & \vdots \\ A_0 h_0[Q-1] & A_1 h_1[Q-1] & \cdots & A_{N-1} h_{N-1}[Q-1] \end{bmatrix}}_{\mathbf{H}} \underbrace{\begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_{N-1} \end{bmatrix}}_{\mathbf{x}} + \underbrace{\begin{bmatrix} w[0] \\ w[1] \\ \vdots \\ w[Q-1] \end{bmatrix}}_{\mathbf{w}}. \quad (1.2)$$

Another example in which (1.1) applies is a discrete-time synchronous multiple antenna system with N transmit antennas and Q receive antennas. As shown in Fig. 1.3, each transmit antenna sends a different complex symbol x_i over the channel. For narrowband transmission, the path from the i th transmit antenna to the j th receive antenna is described by a single flat fading coefficient, h_{ji} , the set of which can be assembled into the matrix \mathbf{H} . Each receive antenna receives a superposition of signals from all transmit antennas in white

noise, so again we encounter the model in (1.1):

$$\underbrace{\begin{bmatrix} r_0 \\ r_1 \\ \vdots \\ r_{Q-1} \end{bmatrix}}_{\mathbf{r}} = \underbrace{\begin{bmatrix} h_{00} & h_{01} & \cdots & h_{0,N-1} \\ h_{10} & h_{11} & \cdots & h_{1,N-1} \\ \vdots & & \ddots & \vdots \\ h_{Q-1,0} & h_{Q-1,1} & \cdots & h_{Q-1,N-1} \end{bmatrix}}_{\mathbf{H}} \underbrace{\begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_{N-1} \end{bmatrix}}_{\mathbf{x}} + \underbrace{\begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_{Q-1} \end{bmatrix}}_{\mathbf{w}}. \quad (1.3)$$

A discrete-time point-to-point channel with intersymbol interference (ISI), depicted in Fig. 1.4, can also be modelled using (1.1). The transmitted data is a stream of complex symbols $x[n]$, which are corrupted by a convolution with the impulse response of the ISI channel, $h[n]$, and by additive noise, $w[n]$, to produce the received symbols

$$r[n] = \sum_k h[n-k]x[k] + w[n]. \quad (1.4)$$

If $r[n]$, $x[n]$, and $w[n]$ are each arranged in vector format, and $\mathbf{H} = [\mathbf{h}_1 | \cdots | \mathbf{h}_P]$ with \mathbf{h}_i being a time-delayed version of $h[n]$ arranged in vector format, then we again obtain the model (1.1). An example with an impulse response of length two is

$$\underbrace{\begin{bmatrix} r[0] \\ r[1] \\ \vdots \\ r[N-1] \end{bmatrix}}_{\mathbf{r}} = \underbrace{\begin{bmatrix} h[0] & 0 & 0 & \cdots & 0 \\ h[1] & h[0] & 0 & & 0 \\ 0 & h[1] & \ddots & \ddots & \vdots \\ \vdots & & \ddots & h[0] & 0 \\ 0 & 0 & \cdots & h[1] & h[0] \end{bmatrix}}_{\mathbf{H}} \underbrace{\begin{bmatrix} x[0] \\ x[1] \\ \vdots \\ x[N-1] \end{bmatrix}}_{\mathbf{x}} + \underbrace{\begin{bmatrix} w[0] \\ w[1] \\ \vdots \\ w[N-1] \end{bmatrix}}_{\mathbf{w}}. \quad (1.5)$$

Note that the \mathbf{H} matrix in this case is square and Toeplitz.

1.1.1 Notation

Before we proceed, let us review some matrix terminology and establish some notation.

The superscripts T and \dagger denote transpose and conjugate transpose respectively. The singular value decomposition of a $Q \times N$ matrix \mathbf{H} is $\mathbf{U}\mathbf{\Sigma}\mathbf{V}^\dagger$, where \mathbf{U} is a $Q \times Q$ unitary matrix, $\mathbf{\Sigma}$ is a $Q \times N$ matrix with singular values on the main diagonal, and \mathbf{V} is an $N \times N$ unitary matrix. The matrix \mathbf{U} is often called the left singular matrix, and \mathbf{V} is

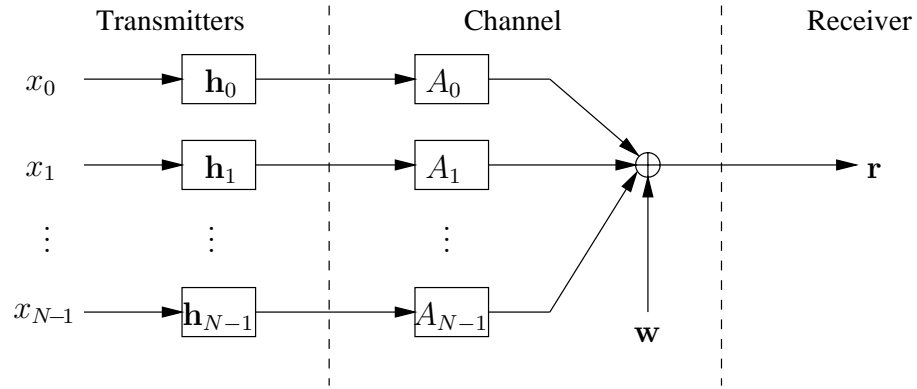


Figure 1.2: Uplink scenario of a CDMA system.

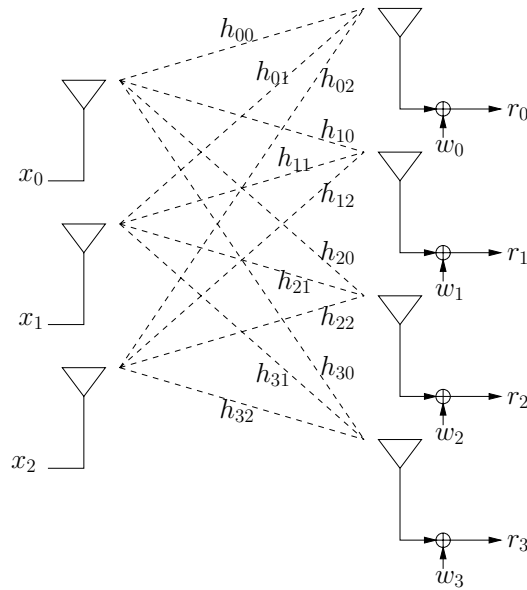


Figure 1.3: Multiple antenna system.

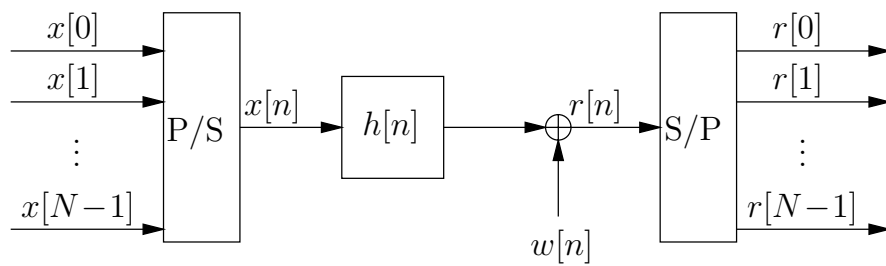


Figure 1.4: ISI channel.

often called the right singular matrix. Since premultiplication of the received vector \mathbf{r} by the matched filter \mathbf{H}^\dagger at the receiver forms a sufficient statistic, and since the cascade of \mathbf{H} and a matched filter \mathbf{H}^\dagger is sometimes more convenient to work with, we shall often refer to $\mathbf{H}^\dagger\mathbf{H}$. The eigenvalue decomposition of the $N \times N$ matrix $\mathbf{H}^\dagger\mathbf{H}$ is then $\mathbf{V}\mathbf{\Lambda}\mathbf{V}^\dagger$, where $\mathbf{\Lambda} = \mathbf{\Sigma}^\dagger\mathbf{\Sigma} = \text{diag}(\lambda_1, \dots, \lambda_N)$. When $N \leq Q$, the N eigenvalues of $\mathbf{H}^\dagger\mathbf{H}$ are equal to the N squared singular values of \mathbf{H} . When $N > Q$, the N eigenvalues of $\mathbf{H}^\dagger\mathbf{H}$ consist of the Q squared singular values of \mathbf{H} and $N - Q$ zeros. The eigenvalues in this context are often called *modes*, and the matrix \mathbf{V} is often called the modal matrix.

The received SNR is defined as

$$\frac{1}{\zeta} \triangleq \frac{\mathcal{E}_s}{\mathcal{N}_0} \cdot \frac{\|\mathbf{H}\|_F^2}{N} \quad (1.6)$$

where \mathcal{E}_s is the average energy of the components in \mathbf{x} , \mathcal{N}_0 is the variance of the noise components in \mathbf{w} , $\|\cdot\|_F^2$ is the squared Frobenius norm of a matrix, and N is the length of \mathbf{x} . The squared Frobenius norm of a matrix is simply the sum of the squared magnitudes of all the components of the matrix [46].

1.2 Information Limits of Communication Channels

In the early 1940s, the general consensus was that increasing the rate of information transmitted over a communication channel increased the probability of error. However, in his 1948 paper “A Mathematical Theory of Communication,” Shannon quantified the maximum rate of information that can be sent error-free over a communication channel [56]. Shannon’s proof involved using random codes with exponentially small error probability for long block lengths, coupled with a nearest neighbor decoding rule. Unfortunately, such structureless codes are extremely difficult to decode in practice, generally requiring an exponentially large lookup table and/or computation. Ever since then, researchers have sought structured codes whose rates are close to theoretical limits and yet can be decoded simply.

There are different information limits for different communication scenarios, and in this section we review the maximum possible information rates for various scenarios and how researchers have tried to achieve those rates with practical modulation schemes. In particular, we look at interference-free channels, interference channels when the transmitter has knowledge of the interference (also known as *channel state information*), and interference

channels when the transmitter does not have such knowledge.

1.2.1 AWGN Channels

Additive white Gaussian noise (AWGN) channels can be thought of as interference-free channels, because they correspond to the case of (1.1) in which \mathbf{H} is the identity matrix \mathbf{I} . The maximum information rate that can be sent over an AWGN channel or, more commonly, the *capacity* of the AWGN channel in bits per two dimensions is well known to be

$$C_{\text{AWGN}} = \log_2 \left(1 + \frac{\mathcal{E}_s}{\mathcal{N}_0} \right). \quad (1.7)$$

A communication system for an AWGN channel is shown in Fig. 1.5. Code design and practical decoding techniques for rates close to AWGN channel capacity are well understood today. In low SNR regimes where low-rate codes suffice, turbo codes [9] and low-density parity-check (LDPC) codes [30, 31] approach theoretical limits and are practically decodable with “message-passing” algorithms like those discussed in Chapter 4. In [17], it was demonstrated that an LDPC code can operate at an SNR that is 0.0045 dB away from the minimum SNR necessary for codes of that rate. In high SNR regimes where high-rate codes are required, multilevel codes with multistage decoding [41, 44] can be used to approach capacity, as discussed in Section 5.2.

1.2.2 Interference Channels with Water Pouring

In communication systems where the interference matrix \mathbf{H} is non-trivial and known at the transmitter, to support the maximum information rate it is necessary to optimally allocate transmit power amongst the various components of \mathbf{x} so that the power transmitted over

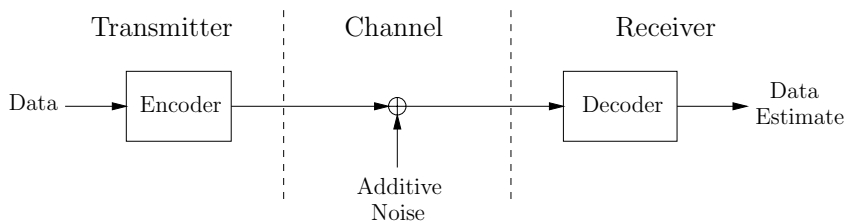


Figure 1.5: A communication system for an AWGN channel.

the k th mode of the channel is given by the “water-pouring” formula [43]

$$\mathcal{E}_{s,k} = \max\left(\mathcal{L} - \frac{N_0}{\lambda_k}, 0\right) \quad (1.8)$$

where \mathcal{L} is chosen such that the average transmit energy is

$$\mathcal{E}_s = \frac{1}{N} \sum_{k=0}^{N-1} \mathcal{E}_{s,k}. \quad (1.9)$$

The idea of water pouring is to transmit more power over large modes, and to transmit less power over small modes. In fact, modes that are too small are allocated no transmit power. If K is the set of modes allotted transmit power via water pouring, then the capacity of the interference channel in bits per two dimensions is

$$C_{\text{int}} = \frac{K}{N} \log_2 \left(\frac{\frac{\mathcal{E}_s}{N_0} \cdot \frac{N}{K} + \left\langle \frac{1}{\lambda_k} \right\rangle_{A,K}}{\left\langle \frac{1}{\lambda_k} \right\rangle_{G,K}} \right) \quad (1.10)$$

where $\langle 1/\lambda_k \rangle_{A,K}$ and $\langle 1/\lambda_k \rangle_{G,K}$ are, respectively, the arithmetic and geometric means of $1/\lambda_k$ over K , given by

$$\left\langle \frac{1}{\lambda_k} \right\rangle_{A,K} = \frac{1}{|K|} \sum_{k \in K} \frac{1}{\lambda_k} \quad (1.11)$$

$$\log \left\langle \frac{1}{\lambda_k} \right\rangle_{G,K} = \frac{1}{|K|} \sum_{k \in K} \log \frac{1}{\lambda_k}. \quad (1.12)$$

Techniques are known today to approach the capacity of interference channels as closely as the capacity of interference-free channels, provided the transmitter is aware of the interference. A communication system that incorporates such techniques is shown in Fig. 1.6. At the transmitter, knowledge of channel state information is used in an interference “pre-canceller” that optimally allocates transmit power to the channel modes. The cascade of the transmitter pre-canceller and the channel appears interference-free, so the coding and decoding techniques for AWGN channels can be exploited to approach capacity. There are two main classes of such techniques. One class treats all the modes as belonging to a single channel [14, 38, 59], while the other class partitions the underlying channel into parallel independent subchannels, over which symbols are transmitted according to the

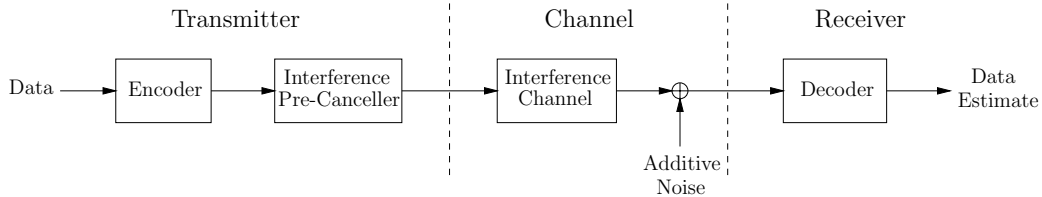


Figure 1.6: A communication system that pre-cancels channel interference.

water-pouring power allocation [55]. For ISI channels, an example in the former class is Tomlinson-Harashima precoding, and an example in the latter is discrete multi-tone (DMT).

1.2.3 Interference Channels without Water Pouring

When the transmitter has no channel state information, water pouring cannot be done to achieve the capacity of the interference channel. Rather, the transmit power is evenly distributed across all the modes of the system, leading to a maximum rate of

$$I_{\text{int}} = \frac{1}{N} \sum_{k=0}^{N-1} \log_2 \left(1 + \frac{\mathcal{E}_s \lambda_k}{\mathcal{N}_0} \right) \quad (1.13)$$

bits per two dimensions. We call this quantity the mutual information of the channel rather than the capacity because the information rate is not optimized at the transmitter via water pouring.

In this scenario, the communication system depicted in Fig. 1.6 can no longer be used, and creating parallel; rather, interference must be dealt with at the receiver. The optimal receiver in a probability of error sense uses a maximum-likelihood (ML) or maximum *a posteriori* (MAP) algorithm that treats the encoder and the interference channel as a single product code and performs joint detection and decoding, shown in Fig. 1.7. Though jointly optimal, the complexity of such a system is usually determined by the product of the complexities of the optimal detector for the corresponding uncoded system and the optimal decoder for the corresponding AWGN channel. Thus, the complexity of such a system is prohibitive.

A classical suboptimal solution is to separate the problems of detection and decoding as shown in Fig. 1.8. The detector should be designed so that the cascade of the interference channel and the detector appears like an AWGN channel, so the complexity of the detector

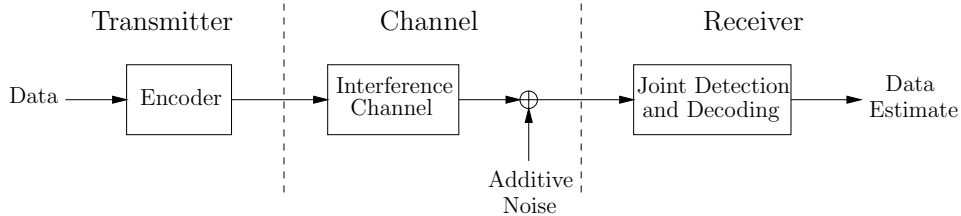


Figure 1.7: Optimal joint detection and decoding.

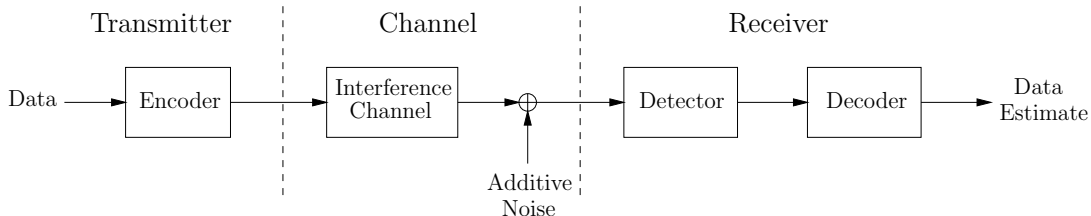


Figure 1.8: Classic suboptimal separation of detection and decoding.

is no different than for uncoded data. However, it is more difficult to make the cascade of the channel and the detector look like an AWGN channel than it is for the cascade of a transmitter pre-canceller and the channel to look like one, because noise from the channel can be amplified or *enhanced* by the detector. Nevertheless, assuming that the detector is designed well, coding and decoding schemes for AWGN channels can be used, so the complexity of the decoder is the same as if there were no interference in the channel. The result is that the overall complexity is the sum, rather than the product, of the complexities of the individual components.

It is this scenario of the three for which techniques to approach the theoretical limits are least developed. This thesis presents practical schemes with low complexity to approach the mutual information of an interference channel without channel knowledge at the transmitter. Before we review some classical detectors, let us gain some insight by comparing the theoretical information limits of the three communication scenarios presented.

1.2.4 A Comparison of Maximum Information Rates

We now compare the maximum achievable rates of different communication scenarios at the same received SNR $1/\zeta$, as defined in (1.6). To facilitate the comparison, we assume that $\|\mathbf{H}\|_F^2 = \sum_{k=1}^N \lambda_k$ is normalized to N , the number of columns in \mathbf{H} , so that the symbol energy \mathcal{E}_s and the noise variance \mathcal{N}_0 are also fixed.

When the transmit power is equally distributed amongst the modes, the interference channel cannot have a mutual information greater than the corresponding AWGN channel with equivalent SNR. We can see this from the concavity of the log function:

$$I_{\text{int}} = \frac{1}{N} \sum_{k=1}^N \log_2 \left(1 + \frac{\mathcal{E}_s \lambda_k}{\mathcal{N}_0} \right) \leq \log_2 \left(1 + \frac{\mathcal{E}_s}{\mathcal{N}_0} \right) = C_{\text{AWGN}} \quad (1.14)$$

where we have used the normalization of $\|\mathbf{H}\|_F^2$, and the rates are per two dimensions. At low SNR, however, we can show that $I_{\text{int}} \approx C_{\text{AWGN}}$ using the approximation $\ln(1 + \alpha) \approx \alpha$:

$$I_{\text{int}} = \frac{1}{N} \sum_{k=1}^N \log_2 \left(1 + \frac{\mathcal{E}_s \lambda_k}{\mathcal{N}_0} \right) \approx \frac{1}{N} \sum_{k=1}^N \frac{\mathcal{E}_s \lambda_k}{\mathcal{N}_0 \ln 2} = \frac{\mathcal{E}_s}{\mathcal{N}_0 \ln 2} \approx \log_2 \left(1 + \frac{\mathcal{E}_s}{\mathcal{N}_0} \right) = C_{\text{AWGN}}. \quad (1.15)$$

At high SNR, the rate loss from C_{int} to I_{int} due to the absence of water pouring is negligible because the capacity-achieving allotment of transmit power to the various modes is asymptotically equal. The term “high SNR” is relative in this context, because not only does the total transmit power need to be high, but the amount allocated to each mode must be high as well. Thus, for C_{int} and I_{int} to be approximately equal, a channel with a few small modes requires a higher SNR than a channel with no small modes. At low SNR, however, the capacity C_{int} of an interference channel can substantially exceed both the mutual information without water pouring I_{int} and the capacity of the AWGN channel C_{AWGN} even with the normalization of $\|\mathbf{H}\|_F^2$. Intuitively, transmit power is selectively loaded onto channel modes that can support the highest rates.

Figure 1.9 compares C_{int} and I_{int} for the three-tap ISI channel with impulse response

$$h[n] = 0.5\delta[n] + 0.707\delta[n - 1] + 0.5\delta[n - 2] \quad (1.16)$$

to C_{AWGN} . As noted earlier, the capacity of an ISI channel exceeds that of the corresponding AWGN channel at low SNR, since transmit power can be loaded onto favorable frequencies. If water pouring is not available, then at low SNR the mutual information of the random ISI channel approaches the capacity of the corresponding AWGN channel. At high SNR, the capacity of the ISI channel becomes less than the AWGN channel capacity, and also the effect of water pouring becomes negligible. The asymptotic slopes of all three curves are equal, implying that the penalty of an interference channel is only a fixed rate loss.

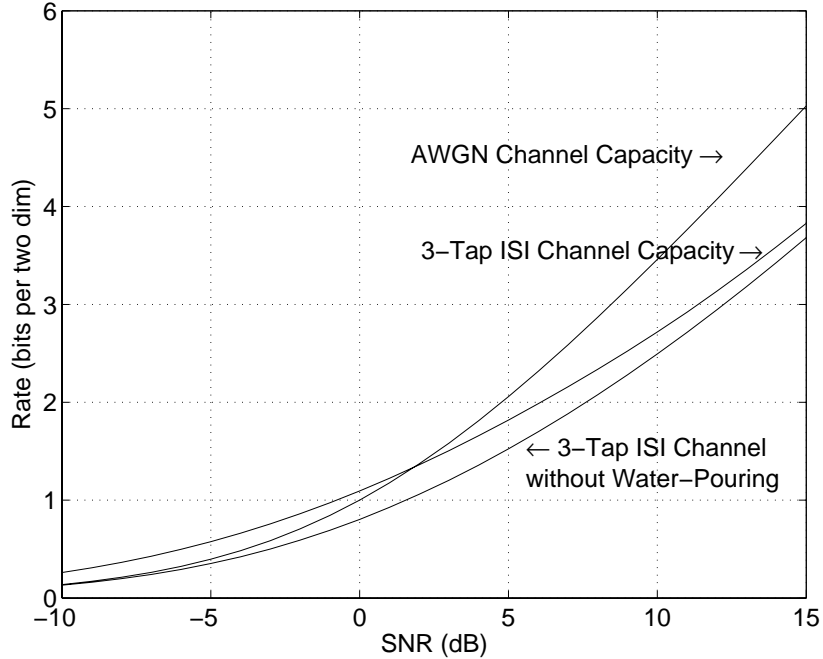


Figure 1.9: Information rates for the three-tap ISI channel of (1.16).

1.3 Classical Detection Schemes

There are a variety of detectors that can be used in the scenario of Fig. 1.8, and in this section we discuss some of the most common. We begin with the maximum-likelihood (ML) detector, which minimizes the probability of vector detection error and can be considered optimal. The rest of the detectors presented in this section are designed to offer an approximate solution to ML detection with lower complexity. While the extent to which these suboptimal detectors trade off accuracy with complexity varies, they all share the idea that the effect of \mathbf{H} should be explicitly “cancelled” or “undone” so that the decision device can treat the channel as an AWGN channel. In the special case of an uncoded system, the detection problem can be decoupled and processed by a simple symbol-by-symbol decision device.

Before we review these classical detectors, it is useful to establish a way by which we can compare detector performance. Although practical communication systems typically use coded transmission, we will compare detectors based on their performance for uncoded transmission, i.e., each component of \mathbf{x} contains independent data and is equiprobably chosen from a finite set \mathcal{X} , and so $\mathbf{x} \in \mathcal{X}^N$. Since detectors typically treat coded data as if

it were uncoded, focusing on uncoded systems allows us to isolate the quality of the detector from the quality of the decoder.

The symbol-error rate for an uncoded system, defined as the probability that a component of $\hat{\mathbf{x}}$ is not equal to the corresponding component of $\mathbf{x} \in \mathcal{X}^N$, is the measure that we use in this thesis when specifically talking about detector performance. We use this metric for several reasons. First, the ML detector, which minimizes the probability of vector detection error, has a symbol-error rate that is almost identical at high signal-to-noise ratio (SNR) to the detector that minimizes symbol-error probability [27]. Thus asymptotically, the ML detector provides a useful lower bound to the symbol-error rate of *any* detector. Second, looking at symbol errors is equivalent to computing the Hamming distance between \mathbf{x} and $\hat{\mathbf{x}}$, which gives a measure of closeness of the detected vector to the transmitted vector. Since the ML vector is closest to \mathbf{x} on average, we can infer how close the solutions provided by other detectors are to the optimal solution. Third, since suboptimal detectors typically attempt to decouple the detection of uncoded symbols by using a symbol-by-symbol minimum-distance decision device or *slicer*, using the symbol-error rate is intuitive. Fourth, codes are often characterized by the number of symbol errors in a codeword that can be corrected. In a system that performs detection followed by decoding, a low symbol-error rate after detection is desirable for good overall performance of the system.

1.3.1 ML Detection

Since all vectors \mathbf{x} are equally likely, the detector that minimizes the probability of vector detection error is the ML detector:

$$\hat{\mathbf{x}} = \arg \max_{\mathbf{x} \in \mathcal{X}^N} f(\mathbf{r}|\mathbf{x}). \quad (1.17)$$

Since the noise is independent of \mathbf{x} , uncorrelated and Gaussian, (1.17) simplifies to the minimum-distance rule

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x} \in \mathcal{X}^N} \|\mathbf{r} - \mathbf{H}\mathbf{x}\|, \quad (1.18)$$

depicted in Fig. 1.10. Thus, the detector computes the most likely vector \mathbf{x} based on knowledge of \mathbf{r} , \mathbf{H} , and the distribution of \mathbf{w} . The set of all possible uncoded vectors \mathcal{X}^N can be represented in N -dimensional Euclidean space as the points of a (shifted) orthogonal lattice bounded within an N -dimensional cube. The set of possible vectors is depicted in

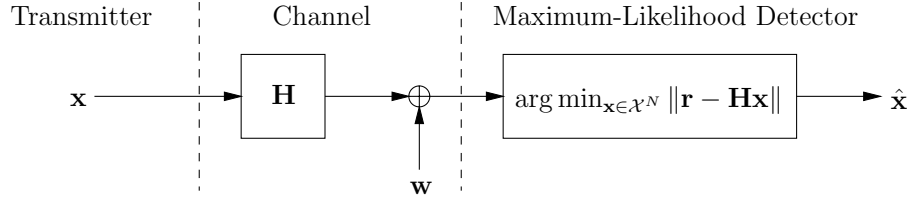


Figure 1.10: Maximum-likelihood (ML) detection.

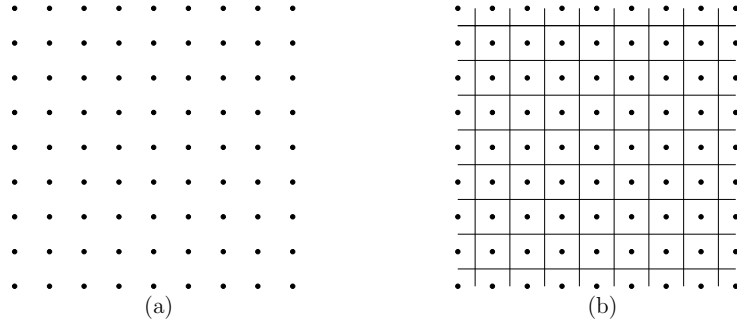


Figure 1.11: (a) Bounded lattice representing the uncoded set of vectors \mathcal{X}^2 . (b) Corresponding decision regions for the AWGN channel.

Fig. 1.11(a) for $N = 2$.

In the special case of an AWGN channel, our model becomes $\mathbf{r} = \mathbf{x} + \mathbf{w}$, and so \mathbf{r} is a noise-perturbed version of \mathbf{x} . The minimum-distance rule (1.18) simplifies to

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x} \in \mathcal{X}^N} \|\mathbf{r} - \mathbf{x}\|. \quad (1.19)$$

Since each component of the uncoded vector \mathbf{x} affects only the corresponding component of \mathbf{r} , and since the noise vector is uncorrelated, the ML detector can be decoupled into a set of symbol-by-symbol optimizations; i.e.,

$$\hat{x}_i = \arg \min_{x_i \in \mathcal{X}} \|r_i - x_i\| \quad \text{for } i = 0, 1, \dots, N - 1, \quad (1.20)$$

which can be solved using a symbol-by-symbol minimum-distance decision device or *slicer*. The decision regions, corresponding to the values of \mathbf{r} for which each of the possible decisions is made, are depicted in Fig. 1.11(b). The ability to decouple the ML detector into componentwise minimizations is indicated by the fact that the boundaries of the decision regions form an orthogonal grid. The minimization for each of the N components of \mathbf{x}

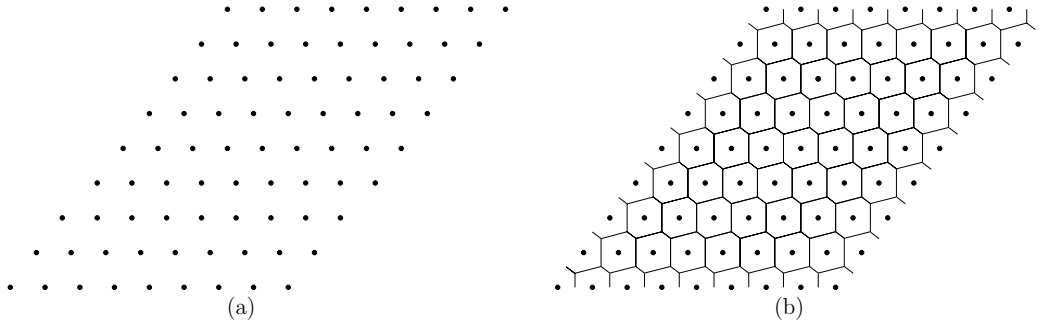


Figure 1.12: (a) Bounded lattice representing all possible vectors $\mathbf{H}\mathbf{x}$ for an interference channel. (b) Corresponding decision regions.

requires the comparison of $|\mathcal{X}|$ differences, so complexity is linear in N .

In the general case in which linear interference is present, we have that $\mathbf{r} = \mathbf{H}\mathbf{x} + \mathbf{w}$, and the ML vector detector of (1.18) generally cannot be decomposed into N smaller problems. We can see this by first recognizing that the action of \mathbf{H} on the set of all possible uncoded vectors $\mathbf{x} \in \mathcal{X}^N$ is to map the points of the bounded orthogonal lattice in Fig. 1.11(a) to the points of a bounded lattice with generators along the directions of the columns of \mathbf{H} , like the bounded lattice in Fig. 1.12(a). The decision regions of (1.18) are now generally polytopes as shown in Fig. 1.12(b), and decoupling of the problem is no longer possible. The minimization of (1.18) requires the comparison of $|\mathcal{X}|^N$ differences, so complexity is exponential in N . In fact, the least-squares integer program in (1.18) for general \mathbf{H} matrices has been shown to be nondeterministic polynomial-time hard (NP-hard) [65]. In the ISI channel case, ML detection can be performed using a dynamic programming algorithm known as the Viterbi algorithm [27], which has complexity proportional to $|\mathcal{X}|^L$ where L is length of the channel impulse response. If L or $|\mathcal{X}|$ is large, then ML detection is still complex.

The high complexity of the ML detector has invariably precluded its use in practice, so lower-complexity detectors that provide approximate solutions to (1.18) are used, which we review in the next section.

1.3.2 Linear Detection

As depicted in Fig. 1.13, linear detectors take the received vector \mathbf{r} and premultiply it by a matrix \mathbf{B}^\dagger . The resulting product, $\tilde{\mathbf{x}}$, is passed to a minimum-distance symbol-by-symbol slicer to produce $\hat{\mathbf{x}}$. The matrix \mathbf{B} can be optimized using different criteria, but

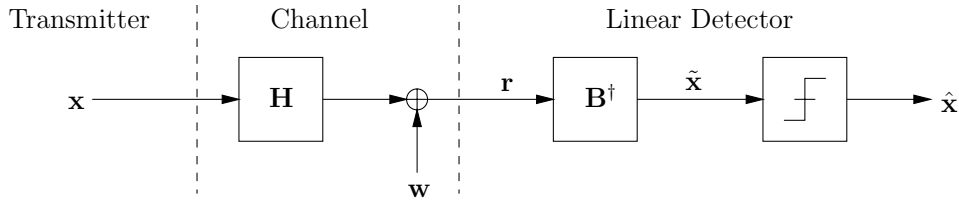


Figure 1.13: Linear detection.

two of the most popular are the zero-forcing (ZF) criterion and the minimum mean-squared error (MMSE) criterion. The ZF criterion, also known as the interference-nulling criterion, chooses \mathbf{B} to completely eliminate interference in $\tilde{\mathbf{x}}$, while the MMSE criterion chooses \mathbf{B} to minimize the variance of $\tilde{\mathbf{x}} - \mathbf{x}$.

The main disadvantage of this low-complexity class of detectors is that they perform quite poorly in a symbol-error rate sense because the matrix \mathbf{B}^\dagger enhances the variance of the noise components in \mathbf{w} and also makes the difference vector $\tilde{\mathbf{x}} - \mathbf{x}$ correlated. These problems are more severe when the ZF criterion is used, but are present nevertheless for the entire class of detectors.

Shnidman [57] considered the problem of eliminating interference in a multiuser context very early on. The MMSE linear detector for CDMA systems was first described by Xie et al. [70] and Madhow and Honig [47]. Tufts [61] derived the ZF and MMSE linear detectors for the ISI channel case, known in that context as linear *equalizers*.

1.3.3 Decision-Feedback Detection

The decision-feedback detector builds upon the linear detector by combining it with a non-linear feedback loop, pictured in Fig. 1.14. As before, the received vector \mathbf{r} is premultiplied by \mathbf{B}^\dagger , but rather than making minimum-distance symbol-by-symbol slicer decisions on the entire output vector, decisions are made sequentially, one component at a time. To begin, the first component of $\mathbf{B}^\dagger \mathbf{r}$, denoted \tilde{x}_1 , is processed by the slicer to produce the detected symbol \hat{x}_1 . Assuming that \hat{x}_1 is equal to x_1 , the feedback loop is used to subtract off the interference caused by x_1 from the remaining components of $\mathbf{B}^\dagger \mathbf{r}$. The second component of the resulting interference-reduced vector, denoted \tilde{x}_2 , is then processed by the slicer to produce \hat{x}_2 . Assuming that \hat{x}_2 is a correct decision, the interference caused by x_2 is subtracted from the remaining components of $\mathbf{B}^\dagger \mathbf{r}$, and the process continues until decisions all

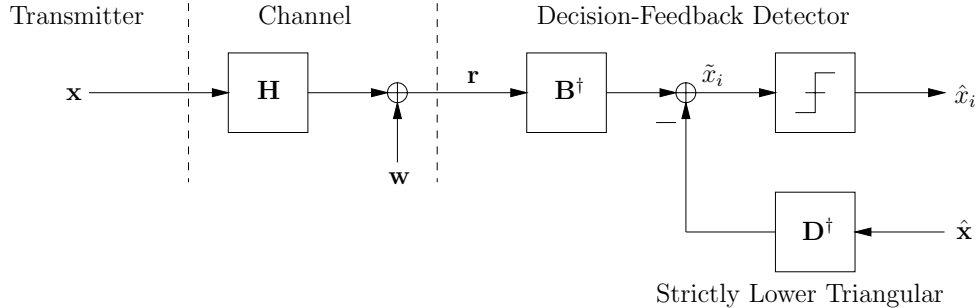


Figure 1.14: Decision-feedback detection.

the components have been made. As with the linear detector, the ZF and MMSE criteria are popular.

Though decision-feedback detectors usually perform better than linear detectors, they still have some serious shortcomings. First, noise enhancement is still an issue, though the situation is not as bad as with linear detection. Second, decisions are made sequentially at the slicer and so are used to improve only future decisions, not past ones. Third, the sequential nature of the decision device means that in practice, incorrect decisions can lead to further incorrect decisions, a phenomenon known as *error propagation*. Moreover, since the matrices \mathbf{B} and \mathbf{D} used in decision-feedback detection are often optimized under the faulty assumption of no error propagation, there may be some mismatch between the desired optimization criterion and the matrices that are used. Fourth, the sequential structure of the decision-feedback detector makes it essentially incompatible for use with ISI channels in conjunction with channel coding (on channels not known at the transmitter, as is the case of interest in this thesis). As a result, use of the decision-feedback equalizer has been largely restricted to uncoded systems.

The idea of feeding back decisions to mitigate the effects of interference for future symbols was first used by Austin [3] in the context of ISI channels. Duel-Hallen [21] introduced the idea to CDMA systems, while Foschini [29] brought the idea to multiple antenna systems via the Bell Labs Layered Space-Time (BLAST) detection algorithm.

1.3.4 Multistage Detection

In an attempt to symmetrize the problem of being able to cancel only future symbols in decision-feedback detectors, multistage detectors process the received vector in block

iterations. In some sense, multistage detectors can be thought of as parallel processors, whereas decision-feedback detectors are sequential processors.

An example of a multistage detector is shown in Fig. 1.15. During the first iteration, the vector \mathbf{r} is premultiplied by the matched filter \mathbf{H}^\dagger to produce $\tilde{\mathbf{x}}^1$, which is then sent to the slicer to generate a first set of tentative decisions $\hat{\mathbf{x}}^1$ on *all* the symbols. During the second iteration, the vector \mathbf{r} is again premultiplied by the matched filter \mathbf{H}^\dagger , but before the result is sent to the slicer, an exact replica of the interference is created and subtracted off assuming $\hat{\mathbf{x}}^1$ is a correct set of decisions. The slicer then takes the resulting vector $\tilde{\mathbf{x}}^2$, and generates a second set of tentative decisions $\hat{\mathbf{x}}^2$. Further sets of tentative decisions $\tilde{\mathbf{x}}^l$ are generated in the same manner, using the tentative decisions of the previous iteration $\hat{\mathbf{x}}^{l-1}$ to subtract off interference. After a sufficient number of iterations, the most recent tentative decisions are taken as the final decisions. The two matrices in this example are fixed during each iteration, and are optimized to maximize the signal-to-interference+noise ratio (SINR) at the slicer input assuming correct tentative decisions. In general, the class of multistage detectors includes detectors with the same structure as in Fig. 1.15 but with alternative pairs of matrices that may change with each iteration.

The problem with multistage detectors is that the decisions typically do not converge to the optimum ones, and limit cycles and divergence are possible. The reason for the poor performance of the multistage detector in Fig. 1.15 is that, like decision-feedback detectors, the two matrices are optimized assuming correct decisions are made at each iteration. Using this faulty assumption causes the incorrect quantities to be subtracted off as interference, which leads to the propagation of errors from iteration to iteration.

Gersho and Lim [32] developed multistage detectors for the ISI channel case, and Varanasi and Aazhang [64] later introduced them for the CDMA system case.

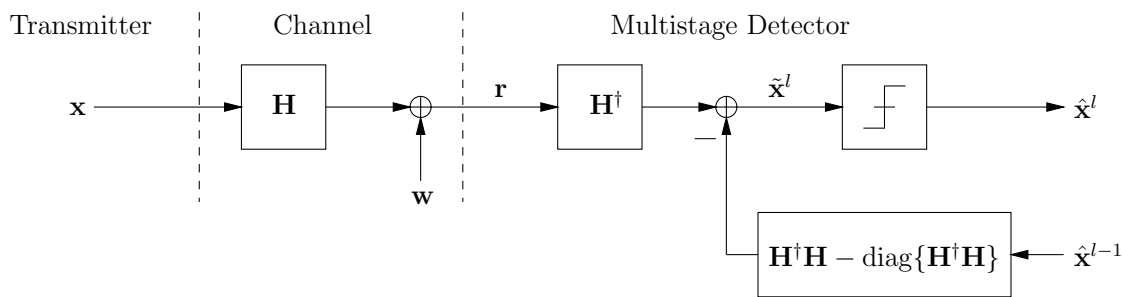


Figure 1.15: A multistage detector.

1.4 Two Challenges with Interference Channels

In this section, we present an example that compares the symbol-error rate performance of the detectors discussed thus far. In doing so, we highlight two difficulties in communication over interference channels that are not present over AWGN channels.

We consider the transmission of an uncoded 2×1 vector \mathbf{x} of binary (2-PAM) symbols over the 2×2 channel

$$\mathbf{H} = \begin{bmatrix} \sqrt{3}/2 & -1/2 \\ -1/2 & \sqrt{3}/2 \end{bmatrix}. \quad (1.21)$$

The bit-error rate performance of ML detection for this channel is depicted in Fig. 1.16 as a function of received SNR.

The first difficulty is that ML detection at the receiving end of an interference channel does not achieve as low a bit-error rate as does ML detection at the receiving end of the corresponding AWGN (interference-free) channel at the same SNR. The bit-error rate of the AWGN channel is also plotted in Fig. 1.16 as the dashed line. Since an AWGN channel has no interference, the corresponding bit-error rate curve is considered a lower bound

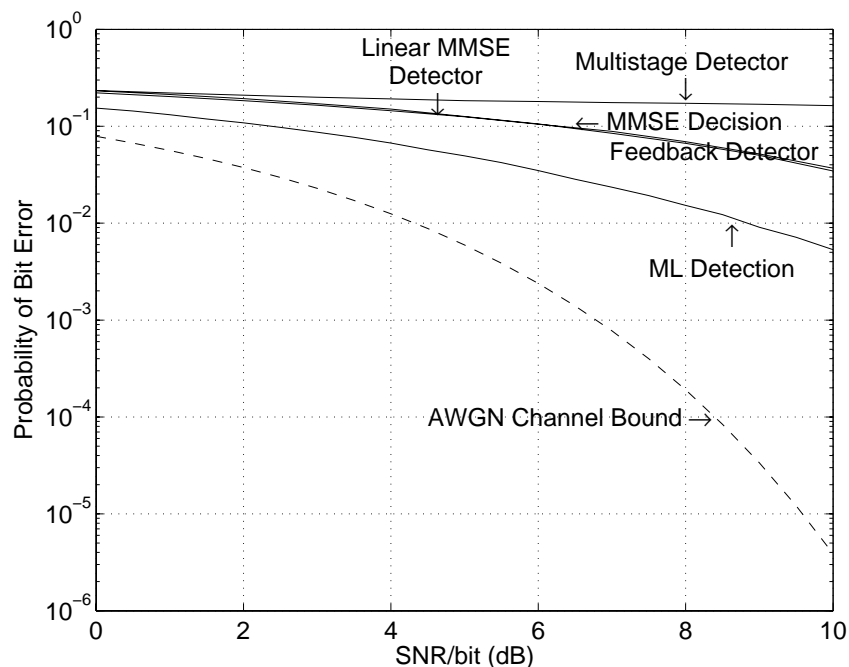


Figure 1.16: Bit-error probability of various detection schemes as a function of SNR for the interference channel of (1.21). The AWGN channel bound is also pictured, corresponding to the interference-free case.

to the bit-error rate curves of all detectors for interference channels. We generically call this bound the *AWGN channel bound*, but it is more commonly known as the matched filter bound in the context of ISI channels and the single-user matched-filter bound in the context of CDMA systems. Comparing the curves for ML detection in the interference and interference-free cases, we can see that they diverge at increasingly higher SNRs.

The second difficulty is that, given an interference channel, the probability of bit error achieved using suboptimal detectors does not approach that of ML detection. The bit-error rate curves for linear MMSE detection, MMSE decision-feedback detection, and multistage detection are plotted in Fig. 1.16. The curves for linear MMSE detection and MMSE decision-feedback detection are similar and appear to diverge from the corresponding curve for ML detection at high SNR. The multistage detector performs even worse than the linear and decision-feedback detectors.

This example highlights not only the potentially huge gap between the bit-error rates of the ML solution for interference and interference-free channels, but also the need for effective low-complexity detectors that can approximate the ML solution for interference channels well.

1.5 Thesis Outline

In this thesis, we develop practical schemes to approach the maximum information rates achievable over interference channels when the transmitter does not know the channel. In light of Section 1.2.4, our results also apply to the scenario in which the transmitter knows the channel but chooses not to do water pouring because the SNR is high enough that $I_{\text{int}} \approx C_{\text{int}}$.

The next two chapters of this thesis deal explicitly with the two difficulties identified in Section 1.4 for communication over interference channels.

In Chapter 2, we introduce a transmitter precoding technique that conditions the interference matrix \mathbf{H} so that the interference becomes as benign as possible. This method, which we call *mode-interleaved precoding*, relies on large-dimensional unitary matrices. Even though the precoding takes place at the transmitter, the transmitter does not need knowledge of the channel. We prove that the pairwise error probability of two (possibly coded) vectors sent over an interference channel is asymptotically equal to the pairwise error prob-

ability of the same vectors over the AWGN channel with the same SNR. Furthermore, it is conjectured (and later demonstrated in Chapter 3) that ML detection in combination with mode-interleaved precoding asymptotically achieves the AWGN channel bound for uncoded systems.

While the use of large-dimensional matrices in mode interleaving renders ML detection intractably complex, the complexity problem is mitigated in Chapter 3. We introduce the iterated-decision detector, which is particularly effective for large systems and thus well-suited for use with mode-interleaved precoding. Iterated-decision detectors belong to the class of multistage detectors, but unlike the multistage detector in Fig. 1.15, iterated-decision detectors are designed to maximize the SINR at the slicer input taking into account the reliability of tentative decisions. When combined with mode-interleaved precoding in uncoded systems, iterated-decision detectors asymptotically achieve the AWGN channel bound (and hence the symbol-error rate performance of ML detection) with considerably lower complexity than ML detection. In fact, the complexity is on the order of linear detection and decision-feedback detection schemes.

In Chapter 4, we interpret iterated-decision detectors in the context of message-passing algorithms [42] when mode-interleaved precoding is used. We develop an interpretation of iterated-decision detection as a low-complexity variant of the max-sum message-passing algorithm, a relative of the sum-product algorithm that implements ML detection.

With this new insight, we discuss in Chapter 5 ways in which mode-interleaved precoding and iterated-decision detection can be integrated into coded communication systems so that information rates close to the theoretical maximum can be achieved, and we also develop analytic tools to predict the behavior of such systems. A class of iterative detector and decoder algorithms for binary codes is first proposed, inspired by the message-passing interpretation of the iterated-decision detector in Chapter 4. We also investigate iterative detector and decoder algorithms for multilevel codes, whose decoding strategy has similarities to iterated-decision detection. Finally, we briefly discuss how the iterated-decision detector can also be generalized in a natural way to be compatible with lattice codes.

In Chapter 6, we summarize the contributions of this thesis and discuss future research directions.

Chapter 2

Mode-Interleaved Precoding

In Section 1.4, we considered the symbol-error rate performance of ML detection for the transmission of uncoded symbols over the 2×2 channel

$$\mathbf{H} = \begin{bmatrix} \sqrt{3}/2 & -1/2 \\ -1/2 & \sqrt{3}/2 \end{bmatrix} \quad (2.1)$$

and showed that there is an unbounded SNR loss compared to the performance of ML detection over the corresponding AWGN channel

$$\mathbf{H} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}. \quad (2.2)$$

This observation is indicative of the fact that for communication systems without transmitter water pouring, any \mathbf{H} matrix with off-diagonal elements is intrinsically worse in terms of ML detection error probability than an \mathbf{H} that is the identity matrix scaled to have the same Frobenius norm. The channel is a part of the communication system that we usually have no control over, so we instead have to find ways to work with it effectively.

For example, we can use special processing at the receiver in an attempt to make the channel appear differently, which may perhaps reduce the probability or error of the corresponding ML detector. However, any processing performed at the receiver can always be considered as part of the detector. As we have already discussed in Section 1.3, the ML detector for the given channel provides a lower bound to the probability of error of *any* detector, and so any additional processing at the receiver cannot reduce the gap between

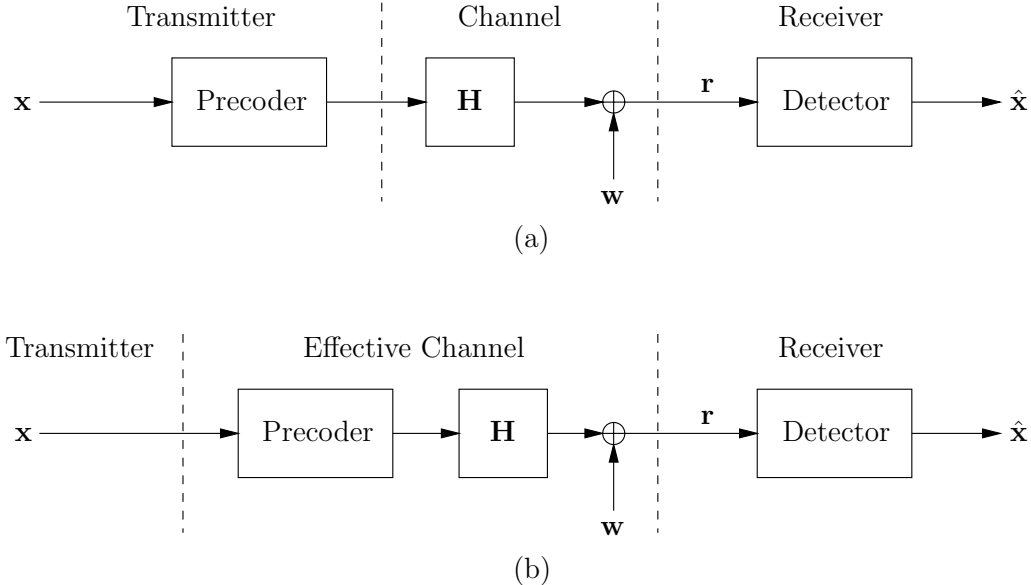


Figure 2.1: (a) A communication system with transmitter precoding. (b) The effective channel created by transmitter precoding.

the error probability of ML detection and the AWGN channel bound.

We can alternatively consider adding some special processing at the transmitter or *precoding* to make the channel appear differently, as shown in Fig. 2.1(a). The challenge here is that the transmitter is assumed to have no knowledge of the channel interference matrix \mathbf{H} . The precoder, then, must be universal in the sense that the ML detector for the effective channel created by the cascade of the precoder and virtually *any* given channel interference matrix \mathbf{H} has a lower error probability than the ML detector for the given channel \mathbf{H} without precoding. The effective channel, shown in Fig. 2.1(b), is assumed to be known at the receiver.

In this chapter, we introduce a universal precoding technique called *mode-interleaved precoding* that effectively transforms a channel matrix such as (2.1) so that the pairwise error probability between two vectors \mathbf{x} and \mathbf{y} , defined as the probability that \mathbf{x} is transmitted but the received vector $\mathbf{r} = \mathbf{H}\mathbf{x} + \mathbf{w}$ is closer to $\mathbf{H}\mathbf{y}$ than it is to $\mathbf{H}\mathbf{x}$, is asymptotically equal to the pairwise error probability for the corresponding AWGN channel (2.2). More generally, almost any $Q \times N$ channel matrix \mathbf{H} can be manipulated so that the pairwise

error probability is asymptotically as if the channel \mathbf{H} were the $N \times N$ matrix

$$\begin{bmatrix} \|\mathbf{H}\|_F / \sqrt{N} & 0 & \cdots & 0 \\ 0 & \|\mathbf{H}\|_F / \sqrt{N} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \|\mathbf{H}\|_F / \sqrt{N} \end{bmatrix}, \quad (2.3)$$

which is the AWGN channel at the same received SNR, defined in (1.6). Thus, with mode-interleaved precoding, it is the Frobenius norm $\|\mathbf{H}\|_F^2$ or equivalently the sum of the eigenvalues of $\mathbf{H}^\dagger \mathbf{H}$ [46] that asymptotically determines pairwise error performance, regardless of the given \mathbf{H} matrix.

In Section 2.1 we present mode-interleaved precoding, which is so named because the precoder is a special unitary matrix that spreads transmit symbols across all modes of the channel. In Section 2.2 we consider a specialized version of mode-interleaved precoding for ISI channels called *frequency-interleaved precoding*, in which the set of possible precoders is restricted to those that maintain the Toeplitz structure of \mathbf{H} . Frequency-interleaved precoding permits the use of low-complexity detectors at the receiver that exploit the Toeplitz structure, and in doing so also reduces the implementational complexity of the general mode-interleaved precoder. Finally, in Section 2.3 we discuss possible detection methods that can be used at the receiver in conjunction with mode- and frequency-interleaved precoding.

2.1 Mode-Interleaved Precoding

In this section, we develop of class of universal transmitter precoders under a set of reasonable design constraints. First, we restrict our attention to linear precoders, which can be represented by a matrix multiplication. As we shall demonstrate later in this section, simple linear precoders are sufficient to make ML detection for interference channels behave in a pairwise sense like ML detection for an AWGN channel with equal Frobenius norm. Second, we focus on linear precoders that correspond to multiplications by square matrices, so as not to change the required physical resources such as bandwidth, transmission duration, number of antennas, etc. Third, we concentrate on unitary matrix precoders, which do not change the statistics of the vector \mathbf{x} , which is typically uncorrelated with equal-variance components for both coded and uncoded cases [11]. In the absence of channel knowledge

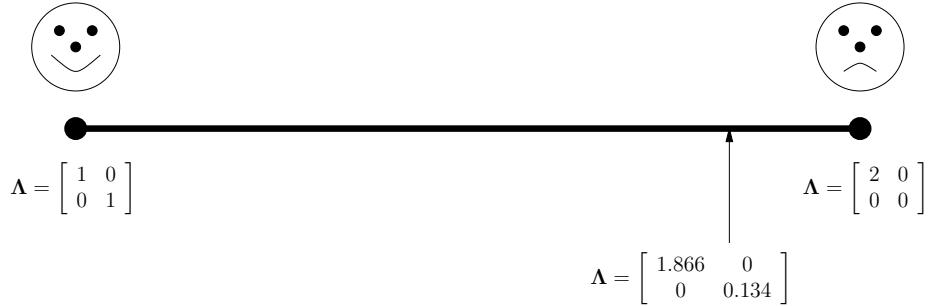


Figure 2.2: A spectrum for 2×2 \mathbf{H} matrices of Frobenius norm 2. At the left end, the two eigenvalues are equal; at the right end, the two eigenvalues are as unequal as possible. Matrices located near the left end tend to be more desirable in the sense that the distance between two vectors does not change as dramatically after being transmitted over the channel. The \mathbf{H} matrix in (2.1), which has eigenvalues of 1.866 and 0.134, is located near the right end of the spectrum.

at the transmitter, it makes sense to keep the statistics of the transmit vector the same.

To develop intuition about how to choose appropriate linear unitary precoders, let us consider a family of 2×2 \mathbf{H} matrices with constant Frobenius norm $\|\mathbf{H}\|_F$. Since the squared Frobenius norm of \mathbf{H} is equal to the sum of the eigenvalues of $\mathbf{H}^\dagger \mathbf{H} = \mathbf{V} \mathbf{\Lambda} \mathbf{V}^\dagger$ [46], it is also a family of 2×2 \mathbf{H} matrices in which $\text{Tr}(\mathbf{\Lambda})$, the trace of $\mathbf{\Lambda}$, is constant.

We can think of these 2×2 \mathbf{H} matrices as lying on a spectrum, as depicted in Fig. 2.2 for matrices with a Frobenius norm of 2. At one end of the spectrum we have equal eigenvalues, and at the other we have the most unequal pair of eigenvalues possible. The matrix in (2.1) is a member of this family and has eigenvalues that are 1.866 and 0.134, making it closer to the right end of the spectrum. Note that the corresponding AWGN channel, in which $\mathbf{H} = \mathbf{I}$, lies on the leftmost part of the spectrum.

The matrices on the right end of the spectrum tend not to be desirable because of the presence of near-zero modes. The action of a near-zero mode can be viewed as a projection of the signal space onto a lower dimensional subspace. In some cases, the rotation caused by the cascade of the unitary precoder and the unitary right singular matrix \mathbf{V} of the channel may be such that the projection due to near-zero modes keeps the \mathbf{x} vectors in the signal constellation distinguishable at the receiver, as shown in Fig. 2.3(a). In other cases, however, the rotation caused by the cascade of the unitary precoder and the unitary matrix \mathbf{V} of the channel may be such that near-zero modes make some \mathbf{x} vectors in the signal constellation virtually indistinguishable at the receiver even in the absence of noise,

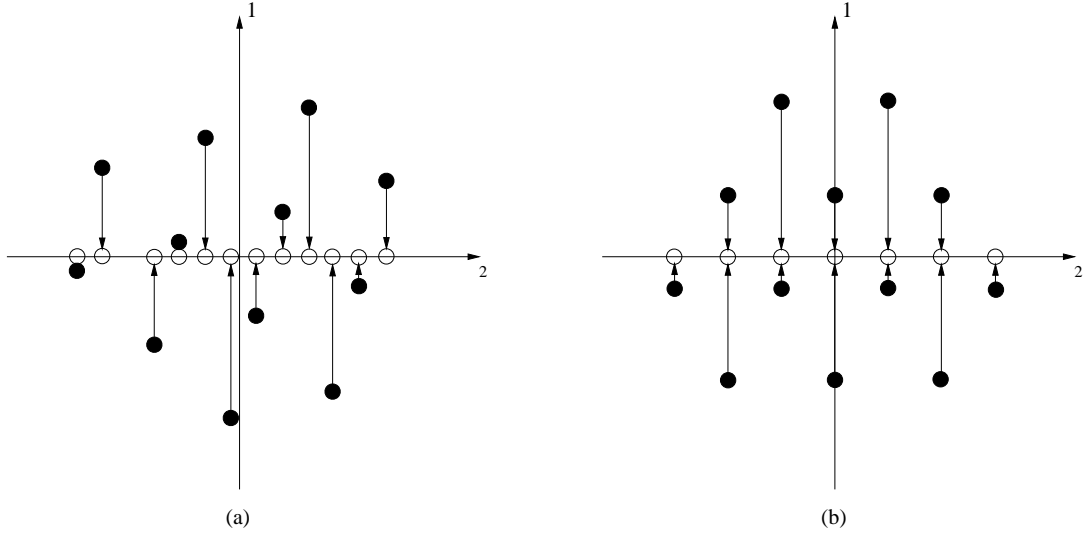


Figure 2.3: A two-dimensional signal constellation after a linear transformation by the precoder in cascade with a matrix \mathbf{H} with a zero mode. The black dots are constellation points after the possible \mathbf{x} vectors are rotated by the precoder and \mathbf{V} , and the white dots are the points after the two-dimensional space is scaled by the modes $\lambda_1 = 0$ and $\lambda_2 = 1$. (a) The combined rotation of the precoder and \mathbf{V} is such that all points remain distinguishable. (b) The combined rotation of the precoder and \mathbf{V} is such that some points become indistinguishable. The difference in rotation angle between the two examples is approximately 45° .

as shown in Fig. 2.3(b).

A pair of vectors becomes virtually indistinguishable at the receiver when, by the combined rotation of the precoder and \mathbf{V} , their difference vector becomes aligned with the eigenvector of $\mathbf{H}^\dagger \mathbf{H}$ corresponding to a near-zero mode. Therefore, to reduce the error probability of ML detection, it is desirable to somehow ensure that the difference vectors between all vector pairs are spread across many modes, so that an unfavorable mode does not significantly degrade the performance of ML detection. In the special case of uncoded vectors, this goal is equivalent to spreading each uncoded symbol in \mathbf{x} across many modes, so that information about any symbol is not blocked by a few near-zero modes from reaching the receiver.

Since the \mathbf{H} matrix is not known at the transmitter, it is not obvious how to choose a precoder so that the difference vectors of an arbitrary N -dimensional signal constellation are well-spread across all modes. When N is large, however, there is a method of choosing a precoder that works well, which further inspires a strategy for small values of N .

2.1.1 Large \mathbf{H} Matrices

Before we detail our precoder selection strategy for large N , let us introduce a couple of concepts.

We introduce a random ensemble of $N \times N$ unitary matrices that is isotropically distributed, meaning that an orthonormal column of a unitary matrix belonging to the ensemble has a marginal distribution corresponding to a uniformly distributed point on the surface of an N -dimensional sphere. Further concepts on isotropically distributed vectors and matrices, which prove useful in the sequel, are reviewed in Appendix A.

We also introduce the notion of pairwise error probability $\Pr(\mathbf{x} \rightarrow \mathbf{y}|\mathbf{H})$ between two vectors \mathbf{x} and \mathbf{y} , which is defined as the probability that \mathbf{x} is transmitted but the received vector $\mathbf{r} = \mathbf{H}\mathbf{x} + \mathbf{w}$ is closer to $\mathbf{H}\mathbf{y}$ than it is to $\mathbf{H}\mathbf{x}$. For AWGN vectors \mathbf{w} in which the one-dimensional noise variance is σ^2 , the pairwise error probability is

$$\Pr(\mathbf{x} \rightarrow \mathbf{y}|\mathbf{H}) = \mathcal{Q} \left(\sqrt{\frac{\|\mathbf{H}(\mathbf{y} - \mathbf{x})\|^2}{4\sigma^2}} \right), \quad (2.4)$$

where

$$\mathcal{Q}(v) = \frac{1}{\sqrt{2\pi}} \int_v^\infty e^{-t^2/2} dt. \quad (2.5)$$

Note that (2.4) is also equal to $\Pr(\mathbf{y} \rightarrow \mathbf{x}|\mathbf{H})$, the probability that \mathbf{y} is transmitted but the received vector $\mathbf{r} = \mathbf{H}\mathbf{y} + \mathbf{w}$ is closer to $\mathbf{H}\mathbf{x}$ than it is to $\mathbf{H}\mathbf{y}$, because of the symmetry in the distribution of \mathbf{w} .

When N is large, fixing the precoder to be a matrix chosen from the random ensemble of isotropically distributed $N \times N$ unitary matrices works well with high probability. We make this notion precise with the following theorem, which states that as $N \rightarrow \infty$, the pairwise error probability for two vectors transmitted over an interference channel is asymptotically equivalent to the pairwise error probability over the corresponding AWGN channel.

Theorem 2.1 *Let $\{\mathbf{H}_N\}$ for $N = 1, 2, 3, \dots$ be a sequence of matrices, where \mathbf{H}_N has N columns and the spectral radius of the eigenvalue matrices $\{\mathbf{\Lambda}_N\}$ of $\{\mathbf{H}_N^\dagger \mathbf{H}_N\}$ is uniformly bounded and $(1/N) \text{Tr}(\mathbf{\Lambda}_N) \rightarrow \bar{\lambda}$ as $N \rightarrow \infty$. Let $\{\mathbf{Y}_N\}$ be a sequence of unitary matrices drawn from statistically independent, isotropically distributed random ensembles. Let $\{\mathbf{x}_N\}$ and $\{\mathbf{y}_N\}$ for $N = 1, 2, 3, \dots$ be two sequences of vectors where N denotes the vector dimension, and let $\{\sigma_N^2\}$ be a sequence of noise variances such that $\|\mathbf{\Delta}_N\|^2/\sigma_N^2$ tends*

to a limit, where $\mathbf{\Delta}_N \triangleq \mathbf{y}_N - \mathbf{x}_N$. Then, as $N \rightarrow \infty$, the pairwise error probability is

$$\Pr(\mathbf{x}_N \rightarrow \mathbf{y}_N | \mathbf{H}_N \mathbf{\Upsilon}_N) = \mathcal{Q} \left(\sqrt{\frac{\|\mathbf{H}_N \mathbf{\Upsilon}_N \mathbf{\Delta}_N\|^2}{4\sigma_N^2}} \right) \xrightarrow{\text{a.s.}} \mathcal{Q} \left(\sqrt{\bar{\lambda} \lim_{N \rightarrow \infty} \frac{\|\mathbf{\Delta}_N\|^2}{4\sigma_N^2}} \right). \quad (2.6)$$

Before we proceed to the proof, let us make a few remarks. The theorem states that the asymptotic pairwise error probability of any two N -dimensional vectors transmitted over an interference channel is equal to the pairwise error probability over the corresponding AWGN channel. The requirement that $(1/N)\text{Tr}(\mathbf{\Lambda}_N) \rightarrow \bar{\lambda}$ as $N \rightarrow \infty$ is a normalization condition that is not critical to the essence of the theorem. It allows us, for example, to consider a sequence $\{\mathbf{H}_N\}$ for which the eigenvalues in $\{\mathbf{\Lambda}_N\}$ have a limiting cumulative distribution function and, thus, a limiting $(1/N)\text{Tr}(\mathbf{\Lambda}_N)$. Similarly, the normalization condition that $\{\|\mathbf{\Delta}_N\|^2/\sigma_N^2\}$ converge to a limit is also not critical to the essence of the theorem. What is important, however, is the condition that the spectral radius of the eigenvalue matrices $\{\mathbf{\Lambda}_N\}$ of $\{\mathbf{H}_N^\dagger \mathbf{H}_N\}$ be uniformly bounded. In combination with the normalization condition that $(1/N)\text{Tr}(\mathbf{\Lambda}_N) \rightarrow \bar{\lambda}$ as $N \rightarrow \infty$, uniformly bounded spectral radii ensure that the number of nonzero eigenvalues available in $\mathbf{\Lambda}_N$ over which to “spread” difference vectors goes to infinity as $N \rightarrow \infty$, allowing the limiting result of the theorem. Note that a growing number of nonzero eigenvalues implies that Q , the number of rows in \mathbf{H}_N , must grow with N .

Proof: The pairwise probability of error is given by

$$\Pr(\mathbf{x}_N \rightarrow \mathbf{y}_N | \mathbf{H}_N \mathbf{\Upsilon}_N) = \mathcal{Q} \left(\sqrt{\frac{\|\mathbf{H}_N \mathbf{\Upsilon}_N \mathbf{\Delta}_N\|^2}{4\sigma_N^2}} \right) = \mathcal{Q} \left(\sqrt{\frac{\mathbf{\Delta}_N^\dagger \mathbf{\Upsilon}_N^\dagger \mathbf{H}_N^\dagger \mathbf{H}_N \mathbf{\Upsilon}_N \mathbf{\Delta}_N}{4\sigma_N^2}} \right). \quad (2.7)$$

Since the eigenvalue decomposition of $\mathbf{H}_N^\dagger \mathbf{H}_N$ is $\mathbf{V}_N \mathbf{\Lambda}_N \mathbf{V}_N^\dagger$,

$$\Pr(\mathbf{x}_N \rightarrow \mathbf{y}_N | \mathbf{H}_N \mathbf{\Upsilon}_N) = \mathcal{Q} \left(\sqrt{\frac{\mathbf{\Delta}_N^\dagger \mathbf{\Upsilon}_N^\dagger \mathbf{V}_N \mathbf{\Lambda}_N \mathbf{V}_N^\dagger \mathbf{\Upsilon}_N \mathbf{\Delta}_N}{4\sigma_N^2}} \right). \quad (2.8)$$

Given that \mathbf{V}_N is unitary and that $\mathbf{\Upsilon}_N$ is an independent, isotropically distributed unitary matrix, it follows from Appendix A that the product $\mathbf{V}_N^\dagger \mathbf{\Upsilon}_N \mathbf{\Delta}_N$ is an isotropically distributed vector of norm $\|\mathbf{\Delta}_N\|$. Such vectors have the same distribution as $(\mathbf{z}_N / \sqrt{\mathbf{z}_N^\dagger \mathbf{z}_N}) \|\mathbf{\Delta}_N\|$, where \mathbf{z}_N is a N -dimensional vector of independent $\mathcal{CN}(0, 1)$ random variables. Thus

$\mathbf{\Delta}_N^\dagger \mathbf{\Upsilon}_N^\dagger \mathbf{V}_N \mathbf{\Lambda}_N \mathbf{V}_N^\dagger \mathbf{\Upsilon}_N \mathbf{\Delta}_N$ has the same distribution as

$$\frac{\frac{1}{N} \mathbf{z}_N^\dagger \mathbf{\Lambda}_N \mathbf{z}_N}{\frac{1}{N} \mathbf{z}_N^\dagger \mathbf{z}_N} \|\mathbf{\Delta}_N\|^2. \quad (2.9)$$

To evaluate the limit of (2.9) when $N \rightarrow \infty$, we rely on the following lemmas.

Lemma 2.1 ([5]) *Let \mathbf{z}_N denote a Gaussian vector with independent, identically distributed (i.i.d.) zero-mean, unit-variance, complex elements, and let $\mathbf{\Phi}_N$ be a deterministic $N \times N$ diagonal matrix. Then,*

$$E[|\mathbf{z}_N^\dagger \mathbf{\Phi}_N \mathbf{z}_N - \text{Tr}(\mathbf{\Phi}_N)|^4] \leq K \cdot \text{Tr}^2(\mathbf{\Phi}_N \mathbf{\Phi}_N^\dagger), \quad (2.10)$$

where K is a constant that does not depend on $\mathbf{\Phi}_N$ or N .

If the eigenvalues of $\mathbf{H}_N^\dagger \mathbf{H}_N$ are bounded by some constant η , then $\text{Tr}^2(\mathbf{\Lambda}_N \mathbf{\Lambda}_N^\dagger) \leq \eta^4 N^2$, so that from Lemma 2.1 we have that

$$E \left[\left| \frac{1}{N} \mathbf{z}_N^\dagger \mathbf{\Lambda}_N \mathbf{z}_N - \frac{1}{N} \text{Tr}(\mathbf{\Lambda}_N) \right|^4 \right] \leq \frac{K \eta^4}{N^2}. \quad (2.11)$$

We therefore have the following lemma.

Lemma 2.2 ([24]) *Let $\{\mathbf{\Phi}_N\}$ be a sequence of diagonal matrices such that $\frac{1}{N} \text{Tr}(\mathbf{\Phi}_N) \rightarrow \bar{\phi}$ as $N \rightarrow \infty$ and the spectral radius of $\{\mathbf{\Phi}_N\}$ is uniformly bounded. Then, with $\{\mathbf{z}_N\}$ denoting a sequence of Gaussian vectors with i.i.d., zero-mean, unit-variance complex elements, we have that $(1/N) \mathbf{z}_N^\dagger \mathbf{\Phi}_N \mathbf{z}_N \xrightarrow{\text{a.s.}} \bar{\lambda}$ as $N \rightarrow \infty$.*

Proof: From Markov's inequality and Lemma 2.1,

$$\Pr \left(\left| \frac{1}{N} \mathbf{z}_N^\dagger \mathbf{\Phi}_N \mathbf{z}_N - \frac{1}{N} \text{Tr}(\mathbf{\Phi}_N) \right| > \epsilon \right) \leq \frac{E \left[\left| \frac{1}{N} \mathbf{z}_N^\dagger \mathbf{\Phi}_N \mathbf{z}_N - \frac{1}{N} \text{Tr}(\mathbf{\Phi}_N) \right|^4 \right]}{\epsilon^4} \leq \frac{C}{N^2}, \quad (2.12)$$

for some constant C that does not depend on $\mathbf{\Phi}_N$ or N . Therefore,

$$\sum_{N=1}^{\infty} \Pr \left(\left| \frac{1}{N} \mathbf{z}_N^\dagger \mathbf{\Phi}_N \mathbf{z}_N - \frac{1}{N} \text{Tr}(\mathbf{\Phi}_N) \right| > \epsilon \right) < \infty, \quad (2.13)$$

and from the first Borel-Cantelli lemma [25], $\frac{1}{N} \mathbf{z}_N^\dagger \mathbf{\Phi}_N \mathbf{z}_N \xrightarrow{\text{a.s.}} \bar{\phi}$. \square

Applying Lemma 2.2 to the numerator of (2.9) and the strong law of large numbers to the denominator gives

$$\Pr(\mathbf{x}_N \rightarrow \mathbf{y}_N | \mathbf{H}_N \mathbf{\Upsilon}_N) \xrightarrow{\text{a.s.}} \mathcal{Q} \left(\sqrt{\bar{\lambda}} \lim_{P \rightarrow \infty} \frac{\|\mathbf{\Delta}_N\|^2}{4\sigma_N^2} \right). \quad (2.14)$$

□

We have defined a class of transmitter precoders for large N that are drawn from a random unitary ensemble. In practice, one chooses a particular precoder from the random ensemble which may be used for many transmissions and perhaps for changing channel conditions. When N is large, we can see from the proof of the theorem that with high probability, the selected precoder spreads *any* difference vector evenly over all modes of virtually *any* channel. Hence, the name we give this kind of transmitter precoding is *mode-interleaved precoding*. Mode-interleaved precoding does not change the unequal modes of the interference channel and does not require knowledge of the channel at the transmitter. Yet under mild conditions, the resulting effective channel has properties which make the pairwise error probability asymptotically the same as for an AWGN channel at the same SNR.

Figure 2.4 shows simulated plots of the pairwise error probability averaged over the ensemble of isotropically distributed unitary precoders,

$$E_{\mathbf{z}_N} \left[\mathcal{Q} \left(\sqrt{\frac{\mathbf{z}_N^\dagger \mathbf{\Delta}_N \mathbf{z}_N / N}{\mathbf{z}_N^\dagger \mathbf{z}_N / N} \alpha^2} \right) \right], \quad (2.15)$$

as a function of $\alpha^2 = \|\mathbf{\Delta}_N\|^2 / 4\sigma_N^2$ for the case in which half of the N eigenvalues of $\mathbf{H}^\dagger \mathbf{H}$ have the value 0.5 and the other half have the value 1.5. For a fixed value of $\alpha^2 = \|\mathbf{\Delta}_N\|^2 / 4\sigma_N^2$, the plot shows that the average pairwise error probability over this class of interference channels approaches the pairwise error probability of the corresponding interference-free channel as $N \rightarrow \infty$.

Theorem 2.1 shows that with a unitary precoding matrix $\mathbf{\Upsilon}$ selected from the ensemble of independent, isotropically distributed unitary matrices, the ability at the receiver to select the more likely of two vectors is not hindered by a transformation by the matrix \mathbf{H} . What the theorem does not show, however, is whether the total probability of ML detection error given that \mathbf{x} is transmitted, $\Pr(\text{error}|\mathbf{x})$, is asymptotically made worse by the transformation

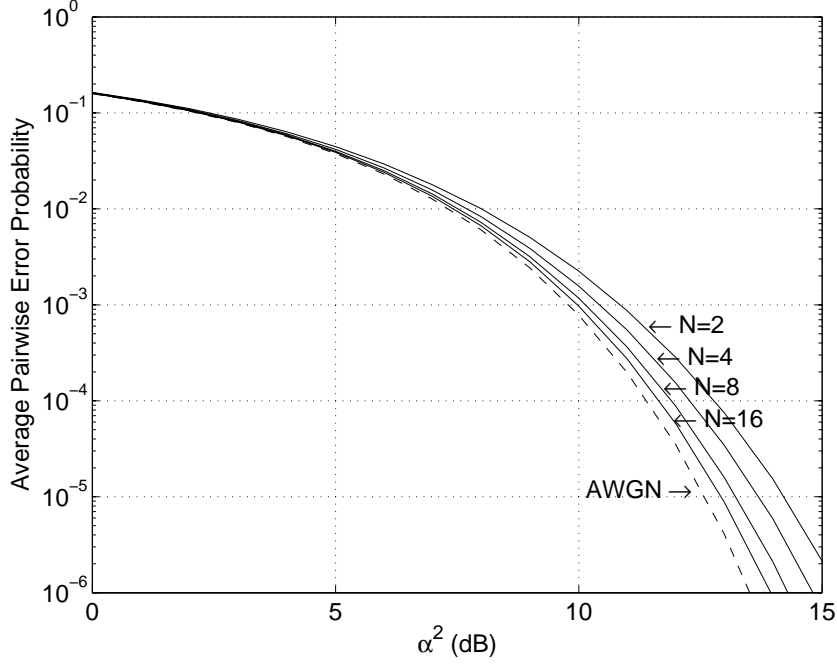


Figure 2.4: Average pairwise error probability as a function of $\alpha^2 = \|\mathbf{\Delta}_N\|^2/4\sigma_N^2$ when half of the eigenvalues of $\mathbf{H}^\dagger\mathbf{H}$ are 0.5 and half are 1.5, and the eigenvector matrix is an independent and isotropically distributed unitary matrix.

by the matrix \mathbf{H} when precoding is present. For fixed N , we can lower bound $\Pr(\text{error}|\mathbf{x}_N)$ using the pairwise error probability between \mathbf{x}_N and the minimum distance vector \mathbf{y}_N :

$$\Pr(\text{error}|\mathbf{x}_N) \geq \max_{\mathbf{y}_N} \Pr(\mathbf{x}_N \rightarrow \mathbf{y}_N | \mathbf{H}_N \mathbf{Y}_N). \quad (2.16)$$

From Theorem 2.1, we know that as $N \rightarrow \infty$ this lower bound converges to the pairwise error probability of the corresponding AWGN channel, provided that the conditions of the theorem are satisfied. For fixed N , we can also obtain an upper bound by applying the union bound:

$$\Pr(\text{error}|\mathbf{x}_N) \leq \sum_{\mathbf{y}_N \neq \mathbf{x}_N} \Pr(\mathbf{x}_N \rightarrow \mathbf{y}_N | \mathbf{H}_N \mathbf{Y}_N). \quad (2.17)$$

If the rate per dimension is kept constant as N grows, then the number of signal vectors in the constellation grows exponentially in N . It is not clear whether the limit $\mathbf{z}_N^\dagger \mathbf{\Lambda}_N \mathbf{z}_N / \mathbf{z}_N^\dagger \mathbf{z}_N \xrightarrow{\text{a.s.}} \bar{\lambda}$ converges fast enough so that the summation of an exponentially growing number of terms in (2.17) approaches the corresponding error probability on an AWGN channel. It is also not clear if it is sufficient to consider only the terms corresponding

to minimum-distance vectors, the so-called “union-bound estimate.” We do know, however, that simulations in Chapter 3 show that the symbol-error rate for ML detection of uncoded symbols asymptotically reaches the AWGN channel bound with mode-interleaved precoding at high SNR. In the case of ML detection of coded vectors with mode-interleaved precoding, we know at least that the probability of ML detection error for an interference channel cannot approach that for the corresponding AWGN channel when transmitting at a rate higher than the mutual information of the interference channel but lower than the capacity of the AWGN channel. We speculate that at high SNR, the probability of ML detection error for an interference channel may be able to approach that for the corresponding AWGN channel at rates up to, say, the cut-off rate for the interference channel, but at this point any formal statements are premature without further analysis.

There are some special cases in which using an isotropically distributed unitary precoder is redundant. The proof of the Theorem 2.1 depends on the fact that the cascade $\mathbf{V}^\dagger \mathbf{Y}$ of the right singular matrix \mathbf{V} and the independent, isotropically distributed unitary precoder \mathbf{Y} is also an isotropically distributed unitary matrix independent of the modes. This property of $\mathbf{V}^\dagger \mathbf{Y}$ ensures that any difference vector is spread across all modes, preventing it from encountering only a few small modes. However, we sometimes deal with systems with a large \mathbf{H} matrix whose right singular vector matrix \mathbf{V} can be modelled as independent and isotropically distributed. In such cases, the addition of a precoder at the transmitter is unnecessary. For example, i.i.d. Gaussian matrices, which are often used to model rich-scattering multiple-input multiple-output (MIMO) channels and CDMA systems, have this property, as shown by the following theorem.

Theorem 2.2 *Let \mathbf{H} be a $Q \times N$ matrix whose elements are i.i.d. Gaussian, and let $\mathbf{V} \mathbf{\Lambda} \mathbf{V}^\dagger$ be the eigenvalue decomposition of the associated Wishart matrix $\mathbf{H}^\dagger \mathbf{H}$. Then \mathbf{V} is an isotropically distributed unitary matrix independent of $\mathbf{\Lambda}$.*

This theorem holds for \mathbf{H} matrices of any dimension, not just when N and Q grow to infinity. Although this result can be found in the statistical literature (see, for example, [49]), we provide a straightforward proof.

Proof: Let $\mathbf{\Psi}$ be an independent, isotropically distributed unitary matrix. By premul-

tipling and postmultiplying $\mathbf{H}^\dagger\mathbf{H}$ by Ψ and Ψ^\dagger respectively, we have that

$$(\mathbf{H}\Psi^\dagger)^\dagger(\mathbf{H}\Psi^\dagger) = (\Psi\mathbf{V})\mathbf{\Lambda}(\Psi\mathbf{V})^\dagger. \quad (2.18)$$

Let us examine the left-hand side of (2.18). Since the elements of \mathbf{H} are i.i.d. Gaussian, \mathbf{H} is an isotropically distributed matrix. With \mathbf{H} being isotropically distributed and Ψ being a unitary matrix, $\mathbf{H}\Psi^\dagger$ has the same distribution as \mathbf{H} , and consequently $(\mathbf{H}\Psi^\dagger)^\dagger(\mathbf{H}\Psi^\dagger)$ has the same distribution as $\mathbf{H}^\dagger\mathbf{H}$.

We now focus on the right-hand side of (2.18). Note that $\Psi\mathbf{V}$ is a unitary matrix and $\mathbf{\Lambda}$ is a diagonal matrix, so the right-hand side of (2.18) is an eigenvalue decomposition. Now, since Ψ is an isotropically distributed unitary matrix and \mathbf{V} is unitary, the matrix $\Psi\mathbf{V}$ is also an isotropically distributed unitary matrix. Furthermore, $\Psi\mathbf{V}$ and $\mathbf{\Lambda}$ are independent since Ψ is independent of $\mathbf{\Lambda}$. Thus we conclude that \mathbf{V} is an isotropically distributed unitary matrix independent of $\mathbf{\Lambda}$. \square

In this i.i.d. Gaussian case and others in which the right singular matrix \mathbf{V} can be modelled as an independent, isotropically distributed unitary matrix, mode-interleaved precoding is redundant and thus unnecessary.

2.1.2 Small \mathbf{H} Matrices

Theorem 2.1 suggests a precoding strategy only for large channel matrices, not small ones. However, when the dimensions Q and N of \mathbf{H} are relatively small, a reasonable strategy is to effectively transform the small system into a large system with new dimensions Q' and N' , and then to apply the precoding strategy already established for large matrices.

We can convert a small channel matrix into a large channel matrix by aggregating consecutive channels uses into one big channel use. The channel matrix may be constant during these consecutive channel uses (e.g. block fading), or the channel matrix may vary from use to use (e.g. fast fading). In this section, for notational convenience we assume the channel is constant, but these techniques are easily extendable to varying channel matrices.

Suppose we wish to aggregate four consecutive uses of a 2×3 channel. The resulting

continues to modulate the same τ data symbols during $\tau - 1$ subsequent symbol periods, but with different i.i.d. Gaussian signature sequences. The received vector during the i th symbol period is thus

$$\mathbf{r}_i = \sum_{k=0}^{\tau-1} \mathbf{H}_{ik} \mathbf{x}_k \quad (2.21)$$

where each of the τ \mathbf{x}_k 's is an $N \times 1$ vector containing one data symbol for each of the N users, and \mathbf{H}_{ik} is the corresponding $Q \times N$ matrix of signatures scaled by channel amplitudes. For each k , there is a new set of data symbols and a new set of signatures. Stacking up τ consecutive received vectors, we get

$$\begin{bmatrix} \mathbf{r}_0 \\ \mathbf{r}_1 \\ \mathbf{r}_2 \\ \vdots \\ \mathbf{r}_{\tau-1} \end{bmatrix} = \begin{bmatrix} \mathbf{H}_{00} & \mathbf{H}_{01} & \mathbf{H}_{02} & \dots & \mathbf{H}_{0,\tau-1} \\ \mathbf{H}_{10} & \mathbf{H}_{11} & \mathbf{H}_{12} & & \mathbf{H}_{1,\tau-1} \\ \mathbf{H}_{20} & \mathbf{H}_{21} & \mathbf{H}_{22} & & \mathbf{H}_{2,\tau-1} \\ \vdots & & & \ddots & \vdots \\ \mathbf{H}_{\tau-1,0} & \mathbf{H}_{\tau-1,1} & \mathbf{H}_{\tau-1,2} & \dots & \mathbf{H}_{\tau-1,\tau-1} \end{bmatrix} \begin{bmatrix} \mathbf{x}_0 \\ \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_{\tau-1} \end{bmatrix} + \begin{bmatrix} \mathbf{w}_0 \\ \mathbf{w}_1 \\ \mathbf{w}_2 \\ \vdots \\ \mathbf{w}_{\tau-1} \end{bmatrix}. \quad (2.22)$$

The effective channel matrix is thus i.i.d. Gaussian (assuming the CDMA system employs power control), and according to Theorem 2.2 mode-interleaved precoding is redundant and thus unnecessary at the transmitter.

In short, no matter the size of the given channel matrix, mode-interleaved precoding uses a large matrix drawn independently from the ensemble of isotropically distributed unitary matrices in order to condition the channel so that the pairwise error probability at the receiver is asymptotically the same as for an AWGN channel.

2.1.3 Properties of the Effective Channel

To give some further intuition as to why the new effective channel asymptotically leads to the same pairwise error probability as the corresponding AWGN channel, let us compare the properties of $\mathbf{\Upsilon}_N^\dagger \mathbf{H}_N^\dagger \mathbf{H}_N \mathbf{\Upsilon}_N$ with $\mathbf{H}_N^\dagger \mathbf{H}_N$ and $\mathbf{G}_N^\dagger \mathbf{G}_N$, where \mathbf{G}_N is the AWGN channel of equal SNR, $\mathbf{G}_N = \frac{1}{\sqrt{N}} \|\mathbf{H}_N\|_F \mathbf{I}_N$.

Using the fact that the Frobenius norm is unitarily invariant, the matrix $\mathbf{G}_N^\dagger \mathbf{G}_N$ corre-

sponding to AWGN channel can be expressed as

$$\begin{aligned}
\mathbf{G}_N^\dagger \mathbf{G}_N &= \frac{1}{N} \|\mathbf{H}_N\|_F^2 \mathbf{I}_N \\
&= \frac{1}{N} \|\boldsymbol{\Sigma}_N\|_F^2 \mathbf{I}_N \\
&= \frac{1}{N} \text{Tr}(\boldsymbol{\Lambda}_N) \mathbf{I}_N.
\end{aligned} \tag{2.23}$$

Thus, under the assumption that $(1/N)\text{Tr}(\boldsymbol{\Lambda}_N) \rightarrow \bar{\lambda}$ for the sequence of interference channels, the diagonal components of $\mathbf{G}_N^\dagger \mathbf{G}_N$ approach $\bar{\lambda}$ and the off diagonal components are equal to zero.

In the matrix $\mathbf{H}_N^\dagger \mathbf{H}_N$, the components have no particular structure other than a Hermitian constraint:

$$\mathbf{H}_N^\dagger \mathbf{H}_N = \begin{bmatrix} \alpha_{00} & \alpha_{01} & \cdots & \alpha_{0,N-1} \\ \alpha_{01}^* & \alpha_{11} & \cdots & \alpha_{1,N-1} \\ \vdots & & \ddots & \vdots \\ \alpha_{0,N-1}^* & \alpha_{1,N-1}^* & \cdots & \alpha_{N-1,N-1} \end{bmatrix}. \tag{2.24}$$

Specifically, some of the diagonal components α_{ii} can be much smaller than $\bar{\lambda}$, and some of the off-diagonal components α_{ij} can be quite significant.

Using the unitary invariance of the Frobenius norm, we have that

$$\|\boldsymbol{\Upsilon}_N^\dagger \mathbf{H}_N^\dagger \mathbf{H}_N \boldsymbol{\Upsilon}_N\|_F = \|\mathbf{H}_N^\dagger \mathbf{H}_N\|_F. \tag{2.25}$$

Thus, the sums of the squared components of each matrix are equal, but the components are distributed differently in each matrix. The matrix $\boldsymbol{\Upsilon}_N^\dagger \mathbf{H}_N^\dagger \mathbf{H}_N \boldsymbol{\Upsilon}_N$ can be written as

$$\boldsymbol{\Upsilon}_N^\dagger \mathbf{H}_N^\dagger \mathbf{H}_N \boldsymbol{\Upsilon}_N = \begin{bmatrix} \omega_{00} & \omega_{01} & \cdots & \omega_{0,N-1} \\ \omega_{01}^* & \omega_{11} & \cdots & \omega_{1,N-1} \\ \vdots & & \ddots & \vdots \\ \omega_{0,N-1}^* & \omega_{1,N-1}^* & \cdots & \omega_{N-1,N-1} \end{bmatrix}. \tag{2.26}$$

Since $\boldsymbol{\Upsilon}_N^\dagger \mathbf{H}_N^\dagger \mathbf{H}_N \boldsymbol{\Upsilon}_N = \boldsymbol{\Upsilon}_N^\dagger \mathbf{V}_N \boldsymbol{\Lambda}_N \mathbf{V}_N^\dagger \boldsymbol{\Upsilon}_N$ and each column of $\mathbf{V}_N^\dagger \boldsymbol{\Upsilon}_N$ is marginally an isotropically distributed unit vector, we know that each diagonal component of $\boldsymbol{\Upsilon}_N^\dagger \mathbf{H}_N^\dagger \mathbf{H}_N \boldsymbol{\Upsilon}_N$

has the same marginal distribution as

$$\frac{\frac{1}{N}\mathbf{z}_N^\dagger\mathbf{\Lambda}_N\mathbf{z}_N}{\frac{1}{N}\mathbf{z}_N^\dagger\mathbf{z}_N}, \quad (2.27)$$

where \mathbf{z}_N is a N -dimensional vector of independent $\mathcal{CN}(0, 1)$ random variables. Then, under the assumptions that $(1/N)\text{Tr}(\mathbf{\Lambda}_N) \rightarrow \bar{\lambda}$ and the spectral radius of $\{\mathbf{\Lambda}_N\}$ is uniformly bounded, we can apply Lemma 2.2 to the numerator of (2.27) and the strong law of large numbers to the denominator to show that each diagonal component approaches $\bar{\lambda}$ with probability 1, i.e., $\omega_{ii} \xrightarrow{\text{a.s.}} \bar{\lambda}$. We can also do a similar analysis for the off-diagonal components of $\mathbf{\Upsilon}_N^\dagger\mathbf{H}_N^\dagger\mathbf{H}_N\mathbf{\Upsilon}_N$ to show that each off-diagonal component approaches zero with probability 1. Since each off-diagonal component involves two distinct columns of the isotropically distributed unitary matrix $\mathbf{V}_N^\dagger\mathbf{\Upsilon}_N$, it follows that any two distinct eigenvectors have a joint distribution identical to that obtained by Gram-Schmidt orthogonalizing two independent Gaussian vectors. Specifically, if \mathbf{y}_N and \mathbf{z}_N are two independent N -dimensional vectors of independent $\mathcal{CN}(0, 1)$ random variables, then one column of $\mathbf{V}_N^\dagger\mathbf{\Upsilon}_N$ has the same distribution as $\mathbf{z}_N/\sqrt{\mathbf{z}_N^\dagger\mathbf{z}_N}$ and the other has the same distribution as

$$\frac{\mathbf{y}_N - \left(\frac{\mathbf{z}_N^\dagger\mathbf{y}_N}{\sqrt{\mathbf{z}_N^\dagger\mathbf{z}_N}} \right) \cdot \frac{\mathbf{z}_N}{\sqrt{\mathbf{z}_N^\dagger\mathbf{z}_N}}}{\sqrt{\left(\mathbf{y}_N - \left(\frac{\mathbf{z}_N^\dagger\mathbf{y}_N}{\sqrt{\mathbf{z}_N^\dagger\mathbf{z}_N}} \right) \cdot \frac{\mathbf{z}_N}{\sqrt{\mathbf{z}_N^\dagger\mathbf{z}_N}} \right)^\dagger \left(\mathbf{y}_N - \left(\frac{\mathbf{z}_N^\dagger\mathbf{y}_N}{\sqrt{\mathbf{z}_N^\dagger\mathbf{z}_N}} \right) \cdot \frac{\mathbf{z}_N}{\sqrt{\mathbf{z}_N^\dagger\mathbf{z}_N}} \right)}}, \quad (2.28)$$

which can be simplified to

$$\frac{\mathbf{y}_N - \left(\frac{\mathbf{z}_N^\dagger\mathbf{y}_N}{\mathbf{z}_N^\dagger\mathbf{z}_N} \right) \mathbf{z}_N}{\sqrt{\mathbf{y}_N^\dagger\mathbf{y}_N - \frac{|\mathbf{z}_N^\dagger\mathbf{y}_N|^2}{\mathbf{z}_N^\dagger\mathbf{z}_N}}}. \quad (2.29)$$

Thus each off-diagonal component of $\mathbf{\Upsilon}_N^\dagger\mathbf{H}_N^\dagger\mathbf{H}_N\mathbf{\Upsilon}_N = \mathbf{\Upsilon}_N^\dagger\mathbf{V}_N\mathbf{\Lambda}_N\mathbf{V}_N^\dagger\mathbf{\Upsilon}_N$ has the same distribution as

$$\frac{\frac{1}{N}\mathbf{z}_N^\dagger\mathbf{\Lambda}_N \left(\mathbf{y}_N - \left(\frac{\mathbf{z}_N^\dagger\mathbf{y}_N}{\mathbf{z}_N^\dagger\mathbf{z}_N} \right) \mathbf{z}_N \right)}{\sqrt{\frac{1}{N}\mathbf{z}_N^\dagger\mathbf{z}_N} \sqrt{\frac{1}{N} \left(\mathbf{y}_N^\dagger\mathbf{y}_N - \frac{|\mathbf{z}_N^\dagger\mathbf{y}_N|^2}{\mathbf{z}_N^\dagger\mathbf{z}_N} \right)}} = \frac{\frac{1}{N}\mathbf{z}_N^\dagger\mathbf{\Lambda}_N\mathbf{y}_N - \frac{1}{N}\mathbf{z}_N^\dagger\mathbf{\Lambda}_N\mathbf{z}_N \left(\frac{\mathbf{z}_N^\dagger\mathbf{y}_N}{\mathbf{z}_N^\dagger\mathbf{z}_N} \right)}{\sqrt{\frac{1}{N}\mathbf{z}_N^\dagger\mathbf{z}_N} \sqrt{\frac{1}{N}\mathbf{y}_N^\dagger\mathbf{y}_N - \frac{1}{N}\frac{|\mathbf{z}_N^\dagger\mathbf{y}_N|^2}{\mathbf{z}_N^\dagger\mathbf{z}_N}}}. \quad (2.30)$$

To evaluate the limit of (2.30) when $N \rightarrow \infty$, we rely on the following lemmas.

Lemma 2.3 ([24]) *Let \mathbf{y}_N and \mathbf{z}_N denote independent Gaussian vectors with i.i.d., zero mean, unit variance, complex elements, and let Φ_N be a deterministic $N \times N$ diagonal matrix. Then,*

$$E[|\mathbf{z}_N^\dagger \Phi_N \mathbf{y}_N|^4] \leq K \text{Tr}^2(\Phi_N \Phi_N^\dagger), \quad (2.31)$$

where K is a constant that does not depend on Φ_N or N .

If the eigenvalues of $\mathbf{H}_N^\dagger \mathbf{H}_N$ are bounded by some constant γ , then $\text{Tr}^2(\Lambda_N \Lambda_N^\dagger) \leq \gamma^4 N^2$, so that from Lemma 2.3 we have that

$$E \left[\left| \frac{1}{N} \mathbf{z}_N^\dagger \Lambda_N \mathbf{y}_N \right|^4 \right] \leq \frac{K \gamma^4}{N^2}. \quad (2.32)$$

We therefore have the following lemma.

Lemma 2.4 ([24]) *Let $\{\Phi_N\}$ be a sequence of diagonal matrices such that the spectral radius of $\{\Phi_N\}$ is uniformly bounded. Then, with $\{\mathbf{y}_N\}$ and $\{\mathbf{z}_N\}$ denoting two independent sequences of Gaussian vectors with i.i.d., zero mean, unit variance complex elements, we have that $(1/N) \mathbf{z}_N^\dagger \Phi_N \mathbf{y}_N \xrightarrow{\text{a.s.}} 0$ as $N \rightarrow \infty$.*

Proof: From Markov's inequality and Lemma 2.3,

$$\Pr \left(\left| \frac{1}{N} \mathbf{z}_N^\dagger \Phi_N \mathbf{y}_N \right| > \epsilon \right) \leq \frac{E \left[\left| \frac{1}{N} \mathbf{z}_N^\dagger \Phi_N \mathbf{y}_N \right|^4 \right]}{\epsilon^4} \leq \frac{C}{N^2}, \quad (2.33)$$

for some constant C that does not depend on Φ_N or N . Therefore,

$$\sum_{N=1}^{\infty} \Pr \left(\left| \frac{1}{N} \mathbf{z}_N^\dagger \Phi_N \mathbf{y}_N \right| > \epsilon \right) < \infty, \quad (2.34)$$

and from the first Borel-Cantelli lemma [25], $\frac{1}{N} \mathbf{z}_N^\dagger \Phi_N \mathbf{y}_N \xrightarrow{\text{a.s.}} 0$. \square

Then, under the assumptions that $(1/N) \text{Tr}(\Lambda_N) \rightarrow \bar{\lambda}$ and the spectral radius of $\{\Lambda_N\}$ is uniformly bounded, we can apply Lemmas 2.2 and 2.4 to (2.30) and conclude that each off-diagonal component approaches zero with probability 1, i.e., $\alpha_{ij} \xrightarrow{\text{a.s.}} 0$.

In summary, the effect of the mode-interleaved precoder Υ is to take the interference components of $\mathbf{H}_N^\dagger \mathbf{H}_N$ and redistribute the same amount of interference so that $\Upsilon^\dagger \mathbf{H}_N^\dagger \mathbf{H}_N \Upsilon$ approaches $\mathbf{G}^\dagger \mathbf{G}$ for the AWGN channel in a componentwise manner. In particular, large off-diagonal interference components in $\mathbf{H}_N^\dagger \mathbf{H}_N$ are spread over the other components so

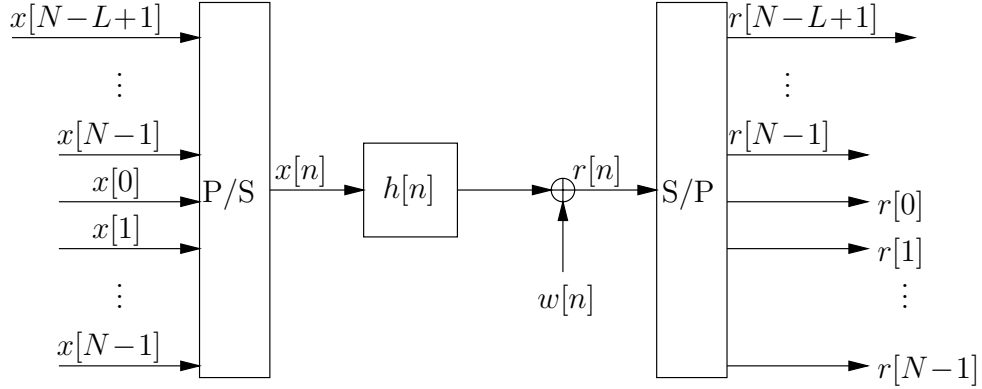


Figure 2.5: ISI channel model with a cyclic prefix.

that the energy becomes more benign. We emphasize that $\|\mathbf{\Upsilon}^\dagger \mathbf{H}_N^\dagger \mathbf{H}_N \mathbf{\Upsilon}\|_F = \|\mathbf{H}_N^\dagger \mathbf{H}_N\|_F$, which means that interference is not eliminated by mode-interleaved precoding, but rather spread out more evenly amongst all the matrix components.

2.2 Frequency-Interleaved Precoding

If we interpret the ISI channel as the matrix equation (1.5), then we may use a mode-interleaved precoder $\mathbf{\Upsilon}$ at the transmitter to randomly rotate the data sequence. However, the effective channel matrix $\mathbf{H}\mathbf{\Upsilon}$ would no longer be Toeplitz, and any low-complexity detection methods at the receiver that take advantage of the Toeplitz structure of the channel would no longer be useful. In this section, we resolve this issue by restricting the class of random unitary precoders so that the effective channel matrix remains Toeplitz.

The first step in defining an appropriate restricted class of unitary matrices is to slightly modify the ISI model of Fig. 1.4 by transmitting the last $L - 1$ symbols in data sequence before the entire sequence of length N is sent, where L greater than or equal to the length of the channel impulse response. This redundancy, depicted in Fig. 2.5, is commonly referred to as a cyclic prefix. Thus, if the original matrix equation for a two-tap ISI channel is

$$\underbrace{\begin{bmatrix} r[0] \\ r[1] \\ r[2] \\ r[3] \end{bmatrix}}_{\mathbf{r}} = \underbrace{\begin{bmatrix} h[0] & 0 & 0 & 0 \\ h[1] & h[0] & 0 & 0 \\ 0 & h[1] & h[0] & 0 \\ 0 & 0 & h[1] & h[0] \end{bmatrix}}_{\mathbf{H}} \underbrace{\begin{bmatrix} x[0] \\ x[1] \\ x[2] \\ x[3] \end{bmatrix}}_{\mathbf{x}} + \underbrace{\begin{bmatrix} w[0] \\ w[1] \\ w[2] \\ w[3] \end{bmatrix}}_{\mathbf{w}}, \quad (2.35)$$

then with a cyclic prefix the resulting matrix equation is

$$\underbrace{\begin{bmatrix} r[0] \\ r[1] \\ r[2] \\ r[3] \end{bmatrix}}_{\mathbf{r}} = \underbrace{\begin{bmatrix} h[0] & 0 & 0 & h[1] \\ h[1] & h[0] & 0 & 0 \\ 0 & h[1] & h[0] & 0 \\ 0 & 0 & h[1] & h[0] \end{bmatrix}}_{\tilde{\mathbf{H}}} \underbrace{\begin{bmatrix} x[0] \\ x[1] \\ x[2] \\ x[3] \end{bmatrix}}_{\mathbf{x}} + \underbrace{\begin{bmatrix} w[0] \\ w[1] \\ w[2] \\ w[3] \end{bmatrix}}_{\mathbf{w}}. \quad (2.36)$$

In addition to being Toeplitz, the modified channel matrix $\tilde{\mathbf{H}}$ is now also circulant, so the ISI channel model now uses a circular convolution as opposed to a linear convolution. A useful property of circulant matrices that we shall exploit is that the eigenvalues of a circulant matrix are exactly the discrete Fourier transform (DFT) coefficients of the first column of the matrix, and the eigenvector matrix is the DFT matrix. This makes $\tilde{\mathbf{H}}$ decomposable as

$$\tilde{\mathbf{H}} = \mathbf{F}^\dagger \mathbf{\Lambda}^{1/2} \mathbf{F} \quad (2.37)$$

where \mathbf{F} is the DFT matrix and $\mathbf{\Lambda}^{1/2} \triangleq \text{diag}(H[0], H[1], H[2], H[3])$.

We constrain \mathbf{Y} to belong to the subset of all unitary matrices that can be expressed as $\mathbf{Y} = \mathbf{F}^\dagger \mathbf{P} \mathbf{F}$ where \mathbf{P} is a permutation matrix, defined as a square matrix containing exactly a single one in each row and column and zeros elsewhere. A precoder is chosen from this restricted set with all permutation matrices \mathbf{P} being equally likely, which means that all unitary matrices in the set are equally likely.

Figure 2.6 shows the entire precoded system. At the transmitter, the data sequence is processed by a DFT, a random permutation, and an inverse DFT. A cyclic prefix is appended, and the resulting sequence is sent through the ISI channel. At the receiver front end, the cyclic prefix is removed and the remaining sequence is processed by a DFT, the inverse of the permutation, and an inverse DFT. Note that the operations performed at the receiver front end are the inverse operations of the transmitter in reverse order. It is important to emphasize that the transmitter has no information about the channel impulse response other than an upper bound L on its length. However, the transmitter and receiver must agree on which random permutation to use *a priori*. Using (2.37), the received vector

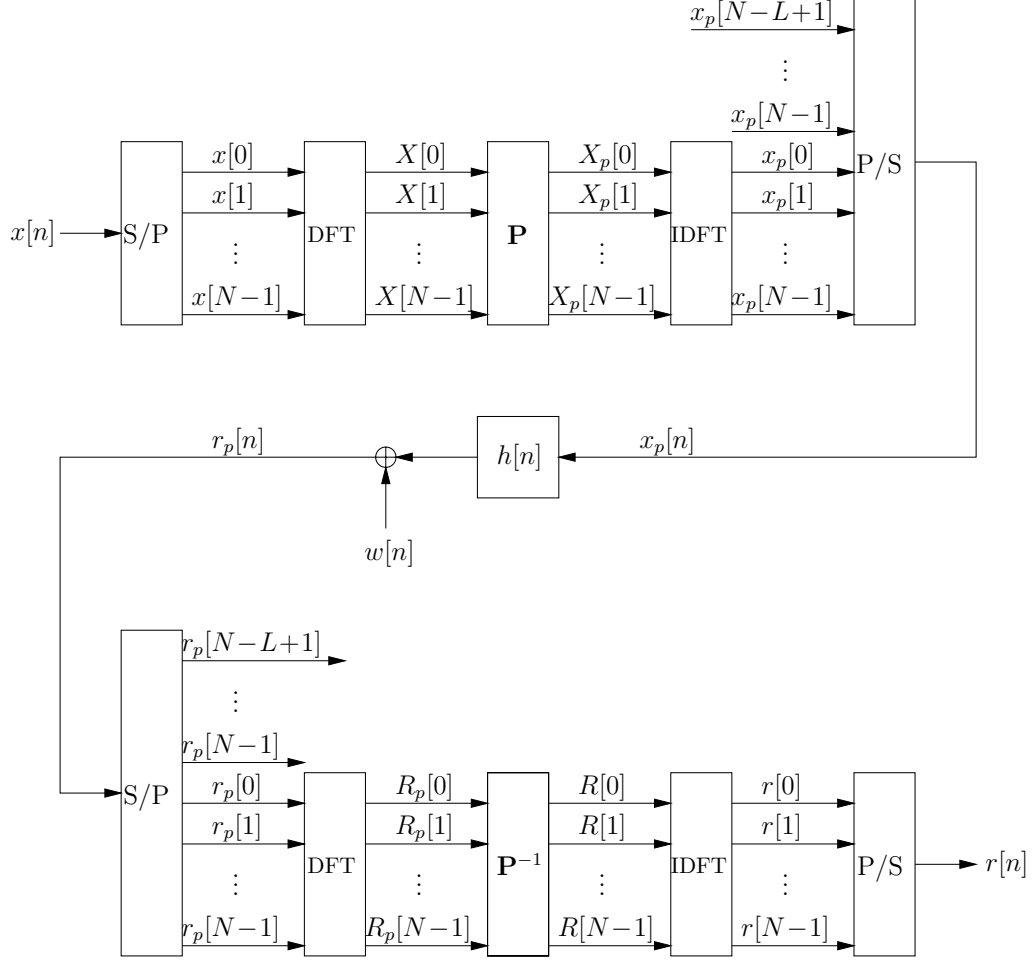


Figure 2.6: Frequency-interleaved precoded system.

is

$$\begin{aligned}
 \mathbf{r} &= (\mathbf{F}^\dagger \mathbf{P}^{-1} \mathbf{F})(\mathbf{F}^\dagger \mathbf{\Lambda}^{1/2} \mathbf{F})(\mathbf{F}^\dagger \mathbf{P} \mathbf{F}) \mathbf{x} + (\mathbf{F}^\dagger \mathbf{P}^{-1} \mathbf{F}) \mathbf{w} \\
 &= \mathbf{F}^\dagger (\mathbf{P}^{-1} \mathbf{\Lambda}^{1/2} \mathbf{P}) \mathbf{F} \mathbf{x} + \mathbf{w}_p,
 \end{aligned} \tag{2.38}$$

where \mathbf{w}_p has the same statistics as \mathbf{w} because $\mathbf{F}^\dagger \mathbf{P}^{-1} \mathbf{F}$ is a unitary matrix. The effective channel matrix $\tilde{\mathbf{H}}_p \triangleq \mathbf{F}^\dagger (\mathbf{P}^{-1} \mathbf{\Lambda}^{1/2} \mathbf{P}) \mathbf{F}$ has \mathbf{F} as its eigenvector matrix, so $\tilde{\mathbf{H}}_p$ is circulant. Furthermore, since the eigenvalue matrix of $\tilde{\mathbf{H}}_p$ is $\mathbf{P}^{-1} \mathbf{\Lambda}^{1/2} \mathbf{P}$, the DFT of the effective ISI channel is the randomly permuted DFT of the original channel. Hence, we give this kind of precoding the name *frequency-interleaved precoding*. So the effect of this restricted class of unitary precoders is to shuffle the modes of $\mathbf{H}^\dagger \mathbf{H}$ so that their order is independent of the modal (DFT) matrix. Since these precoders preserve the Toeplitz structure of the channel

matrix (provided there is a cyclic prefix), low-complexity detectors that exploit the Toeplitz structure can be used at the receiver.

We can alternatively view the effect of frequency-interleaved precoding in the time and DFT domains. At the transmitter in Fig. 2.6, a sequence of N data symbols $x[n]$ is interleaved in the frequency domain, so that the time sequence $x_p[n]$ corresponding to the N -point DFT $X_p[k]$ is obtained, where $X_p[k] \triangleq X[p(k)]$, $p(\cdot)$ is a random permutation of the set $\mathcal{S} = \{0, 1, 2, \dots, N - 1\}$, and $X[k]$ is the DFT of $x[n]$. Appended to the time sequence is a cyclic prefix of length $L - 1$. The resulting signal $x_p[n]$ is transmitted over the ISI channel. The symbols $x_p[n]$ are corrupted by a convolution with the impulse response of the channel, $h[n]$, and by additive noise, $w[n]$, to produce the received symbols

$$r_p[n] = \sum_m h[m]x_p[n - m] + w[n]. \quad (2.39)$$

At the receiver front end, the cyclic prefix is removed from the sequence $r_p[n]$, and the remaining sequence is deinterleaved in the frequency domain, so that the N -point DFT $R[k]$ is obtained. The relationship between $R[k]$ and $X[k]$ is then

$$R[k] = H_{p^{-1}}[k]X[k] + W_{p^{-1}}[k], \quad (2.40)$$

where $H_{p^{-1}}[k] \triangleq H[p^{-1}(k)]$ and $W_{p^{-1}}[k] \triangleq W[p^{-1}(k)]$, $p^{-1}(\cdot)$ is the inverse permutation of $p(\cdot)$, and $H[k]$ and $W[k]$ are the DFTs of $h[n]$ and $w[n]$ respectively. Note that the use of a cyclic prefix causes $X[k]$ to be pointwise multiplied by $H_{p^{-1}}[k]$, corresponding to a circular convolution in the time domain. We again come to the conclusion that frequency-interleaved precoding shuffles the DFT coefficients of the original channel.

For general mode-interleaved precoding, Theorem 2.1 states that the pairwise error probability of any two transmit vectors over an interference channel is asymptotically the same as over the corresponding AWGN channel. However, for frequency-interleaved precoding a similar result can be shown only for *most*, not all, pairs of transmit vectors. The transmit pairs for which the result is not true have error vectors whose DFTs contains many zeros. Such error vectors cannot be spread across the different frequencies, and may lead to indistinguishable transmit vectors at the receiver. This problem is due to the use of a restricted subset of mode-interleaved precoders defined using DFT matrices. Nevertheless, we shall

see in Section 3.2 that in practice, such error events seem rare and do not have a noticeable impact on overall performance.

Note that this class of precoders are defined using DFTs, they can be implemented using fast Fourier transforms (FFTs) with the low complexity of $N \log N$ rather than N^3 for general rotations. This observation also motivates the use of this restricted subset as a low-complexity alternative to isotropically distributed unitary matrices even for general channel matrices that do not have Toeplitz structure.

2.2.1 Properties of the Effective Channel

As was similarly done for mode-interleaved precoding in Section 2.1.3, we now compare the properties of $\tilde{\mathbf{H}}_p^\dagger \tilde{\mathbf{H}}_p$ with the corresponding $\mathbf{H}^\dagger \mathbf{H}$ for frequency-interleaved precoding.

The original channel matrix associated with a two-tap unit-energy channel impulse response is of the form

$$\mathbf{H} = \begin{bmatrix} h[0] & 0 & 0 & \dots & 0 \\ h[1] & h[0] & 0 & & 0 \\ 0 & h[1] & \ddots & & 0 \\ \vdots & & \ddots & h[0] & \vdots \\ 0 & 0 & \dots & h[1] & h[0] \end{bmatrix}. \quad (2.41)$$

The cascade of the channel matrix with its matched filter is thus the tridiagonal matrix

$$\mathbf{H}^\dagger \mathbf{H} = \begin{bmatrix} 1 & \alpha & 0 & \dots & 0 \\ \alpha^* & 1 & \alpha & & 0 \\ 0 & \alpha^* & \ddots & \ddots & \vdots \\ \vdots & & \ddots & 1 & \alpha \\ 0 & 0 & \dots & \alpha^* & 1 \end{bmatrix} \quad (2.42)$$

where $\alpha = h[0]h^*[1]$. As N increases, the constants on the two non-zero off-diagonals do not change.

In contrast, the effective channel matrix has the form

$$\tilde{\mathbf{H}}_p = \begin{bmatrix} h_{p-1}[0] & h_{p-1}[N-1] & h_{p-1}[N-2] & \dots & h_{p-1}[1] \\ h_{p-1}[1] & h_{p-1}[0] & h_{p-1}[N-1] & & h_{p-1}[2] \\ h_{p-1}[2] & h_{p-1}[1] & \ddots & & h_{p-1}[3] \\ \vdots & & \ddots & h_{p-1}[0] & \vdots \\ h_{p-1}[N-1] & h_{p-1}[N-2] & \dots & h_{p-1}[1] & h_{p-1}[0] \end{bmatrix} \quad (2.43)$$

where the effective channel impulse response can be expressed as

$$h_{p-1}[n] = \frac{1}{N} \sum_{k=0}^{N-1} H_{p-1}[k] e^{j \frac{2\pi k}{N} n} = \frac{1}{N} \sum_{k=0}^{N-1} H[k] e^{j \frac{2\pi p(k)}{N} n} \quad (2.44)$$

for $0 \leq n \leq N-1$. The second-order statistics of $h_{p-1}[n]$ are given by

$$\begin{aligned} E[h_{p-1}[n]] &= \frac{1}{N} \sum_{k=0}^{N-1} H[k] E \left[e^{j \frac{2\pi p(k)}{N} n} \right] \\ &= \left(\frac{1}{N} \sum_{k=0}^{N-1} H[k] \right) \delta[n] \\ &= h[0] \delta[n] \end{aligned} \quad (2.45)$$

and

$$\begin{aligned} E[h_{p-1}[n] h_{p-1}^*[m]] &= \frac{1}{N^2} \sum_{k=0}^{N-1} \sum_{i=0}^{N-1} H[k] H^*[i] E \left[e^{\frac{2\pi}{N} (p(k)n - p(i)m)} \right] \\ &= \begin{cases} |h[0]|^2 & n = m = 0 \\ \frac{1}{N-1} \sum_{i=1}^{N-1} |h[i]|^2 & n = m \neq 0 \\ 0 & \text{otherwise} \end{cases} \end{aligned} \quad (2.46)$$

where the expectation is with respect to all permutations. Thus, the coefficients of the resulting effective channel $h_{p-1}[n]$ are uncorrelated, with the energy in all but the zeroth tap of the channel impulse response statistically spread out evenly. Using the second-order statistics along with tools similar to those developed in Appendix B, the cascade of a filter

matched to the frequency-interleaved precoded ISI channel can be shown to have the form

$$\tilde{\mathbf{H}}_p^\dagger \tilde{\mathbf{H}}_p = \begin{bmatrix} 1 & \epsilon_1 & \epsilon_2 & \cdots & \epsilon_{N-1} \\ \epsilon_1^* & 1 & \epsilon_1 & & \epsilon_{N-2} \\ \epsilon_2^* & \epsilon_1^* & \ddots & \ddots & \vdots \\ \vdots & & \ddots & 1 & \epsilon_1 \\ \epsilon_{N-1}^* & \epsilon_{N-2}^* & \cdots & \epsilon_1^* & 1 \end{bmatrix} \quad (2.47)$$

where $E[\epsilon_i = 0]$ and $\lim_{N \rightarrow \infty} \text{var}(\epsilon_i) = 0$. It is important to note that the Frobenius norms (sum of the off-diagonal energies) of both $\mathbf{H}^\dagger \mathbf{H}$ and $\tilde{\mathbf{H}}_p^\dagger \tilde{\mathbf{H}}_p$ are always equal for the same N . In other words, large off-diagonal components in $\mathbf{H}^\dagger \mathbf{H}$ are spread out more evenly by frequency-interleaved precoding and thus made more benign.

2.3 Detection for Mode- and Frequency-Interleaved Precoded Systems

Mode- and frequency-interleaved precoding can potentially be used in conjunction with any detection scheme, and in this section we examine some of the possibilities.

ML detection by brute force methods can deal arbitrary channels, but with a data vector of length N , the complexity is exponential in N , making ML detection generally intractable. For ISI channels, maximum-likelihood sequence detection (MLSD) can usually be efficiently implemented using the Viterbi algorithm (linear in the data vector length and exponential in the channel impulse response length), but the Viterbi algorithm would need to be modified somehow to account for the circular rather than linear convolution of the effective channel. Even if the circular convolution could be taken into account, the length of the effective channel would be equal to the length N of the data sequence, and the complexity would still be exponential in N . Because of this extreme complexity, we have not attempted simulations of mode-interleaved precoding with ML detection. Rather, we will show in the next chapter that such a system would theoretically achieve AWGN channel performance.

Suboptimal detection schemes such as linear detection can still be used. In the ISI case, the filters can be easily modified to deal with a channel corresponding to a circular convolution by simply implementing the filters in the DFT domain. Essentially, the input

signal to the filter is transformed to the DFT domain and then multiplied by the DFT of the filter to generate the DFT of the output signal. However, mode- and frequency-interleaved precoding do not change the mean-square error (MSE) for either ZF or MMSE linear detection. The MSE for ZF linear detection is

$$\varepsilon_{\text{ZF-LE}} = \frac{1}{N} \text{Tr} (\mathcal{N}_0 \mathbf{\Lambda}_N^{-1}), \quad (2.48)$$

and that for MMSE linear detection is

$$\varepsilon_{\text{MMSE-LE}} = \frac{1}{N} \text{Tr} \left(\mathcal{N}_0 \left(\frac{\mathcal{N}_0}{\mathcal{E}_s} \mathbf{I} + \mathbf{\Lambda}_N \right)^{-1} \right). \quad (2.49)$$

Mode and frequency interleaving leave the set of eigenvalues of $\mathbf{H}^\dagger \mathbf{H}$ unchanged, so (2.48) and (2.49) remain unchanged.

While decision-feedback detectors are hard to analyze in the general case, we can gain some insight regarding their effectiveness by examining decision-feedback equalizers in the ISI case. The feedforward filter of a decision-feedback equalizer can also be implemented in the DFT domain to deal with the circular convolution of the effective channel, but the situation for the feedback filter is not so straightforward. Processing in the DFT domain requires the entire input block to be available before any output information is determined, but part of the feedback filter output is required while the input to the feedback filter is still arriving sequentially. While some additional modification to the decision-feedback equalizer may make it compatible with frequency-interleaved precoding, there may not be any motivation to do so. Like the MSE of its linear equalization counterparts, the MSE of the ZF-DFE and the MMSE-DFE remain unchanged with frequency interleaving. From the MSE expressions

$$\varepsilon_{\text{ZF-DFE}} = \exp \left\{ \frac{1}{N} \sum_{k=0}^{N-1} \ln \frac{\mathcal{N}_0}{|H[k]|^2} \right\} \quad (2.50)$$

and

$$\varepsilon_{\text{MMSE-DFE}} = \exp \left\{ \frac{1}{N} \sum_{k=0}^{N-1} \ln \frac{\mathcal{N}_0}{\mathcal{N}_0/\mathcal{E}_s + |H[k]|^2} \right\}, \quad (2.51)$$

it is clear that frequency interleaving has no effect on the MSE.

Chapter 3

The Iterated-Decision Detector

In this chapter, we introduce iterated-decision detection, which is designed to exploit the kind of large-system randomness created by mode-interleaved precoding. The iterated-decision detector uses the large-system randomness to make convenient Gaussian assumptions (central limit theorem) and to predict certain key parameters (law of large numbers). Figure 3.1 shows that the iterated-decision detector achieves the AWGN channel bound for uncoded transmission at high SNR when mode-interleaved precoding is used. By implication, ML detection in combination with precoding must also achieve the AWGN channel bound, since ML detection is asymptotically optimal in a symbol-error rate sense. Therefore, for uncoded transmission, mode-interleaved precoding simultaneously ensures that ML detection achieves the AWGN channel bound when given virtually any channel, and that iterated-decision detection provides an excellent approximation in a symbol-error rate sense to ML detection at high SNR.

In Sections 3.1 and 3.2, we focus on the basic theory and fundamental limits of the iterated-decision detector for uncoded systems in which the receiver has accurate knowledge of \mathbf{H} . Specifically, in Section 3.1 we describe the structure of the iterated-decision detector and optimize to maximize the signal-to-interference+noise ratio (SINR), taking into account the reliability of tentative decisions. We also present an asymptotic performance analysis of the iterated-decision detector when a mode-interleaved precoder is used at the transmitter. The analysis shows that the AWGN channel bound can be reached for uncoded transmission over a wide range of practical channels, and the validity of these theoretical results are verified by simulations. Section 3.2 is the corresponding section for the special case of iterated-decision equalization for ISI channels. Because of the special Toeplitz structure of the ISI channel matrix, iterated-decision equalization can be imple-

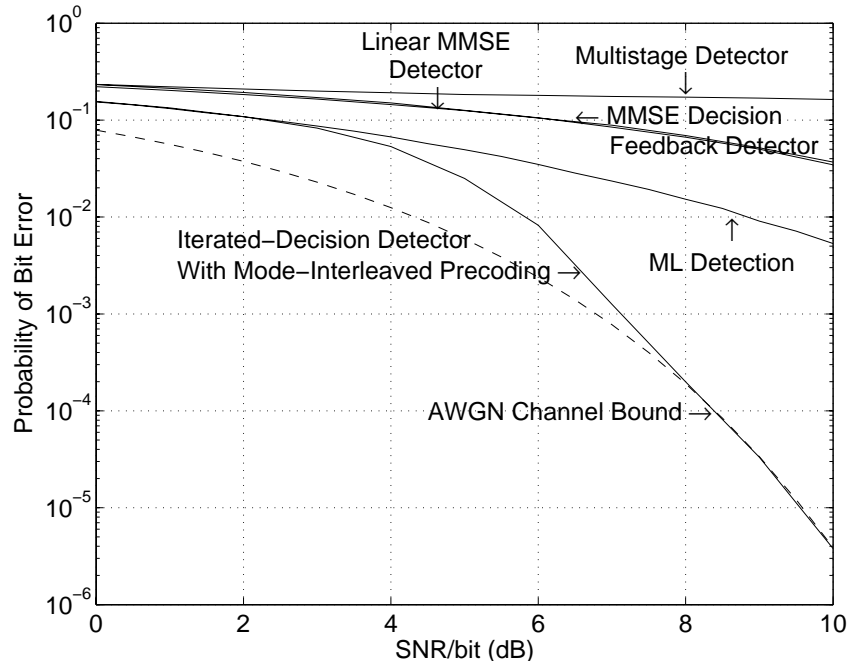


Figure 3.1: Bit-error probability of various detection schemes as a function of SNR for the interference channel of (2.1). Except for the iterated-decision detector, the various detectors are not used in conjunction with mode-interleaved precoding. Since the iterated-decision detector with mode-interleaved precoding achieves the AWGN channel bound at high SNR, we can infer that ML detection with mode-interleaved precoding (not pictured) would also achieve the AWGN channel bound.

mented with lower complexity. In Section 3.3, we describe adaptive implementations of both iterated-decision detection and iterated-decision equalization in which \mathbf{H} is not known *a priori* at the receiver. Examining the fixed and adaptive scenarios separately and comparing their results allows us to isolate the effects of tracking \mathbf{H} from overall detector behavior. It turns out that only a modest amount of training symbols is required at high SNR for the adaptive detector to perform as if \mathbf{H} were known exactly at the receiver. We reemphasize that in all these cases, we restrict our attention to transmitters that have no knowledge of \mathbf{H} .

3.1 Iterated-Decision Detection

Iterated-decision detectors are structurally related to the multistage detectors [64, 32] discussed in Section 1.3.4, in that they both generate tentative decisions for all symbols at each iteration and subsequently use these to cancel interference at the next iteration. However, unlike the multistage detector in Fig. 1.15, the iterated-decision detectors explored in this

chapter are optimized to maximize the signal-to-interference+noise ratio (SINR) at each iteration while explicitly taking into account the reliability of tentative decisions. With each successive iteration, increasingly refined hard decisions are generated using this strategy, with no limit cycles or divergence.

3.1.1 Structure

The structure of the iterated-decision detector is depicted in Fig. 3.2. The parameters of all systems and signals associated with the l th pass are denoted using the superscript l . On the l th pass of the equalizer where $l = 1, 2, 3, \dots$, the received vector \mathbf{r} is first premultiplied by a $N \times Q$ matrix $\mathbf{B}^{l\dagger} = [\mathbf{b}_1^l | \dots | \mathbf{b}_N^l]^\dagger$, producing the $N \times 1$ vector

$$\tilde{\mathbf{r}}^l = \mathbf{B}^{l\dagger} \mathbf{r}. \quad (3.1)$$

The matrix \mathbf{B}^l is constrained such that $\tilde{\mathbf{r}}^l$ is an unbiased estimate of \mathbf{x} , which means that $\mathbf{b}_i^{l\dagger} \mathbf{h}_i = 1$ for $i = 1, 2, \dots, N$. Next, an appropriately constructed estimate $\hat{\mathbf{z}}^l$ of the interference is constructed, where

$$\hat{\mathbf{z}}^l = \mathbf{D}^{l\dagger} \hat{\mathbf{x}}^{l-1}, \quad (3.2)$$

with $\mathbf{D}^l = [\mathbf{d}_1^l | \dots | \mathbf{d}_N^l]$ being a $N \times N$ matrix. (In subsequent analysis, we will show that $\hat{\mathbf{x}}^0$ is never required for the first iteration, so the vector may remain undefined.) Since $\hat{\mathbf{z}}^l$ is intended to be some kind of interference estimate, we restrict attention to the case in which the diagonal of \mathbf{D}^l is zero:

$$(\mathbf{D}^l)_{11} = (\mathbf{D}^l)_{22} = \dots = (\mathbf{D}^l)_{NN} = 0. \quad (3.3)$$

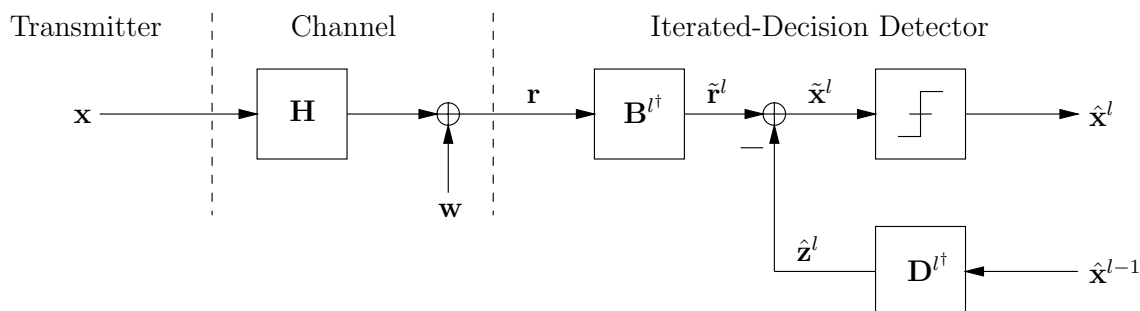


Figure 3.2: Iterated-decision detection.

The estimate $\hat{\mathbf{z}}^l$ is subtracted from $\tilde{\mathbf{r}}^l$, i.e.,

$$\tilde{\mathbf{x}}^l = \tilde{\mathbf{r}}^l - \hat{\mathbf{z}}^l, \quad (3.4)$$

and a bank of slicers then generates the $N \times 1$ vector of hard decisions $\hat{\mathbf{x}}^l$ from $\tilde{\mathbf{x}}^l$ using a symbol-wise minimum-distance rule.

3.1.2 Characterizing the Slicer Input

The composite system consisting of \mathbf{H} in cascade with l iterations of the multipass detector can be conveniently characterized when \mathbf{H} is known at the receiver. Let \mathbf{x} and $\hat{\mathbf{x}}^{l-1}$ be vectors of zero-mean uncorrelated symbols with energy \mathcal{E}_s , and let their normalized correlation matrix be expressed in the form

$$\frac{E[\mathbf{x} \cdot \hat{\mathbf{x}}^{l-1\dagger}]}{\mathcal{E}_s} = \boldsymbol{\rho}^{l-1} \triangleq \text{diag}\{\rho_1^{l-1}, \rho_2^{l-1}, \dots, \rho_N^{l-1}\}, \quad (3.5)$$

where ρ_i^{l-1} can be interpreted as a measure of the reliability of \hat{x}_i^{l-1} . Moreover, let \mathbf{D}^l satisfy the natural requirement (3.3). Then, the slicer input \tilde{x}_i^l defined via (3.4) with (3.1), (3.2) and (1.1) satisfies, for $i = 1, 2, \dots, N$,

$$\tilde{x}_i^l = x_i + u_i^l \quad (3.6)$$

where u_i^l is complex-valued, zero-mean, and uncorrelated with x_i , having variance

$$\begin{aligned} \text{var } u_i^l &= \mathcal{N}_0 \mathbf{b}_i^{l\dagger} \mathbf{b}_i^l + \mathcal{E}_s \left(\mathbf{d}_i^l - \boldsymbol{\rho}^{l-1\dagger} (\mathbf{b}_i^{l\dagger} \mathbf{H} - \mathbf{e}_i^\dagger)^\dagger \right)^\dagger \left(\mathbf{d}_i^l - \boldsymbol{\rho}^{l-1\dagger} (\mathbf{b}_i^{l\dagger} \mathbf{H} - \mathbf{e}_i^\dagger)^\dagger \right) \\ &\quad + \mathcal{E}_s (\mathbf{b}_i^{l\dagger} \mathbf{H} - \mathbf{e}_i^\dagger) (\mathbf{I} - \boldsymbol{\rho}^{l-1} \boldsymbol{\rho}^{l-1\dagger}) (\mathbf{b}_i^{l\dagger} \mathbf{H} - \mathbf{e}_i^\dagger)^\dagger, \end{aligned} \quad (3.7)$$

with \mathbf{I} being the identity matrix, and \mathbf{e}_i being the $N \times 1$ column vector that is zero except for the i th element, which is unity.

The variance in (3.7) is determined as follows. First, we write

$$u_i^l = \mathbf{b}_i^{l\dagger} \mathbf{w} + \tilde{\mathbf{c}}_i^{l\dagger} \mathbf{x} - \mathbf{d}_i^{l\dagger} \hat{\mathbf{x}}^{l-1} \quad (3.8)$$

with

$$\tilde{\mathbf{c}}_i^l = \mathbf{c}_i^l - \mathbf{e}_i \quad (3.9)$$

and

$$\mathbf{c}_i^{l\dagger} = \mathbf{b}_i^{l\dagger} \mathbf{H}. \quad (3.10)$$

We obtain the mean of u_i^l as

$$E[u_i^l] = \mathbf{b}_i^{l\dagger} E[\mathbf{w}] + \tilde{\mathbf{c}}_i^{l\dagger} E[\mathbf{x}] - \mathbf{d}_i^{l\dagger} E[\hat{\mathbf{x}}^{l-1}] = 0. \quad (3.11)$$

Since \mathbf{w} is statistically independent of \mathbf{x} and can be assumed to be independent of $\hat{\mathbf{x}}_i^{l-1}$,

$$\begin{aligned} \text{var } u_i^l &= \mathcal{N}_0 \mathbf{b}_i^{l\dagger} \mathbf{b}_i^l + \mathcal{E}_s (\mathbf{d}_i^{l\dagger} \mathbf{d}_i^l - \tilde{\mathbf{c}}_i^{l\dagger} \boldsymbol{\rho}^{l-1} \mathbf{d}_i^l - \mathbf{d}_i^{l\dagger} \boldsymbol{\rho}^{l-1\dagger} \tilde{\mathbf{c}}_i^l + \tilde{\mathbf{c}}_i^{l\dagger} \tilde{\mathbf{c}}_i^l) \\ &= \mathcal{N}_0 \mathbf{b}_i^{l\dagger} \mathbf{b}_i^l + \mathcal{E}_s \left(\mathbf{d}_i^l - \boldsymbol{\rho}^{l-1\dagger} \tilde{\mathbf{c}}_i^l \right)^\dagger \left(\mathbf{d}_i^l - \boldsymbol{\rho}^{l-1\dagger} \tilde{\mathbf{c}}_i^l \right) + \mathcal{E}_s \tilde{\mathbf{c}}_i^{l\dagger} (\mathbf{I} - \boldsymbol{\rho}^{l-1} \boldsymbol{\rho}^{l-1\dagger}) \tilde{\mathbf{c}}_i^l \end{aligned} \quad (3.12)$$

which, using (3.9) and (3.10), yields (3.7).

The second-order model of (3.6) and (3.7) turns out to be a useful one for analyzing and optimizing the performance of the iterated-decision detector. Using (3.6), the signal-to-interference+noise ratio (SINR) at the i th slicer input during each pass can be written as

$$\gamma_i^l(\mathbf{b}_i^l, \mathbf{d}_i^l) = \frac{\mathcal{E}_s}{\text{var } u_i^l}. \quad (3.13)$$

Since the detector uses a symbol-by-symbol decision device, a natural detector design strategy involves maximizing the SINR of the i th data symbol over all \mathbf{b}_i^l and \mathbf{d}_i^l .

3.1.3 Filter Optimization

For a given filter \mathbf{b}_i^l , it is straightforward to find the optimal filter \mathbf{d}_i^l . In particular, note that \mathbf{d}_i^l appears only in a non-negative denominator term of the SINR expression given by (3.13) and (3.7), and that term can be made exactly zero by setting

$$\mathbf{d}_i^l = \boldsymbol{\rho}^{l-1\dagger} (\mathbf{b}_i^{l\dagger} \mathbf{H} - \mathbf{e}_i^\dagger)^\dagger \quad \text{for } i = 1, 2, \dots, N \quad (3.14)$$

or, equivalently,

$$\mathbf{D}^l = \boldsymbol{\rho}^{l-1\dagger} \left(\mathbf{B}^{l\dagger} \mathbf{H} - \mathbf{I} \right)^\dagger. \quad (3.15)$$

Using (3.14), the SINR expression in (3.13) now simplifies to

$$\gamma_i^l(\mathbf{b}_i^l) = \frac{\mathcal{E}_s}{\mathcal{N}_0 \mathbf{b}_i^{l\dagger} \mathbf{b}_i^l + \mathcal{E}_s (\mathbf{b}_i^{l\dagger} \mathbf{H} - \mathbf{e}_i^\dagger) (\mathbf{I} - \boldsymbol{\rho}^{l-1} \boldsymbol{\rho}^{l-1\dagger}) (\mathbf{b}_i^{l\dagger} \mathbf{H} - \mathbf{e}_i^\dagger)^\dagger}. \quad (3.16)$$

This result for \mathbf{d}_i^l is intuitively satisfying. If $\hat{\mathbf{x}}^{l-1} = \mathbf{x}$ so that $\boldsymbol{\rho}^{l-1} = \mathbf{I}$, then the inner product $\mathbf{D}^{l\dagger} \hat{\mathbf{x}}^{l-1}$ exactly reproduces the interference component of $\tilde{\mathbf{r}}^l$. More generally, $\boldsymbol{\rho}^{l-1}$ describes our confidence in the quality of the estimate $\hat{\mathbf{x}}^{l-1}$. If $\hat{\mathbf{x}}^{l-1}$ is a poor estimate of \mathbf{x} , then $\boldsymbol{\rho}^{l-1}$ will in turn be close to zero, and consequently a smaller weighting is applied to the interference estimate that is to be subtracted from $\tilde{\mathbf{r}}^l$. On the other hand, if $\hat{\mathbf{x}}^{l-1}$ is an excellent estimate of \mathbf{x} , then $\boldsymbol{\rho}^{l-1} \approx \mathbf{I}$, and nearly all of the interference is subtracted from $\tilde{\mathbf{r}}^l$. Note that the diagonal of \mathbf{D}^l is indeed zero, as stipulated by (3.3).

Next, we optimize the vector \mathbf{b}_i^l . The identity

$$\begin{aligned} & (\mathbf{b}_i^{l\dagger} \mathbf{H} - \mathbf{e}_i^\dagger) (\mathbf{I} - \boldsymbol{\rho}^{l-1} \boldsymbol{\rho}^{l-1\dagger}) (\mathbf{b}_i^{l\dagger} \mathbf{H} - \mathbf{e}_i^\dagger)^\dagger \\ &= (\mathbf{b}_i^{l\dagger} \mathbf{H}) (\mathbf{I} - \boldsymbol{\rho}^{l-1} \boldsymbol{\rho}^{l-1\dagger}) (\mathbf{b}_i^{l\dagger} \mathbf{H})^\dagger - (1 - (\rho_i^{l-1})^2) \end{aligned} \quad (3.17)$$

can be used to rewrite (3.16) as

$$\gamma_i^l(\mathbf{b}_i^l) = \frac{1}{\frac{1}{\phi_i^l(\mathbf{b}_i^l)} - (1 - (\rho_i^{l-1})^2)}, \quad (3.18)$$

where

$$\phi_i^l(\mathbf{b}_i^l) = \frac{\mathcal{E}_s}{\mathbf{b}_i^{l\dagger} [\mathcal{N}_0 \mathbf{I} + \mathcal{E}_s \mathbf{H} (\mathbf{I} - \boldsymbol{\rho}^{l-1} \boldsymbol{\rho}^{l-1\dagger}) \mathbf{H}^\dagger] \mathbf{b}_i^l}. \quad (3.19)$$

Using the constraint that $\tilde{\mathbf{r}}^l$ be an unbiased estimate of \mathbf{x} , followed by the Schwarz inequality, we have¹

$$\begin{aligned} 1 &= |\mathbf{b}_i^{l\dagger} \mathbf{h}_i|^2 \\ &= \left| \mathbf{b}_i^{l\dagger} [\mathcal{N}_0 \mathbf{I} + \mathcal{E}_s \mathbf{H} (\mathbf{I} - \boldsymbol{\rho}^{l-1} \boldsymbol{\rho}^{l-1\dagger}) \mathbf{H}^\dagger]^{1/2} \right. \\ &\quad \left. \times [\mathcal{N}_0 \mathbf{I} + \mathcal{E}_s \mathbf{H} (\mathbf{I} - \boldsymbol{\rho}^{l-1} \boldsymbol{\rho}^{l-1\dagger}) \mathbf{H}^\dagger]^{-1/2} \mathbf{h}_i \right|^2 \\ &\leq \mathbf{b}_i^{l\dagger} [\mathcal{N}_0 \mathbf{I} + \mathcal{E}_s \mathbf{H} (\mathbf{I} - \boldsymbol{\rho}^{l-1} \boldsymbol{\rho}^{l-1\dagger}) \mathbf{H}^\dagger] \mathbf{b}_i^l \\ &\quad \times \mathbf{h}_i^\dagger [\mathcal{N}_0 \mathbf{I} + \mathcal{E}_s \mathbf{H} (\mathbf{I} - \boldsymbol{\rho}^{l-1} \boldsymbol{\rho}^{l-1\dagger}) \mathbf{H}^\dagger]^{-1} \mathbf{h}_i \end{aligned} \quad (3.20)$$

¹ $\mathbf{J}^{1/2}$, a square root matrix of the positive semidefinite matrix \mathbf{J} , satisfies $\mathbf{J} = \mathbf{J}^{1/2\dagger} \mathbf{J}^{1/2}$.

with equality if and only if

$$\mathbf{b}_i^l \propto [\mathcal{N}_0 \mathbf{I} + \mathcal{E}_s \mathbf{H}(\mathbf{I} - \boldsymbol{\rho}^{l-1} \boldsymbol{\rho}^{l-1\dagger}) \mathbf{H}^\dagger]^{-1} \mathbf{h}_i \quad \text{for } i = 1, 2, \dots, N. \quad (3.21)$$

Substituting (3.20) into (3.19), we see that (3.21) maximizes (3.19) and, in turn, (3.18). To ensure that $\mathbf{b}_i^{l\dagger} \mathbf{h}_i = 1$, we choose the constant of proportionality to be

$$\mu_i^l = \frac{1}{\mathbf{h}_i^\dagger [\mathcal{N}_0 \mathbf{I} + \mathcal{E}_s \mathbf{H}(\mathbf{I} - \boldsymbol{\rho}^{l-1} \boldsymbol{\rho}^{l-1\dagger}) \mathbf{H}^\dagger]^{-1} \mathbf{h}_i}. \quad (3.22)$$

Thus, we may write ²

$$\mathbf{B}^l = [\mathcal{N}_0 \mathbf{I} + \mathcal{E}_s \mathbf{H}(\mathbf{I} - \boldsymbol{\rho}^{l-1} \boldsymbol{\rho}^{l-1\dagger}) \mathbf{H}^\dagger]^{-1} \mathbf{H} \text{diag} \left\{ \mu_1^l, \mu_2^l, \dots, \mu_N^l \right\}. \quad (3.23)$$

Some comments can be made about the special case when $l = 1$. During the first pass, feedback is not used because $\boldsymbol{\rho}^0 = \mathbf{0}$, so the vector $\hat{\mathbf{x}}^0$ does not need to be defined. Moreover, the filter \mathbf{B}^1 takes the form

$$\mathbf{B}^1 \propto [\mathcal{N}_0 \mathbf{I} + \mathcal{E}_s \mathbf{H} \mathbf{H}^\dagger]^{-1} \mathbf{H} \text{diag} \left\{ \mu_1^1, \mu_2^1, \dots, \mu_N^1 \right\}, \quad (3.24)$$

which is an expression for the unbiased linear MMSE detector. Thus the performance of the iterated-decision detector, after just one iteration, is identical to the performance of the linear MMSE detector. At the end of this section, we show that the iterated-decision detector, when using multiple iterations, performs significantly better than the linear MMSE detector.

The iterated-decision detector also has an interesting relationship with another detector. If $\boldsymbol{\rho}^{l-1}$ is set to \mathbf{I} , then the matrices (3.23) and (3.15) for the iterated-decision detector become the matrices used for the multistage detector [64]. In other words, the iterated-decision detector explicitly takes into account the reliability of tentative decisions, while the

²Using the special case of the matrix inversion lemma

$$\mathbf{Y}(\mathbf{I} + \mathbf{X}\mathbf{Y})^{-1} = (\mathbf{I} + \mathbf{Y}\mathbf{X})^{-1} \mathbf{Y},$$

we may alternatively write

$$\mathbf{B}^l \propto \mathbf{H}[\mathcal{N}_0 \mathbf{I} + \mathcal{E}_s (\mathbf{I} - \boldsymbol{\rho}^{l-1} \boldsymbol{\rho}^{l-1\dagger}) \mathbf{H}^\dagger \mathbf{H}]^{-1} \text{diag} \left\{ \mu_1^l, \mu_2^l, \dots, \mu_N^l \right\},$$

which may be easier to evaluate depending on the relative sizes of N and Q .

multistage detector assumes that all tentative decisions are correct. As we will see at the end of this section, this difference is the reason that the decisions of the former asymptotically converge to the optimum ones, while the decisions of the latter often diverge.

We now proceed to simplify the SINR expression that characterizes the resulting performance for the i th user. With the optimum \mathbf{b}_i^l and \mathbf{d}_i^l , we have, substituting (3.20) into (3.19),

$$\phi_i^l = \mathcal{E}_s \mathbf{h}_i^\dagger [\mathcal{N}_0 \mathbf{I} + \mathcal{E}_s \mathbf{H} (\mathbf{I} - \boldsymbol{\rho}^{l-1} \boldsymbol{\rho}^{l-1\dagger}) \mathbf{H}^\dagger]^{-1} \mathbf{h}_i. \quad (3.25)$$

After some algebraic manipulation, the SINR from (3.18), with (3.25), then becomes

$$\gamma_i^l = \left(\frac{1}{([\mathbf{I} + \boldsymbol{\alpha}^l]^{-1})_{ii}} - 1 \right) \cdot \frac{1}{1 - (\rho_i^{l-1})^2} \quad (3.26)$$

where

$$\boldsymbol{\alpha}^l = \frac{\mathcal{E}_s (\mathbf{I} - \boldsymbol{\rho}^{l-1} \boldsymbol{\rho}^{l-1\dagger}) \mathbf{H}^\dagger \mathbf{H}}{\mathcal{N}_0}. \quad (3.27)$$

3.1.4 Computing the Correlation Coefficient $\boldsymbol{\rho}$

If \tilde{x}_i^l is treated as the output of an additive white Gaussian noise (AWGN) channel with input x_i , then we have the following convenient iterative algorithm for computing the set of correlation matrices $\boldsymbol{\rho}^l$, and in turn predicting the sequence of symbol error probabilities:

1. Set $\rho_i^0 = 0$ for $i = 1, 2, \dots, N$ and let $l = 1$.
2. Compute the SINR γ_i^l for $i = 1, 2, \dots, N$ at the slicer input on the l th decoding pass from $\boldsymbol{\rho}^{l-1}$ via (3.26) and (3.27).
3. Approximate the symbol error probability $\Pr(\epsilon_i^l)$ at the slicer output from γ_i^l for $i = 1, 2, \dots, N$ using the appropriate high-SNR formula for the symbol error rate of a symbol-by-symbol threshold detector for AWGN channels [51]. For M -PSK,

$$\Pr(\epsilon_i^l) \approx 2\mathcal{Q} \left(\sin \left(\frac{\pi}{M} \right) \sqrt{2\gamma_i^l} \right), \quad (3.28)$$

where $\mathcal{Q}(\cdot)$ is defined in (2.5). For square M -QAM,

$$\Pr(\epsilon_i^l) \approx 1 - \left(1 - 2 \left(1 - \frac{1}{\sqrt{M}} \right) \mathcal{Q} \left(\sqrt{\frac{3\gamma_i^l}{M-1}} \right) \right)^2. \quad (3.29)$$

4. Approximate ρ_i^l from $\Pr(\epsilon_i^l)$ for $i = 1, 2, \dots, N$. For M -PSK [7]

$$\rho_i^l \approx 1 - 2 \sin^2 \left(\frac{\pi}{M} \right) \Pr(\epsilon_i^l), \quad (3.30)$$

and for square M -QAM

$$\rho_i^l \approx 1 - \frac{3}{M-1} \Pr(\epsilon_i^l). \quad (3.31)$$

Both (3.30) and (3.31) can be expressed as

$$\rho_i^l \approx 1 - \frac{d_{\min}^2}{2\mathcal{E}_s} \Pr(\epsilon_i^l), \quad (3.32)$$

where d_{\min} is the minimum distance between constellation points.³

5. Increment l and go to step 2.

3.1.5 Asymptotic Performance with Mode-Interleaved Precoding

The iterated-decision detector, when used with the mode-interleaved precoding of Chapter 2, has some attractive properties. The following theorem characterizes the composite system consisting of the cascade of a mode-interleaved precoder \mathbf{Y} , the channel \mathbf{H} , and the iterated-decision detector after l iterations.

Theorem 3.1 *Let $\{\mathbf{H}_N\}$ for $N = 1, 2, 3, \dots$ be a sequence of $Q \times N$ channel matrices, where the eigenvalue matrices $\{\mathbf{\Lambda}_N\}$ of $\{\mathbf{H}_N^\dagger \mathbf{H}_N\}$ satisfy*

$$K_N^l \triangleq \frac{1}{N} \text{Tr} \left(\left(\mathbf{I}_N + \frac{1}{\xi^l} \mathbf{\Lambda}_N \right)^{-1} \right) \rightarrow K^l \quad (3.33)$$

as $N \rightarrow \infty$, where

$$\frac{1}{\xi^l} \triangleq \frac{(1 - (\rho^{l-1})^2)}{\zeta} \quad (3.34)$$

with the received SNR $1/\zeta$ defined in (1.6). Let $\{\mathbf{Y}_N\}$ be a sequence of unitary matrices drawn from statistically independent, isotropically distributed random ensembles, used as the corresponding mode-interleaved precoders. Let $\{\mathbf{x}_N\}$ and $\{\hat{\mathbf{x}}_N^{l-1}\}$ for $N = 1, 2, 3, \dots$ be two sequences of zero-mean uncorrelated symbols with energy \mathcal{E}_s , where N denotes the vector dimension; and let the normalized correlation matrix of the two vectors be expressed in the

³In the special case of QPSK (4-PSK), it can be shown that the algorithm can be streamlined by eliminating Step 3 and replacing the approximation (3.32) with the exact formula $\rho_i^l = 1 - 2\mathcal{Q}(\sqrt{\gamma_i^l})$.

form

$$\frac{E[\mathbf{x}_N \cdot \hat{\mathbf{x}}_N^{l-1\dagger}]}{\mathcal{E}_s} = \rho^{l-1} \triangleq \rho^{l-1} \mathbf{I}. \quad (3.35)$$

Finally, let $\{\mathbf{B}_N^l\}$ and $\{\mathbf{D}_N^l\}$ be the matrices given in (3.23) and (3.15). From (3.6), the corresponding slicer input can be expressed as

$$\tilde{\mathbf{x}}_N^l = \mathbf{x}_N + \mathbf{u}_N^l, \quad (3.36)$$

where \mathbf{u}_N^l is complex-valued, zero-mean, and uncorrelated with \mathbf{x}_N . Then, as $N \rightarrow \infty$, \mathbf{u}_N^l converges in distribution to a marginally Gaussian white noise vector, with each component having variance

$$\text{var } u_i^l \xrightarrow{\text{a.s.}} \mathcal{E}_s (1 - (\rho^{l-1})^2) \left(\frac{1}{1 - K^l} - 1 \right). \quad (3.37)$$

Furthermore,

$$\gamma_i^l \xrightarrow{\text{a.s.}} \left(\frac{1}{K^l} - 1 \right) \cdot \frac{1}{1 - (\rho^{l-1})^2}. \quad (3.38)$$

Proof: Let us begin by considering μ_i^l as defined by (3.22), with $\mathbf{H}_N \mathbf{\Upsilon}_N$ being the effective channel. The eigenvector matrix of $\mathbf{\Upsilon}_N^\dagger \mathbf{H}_N^\dagger [\mathcal{N}_0 \mathbf{I}_N + \mathcal{E}_s (1 - (\rho^{l-1})^2) \mathbf{H}_N \mathbf{H}_N^\dagger]^{-1} \mathbf{H}_N \mathbf{\Upsilon}_N$ is $\mathbf{V}_N^\dagger \mathbf{\Upsilon}_N$ where \mathbf{V}_N is the right singular matrix of \mathbf{H} , and the eigenvalue matrix is

$$\frac{1}{\mathcal{E}_s (1 - (\rho^{l-1})^2)} \cdot \frac{1}{\xi^l} \mathbf{\Lambda}_N \left(\mathbf{I}_N + \frac{1}{\xi^l} \mathbf{\Lambda}_N \right)^{-1}. \quad (3.39)$$

Given that the eigenvector matrix $\mathbf{V}_N^\dagger \mathbf{\Upsilon}_N$ is an independent, isotropically distributed unitary matrix, it follows that any eigenvector is an isotropically distributed unit vector. Such vectors have the same distribution as $\mathbf{z}_N / \sqrt{\mathbf{z}_N^\dagger \mathbf{z}_N}$, where \mathbf{z}_N is a N -dimensional vector of independent $\mathcal{CN}(0, 1)$ random variables. Thus μ_i^l has the same distribution as

$$\left\{ \frac{1}{\mathcal{E}_s (1 - (\rho^{l-1})^2)} \cdot \frac{\frac{1}{N} \mathbf{z}_N^\dagger \left[\frac{1}{\xi^l} \mathbf{\Lambda}_N \left(\mathbf{I}_N + \frac{1}{\xi^l} \mathbf{\Lambda}_N \right)^{-1} \right] \mathbf{z}_N}{\frac{1}{N} \mathbf{z}_N^\dagger \mathbf{z}_N} \right\}. \quad (3.40)$$

Since the spectral radius of the matrices $\left\{ \frac{1}{\xi^l} \mathbf{\Lambda}_N \left(\mathbf{I}_N + \frac{1}{\xi^l} \mathbf{\Lambda}_N \right)^{-1} \right\}$ is less than unity and

$$\frac{1}{N} \text{Tr} \left(\frac{1}{\xi^l} \mathbf{\Lambda}_N \left(\mathbf{I}_N + \frac{1}{\xi^l} \mathbf{\Lambda}_N \right)^{-1} \right) = 1 - \frac{1}{N} \text{Tr} \left(\left(\mathbf{I}_N + \frac{1}{\xi^l} \mathbf{\Lambda}_N \right)^{-1} \right), \quad (3.41)$$

we can apply Lemma 2.2 to the numerator of (3.40) and the strong law of large numbers to the denominator, giving

$$\mu_i^l \xrightarrow{\text{a.s.}} \frac{\mathcal{E}_s(1 - (\rho^{l-1})^2)}{1 - K^l}. \quad (3.42)$$

Next, let us consider any component u_i^l as defined in (3.8). We obtain the mean and variance, respectively, of u_i^l given $\mathbf{H}_N \mathbf{\Upsilon}_N$ as

$$E[u_i^l | \mathbf{H}_N \mathbf{\Upsilon}_N] = 0 \quad (3.43)$$

and

$$E[u_i^l u_i^{l*} | \mathbf{H}_N \mathbf{\Upsilon}_N] = \mathcal{N}_0 \mathbf{b}_i^{l\dagger} \mathbf{b}_i^l + \mathcal{E}_s(1 - (\rho^{l-1})^2) \tilde{\mathbf{c}}_i^{l\dagger} \tilde{\mathbf{c}}_i^l. \quad (3.44)$$

From (3.23) with $\mathbf{H}_N \mathbf{\Upsilon}_N$ being the effective channel, the eigenvector matrix of

$$\mathcal{N}_0 \text{diag}^{-1} \{ \mu_1^l, \mu_2^l, \dots, \mu_N^l \} \mathbf{B}_N^{l\dagger} \mathbf{B}_N^l \text{diag}^{-1} \{ \mu_1^l, \mu_2^l, \dots, \mu_N^l \} \quad (3.45)$$

is $\mathbf{V}_N^\dagger \mathbf{\Upsilon}_N$, and the eigenvalue matrix is

$$\frac{1}{\mathcal{E}_s(1 - (\rho^{l-1})^2)} \cdot \frac{1}{\xi^l} \mathbf{\Lambda}_N \left(\mathbf{I}_N + \frac{1}{\xi^l} \mathbf{\Lambda}_N \right)^{-2}. \quad (3.46)$$

Similarly, from (3.9), (3.10), and (3.23) with $\mathbf{H}_N \mathbf{\Upsilon}_N$ being the effective channel, the eigenvector matrix of

$$\mathcal{E}_s(1 - (\rho^{l-1})^2) \text{diag}^{-1} \{ \mu_1^l, \mu_2^l, \dots, \mu_N^l \} \tilde{\mathbf{C}}_N^{l\dagger} \tilde{\mathbf{C}}_N^l \text{diag}^{-1} \{ \mu_1^l, \mu_2^l, \dots, \mu_N^l \} \quad (3.47)$$

is $\mathbf{V}_N^\dagger \mathbf{\Upsilon}_N$, and the eigenvalue matrix is

$$\frac{1}{\mathcal{E}_s(1 - (\rho^{l-1})^2)} \left[\frac{1}{\xi^l} \mathbf{\Lambda}_N \left(\mathbf{I}_N + \frac{1}{\xi^l} \mathbf{\Lambda}_N \right)^{-1} - \frac{1}{N} \text{Tr} \left(\frac{1}{\xi^l} \mathbf{\Lambda}_N \left(\mathbf{I}_N + \frac{1}{\xi^l} \mathbf{\Lambda}_N \right)^{-1} \right) \mathbf{I}_N \right]^2. \quad (3.48)$$

Given that the eigenvector matrix $\mathbf{V}_N^\dagger \mathbf{\Upsilon}_N$ is an independent, isotropically distributed unitary matrix, it follows that any eigenvector is an isotropically distributed unit vector. Such vectors have the same distribution as $\mathbf{z}_N / \sqrt{\mathbf{z}_N^\dagger \mathbf{z}_N}$, where \mathbf{z}_N is a N -dimensional vector of independent $\mathcal{CN}(0, 1)$ random variables. Thus $E[u_i^l u_i^{l*} | \mathbf{H}_N \mathbf{\Upsilon}_N]$ has the same distribution

as

$$\frac{|\mu_i^l|^2}{\mathcal{E}_s(1 - (\rho^{l-1})^2)} \cdot \frac{\frac{1}{N} \mathbf{z}_N^\dagger \mathbf{G}_N \mathbf{z}_N}{\frac{1}{N} \mathbf{z}_N^\dagger \mathbf{z}_N} \quad (3.49)$$

where

$$\begin{aligned} \mathbf{G}_N &\triangleq \frac{1}{\xi^l} \mathbf{\Lambda}_N \left(\mathbf{I}_N + \frac{1}{\xi^l} \mathbf{\Lambda}_N \right)^{-2} \\ &\quad + \left[\frac{1}{\xi^l} \mathbf{\Lambda}_N \left(\mathbf{I}_N + \frac{1}{\xi^l} \mathbf{\Lambda}_N \right)^{-1} - \frac{1}{N} \text{Tr} \left(\frac{1}{\xi^l} \mathbf{\Lambda}_N \left(\mathbf{I}_N + \frac{1}{\xi^l} \mathbf{\Lambda}_N \right)^{-1} \right) \mathbf{I}_N \right]^2 \\ &= \left(\mathbf{I}_N + \frac{1}{\xi^l} \mathbf{\Lambda}_N \right)^{-1} - \left(\mathbf{I}_N + \frac{1}{\xi^l} \mathbf{\Lambda}_N \right)^{-2} \\ &\quad + \left[\frac{1}{N} \text{Tr} \left(\left(\mathbf{I}_N + \frac{1}{\xi^l} \mathbf{\Lambda}_N \right)^{-1} \right) \mathbf{I}_N - \left(\mathbf{I}_N + \frac{1}{\xi^l} \mathbf{\Lambda}_N \right)^{-1} \right]^2 \\ &= \left(\mathbf{I}_N + \frac{1}{\xi^l} \mathbf{\Lambda}_N \right)^{-1} + \left[\frac{1}{N} \text{Tr} \left(\left(\mathbf{I}_N + \frac{1}{\xi^l} \mathbf{\Lambda}_N \right)^{-1} \right) \mathbf{I}_N \right]^2 \\ &\quad - 2 \cdot \frac{1}{N} \text{Tr} \left(\left(\mathbf{I}_N + \frac{1}{\xi^l} \mathbf{\Lambda}_N \right)^{-1} \right) \left(\mathbf{I}_N + \frac{1}{\xi^l} \mathbf{\Lambda}_N \right)^{-1}. \end{aligned} \quad (3.50)$$

Since the spectral radius of the matrices $\left\{ \left(\mathbf{I}_N + \frac{1}{\xi^l} \mathbf{\Lambda}_N \right)^{-1} \right\}$ is less than unity, we have that the spectral radius of $\{\mathbf{G}_N\}$ is less than 2. Furthermore, the average of the eigenvalues of \mathbf{G}_N is

$$\begin{aligned} \frac{1}{N} \text{Tr}(\mathbf{G}_N) &= \frac{1}{N} \text{Tr} \left(\left(\mathbf{I}_N + \frac{1}{\xi^l} \mathbf{\Lambda}_N \right)^{-1} \right) + \left[\frac{1}{N} \text{Tr} \left(\left(\mathbf{I}_N + \frac{1}{\xi^l} \mathbf{\Lambda}_N \right)^{-1} \right) \right]^2 \\ &\quad - 2 \left[\frac{1}{N} \text{Tr} \left(\left(\mathbf{I}_N + \frac{1}{\xi^l} \mathbf{\Lambda}_N \right)^{-1} \right) \right]^2 \\ &= \frac{1}{N} \text{Tr} \left(\left(\mathbf{I}_N + \frac{1}{\xi^l} \mathbf{\Lambda}_N \right)^{-1} \right) - \left[\frac{1}{N} \text{Tr} \left(\left(\mathbf{I}_N + \frac{1}{\xi^l} \mathbf{\Lambda}_N \right)^{-1} \right) \right]^2. \end{aligned} \quad (3.51)$$

Thus, we can apply Lemma 2.2 and (3.42) to the numerator of (3.49) and the strong law of large numbers to the denominator, giving

$$E[u_i^l u_i^{l*} | \mathbf{H}_N \mathbf{Y}_N] \xrightarrow{\text{a.s.}} \mathcal{E}_s(1 - (\rho^{l-1})^2) \left(\frac{1}{1 - K^l} - 1 \right). \quad (3.52)$$

The limit of (3.13) is thus

$$\gamma_i^l \xrightarrow{\text{a.s.}} \left(\frac{1}{K^l} - 1 \right) \cdot \frac{1}{(1 - (\rho^{l-1})^2)}. \quad (3.53)$$

The covariance of u_i^l and u_j^l for any $i \neq j$ given $\mathbf{H}_N \boldsymbol{\Upsilon}_N$ is

$$E[u_i^l u_j^{l*} | \mathbf{H}_N \boldsymbol{\Upsilon}_N] = \mathcal{N}_0 \mathbf{b}_i^{l\dagger} \mathbf{b}_j^l + \mathcal{E}_s (1 - (\rho^{l-1})^2) \tilde{\mathbf{c}}_i^{l\dagger} \tilde{\mathbf{c}}_j^l. \quad (3.54)$$

Given that the eigenvector matrix of

$$\text{diag}^{-1} \left\{ \mu_1^l, \mu_2^l, \dots, \mu_N^l \right\} \left(\mathcal{N}_0 \mathbf{B}_N^{l\dagger} \mathbf{B}_N^l + \mathcal{E}_s (1 - (\rho^{l-1})^2) \tilde{\mathbf{C}}_N^{l\dagger} \tilde{\mathbf{C}}_N^l \right) \text{diag}^{-1} \left\{ \mu_1^l, \mu_2^l, \dots, \mu_N^l \right\} \quad (3.55)$$

is the independent, isotropically distributed unitary matrix $\mathbf{V}_N^\dagger \boldsymbol{\Upsilon}_N$, it follows that any two distinct eigenvectors have a joint distribution identical to that obtained by Gram-Schmidt orthogonalizing two independent Gaussian vectors. Specifically, if \mathbf{y}_N and \mathbf{z}_N are two independent N -dimensional vectors of independent $\mathcal{CN}(0, 1)$ random variables, then one eigenvector of $\mathbf{V}_N^\dagger \boldsymbol{\Upsilon}_N$ has the same distribution as $\mathbf{z}_N / \sqrt{\mathbf{z}_N^\dagger \mathbf{z}_N}$ and the other has the same distribution as (2.29). Thus $E[u_i^l u_j^{l*} | \mathbf{H}_N \boldsymbol{\Upsilon}_N]$ has the same distribution as

$$\begin{aligned} & \frac{\mu_i^{l*} \mu_j^l}{\mathcal{E}_s (1 - (\rho^{l-1})^2)} \cdot \frac{\frac{1}{N} \mathbf{z}_N^\dagger \mathbf{G}_N \left(\mathbf{y}_N - \left(\frac{\mathbf{z}_N^\dagger \mathbf{y}_N}{\mathbf{z}_N^\dagger \mathbf{z}_N} \right) \mathbf{z}_N \right)}{\sqrt{\frac{1}{N} \mathbf{z}_N^\dagger \mathbf{z}_N} \sqrt{\frac{1}{N} \left(\mathbf{y}_N^\dagger \mathbf{y}_N - \frac{|\mathbf{z}_N^\dagger \mathbf{y}_N|^2}{\mathbf{z}_N^\dagger \mathbf{z}_N} \right)}} \\ &= \frac{\mu_i^{l*} \mu_j^l}{\mathcal{E}_s (1 - (\rho^{l-1})^2)} \cdot \frac{\frac{1}{N} \mathbf{z}_N^\dagger \mathbf{G}_N \mathbf{y}_N - \frac{1}{N} \mathbf{z}_N^\dagger \mathbf{G}_N \mathbf{z}_N \left(\frac{\mathbf{z}_N^\dagger \mathbf{y}_N}{\mathbf{z}_N^\dagger \mathbf{z}_N} \right)}{\sqrt{\frac{1}{N} \mathbf{z}_N^\dagger \mathbf{z}_N} \sqrt{\frac{1}{N} \mathbf{y}_N^\dagger \mathbf{y}_N - \frac{\frac{1}{N} |\mathbf{z}_N^\dagger \mathbf{y}_N|^2}{\frac{1}{N} \mathbf{z}_N^\dagger \mathbf{z}_N}}}. \end{aligned} \quad (3.56)$$

Applying Lemmas 2.2 and 2.4 to (3.56), we conclude that

$$E[u_i^l u_j^{l*} | \mathbf{H}_N \boldsymbol{\Upsilon}_N] \xrightarrow{\text{a.s.}} 0. \quad (3.57)$$

□

Unlike Theorem 2.1, there is no explicit requirement here that Q must go to infinity with N . However, if Q does not go to infinity, then there are only a finite number of nonzero eigenvalues, in which case $K_N^l \rightarrow 1$ and $\gamma_i^l \xrightarrow{\text{a.s.}} 0$.

The assumption in (3.35) that ρ^{l-1} is a scaled identity matrix is valid for $l = 1, 2, 3, \dots$ for the following reason. For $l = 1$, $\rho^0 = \mathbf{0}$, which is consistent with (3.35). Since the components of the effective noise vector \mathbf{u}_N^1 are marginally Gaussian, have the same asymptotic variance, and are asymptotically uncorrelated, ρ^1 must asymptotically have the form $\rho^1 \mathbf{I}$. The same argument can be repeated for $l = 2, 3, \dots$

Note that in Step 2 of the iterative algorithm for determining ρ^l , the SINRs γ_i^l are now equal and can be alternatively computed for large N from ρ^{l-1} via (3.38), (3.33), (3.34), and (1.6). Moreover, the whiteness of \mathbf{u}_N^l strongly suggests that treating \tilde{x}_i^l as if it were the output of an AWGN channel with input x_i in Step 3 is a good approximation.

We can obtain a corollary for the special case in which the components of \mathbf{H} are mutually independent, zero-mean, complex-valued, circularly symmetric Gaussian random variables with variance $1/Q$, i.e., \mathbf{H} is an i.i.d. Gaussian matrix. (In CDMA systems this case corresponds to the use of random signatures, accurate power control, and normalized channel gains, while for multiple antenna systems this case corresponds to a rich scattering environment.) As shown in Theorem 2.2, such an \mathbf{H} matrix has an isotropically distributed right singular vector matrix, which makes additional mode-interleaved precoding at the transmitter unnecessary. The following corollary characterizes the composite system consisting of this special \mathbf{H} in cascade with the iterated-decision detector after l iterations.

Corollary 3.1 *Let $\{\mathbf{H}_N\}$ be a sequence of $Q \times N$ matrices whose components are mutually independent, zero-mean, complex-valued, circularly symmetric Gaussian random variables with variance $1/Q$. Then, as $N \rightarrow \infty$ with $\beta \triangleq N/Q$ converging to a constant, each component of \mathbf{u}_N^l has variance*

$$\text{var } u_i^l \xrightarrow{\text{a.s.}} \mathcal{E}_s(1 - (\rho^{l-1})^2) \left(\frac{1}{\frac{\xi^l}{4\beta} \mathcal{F}\left(\frac{1}{\xi^l}, \beta\right)} - 1 \right) \quad (3.58)$$

and

$$\gamma_i^l \xrightarrow{\text{a.s.}} \left(\frac{1}{1 - \frac{\xi^l}{4\beta} \mathcal{F}\left(\frac{1}{\xi^l}, \beta\right)} - 1 \right) \cdot \frac{1}{1 - (\rho^{l-1})^2} \quad (3.59)$$

where

$$\mathcal{F}(y, z) \triangleq \left(\sqrt{y(1 + \sqrt{z})^2 + 1} - \sqrt{y(1 - \sqrt{z})^2 + 1} \right)^2. \quad (3.60)$$

Proof: The proof requires the following lemma.

Lemma 3.1 ([6]) *Let $\{\mathbf{H}_N\}$ be a sequence of $Q \times N$ matrices whose elements are independent $\mathcal{CN}(0, 1/Q)$. If $\beta = N/Q$ converges to a constant, then the percentage of the N eigenvalues of $\mathbf{H}_N^\dagger \mathbf{H}_N$ that lie below x converges to the cumulative distribution function of the probability density function*

$$f_\beta(x) = [1 - \beta^{-1}]^+ \delta(x) + \frac{\sqrt{[x - \eta_1]^+ [\eta_2 - x]^+}}{2\pi\beta x} \quad (3.61)$$

where

$$\eta_1 = (1 - \sqrt{\beta})^2 \quad (3.62)$$

$$\eta_2 = (1 + \sqrt{\beta})^2, \quad (3.63)$$

and the operator $[\cdot]^+$ is defined according to

$$[u]^+ \triangleq \max\{0, u\}. \quad (3.64)$$

Using Lemma 3.1, we have that [34]

$$\begin{aligned} K^l &= \lim_{N \rightarrow \infty} K_N^l \\ &= \lim_{N \rightarrow \infty} \frac{1}{N} \text{Tr} \left(\left(\mathbf{I}_N + \frac{1}{\xi^l} \mathbf{\Lambda}_N \right)^{-1} \right) \\ &= \int_0^\infty \left(\frac{1}{1 + \lambda/\xi^l} \right) f_\beta(\lambda) dt \\ &= 1 - \frac{\xi^l}{4\beta} \mathcal{F} \left(\frac{1}{\xi^l}, \beta \right) \end{aligned} \quad (3.65)$$

where $\mathcal{F}(\cdot, \cdot)$ is given by (3.60). Substituting (3.65) into (3.37) and (3.38) we obtain the desired result. \square

We now present some results for the iterated-decision detector when the channel is a $Q \times N$ \mathbf{H} matrix whose components are i.i.d. Gaussian with variance $1/Q$. Note that the insights we gain are also applicable to other channels as long as iterated-decision detection and mode-interleaved precoding are used.

From Steps 2 and 3 of the algorithm to compute $\boldsymbol{\rho}$ in Section 3.1.4, we see that $\Pr(\epsilon^l)$ can be expressed as

$$\Pr(\epsilon^l) = \mathcal{G}(\zeta, \beta, \rho^{l-1}), \quad (3.66)$$

where $\mathcal{G}(\cdot, \cdot, \cdot)$ is a monotonically decreasing function in both SNR $1/\zeta$ and correlation ρ^{l-1} , but a monotonically increasing function in β . The monotonicity of $\mathcal{G}(\cdot, \cdot, \cdot)$ is illustrated in Fig. 3.3 where the successively lower solid curves plot $\mathcal{G}(\zeta, \beta, \rho)$ as a function of $1/(1-\rho)$ for various values of β , with an SNR per bit of 7 dB. Meanwhile, from Step 4 of the algorithm, we see that we can also express $\Pr(\epsilon^l)$ as

$$\Pr(\epsilon^l) = \mathcal{H}(\rho^l), \quad (3.67)$$

where $\mathcal{H}(\cdot)$ is a monotonically decreasing function of ρ^l . The dashed line in Fig. 3.3 plots $\mathcal{H}(\rho)$ as a function of $1/(1-\rho)$.

At a given $1/\zeta$ and β , the sequence of error probabilities $\Pr(\epsilon^l)$ and correlation coefficients ρ^l can be obtained by starting at the left end of the solid curve (corresponding to $\rho^0 = 0$) and then successively moving horizontally to the right from the solid curve to the dashed line, and then moving downward from the dashed line to the solid curve. Each “step” of the resulting descending staircase corresponds to one pass of the iterated-decision detector. In Fig. 3.3, the sequence of operating points is indicated on the solid curves with

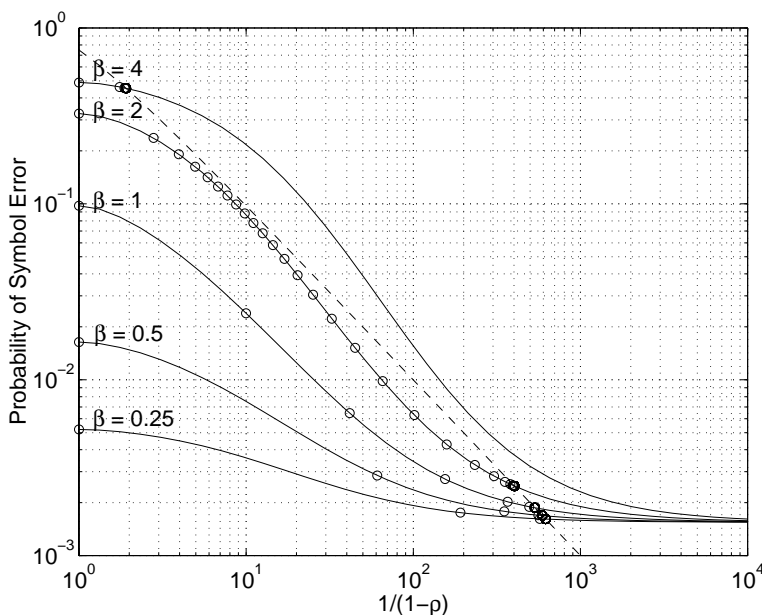


Figure 3.3: Iterated-decision detector performance for i.i.d. Gaussian \mathbf{H} . The solid curves plot QPSK symbol error rate as a function of the correlation coefficient ρ for various $\beta = N/Q$ values, with an SNR per bit of 7 dB. Along each curve, \circ 's identify the theoretically predicted decreasing error rates achieved with $l = 1, 2, \dots$ decoding passes, and the intersections with the dashed line are the steady-state values ($l \rightarrow \infty$).

the \circ symbols. That the sequence of error probabilities $\Pr(\epsilon^1), \Pr(\epsilon^2), \dots$ obtained by the recursive algorithm is monotonically decreasing suggests that additional iterations always improve performance. The error rate performance for a given SNR of $1/\zeta$ and a given β eventually converges to a steady-state value of $\Pr(\epsilon^\infty)$, which is the unique solution to the equation

$$\Pr(\epsilon^\infty) = \mathcal{G}(\zeta, \beta, \mathcal{H}^{-1}(\Pr(\epsilon^\infty))), \quad (3.68)$$

corresponding to the intersection of the dashed line and the appropriate solid curve in Fig. 3.3.

The convergence properties of the iterative detector are closely related to the value of β and warrant further investigation. If β is relatively small, Fig. 3.3 suggests that steady-state performance is approximately achieved with comparatively few iterations, after which additional iterations provide only negligibly small gains in performance. This observation can also be readily made from Fig. 3.4, where bit-error rate is plotted as a function of SNR per bit for 1, 2, 3, 5, and an infinite number of iterations, with $\beta = N/Q = 0.77$. It is significant that, for small β , few passes are required to converge to typical target bit-error rates, since the amount of computation is directly proportional to the number of passes

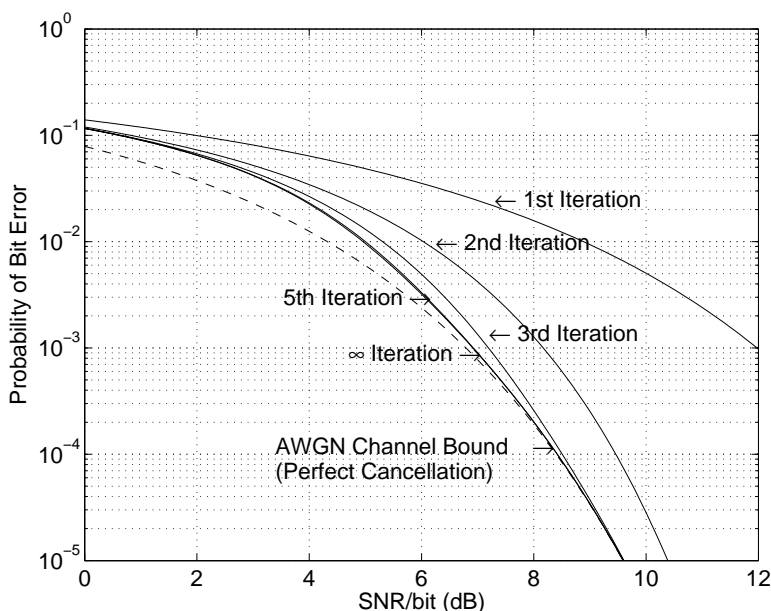


Figure 3.4: Theoretical iterated-decision detector performance for i.i.d. Gaussian \mathbf{H} , as a function of SNR per bit. The solid curves depict the QPSK bit-error rate with $\beta = N/Q = 0.77$ as a function of SNR per bit for 1, 2, 3, 5, and ∞ decoding iterations.

required; we emphasize that the complexity of a single pass of the iterated-decision detector is comparable to that of a decision-feedback detector or a linear detector. As β increases, Fig. 3.3 shows that the gap between the solid curve and the dashed curve decreases. Thus the “steps” of the descending staircase get smaller, and there is a significant increase in the number of iterations required to approximately achieve steady-state performance. Moreover, the probability of error at steady-state becomes slightly larger at the same SNR. When β is greater than some SNR-dependent threshold, not only can (3.68) have multiple solutions, but one of the solutions occurs at a high probability of error, as illustrated by the curve in Fig. 3.3 corresponding to $\beta = 4$. The dependence of the threshold on SNR is shown in Fig. 3.5. As the SNR increases, the β threshold increases, and the bit-error rate curve becomes much sharper at the threshold. Our experiments show that in the high SNR regime the threshold for QPSK is near $\beta \approx e$. In general, the threshold is also dependent on the signal constellation used.

We can gain some insight into this thresholding effect by studying the high-SNR ($\zeta \rightarrow 0$) limit of (3.38) for fixed ρ . For a particular eigenvalue matrix $\mathbf{\Lambda}_N$ of $\mathbf{H}_N^\dagger \mathbf{H}_N$, let R be the number of nonzero eigenvalues of $\mathbf{\Lambda}_N$, and let $R/N \rightarrow 1/\kappa$ as $N \rightarrow \infty$. (Note from (3.61) that $1 - R/N \geq [1 - 1/\beta]^+$.) Then the SINR expression in (3.38) with (3.33) becomes

$$\begin{aligned} \gamma &= \left(\frac{1}{\lim_{N \rightarrow \infty} \frac{1}{N} \sum_{k=1}^N \frac{1}{1 + \lambda_k/\xi}} - 1 \right) \cdot \frac{1}{1 - \rho^2} \\ &= \left(\frac{1}{\left(1 - \frac{1}{\kappa}\right) + \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{k=1}^R \frac{1}{1 + \lambda_k/\xi}} - 1 \right) \cdot \frac{1}{1 - \rho^2} \\ &= \left(\frac{\frac{1}{\kappa} - \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{k=1}^R \frac{1}{1 + \lambda_k/\xi}}{\left(1 - \frac{1}{\kappa}\right) + \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{k=1}^R \frac{1}{1 + \lambda_k/\xi}} \right) \cdot \frac{1}{1 - \rho^2}. \end{aligned} \quad (3.69)$$

In the limit as $\zeta \rightarrow 0$ with ρ fixed, we have from (3.34) that $\xi \rightarrow 0$. Thus

$$\lim_{\zeta \rightarrow 0} \gamma = \frac{\frac{1}{\kappa}}{1 - \frac{1}{\kappa}} \cdot \frac{1}{1 - \rho^2} = \frac{1}{\kappa - 1} \cdot \frac{1}{1 - \rho^2}. \quad (3.70)$$

Since this limit is approached from below, it is a convenient upper bound on γ for any ζ , corresponding to lower bounds on $\Pr(\epsilon)$. This bound, which we call the *zero-modes bound*, is illustrated in Fig. 3.6 for a channel with $\kappa = 1.641$ and all nonzero eigenvalues equal to κ . The solid curve corresponds to (3.38) and is indeed lower-bounded by the zero-modes bound.

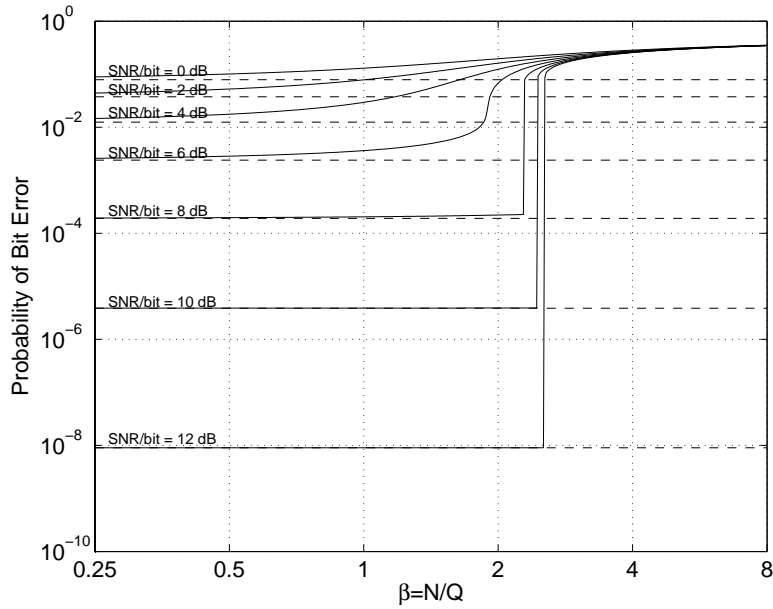


Figure 3.5: Theoretical iterated-decision detector performance for i.i.d. Gaussian \mathbf{H} , as a function of $\beta = N/Q$. The solid curves depict the QPSK bit-error rate as a function of β for various values of SNR per bit, while the corresponding dashed curves depict the AWGN channel bound.

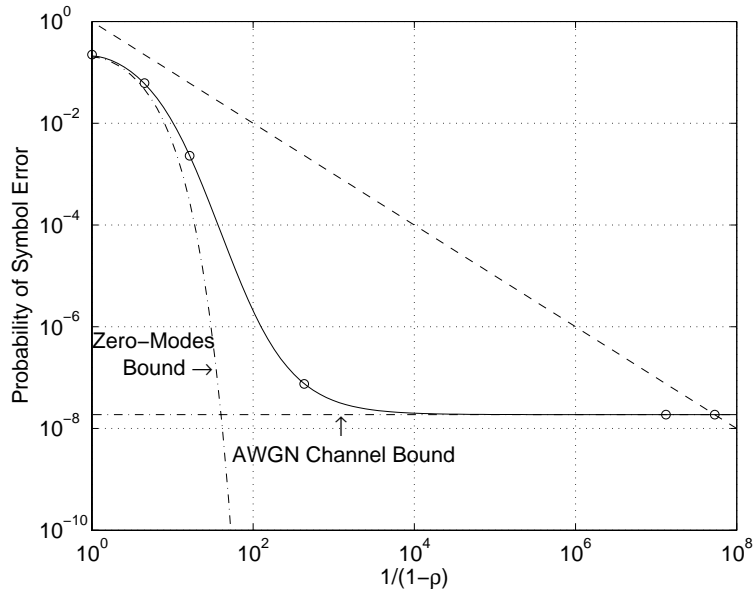


Figure 3.6: High SNR performance limits of the iterated-decision detector. The solid curve plots the QPSK symbol error rate as a function of ρ for a channel with $\kappa = 1.641$ and all nonzero eigenvalues equal to κ . The SNR per bit is 12 dB, and \circ 's identify the theoretically predicted decreasing error rates. The dash-dot curves indicate the high-SNR limits of performance.

For channels with larger values of κ , the bound and the dashed line intersect, leading to the thresholding phenomenon. Since the dashed line is a function of the signal constellation used, the threshold is in turn dependent on the signal constellation. An intriguing open issue is whether these thresholds are a phenomenon specific to the iterative detector, or rather a fundamental limit of detection in general.

We can also determine another upper bound on γ by taking the limit of (3.38) as $\rho \rightarrow 1$ for fixed ζ . The SINR expression in (3.38) with (3.33) can be rewritten as

$$\begin{aligned} \gamma &= \left(\frac{1}{\lim_{N \rightarrow \infty} \frac{1}{N} \sum_{k=1}^N \frac{1}{1+\lambda_k/\xi}} - 1 \right) \cdot \frac{1}{1-\rho^2} \\ &= \left(\frac{1 - \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{k=1}^N \frac{1}{1+\lambda_k/\xi}}{\lim_{N \rightarrow \infty} \frac{1}{N} \sum_{k=1}^N \frac{1}{1+\lambda_k/\xi}} \right) \cdot \frac{1}{1-\rho^2} \\ &= \left(\frac{\lim_{N \rightarrow \infty} \frac{1}{N} \sum_{k=1}^N \frac{\lambda_k}{\xi+\lambda_k}}{\lim_{N \rightarrow \infty} \frac{1}{N} \sum_{k=1}^N \frac{1}{\xi+\lambda_k}} \right) \cdot \frac{1}{\zeta}, \end{aligned} \quad (3.71)$$

where we have used (3.34). As $\rho \rightarrow 1$ with ζ fixed, we have from (3.34) that $\xi \rightarrow \infty$. Thus,

$$\lim_{\rho \rightarrow 1} \gamma = \left(\frac{\lim_{N \rightarrow \infty} \frac{1}{N} \sum_{k=1}^N \lambda_k}{\lim_{N \rightarrow \infty} \frac{1}{N} \sum_{k=1}^N 1} \right) \frac{1}{\zeta} = \frac{1}{\zeta}. \quad (3.72)$$

Thus (3.72) is another upper bound on γ , corresponding to a lower bound on $\Pr(\epsilon)$. In fact, this bound is the AWGN channel bound, and is illustrated in Fig. 3.6.

We now examine the conditions under which the iterated-decision detector with mode-interleaved precoding can achieve AWGN channel performance. First, from Fig. 3.6, we observe that a necessary condition for the AWGN channel bound to be achieved is that the zero-mode bound lie completely below the dashed line. This is the case for virtually all practical channels, for which $\kappa \approx 1$. Second, the convergence in (3.72) must occur for small enough values of $1/(1-\rho)$ so that $\gamma^\infty \approx 1/\zeta$; i.e., convergence of the solid curve to the AWGN channel bound in Fig. 3.6 must occur to the left of the dashed line. This is indeed the case at high SNR, as we now show. Since the spectral radius of $\{\mathbf{\Lambda}_N\}$ is bounded, there exists a universal ξ^* such that

$$\gamma = \left(\frac{\lim_{N \rightarrow \infty} \frac{1}{N} \sum_{k=1}^N \frac{\lambda_k}{\xi+\lambda_k}}{\lim_{N \rightarrow \infty} \frac{1}{N} \sum_{k=1}^N \frac{1}{\xi+\lambda_k}} \right) \cdot \frac{1}{\zeta} \geq (1-\epsilon) \frac{1}{\zeta} \quad (3.73)$$

for all ζ . At a given ζ , the value of ρ corresponding to ξ^* is

$$\rho^* = \sqrt{1 - \frac{\zeta}{\xi^*}} \approx 1 - \frac{\zeta}{2\xi^*}, \quad (3.74)$$

and thus the value of $1/(1 - \rho)$ such that $\gamma \geq (1 - \epsilon)1/\zeta$ is

$$\frac{1}{1 - \rho^*} \propto \frac{1}{\zeta}. \quad (3.75)$$

Next, looking at (3.32) and (3.28)–(3.29) with the value $\gamma = 1/\zeta$, we see that the value of $1/(1 - \rho)$ at which the dashed line intersects the AWGN channel bound is

$$\frac{1}{1 - \rho} \propto \exp\left(\frac{1}{\zeta}\right) \quad (3.76)$$

where we have used the approximation $\mathcal{Q}(v) \propto \exp(-v^2/2)$. As $\zeta \rightarrow 0$, (3.75) becomes less than (3.76), so the AWGN channel bound is achieved at high SNR.

We now compare the performance of the iterated-decision detector to other detectors for i.i.d. Gaussian \mathbf{H} . In Fig. 3.7, the theoretical ($Q \rightarrow \infty$) and simulated ($Q = 128$) bit-error rates of various detectors are plotted as a function of SNR with $\beta = 1$. The iterated-decision detector significantly outperforms the other detectors at moderate to high SNR, and asymptotically approaches the AWGN channel bound. Next, in Fig. 3.8, the effect of β on the theoretical ($Q \rightarrow \infty$) and simulated ($Q = 128$) bit-error rates is compared for the various detectors⁴ when \mathbf{H} is i.i.d. Gaussian with $Q = 128$ at an SNR per bit of 10 dB. The iterated-decision detector has clearly superior performance when $\beta \lesssim 1.5$.

Finally, we plot in Fig. 3.9 the performance of the 2×2 channel in (2.1) when the iterated-decision detector is used in conjunction with mode interleaving. Mode-interleaved precoding is applied to a block diagonal matrix that consists of 64 channel uses stacked as in (2.20). The iterated-decision detector with precoding outperforms the ML detector without precoding, and even achieves the AWGN channel bound at high SNR. Although ML detection is too complex to implement in the precoded case, we can infer that the ML performance curve must lie between the performance curve for the iterated-decision detector with precoding and the AWGN channel bound. Note that the iterated-decision detector

⁴The theoretical large system performance of the decorrelator for the case $\beta > 1$ is derived in [22], where the decorrelator is defined as the Moore-Penrose generalized inverse [35] of \mathbf{H} .

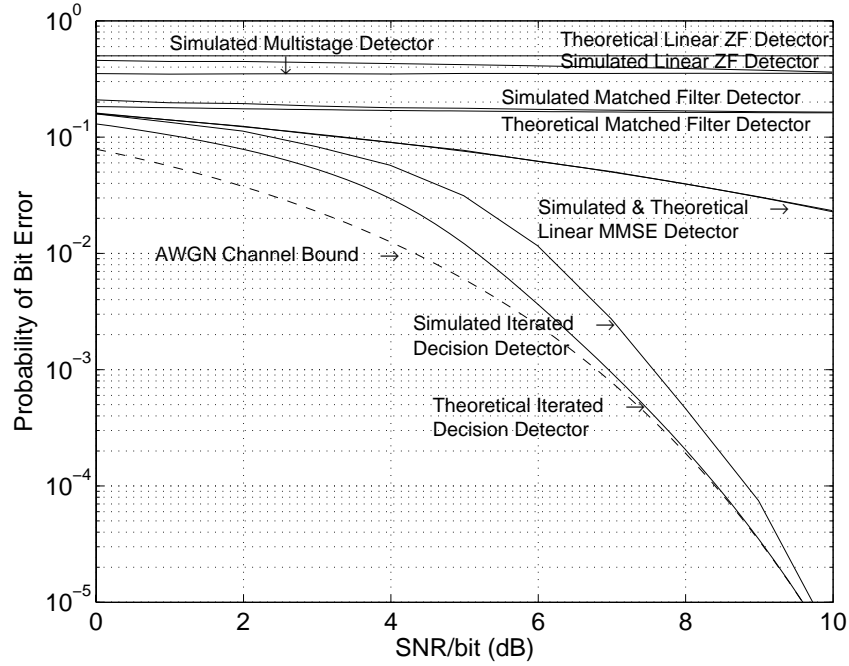


Figure 3.7: Theoretical ($Q \rightarrow \infty$) and experimentally observed ($Q = 128$) performance for various detectors for i.i.d. Gaussian \mathbf{H} . The solid curves depict QPSK bit-error rates with $\beta = N/Q = 1$ as a function of SNR per bit.

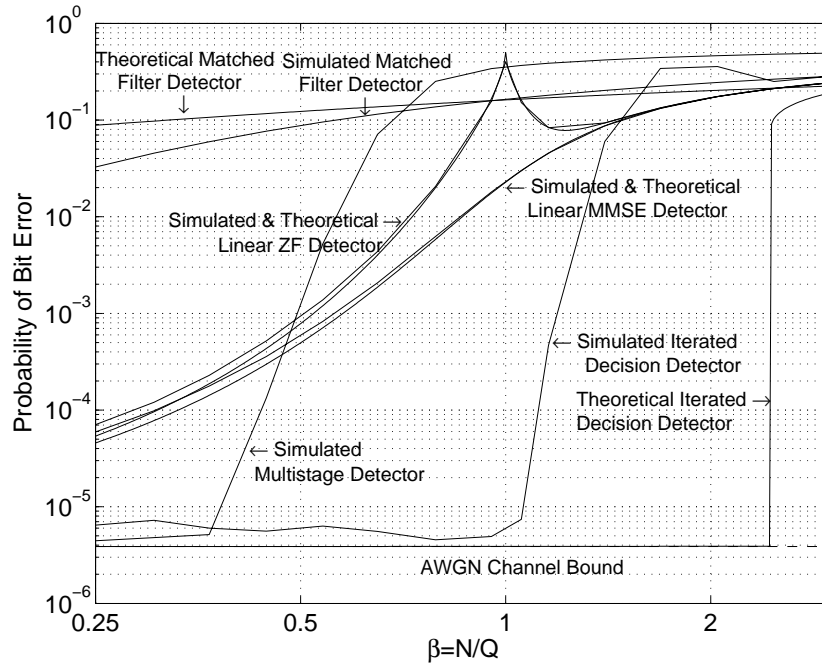


Figure 3.8: Theoretical ($Q \rightarrow \infty$) and experimentally observed ($Q = 128$) performance for various detectors, for i.i.d. Gaussian \mathbf{H} . The solid curves depict QPSK bit-error rates at an SNR per bit of 10 dB as a function of $\beta = N/Q$.

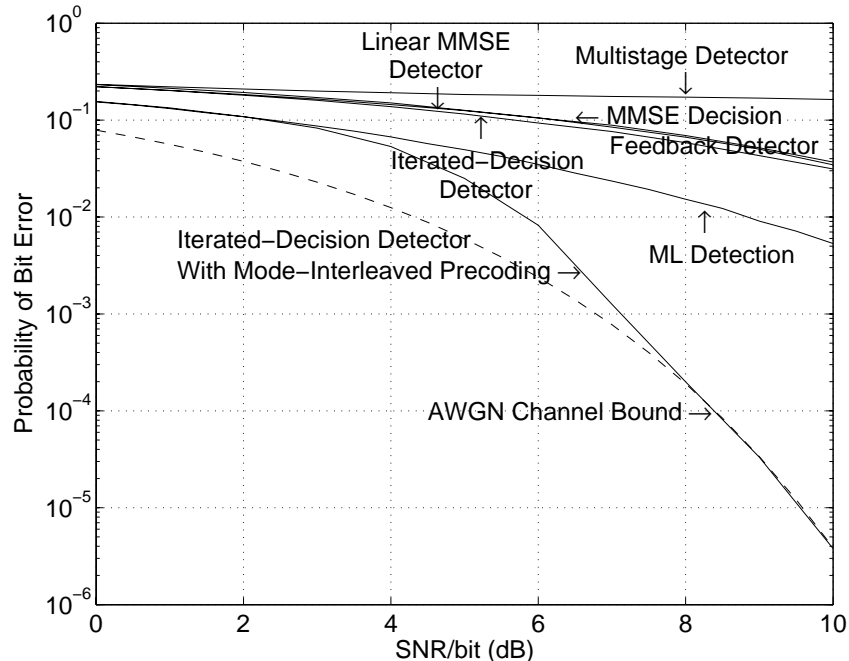


Figure 3.9: Experimentally observed QPSK bit-error probability of various detection schemes as a function of SNR for the interference channel of (2.1). Except for the iterated-decision detector which is used both with and without mode-interleaved precoding, the various detectors are not used in conjunction with mode-interleaved precoding. Since the iterated-decision detector with mode-interleaved precoding achieves the AWGN channel bound at high SNR, we can infer that ML detection with mode-interleaved precoding (not pictured) would also achieve the AWGN channel bound.

without mode-interleaved precoding performs rather poorly, since the small system does not lend itself to Gaussian approximations for the effective noise at the slicer input.

3.2 Iterated-Decision Equalization

In the Toeplitz case, the optimal matrices given by (3.23) and (3.15) can be considerably simplified when the length L of the channel impulse response $h[n]$ is much smaller than the block length N of $x[n]$ (N is also the dimensions of the \mathbf{H} matrix), which is typically the case. The columns of \mathbf{H} are essentially shifted versions of the channel impulse response $h[n]$, and so the corresponding columns of the optimal \mathbf{B}^l matrix in (3.23) are also similarly shifted versions of each other, with the scaling factor μ_i^l in (3.22) equal to a constant μ^l for all i . With \mathbf{B}^l and \mathbf{H} both Toeplitz, \mathbf{D}^l in (3.15) also is a Toeplitz matrix. The SINR γ_i^l at the slicer input is equal for all transmitted symbols, and so a reasonable assumption, then,

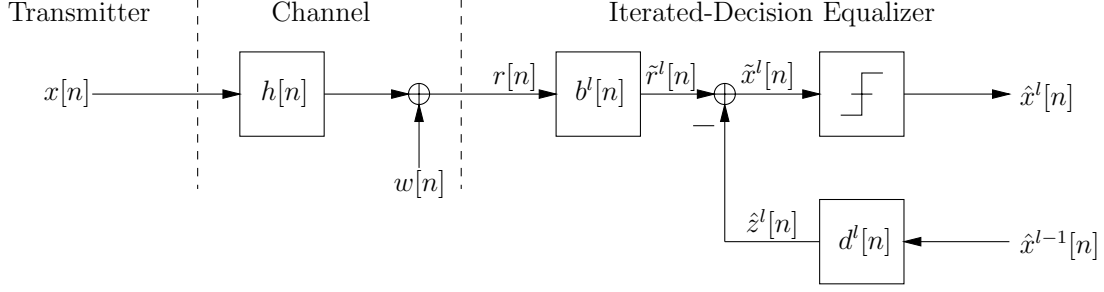


Figure 3.10: Iterated-decision equalizer.

is that the normalized correlation matrix of \mathbf{x} and $\hat{\mathbf{x}}^{l-1}$ is of the form

$$\boldsymbol{\rho}^{l-1} = \frac{E[\mathbf{x} \cdot \hat{\mathbf{x}}^{l-1\dagger}]}{\mathcal{E}_s} = \rho^{l-1} \mathbf{I}. \quad (3.77)$$

With \mathbf{B}^l and \mathbf{D}^l both Toeplitz, the iterated-decision detector can be implemented using linear time-invariant filters (convolutions) as opposed to linear time-varying filters (matrix-vector multiplications). We call this special version of the detector the iterated-decision equalizer.

The detailed structure of the iterated-decision equalizer is depicted in Fig. 3.10. On the l th pass of the equalizer, the received data $r[n]$ is first processed by a linear filter $b^l[n]$ with frequency response

$$B^l(\omega) = \frac{\mu^l H^*(\omega)}{\mathcal{N}_0 + \mathcal{E}_s(1 - (\rho^{l-1})^2)|H(\omega)|^2} \quad (3.78)$$

with scale factor

$$\mu^l = \frac{1}{\frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{|H(\omega)|^2}{\mathcal{N}_0 + \mathcal{E}_s(1 - (\rho^{l-1})^2)|H(\omega)|^2} d\omega}, \quad (3.79)$$

producing the sequence

$$\tilde{r}^l[n] = \sum_k b^l[n-k]r[k]. \quad (3.80)$$

Next, an estimate $\hat{z}^l[n]$ of the ISI constructed, where

$$\hat{z}^l[n] = \sum_k d^l[n-k]\hat{x}^{l-1}[k] \quad (3.81)$$

with

$$D^l(\omega) = \rho^{l-1} (B^l(\omega)H(\omega) - 1). \quad (3.82)$$

While the strictly causal feedback filter of the DFE subtracts out only postcursor ISI, the noncausal nature of the filter $d^l[n]$ allows the iterated-decision equalizer to cancel both precursor and postcursor ISI. Also, since $\hat{z}^l[n]$ is intended to be an ISI estimate, it makes sense that $d^l[0] = 0$. The estimate $\hat{z}^l[n]$ is subtracted from $\tilde{r}^l[n]$, i.e.,

$$\tilde{x}^l[n] = \tilde{r}^l[n] - \hat{z}^l[n]. \quad (3.83)$$

The slicer input $\tilde{x}^l[n]$ defined via (3.83) satisfies, for each n ,

$$\tilde{x}^l[n] = x[n] + u^l[n] \quad (3.84)$$

where

$$u^l[n] = b^l[n] * w[n] + c^l[n] * x[n] - d^l[n] * \hat{x}^{l-1}[n] \quad (3.85)$$

is complex-valued and zero-mean, having variance

$$\text{var } u^l[n] = \mathcal{E}_s(1 - (\rho^{l-1})^2) \left(\frac{1}{1 - K^l} - 1 \right) \quad (3.86)$$

where

$$K^l = \frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{1}{1 + |H(\omega)|^2/\xi^l} d\omega \quad (3.87)$$

with ξ^l given by (3.34). The resulting SINR is

$$\gamma^l = \left(\frac{1}{K^l} - 1 \right) \cdot \frac{1}{1 - (\rho^{l-1})^2}. \quad (3.88)$$

The slicer generates the hard decisions $\hat{x}^l[n]$ from $\tilde{x}^l[n]$ using a symbol-wise minimum-distance rule. Note that Step 2 of the iterative algorithm in Section 3.1.4 now uses (3.88).

Using the fact that the eigenvalues of a square ISI matrix \mathbf{H} correspond to DFT samples $H[k]$ of the frequency response $H(\omega)$ and the eigenvector matrix corresponds to a discrete Fourier transform (DFT) matrix, the iterated-decision equalizer can be alternatively implemented in the DFT domain as illustrated in Fig. 3.11. The optimal filters are

$$B^l[k] = \frac{\mu^l H^*[k]}{\mathcal{N}_0 + \mathcal{E}_s(1 - (\rho^{l-1})^2)|H[k]|^2} \quad (3.89)$$

$$D^l[k] = \rho^{l-1} \left(B^l[k]H[k] - 1 \right) \quad (3.90)$$

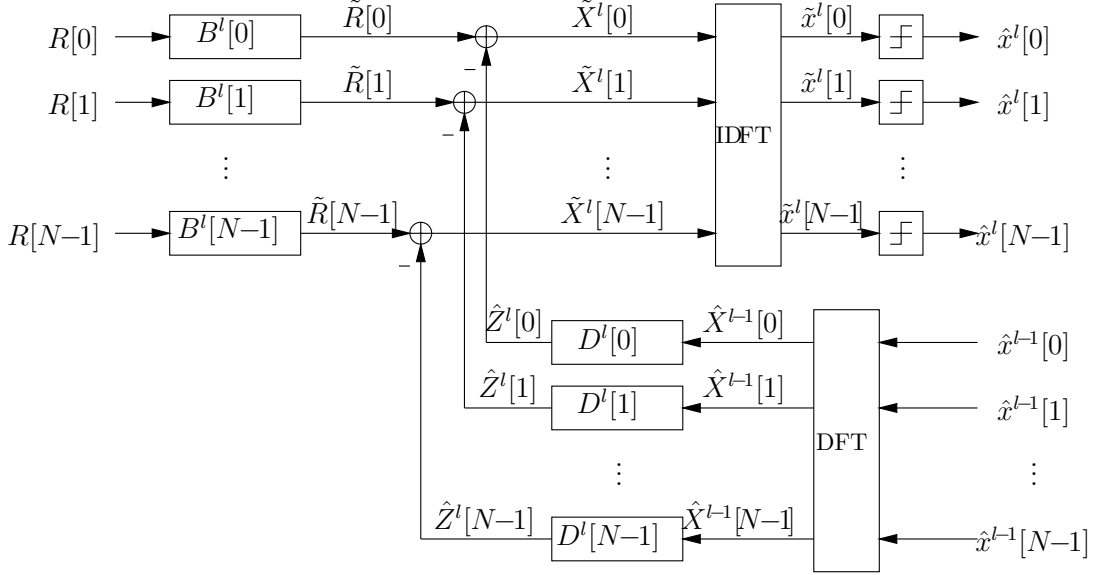


Figure 3.11: Iterated-decision equalizer implemented in the DFT domain.

where

$$\mu^l = \frac{1}{\frac{1}{N} \sum_{k=0}^{N-1} \frac{|H[k]|^2}{\mathcal{N}_0 + \mathcal{E}_s (1 - (\rho^{l-1})^2) |H[k]|^2}}. \quad (3.91)$$

The iterative algorithm for determining the sequence of ρ^l remains the same as in Section 3.1.4, except that γ^l is now computed in Step 2 from ρ^{l-1} via (3.88) with

$$K^l \approx \frac{1}{N} \sum_{k=0}^{N-1} \frac{1}{1 + |H[k]|^2 / \xi^l}. \quad (3.92)$$

When implemented in the DFT domain, iterated-decision equalization for a block length N has a complexity that is proportional to $N \log N$ for a given channel and SNR. Moreover, the complexity does not depend directly on the constellation size. By contrast, the Viterbi algorithm which implements MLSD has complexity NM^L , where M is the signal constellation size and L is the length of the channel. The actual savings can be dramatic in practice on typical channels. For example, when $N = 256$, $M = 4$, and $L = 5$, and we perform 10 iterations of the iterated-decision equalizer (which is typically more iterations than needed), the iterated-decision equalizer is roughly an order of magnitude less complex than the Viterbi algorithm with the same parameters. The difference is even more dramatic with larger signal constellations. When M is increased to 64, the complexity of the iterated-decision equalizer remains unchanged, but the Viterbi algorithm becomes an

additional 6 orders of magnitude more complex.

The use of frequency-interleaved precoding does not increase the complexity of the system from $N \log N$. In fact, as we show in the next section, such precoding allows the iterated-decision equalizer to perform asymptotically as well as the Viterbi algorithm for uncoded systems.

3.2.1 Asymptotic Performance with Frequency-Interleaved Precoding

In Section 2.2.1, it was shown that with frequency-interleaved precoding, the coefficients of the resulting effective channel $h_{p-1}[n]$ are uncorrelated. In light of this fact, it is reasonable to expect that the effective noise process at the slicer input becomes white in the limit as $N \rightarrow \infty$. This is indeed the case as described by the following theorem, whose proof is in Appendix B.

Theorem 3.2 *Let $x_N[n]$ and $\hat{x}_N^{l-1}[n]$ for $n = 0, 1, 2, \dots, N-1$ be two sequences of zero-mean uncorrelated symbols with energy \mathcal{E}_s ; and let the normalized correlation between the two sequences be expressed in the form*

$$\frac{E[x_N^*[n] \cdot \hat{x}_N^{l-1}[k]]}{\mathcal{E}_s} = \rho^{l-1} \delta[n - k]. \quad (3.93)$$

Let $\{x_N[n]\}$ and $\{\hat{x}_N^{l-1}[n]\}$ for $N = 1, 2, 3, \dots$ be two sets of such sequences. Let the frequency interleaver $F_N(\cdot)$ be defined as

$$F_N \left(\frac{2\pi k}{N} + \theta \right) = \frac{2\pi p_N(k)}{N} + \theta \quad (3.94)$$

where $p_N(\cdot)$ is a permutation of the set $\mathcal{S}_N = \{0, 1, 2, \dots, N-1\}$, $k \in \mathcal{S}_N$, and $\theta \in [0, 2\pi/N)$. Let $\{F_N(\cdot)\}$ for $N = 1, 2, 3, \dots$ be a sequence of frequency interleavers with all permutations in \mathcal{S}_N being equally likely, and let $\{F_N^{-1}(\cdot)\}$ be the corresponding inverse frequency interleavers. Let the physical channel $h[n]$ and the frequency-interleaved channels $\{h_{F_N^{-1}}[n]\}$ have frequency responses of $H(\omega)$ and $\{H_{F_N^{-1}}(\omega) \triangleq H(F_N^{-1}(\omega))\}$ respectively, and let $H(\omega)$ be continuous almost everywhere on $[-\pi, \pi]$. Finally, let $\{B_{F_N^{-1}}^l(\omega) \triangleq B^l(F_N^{-1}(\omega))\}$ and $\{D_{F_N^{-1}}^l(\omega) \triangleq D^l(F_N^{-1}(\omega))\}$ be the frequency-interleaved versions of the filters given in (3.78) and (3.82). From (3.84), the slicer input can be expressed, for each n , as

$$\tilde{x}_N^l[n] = x_N[n] + u_N^l[n] \quad (3.95)$$

where $u_N^l[n]$ is complex-valued and zero-mean. Then, as $N \rightarrow \infty$, for a frequency-interleaved precoded system $u_N^l[n]$ converges in distribution to a marginally Gaussian white noise sequence that is uncorrelated with the entire sequence $x_N[n]$.

Thus, treating the cascade of the frequency-interleaved channel with the equalizer as an AWGN channel in Step 3 of the iterative algorithm for determining ρ^l is reasonably well justified. Note also that $(L - 1)/N \rightarrow 0$ as $N \rightarrow \infty$, so the overhead for the cyclic prefix becomes negligible.

As increasingly aggressive data rates are pursued in wideband systems to meet escalating traffic requirements, ISI becomes increasingly severe. We thus consider a corollary to Theorem 3.2 for the special case of severe-ISI channels. For the purposes of analysis, a convenient severe-ISI channel model⁵ we will exploit is one in which $h[n]$ is a finite impulse response (FIR) filter of length L , where L is large and the taps are mutually independent, zero-mean, complex-valued, circularly symmetric Gaussian random variables with variance $1/L$, independent of the data $x[n]$ and the noise $w[n]$.

When $L \rightarrow \infty$, the channel frequency response denoted by

$$H(\omega) = \sum_{n=0}^{L-1} h[n]e^{-j\omega n} \quad (3.96)$$

has the property that the channel frequency response at different frequencies is effectively uncorrelated. Specifically, the normalized correlation function of $H(\omega)$ approaches zero, i.e.⁶,

$$\rho_H(\omega, \nu) = \frac{\text{cov}(H(\omega), H(\nu))}{\sqrt{\text{var}H(\omega)}\sqrt{\text{var}H(\nu)}} \xrightarrow{p.w.a.e.} 0, \quad \text{as } L \rightarrow \infty \text{ when } \omega \neq \nu. \quad (3.97)$$

Since $H(\omega)$ is a Gaussian random process, (3.97) implies that arbitrarily close samples of $H(\omega)$ are independent, i.e., for every ω and ν such that $\omega \neq \nu$ we have that $H(\omega)$ and $H(\nu)$ are independent.

To verify this property, we first note that

$$E[H(\omega)] = E[H(\nu)] = 0 \quad (3.98)$$

$$\text{var}H(\omega) = \text{var}H(\nu) = 1 \quad (3.99)$$

⁵The theoretical information limits of such a channel are derived in Appendix C.

⁶We use $\xrightarrow{p.w.a.e.}$ to denote pointwise convergence almost everywhere.

where we have used (3.96) and the fact that $h[n]$ is a white zero-mean sequence. Similarly, we obtain

$$\begin{aligned}
\text{cov}(H(\omega), H(\nu)) &= \sum_{n=0}^{L-1} \sum_{m=0}^{L-1} E[h[n]h^*[m]]e^{-j\omega n}e^{j\nu m} \\
&= \sum_{n=0}^{L-1} E[h[n]h^*[n]]e^{-j(\omega-\nu)n} \\
&= \frac{1}{L} \sum_{n=0}^{L-1} e^{-j(\omega-\nu)n} \\
&= \frac{1}{L} e^{-j(\omega-\nu)(L-1)/2} \frac{\sin[(\omega-\nu)L/2]}{\sin[(\omega-\nu)/2]}. \tag{3.100}
\end{aligned}$$

Using (3.99) and (3.100) in (3.97), we get

$$|\rho_H(\omega, \nu)| = \begin{cases} 1 & \omega = \nu \\ \left| \frac{\sin[(\omega-\nu)L/2]}{L \sin[(\omega-\nu)/2]} \right| & \omega \neq \nu. \end{cases} \tag{3.101}$$

and, taking the limit as $L \rightarrow \infty$, we see that the property is verified.

Since arbitrarily close samples of $H(\omega)$ for an i.i.d. Gaussian ISI channel are independent, Theorem 3.2 applies in the limit as $L \rightarrow \infty$ and it is redundant to have a frequency-interleaved precoder at the transmitter. The following corollary characterizes the composite system consisting of this special channel in cascade with the multipass equalizer after l iterations [15].

Corollary 3.2 *Let $\{h_L[n]\}$ be a sequence L mutually independent, zero-mean, complex-valued, circularly symmetric Gaussian random variables with variance $1/L$. Then, as $L \rightarrow \infty$,*

$$\text{var } u^l[n] = \mathcal{E}_s(1 - (\rho^{l-1})^2) \left(\frac{1}{1 - \xi^l e^{\xi^l} E_1(\xi^l)} - 1 \right) \tag{3.102}$$

and

$$\gamma^l \xrightarrow{\text{m.s.}} \left(\frac{1}{\xi^l e^{\xi^l} E_1(\xi^l)} - 1 \right) \cdot \frac{1}{1 - (\rho^{l-1})^2}. \tag{3.103}$$

where

$$E_1(s) = \int_s^\infty \frac{e^{-t}}{t} dt \tag{3.104}$$

is the exponential integral.

Proof: Arbitrarily close samples of the asymptotic random process $\lim_{L \rightarrow \infty} H_L(\omega)$ are independent, complex-valued, circularly symmetric Gaussian random variables with zero mean and unity variance. Thus, arbitrarily close samples of $\lim_{L \rightarrow \infty} |H_L(\omega)|^2/\xi^l$ are independent, real-valued, exponentially distributed random variables with mean $\frac{1}{\xi^l}$. From (3.87),

$$\begin{aligned}
K^l &= \lim_{L \rightarrow \infty} \frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{1}{1 + |H_L(\omega)|^2/\xi^l} d\omega \\
&= \frac{1}{2\pi} \int_{-\pi}^{\pi} \lim_{L \rightarrow \infty} \frac{1}{1 + |H_L(\omega)|^2/\xi^l} d\omega \\
&= \int_0^{\infty} \left(\frac{1}{1 + \alpha} \right) \xi^l e^{-\xi^l \alpha} d\alpha \\
&= \xi^l e^{\xi^l} E_1(\xi^l)
\end{aligned} \tag{3.105}$$

where the second equality is an application of the bounded convergence theorem and the fourth equality is from the identity [1]

$$\int_0^{\infty} \frac{e^{-st}}{1+t} dt = e^s E_1(s). \tag{3.106}$$

Substituting (3.105) into (3.86) and (3.88) we obtain the desired result. \square

Note that the requirement that L go to infinity implies that the block length N must also go to infinity. Also, in Step 2 of the iterative algorithm for determining ρ^l , the SINR γ^l can be computed from ρ^l via (3.103), (3.34), and (1.6).

We now present some simulation results, starting with the random ISI channel model. In the ISI case, we have a square Toeplitz matrix, which means that β is unity. In Fig. 3.12, we plot the curves for the random ISI channel analogous to Fig. 3.3. The sequence of operating points is indicated on the solid curves with the \circ symbols. The set of operating points obtained from simulations is also indicated in Fig. 3.12 by the \times symbols. These results suggest that the theoretical predictions are quite accurate. In Fig. 3.13, bit-error rate is plotted as a function of SNR per bit for 1, 2, 3, 5, and an infinite number of iterations. We observe that steady-state performance is approximately achieved with comparatively few iterations. We emphasize that the complexity of a single pass of the iterated-decision equalizer is comparable to that of the DFE or the linear equalizer.

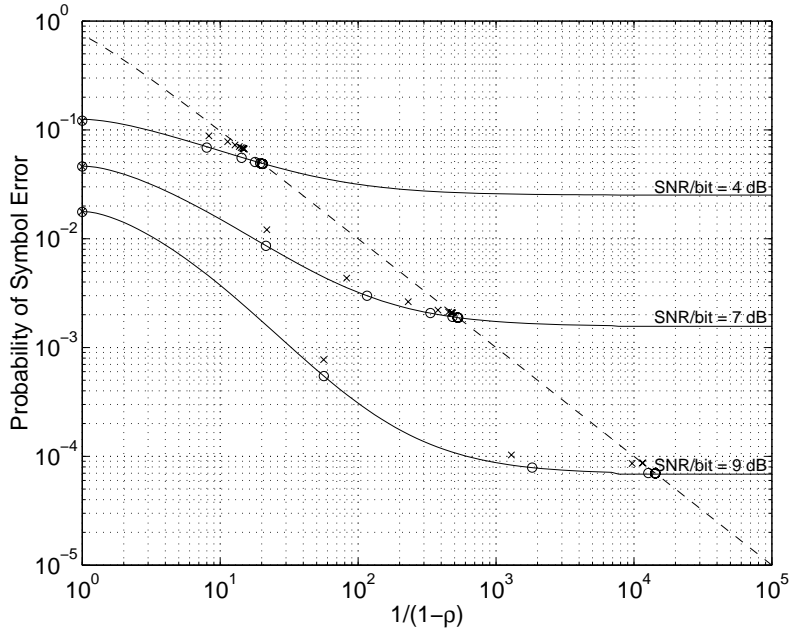


Figure 3.12: Iterated-decision equalizer performance. The curves plot QPSK symbol error rate as a function of the correlation coefficient ρ for various SNRs. Along each curve, \circ 's identify the theoretically predicted decreasing error rates achieved with $l = 1, 2, \dots$ decoding passes, and the intersections with the dashed line are the steady-state values ($l \rightarrow \infty$). The associated experimentally obtained values for $L = 256$ are depicted using \times 's.

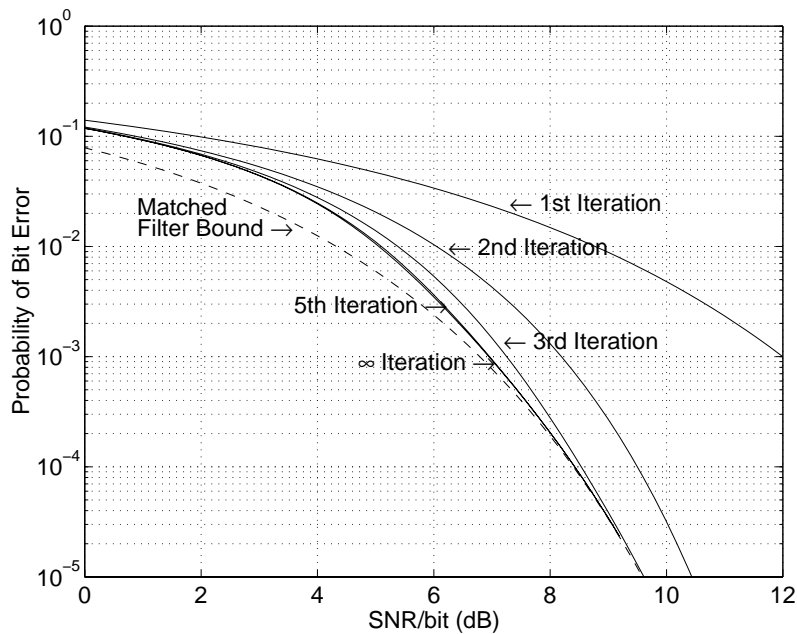


Figure 3.13: Theoretical iterated-decision equalizer performance as a function of SNR per bit. The successively lower solid curves depict the QPSK bit-error rate as a function of SNR per bit for 1, 2, 3, 5, and ∞ decoding iterations.

We now discuss the high-SNR ($\zeta \rightarrow 0$) performance of the equalizer. We first note from (3.34) that if $\rho \rightarrow 1$, then $\xi \rightarrow \infty$. Using the asymptotic series expansion [1]

$$E_1(t) = e^{-t} \sum_{k=0}^{\infty} (-1)^k \frac{k!}{t^{k+1}} \quad \text{for large } t \quad (3.107)$$

in (3.103) and retaining only the first two terms of the series, our SINR expression as $\rho \rightarrow 1$ becomes

$$\gamma \rightarrow \left(\frac{1}{\xi}\right) \cdot \frac{1}{1 - (\rho)^2} = \frac{1}{\zeta}, \quad (3.108)$$

where the equality follows from (3.34). When (3.108) is substituted into (3.28) or (3.29), we get the matched filter bound. Using the same reasoning as in Section 3.1.5, we can show that perfect ISI cancellation is approached at high SNR.

Figure 3.14 compares the theoretical performance ($L \rightarrow \infty$) of the iterated-decision equalizer with experimentally obtained results ($L = 256$). The experimental results are indeed consistent with theoretical predictions, especially at high SNR ($\zeta \rightarrow 0$) where the equalizer achieves the matched filter bound, i.e., $\gamma \rightarrow 1/\zeta$. For comparison, in Fig. 3.14 we also plot the theoretical error rates of the ideal MMSE-DFE, the MMSE linear equalizer, and the ZF linear equalizer, based on their asymptotic SINRs in the large ISI limit [15]

$$\gamma_{\text{MMSE-DFE}} = \exp\left\{e^{\zeta} E_1(\zeta)\right\} - 1 \quad (3.109)$$

$$\gamma_{\text{MMSE-LE}} = \frac{1}{\zeta e^{\zeta} E_1(\zeta)} - 1 \quad (3.110)$$

$$\gamma_{\text{ZF-LE}} = 0. \quad (3.111)$$

We can readily see that at moderate to high SNR, the iterated-decision equalizer requires significantly less transmit power than any of the other equalizers to achieve the same probability of error. Specifically, at high SNR ($\zeta \rightarrow 0$), we have from [15] that $\gamma_{\text{MMSE-DFE}} \rightarrow 1/\zeta e^{\Gamma_0}$ and $\gamma_{\text{MMSE-LE}} \rightarrow 1/[\zeta(-\Gamma_0 - \ln \zeta)] - 1$, where $\Gamma_0 = 0.57721 \dots$ denotes Euler's constant. Thus, the MMSE-DFE theoretically requires e^{Γ_0} times or $10\Gamma_0 \log e \approx 2.507$ dB more transmit power to achieve the same probability of error as the iterated-decision equalizer. Moreover, as $\zeta \rightarrow 0$, the MMSE-LE requires increasingly more transmit power than the iterated-decision equalizer to achieve the same probability of error. The ZF-LE is even worse: $\gamma_{\text{ZF-LE}} = 0$ for all ζ , which is expected since the zeros of the random channel con-

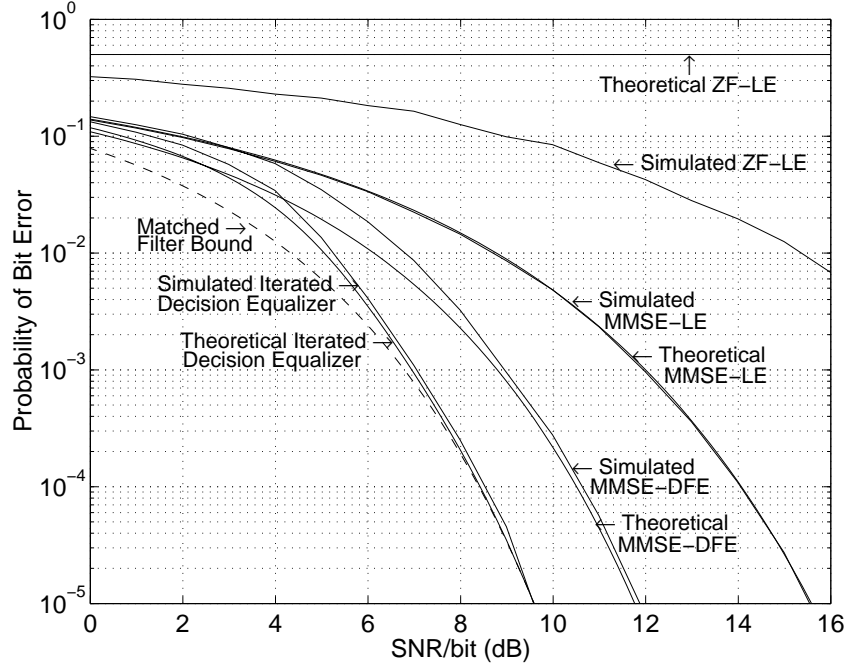


Figure 3.14: Theoretical ($L \rightarrow \infty$) and experimentally observed ($L = 256$) performance for various equalizers. The solid curves depict QPSK bit-error rates as a function of SNR per bit.

verge uniformly on the unit circle in the long ISI limit [10]. These results emphasize the strong suboptimality of conventional equalizers.

Our simulations and plots in the remainder of this section are based on the three-tap channel with impulse response

$$h[n] = 0.5\delta[n] + 0.707\delta[n - 1] + 0.5\delta[n - 2] \quad (3.112)$$

with $N = 8192$ and frequency-interleaved precoding. Figure 3.15 shows the probability of bit error as a function of SNR after a different number of iterations. We see that there is a sharp performance threshold between 6 and 7 dB; below the threshold performance is poor, but above the threshold the matched filter bound is achieved. We find the explanation for this threshold in Fig. 3.16, which shows that as the SNR decreases, the space between the solid convergence curve and the dashed convergence curve gets narrower until the two curves meet, resulting in poor performance. This phenomenon is similar to the β threshold observed in Fig. 3.5. In Fig. 3.17, we compare the bit-error rate as a function of SNR for the iterated-decision equalizer with frequency-interleaved precoding with

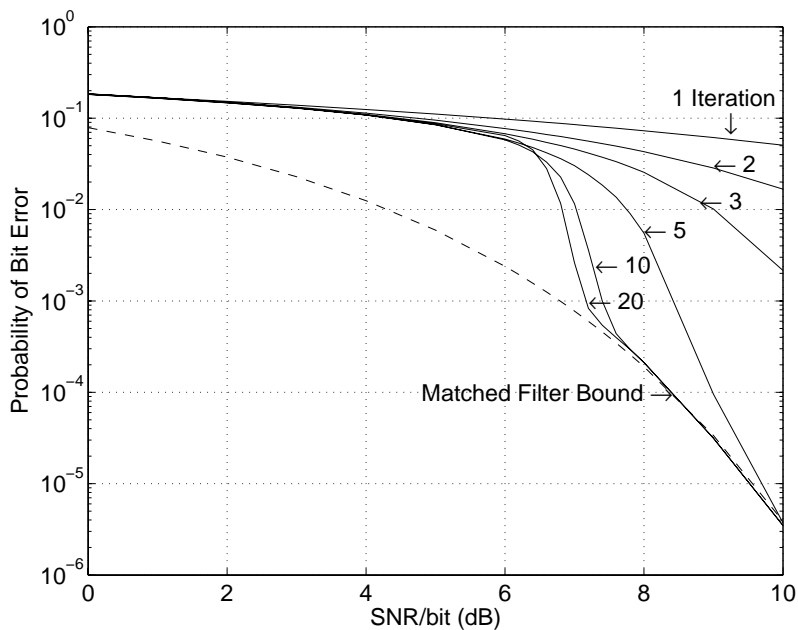


Figure 3.15: Theoretical iterated-decision equalizer performance with frequency-interleaved precoding for the three-tap channel (3.112). The successively lower curves plot the QPSK symbol error rate as a function of SNR per bit for 1, 2, 3, 5, 10, and 20 detection iterations.

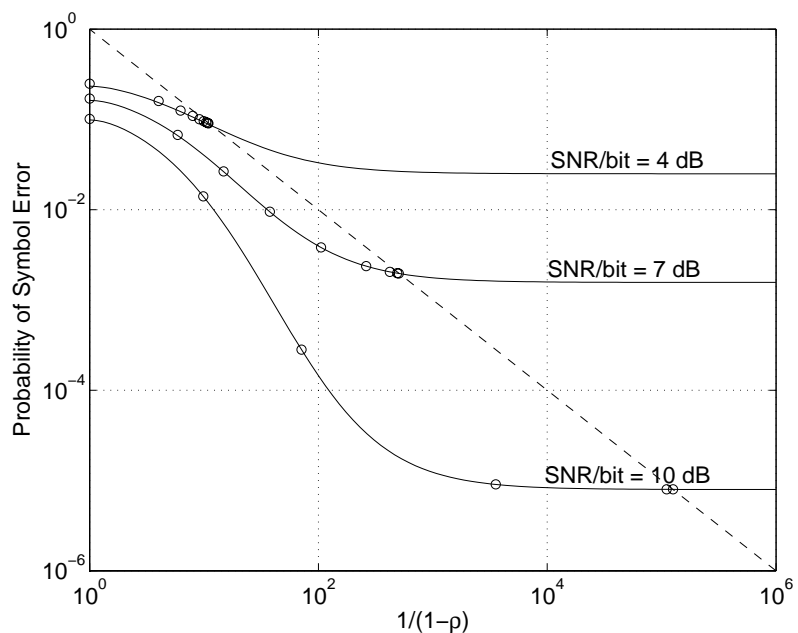


Figure 3.16: Iterated-decision equalizer performance with frequency-interleaved precoding. The solid curves plot QPSK symbol error rate as a function of ρ for the three-tap channel (3.112) at various SNRs. Along the curve, \circ 's identify the theoretically predicted decreasing error rates achieved with $l = 1, 2, \dots$ decoding passes, and the intersections with the dashed line are the steady-state values ($l \rightarrow \infty$).

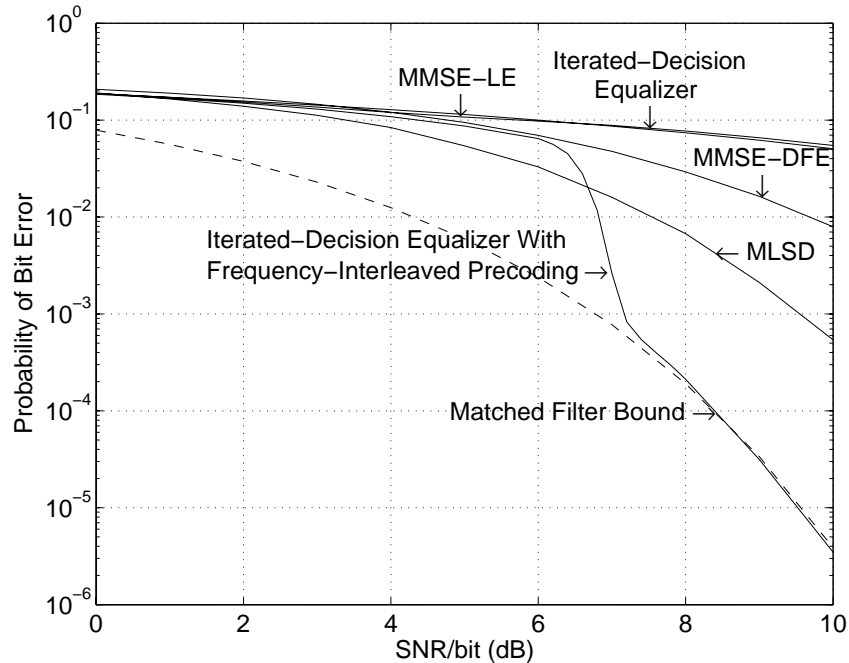


Figure 3.17: Experimentally observed performance for various equalizers. The solid curves depict QPSK bit-error rates for the 3-tap channel in (3.112) as a function of SNR per bit. Except for the iterated-decision detector which is used with and without frequency-interleaved precoding, the various other detectors are not used in conjunction with precoding. Since the iterated-decision detector with frequency-interleaved precoding achieves the matched filter bound at high SNR, we can infer that ML detection with precoding (not pictured) would also achieve the matched filter bound.

various other equalizers. The iterated-decision equalizer with frequency-interleaved precoding clearly outperforms various other equalizers, including MLSD and the iterated-decision equalizer without precoding, and indeed approaches the matched filter bound at high SNR. Although MLSD with precoding is too complex to implement, we can infer that its performance lies between the matched filter bound and the curve for the iterated-decision equalizer with precoding. Note that the SNR gap between the MLSD performance without precoding and the matched filter bound is rather significant. In fact, it can be shown that for square QAM constellations this particular channel is the worst unit-energy three-tap channel in terms of asymptotic SNR loss [51], the gap between the MLSD performance and the matched filter bound at high SNR. At high SNR, the SNR loss approaches 2.3 dB. The iterated-decision equalizer without precoding also performs poorly, because the Gaussian assumption for the effective noise process at the slicer input is not valid without frequency interleaving. As discussed in Section 2.3, if frequency-interleaved precoding is used in con-

junction with a linear equalizer (LE) or DFE, there is no change in the minimum-mean square slicer error [43] when $h[n]$ is transformed into $h_{p-1}[n]$, so there is no obvious benefit in combining precoding with an LE or DFE.

3.3 Adaptive Iterated-Decision Detector

In Section 3.1, we derived the optimal matrices \mathbf{B}^l and \mathbf{D}^l given that the channel matrix \mathbf{H} is known. We now develop an adaptive implementation of the iterated-decision detector, in which optimal matrices are selected automatically (from the received data) without explicit knowledge of the channel. The received vector sequence is

$$\mathbf{r}[n] = \mathbf{H}\mathbf{x}[n] + \mathbf{w}[n], \quad (3.113)$$

for $n = 0, 1, \dots, \tau - 1$, where τ is the length of the packet. We assume that the packet length is chosen small enough such that the channel encountered by each packet appears fixed. Some of the symbols in the vector sequence $\mathbf{x}[n]$ (not necessarily at the head of the packet) are for training and are known at the receiver, while the rest are data symbols that are unknown at the receiver.

Before the first pass ($l = 1$) of the adaptive iterated-decision detector, we need to initialize the hard decisions $\hat{x}_i^0[n]$. Since the locations and values of the training symbols in each packet are known at the receiver, we set $\hat{x}_i^0[n] = x_i[n]$ for the i and n corresponding to those locations. For all other locations in the packets, we set $\hat{x}_i^0[n]$ to be the mean of the signal constellation, typically zero.

On the l th pass of the detector where $l = 1, 2, 3, \dots$, the i th component of the slicer input $\tilde{\mathbf{x}}^l[n]$ can be expressed as

$$\tilde{x}_i^l[n] = \mathbf{a}_i^{l\dagger} \mathbf{k}_i^l[n] \quad (3.114)$$

where

$$\mathbf{a}_i^l = \left[b_{1,i}^l \ b_{2,i}^l \ \cdots \ b_{Q,i}^l \ -d_{1,i}^l \ \cdots \ -d_{i-1,i}^l \ -d_{i+1,i}^l \ \cdots \ -d_{N,i}^l \right]^T \quad (3.115)$$

$$\mathbf{k}_i^l[n] = \left[r_1[n] \ r_2[n] \ \cdots \ r_Q[n] \ \hat{x}_1^{l-1}[n] \ \cdots \ \hat{x}_{i-1}^{l-1}[n] \ \hat{x}_{i+1}^{l-1}[n] \ \cdots \ \hat{x}_N^{l-1}[n] \right]^T \quad (3.116)$$

with $b_{j,k}^l$ and $d_{j,k}^l$ being the jk th elements of \mathbf{B}^l and \mathbf{D}^l respectively. The slicer then gener-

ates the hard decisions $\hat{x}_i^l[n]$ from $\tilde{x}_i^l[n]$ for all i and n , except for those values corresponding to the locations of training symbols in $x_i[n]$. For those n , we set $\hat{x}_i^l[n] = x_i[n]$.

In the l th iteration, there are two sets of data available to the receiver: $\mathbf{r}[n]$ and $\hat{\mathbf{x}}^{l-1}[n]$, $n = 0, 1, \dots, \tau - 1$. If we assume that $x_i[n] \approx \hat{x}_i^{l-1}[n]$ for all i and all n for the purposes of determining the optimal matrices, then it is reasonable to choose \mathbf{b}_i and \mathbf{d}_i so as to minimize the sum of error squares:

$$\mathcal{E}(\mathbf{a}_i^l) = \sum_{n=-\infty}^{\infty} |\hat{x}_i^{l-1}[n] - \mathbf{a}_i^{l\dagger} \mathbf{k}_i^l[n]|^2. \quad (3.117)$$

Since this is a linear least-squares estimation problem, the optimum \mathbf{a}_i^l is [39]

$$\mathbf{a}_{i,\text{opt}}^l = [\Phi_i^l]^{-1} \mathbf{u}_i^l, \quad (3.118)$$

where $\Phi_i^l = \sum_{n=-\infty}^{\infty} \mathbf{k}_i^l[n] \mathbf{k}_i^{l\dagger}[n]$ and $\mathbf{u}_i^l = \sum_{n=-\infty}^{\infty} \hat{x}_i^{l-1*}[n] \mathbf{k}_i^l[n]$. The matrices Φ_i^l can be efficiently obtained by eliminating the $(Q+i)$ th row and column of $\Phi^l = \sum_{n=-\infty}^{\infty} \mathbf{k}^l[n] \mathbf{k}^{l\dagger}[n]$ where $(\mathbf{k}^l[n])^T = [(\mathbf{r}[n])^T (\hat{\mathbf{x}}^{l-1}[n])^T]$, and $[\Phi_i^l]^{-1}$ can be efficiently computed using formulas for the inversion of a partitioned matrix [46].

We now present some simulation results for adaptive iterated-decision detection for i.i.d. Gaussian \mathbf{H} . The block-iterative nature of the detector allows the training symbols to be located anywhere in the packet. Since the locations do not appear to affect performance, we arbitrarily choose to uniformly space vectors of training symbols within the sequence $\mathbf{x}[n]$ for $n = 0, 1, \dots, \tau - 1$. In Fig. 3.18, we plot the bit-error rate of the adaptive iterated-decision equalizer as a function of the number of iterations, for varying amounts of training data. The graph strongly suggests that there is a threshold for the number of training symbols, below which the adaptive detector performs poorly and above which the bit-error rate consistently converges to approximately the same steady-state value regardless of the exact number of training symbols. The excess training data is still important though, since the bit-error rate converges quicker with more training data. In Fig. 3.19, we plot the probability of bit error as a function of SNR for varying amounts of training data. We see that, as expected, performance improves as the amount of training data is increased. Moreover, only a modest amount of training symbols is required at high SNR for the adaptive detector to perform as if the channel were exactly known at the receiver. For comparison purposes, we also

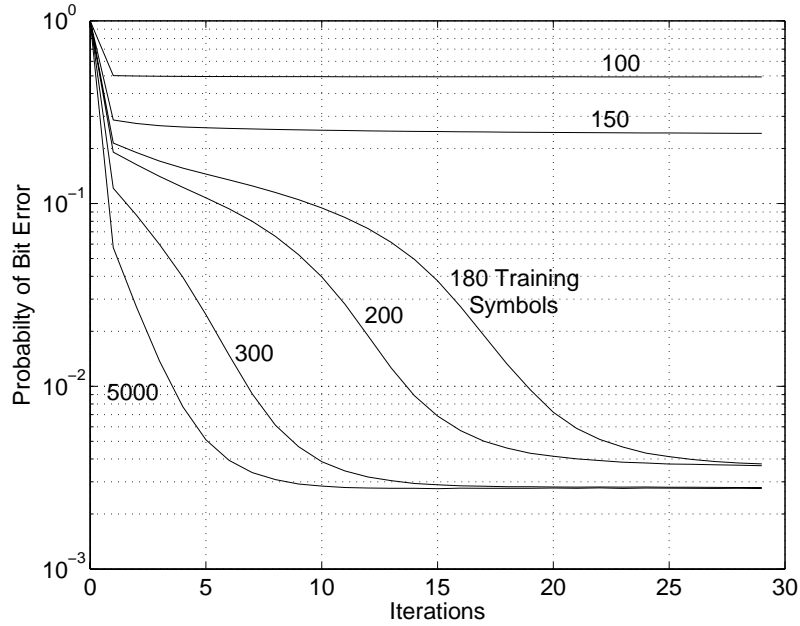


Figure 3.18: Experimentally observed QPSK bit-error rate for the adaptive iterated-decision detector as a function of the number of decoding iterations and the number of training vectors transmitted with each packet of 10000 data vectors at an SNR per bit of 7 dB. The channel is an i.i.d. Gaussian matrix with $\beta = N/Q = 1$.

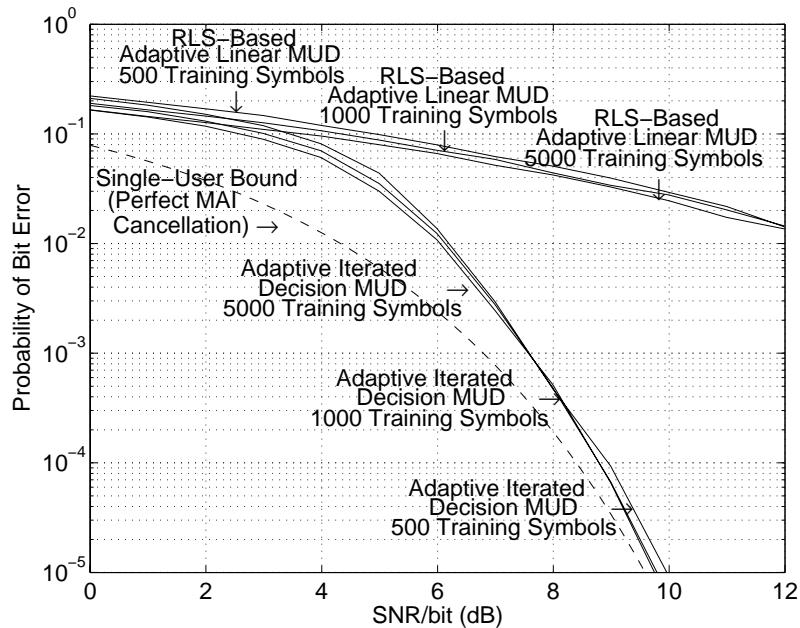


Figure 3.19: Experimentally observed ($Q = 128$) QPSK bit-error rates for the adaptive iterated-decision detector and the RLS-based adaptive linear detector (forgetting factor $\lambda = 1$), with $\beta = N/Q = 1$. Each packet consists of 10000 data vectors plus either 500, 1000, or 5000 additional vectors of training symbols.

plot in Fig. 3.19 the performance of the RLS-based implementation of the adaptive linear detector [40]. The linear detector performs significantly worse than the iterated-decision detector for comparable amounts of training data.

This adaptive iterated-decision detector can also be applied to a frequency-interleaved precoded ISI channel, for which \mathbf{H} is a circulant matrix. One issue that remains to be explored is the development of an adaptive version of the frequency-interleaved iterated-decision equalizer that exploits the circulant structure. For example, it is plausible to somehow use the least-squares criterion to solve for the optimal $B^l[k]$ and $D^l[k]$ in the DFT domain, with the constraint that $\frac{1}{N} \sum_{k=0}^{N-1} D^l[k] = 0$. Or perhaps the least-squares criterion can be used to solve for the optimal filter taps $b^l[n]$ and $d^l[n]$ corresponding to the original ISI channel taps $h[n]$, and then frequency interleaving those optimal filters to match the frequency-interleaved ISI channel.

Solving for the optimal filter taps $b^l[n]$ and $d^l[n]$ corresponding to the original ISI channel taps $h[n]$ is particularly simple because the structure of the \mathbf{H} matrix is a banded Toeplitz matrix, with the length of the channel impulse response L much less than the block length N . Since the filters $b^l[n]$ and $d^l[n]$ for the l th iteration are finite-length filters, we would ideally like them to approximate (3.78) and (3.82), which are infinite length, without using an excessive number of taps. Since the optimal $b^l[n]$ in (3.78) includes a filter matched to $h[n]$, and the optimal $d^l[n]$ in (3.82) includes a cascade of $h[n]$ and the corresponding matched filter, a reasonable rule of thumb is to select L strictly anticausal taps and L strictly causal taps for each filter. The slicer input $\tilde{x}^l[n]$ can then be expressed as

$$\tilde{x}^l[n] = \mathbf{a}^l \mathbf{k}^l[n] \quad (3.119)$$

where

$$\mathbf{a}^{l \dagger} = \left[b^l[-L] \cdots b^l[0] \cdots b^l[L] \quad -d^l[-L] \cdots -d^l[-1] \quad -d^l[1] \cdots -d^l[L] \right] \quad (3.120)$$

$$\mathbf{k}^l[n] = \left[r[n+L] \cdots r[n] \cdots r[n-L] \quad \hat{x}^{l-1}[n+L] \cdots \hat{x}^{l-1}[n+1] \quad \hat{x}^{l-1}[n-1] \cdots \hat{x}^{l-1}[n-L] \right]^T. \quad (3.121)$$

The optimum \mathbf{a}^l is [39]

$$\mathbf{a}_{\text{opt}}^l = [\mathbf{\Phi}^l]^{-1} \mathbf{u}^l, \quad (3.122)$$

where $\mathbf{\Phi}^l = \sum_{n=-\infty}^{\infty} \mathbf{k}^l[n] \mathbf{k}^{l \dagger}[n]$ and $\mathbf{u}^l = \sum_{n=-\infty}^{\infty} \hat{x}^{l-1*}[n] \mathbf{k}^l[n]$. The resulting equalizer

lends itself readily to practical implementation, even for large filter lengths. In particular, the matrix Φ^l can be efficiently computed using correlation functions involving $r[n]$ and $\hat{x}^{l-1}[n]$ [39], and $[\Phi^l]^{-1}$ can be efficiently computed using formulas for the inversion of a partitioned matrix [46].

We plot in Fig. 3.20 the performance of the adaptive iterated-decision equalizer and the recursive least squares (RLS) based implementation of the adaptive DFE [39] for 128-tap random ISI channels. The DFE performs significantly worse than the iterated-decision equalizer for comparable amounts of training data. Indeed, the high SNR gap is even larger than the 2.507 dB determined for the nonadaptive case. This is because, as Figs. 3.14 and 3.20 show, the performance of the adaptive DFE is not accurately predicted by the nonadaptive MMSE-DFE, even in the limit as $L \rightarrow \infty$. It is also worth stressing that the RLS-based adaptive DFE is much more computationally expensive than the adaptive iterated-decision equalizer because the RLS-based DFE requires the multiplication of large

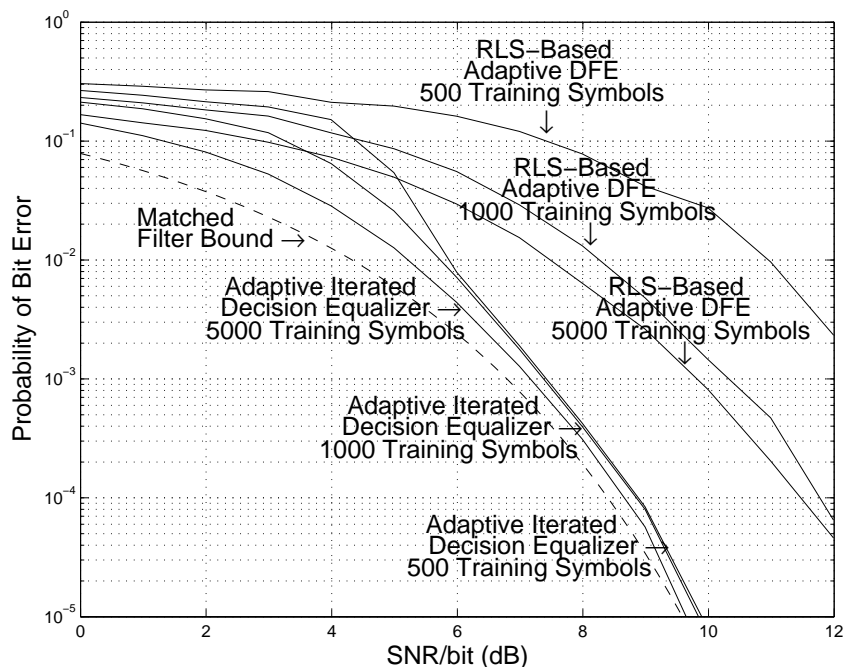


Figure 3.20: Experimentally observed QPSK bit-error rate for the adaptive iterated-decision equalizer and the RLS-based adaptive DFE (with forgetting factor $\lambda = 1$) as a function of SNR per bit. Blocks of 10000 data symbols were transmitted through 128-tap channels, which were equalized using 257 feedforward taps and 256 noncausal feedback taps in the case of the iterated-decision equalizer, and using 257 feedforward taps and 128 strictly causal feedback taps in the case of the DFE.

matrices for *each* transmitted symbol, whereas the iterated-decision equalizer essentially requires the computation of one large matrix inverse per iteration for *all* the symbols in the packets, with the number of required iterations being typically small.

Chapter 4

Iterated-Decision Detection as Low-Bandwidth Message Passing

In the previous chapter, we showed that asymptotically all interference can be cancelled by using the iterated-decision detector with mode-interleaved precoding. Thus, with low complexity, performance close to the AWGN channel bound and the more complex ML detector can be achieved.

In this chapter, we explain why both the iterated-decision detector and ML detection asymptotically perform the same despite the gap in complexity when mode-interleaved precoding is present. We show that the iterated-decision detector is a simplified version of a relative of the sum-product algorithm [42] that approximates ML detection. For convenience, we focus our analysis primarily on the iterated-decision equalizer, and comment on the general detector case at the end of the chapter.

In Section 4.1, we review the sum-product algorithm. In Section 4.2, we apply the sum-product algorithm to the problem of detection over ISI channels. When the sum-product algorithm is applied to a trellis as is commonly done, the well-known forward/backward algorithm [4] results, which computes the exact *a posteriori* probabilities. When the sum-product algorithm is alternatively applied to a graph with cycles, an algorithm that approximates the *a posteriori* probabilities is obtained. In Section 4.3, we describe the max-log simplification to the sum-product algorithm, which turns out to provide an exact solution to a slightly different problem. This modified algorithm, the max-sum algorithm, is the sum-product algorithm in the max-sum semiring. We show that the max-sum algorithm on

a trellis solves the ML detection problem, so the max-sum algorithm on a graph with cycles can be expected to approximately solve the ML detection problem. In Section 4.4, we show that with mode-interleaved precoding, the iterated-decision equalizer has strong similarities to the max-sum algorithm on a particular graph with cycles, thereby providing an intuitive explanation for the asymptotically optimal performance of the iterated-decision equalizer. We conclude the chapter by generalizing the analysis to the iterated-decision detector in Section 4.5.

4.1 The Sum-Product Algorithm

Often times we may wish to compute a marginal function $g_n(x_n)$ of a multivariate function $g(x_1, \dots, x_N)$, defined as

$$g_n(x_n) = \sum_{x_1} \cdots \sum_{x_{n-1}} \sum_{x_{n+1}} \cdots \sum_{x_N} g(x_1, \dots, x_N). \quad (4.1)$$

We may do this by exhaustively summing over all possible combinations of the summation variables, but potentially dramatic computational savings can be obtained by exploiting the distributive law if $g(x_1, \dots, x_N)$ can be factored into a product of local functions:

$$g(x_1, \dots, x_N) = \prod_{X \in Q} g_X(X) \quad (4.2)$$

where Q is set of all subsets of $\{x_1, \dots, x_N\}$.

For example, suppose g is a function of x_1, x_2, x_3, x_4, x_5 and can be expressed as the product

$$g(x_1, x_2, x_3, x_4, x_5) = g_A(x_1, x_3)g_B(x_2)g_C(x_2, x_3, x_4)g_D(x_4, x_5). \quad (4.3)$$

The marginal function $g_1(x_1)$ can be expressed as

$$\begin{aligned} g_1(x_1) &= \sum_{x_2} \sum_{x_3} \sum_{x_4} \sum_{x_5} g_A(x_1, x_3)g_B(x_2)g_C(x_2, x_3, x_4)g_D(x_4, x_5) \\ &= \sum_{x_3} g_A(x_1, x_3) \left(\sum_{x_2} g_B(x_2) \left(\sum_{x_4} g_C(x_2, x_3, x_4) \left(\sum_{x_5} g_D(x_4, x_5) \right) \right) \right). \end{aligned} \quad (4.4)$$

Alternatively, we may rewrite (4.4) using “not-sum” notation [42], whereby summations of

a function are performed over all but one of the function's arguments. In not-sum notation, the marginal function $g_1(x_1)$ is

$$g_1(x_1) = \sum_{\sim\{x_1\}} g_A(x_1, x_3) \left(\sum_{\sim\{x_3\}} g_B(x_2) g_C(x_2, x_3, x_4) \left(\sum_{\sim\{x_4\}} g_D(x_4, x_5) \right) \right), \quad (4.5)$$

where $\sum_{\sim\{x_n\}}$ is the summation operator over all arguments of the function except for x_n .

Similarly, the marginal function $g_2(x_2)$ is

$$g_2(x_2) = g_B(x_2) \left(\sum_{\sim\{x_2\}} g_C(x_2, x_3, x_4) \left(\sum_{\sim\{x_3\}} g_A(x_1, x_3) \right) \left(\sum_{\sim\{x_4\}} g_D(x_4, x_5) \right) \right). \quad (4.6)$$

The purpose of the sum-product algorithm is to efficiently compute marginal functions using expressions such as (4.5) and (4.6) that are derived from the distributive law.

4.1.1 Single-Marginal Sum-Product Algorithm

Expressions like (4.5) and (4.6) can be represented by ordered rooted trees in which internal nodes represent operators and leaf nodes represent variables or constants [54]. For example, the expression $(x + y) \times z$ can be represented by the tree in Fig. 4.1. With expressions consisting exclusively of symmetric operators like multiplication and addition, an unordered rooted tree unambiguously represents the expression.

A factor graph [42] describes the structure of a specific factorization of a multivariate function using variable nodes, factor nodes, and connecting edges. In the factor graph for the factorization, there exists a variable node for each variable x_n in the multivariate function, a factor node for each local function g_X , and an edge connecting a variable node x_n to a factor node g_X if and only if x_n is an argument of g_X . Factor graphs can be classified

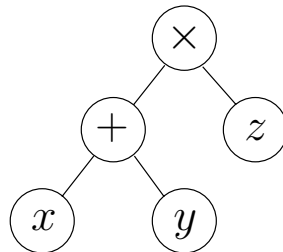


Figure 4.1: An expression tree representing $(x + y) \times z$.

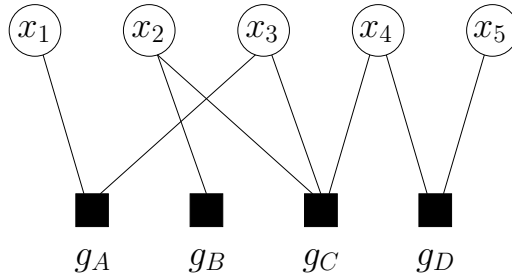


Figure 4.2: A factor graph for the factorization $g_A(x_1, x_3)g_B(x_2)g_C(x_2, x_3, x_4)g_D(x_4, x_5)$.

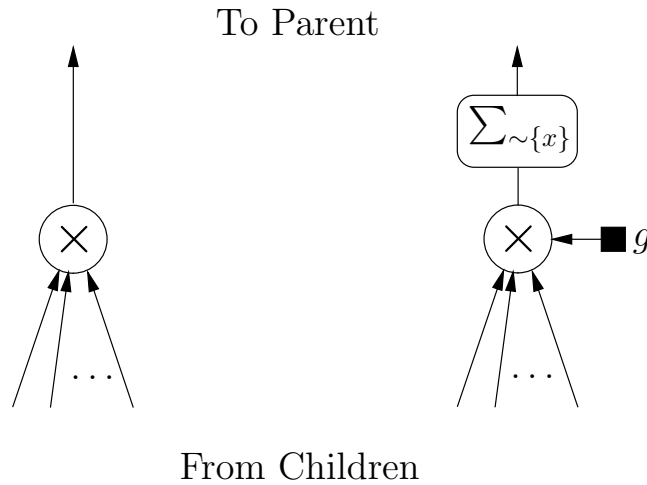


Figure 4.3: Local substitutions that transform a rooted cycle-free graph to an expression tree for a marginal function. On the left is the substitution at a variable node; on the right is the substitution at a factor node g with parent x .

as bipartite graphs, which are defined as graphs in which all vertices are partitioned into two mutually exclusive sets such that no edges exist between vertices in the same set. The factor graph corresponding to the particular factorization of $g(x_1, x_2, x_3, x_4, x_5)$ in (4.3) is depicted in Fig. 4.2.

When a factor graph has no cycles as in Fig. 4.2, the expression tree for the marginal function $g_n(x_n)$ has a direct relationship with the factor graph redrawn as a tree rooted at x_n . In this redrawn factor graph, if the variable nodes are replaced by a product operator, the factor nodes are replaced by a “form product and multiply by g ” operator, and the edges between each variable node x and its children are associated with the not-sum operator $\sum_{\sim\{x\}}$, then we obtain exactly the expression tree for $g_n(x_n)$. These local substitutions are illustrated in Fig. 4.3. The factor graph of Fig. 4.2 redrawn as a tree rooted at x_1 is shown in Fig. 4.4(a), and the corresponding expression tree for $g_1(x_1)$ is shown in Fig. 4.4(b).

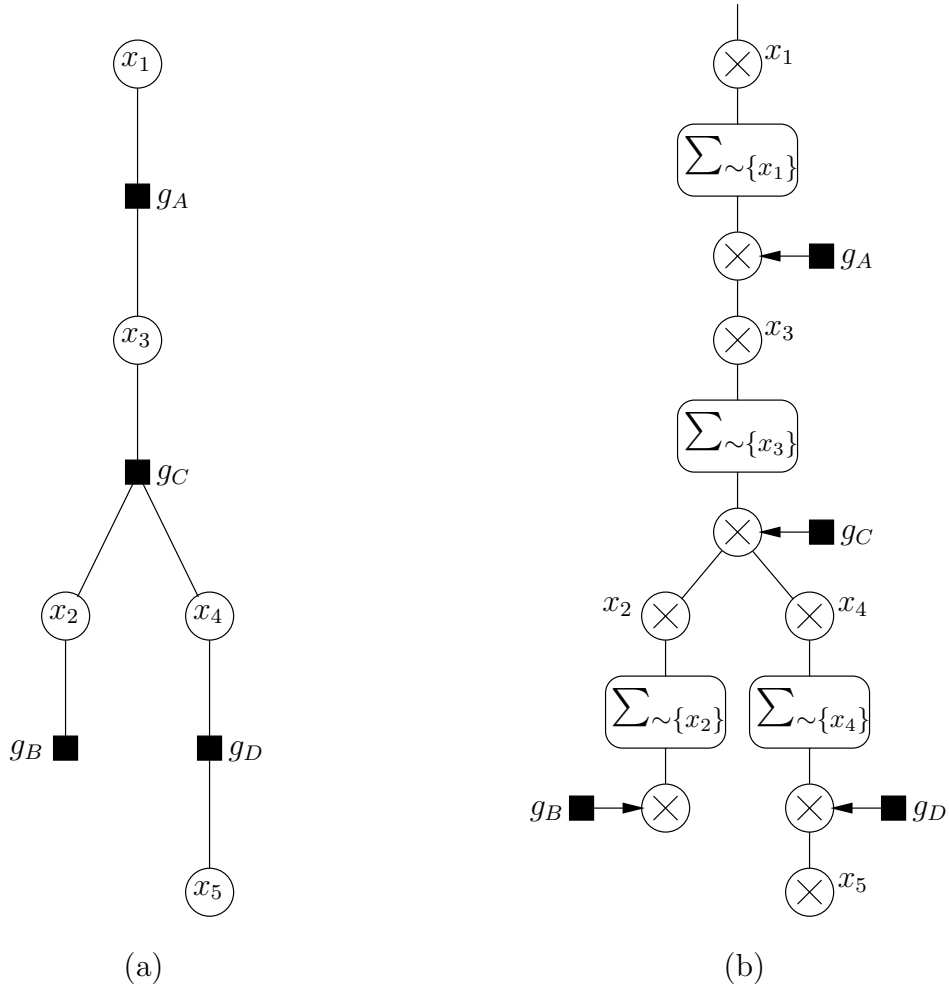


Figure 4.4: (a) The factor graph of Fig. 4.2 redrawn as a tree rooted at x_1 . (b) The corresponding expression tree for $g_1(x_1)$.

Similarly, the factor graph redrawn as a tree rooted at x_2 is shown in Fig. 4.5(a), and the corresponding expression tree for $g_2(x_2)$ is shown in Fig. 4.5(b).

This rather remarkable relationship allows us to use the rooted factor graph to visualize and describe the algorithm for computing a marginal function $g_n(x_n)$ as a sequence of “messages” sent along all edges from the leaf nodes to the root. These messages are always a description of some function, and a “product of messages” is always interpreted as the product of the functions described by the messages, not the product of the messages themselves. Similarly, a not-sum for x operates not on literal messages, but of the functions described by the messages.

In this algorithm, each node must wait for messages sent by all of its children before

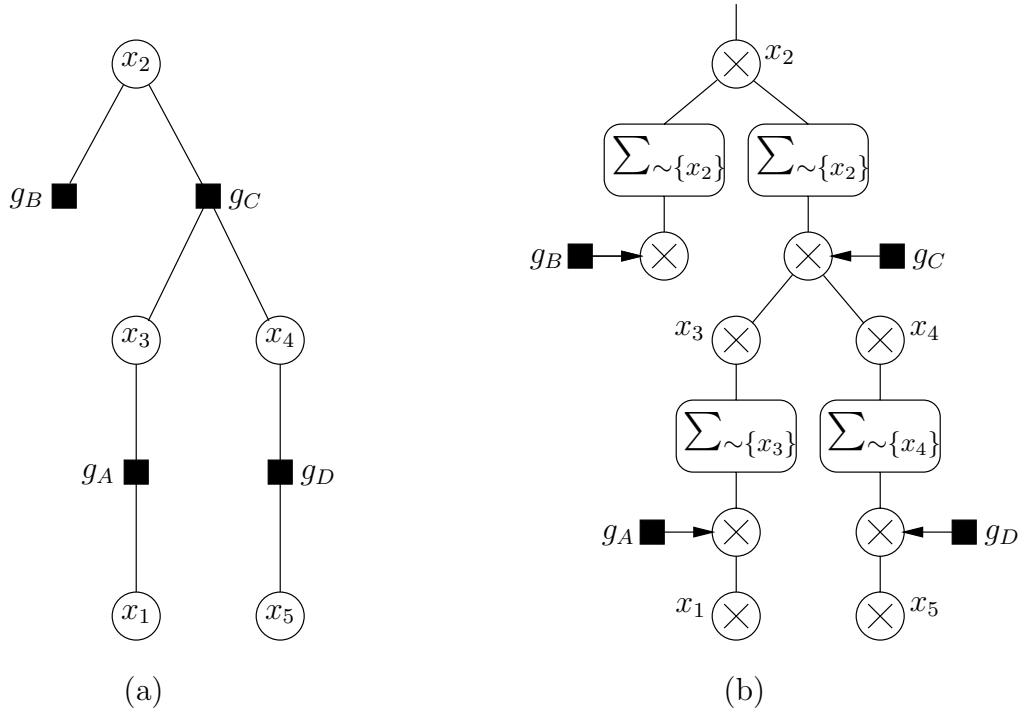


Figure 4.5: (a) The factor graph of Fig. 4.2 redrawn as a tree rooted at x_2 . (b) The corresponding expression tree for $g_2(x_2)$.

sending its own message to its parent. To begin, each leaf variable node sends the identity function (the constant unity function) to its parent, and each leaf factor node g sends to its parent x the result of a not-sum for x of the function g . Each internal variable node sends the product of the functions arriving from all its children upwards to its parent, while each internal factor node g sends to its parent x the output of the not-sum operation for x of the product of g with all the functions from its children. Products with only one operand act as identity operators. These operations are shown in Fig. 4.3. Eventually, the algorithm terminates at the root when the product of the functions sent by its children gives the desired marginal function $g_n(x_n)$. This algorithm shall be referred to as the “single-marginal sum-product algorithm,” since only one marginal is computed using various sum and product operators. Note that a function passed on the edge $\{x, g\}$ is always a function of only x , regardless of whether the function is passed from x to g or from g to x . We can interpret these single-argument messages between child and parent as a summary for x of the product of local functions in the child’s subtree.

4.1.2 The Complete-Marginal Sum-Product Algorithm

To obtain all the marginal functions of a function $g(x_1, \dots, x_n)$ associated with a cycle-free factor graph, one could use the “single-marginal sum-product algorithm” for each variable x_n . However, doing so is likely to be inefficient since one can take advantage of the fact that many intermediate computations can be recycled amongst the various single-marginal sum-product algorithms.

The sum-product algorithm, which efficiently combines the various single-marginal sum-product algorithms, can be visualized and described on an unrooted factor graph. Essentially, all of the rooted, cycle-free factor trees are overlaid onto the unrooted, cycle-free factor graph, so messages are passed along the edges in both directions. Hence, parent-child relationships are no longer permanent; the temporary designations of parent and child depend on the direction a message is sent along an edge.

The rule that governs the algorithm is that a node sends a message to a neighbor (temporary parent) when messages from each of the other neighbors (temporary children) have been received and appropriately combined, just as in the single-marginal sum-product algorithm. Thus, the algorithm begins at the leaf nodes, where messages are passed to their neighbors. Any internal node remains idle until all but one of its neighbors have sent it a message. At this point, the node appropriately combines the received messages and sends the resulting message to the remaining neighbor (temporary parent). The node becomes idle once again, until the remaining neighbor (now a temporary child) responds with a message. Having now received messages from all its neighbors, the node can finish sending messages to all its neighbors by appropriately combining received messages. The algorithm terminates when messages have been passed exactly twice along each edge of the factor graph, once in each direction. Each marginal function is obtained by taking the product of all messages sent to the corresponding variable node by all its neighbors.

As with the single-marginal algorithm, both messages passed in opposite directions on the edge $\{x, g\}$ are always a function of only the variable x , and we can interpret a single-argument message between neighbors as a summary for x of the product of local functions in the message sender’s subgraph.

4.2 Applying Sum-Product Algorithm to ISI Channels

Consider a sequence of symbols¹ $\mathbf{x} = \{x_n\}$ selected from corresponding discrete sets $\{\mathcal{X}_n\}$ and transmitted over an ISI channel with output

$$r_n = \sum_k h_{n-k}x_k + w_n. \quad (4.7)$$

In this case, the global function we are interested in is the *a posteriori* joint probability mass function for $\mathbf{x} = \{x_n\}$ given the fixed observation $\mathbf{r} = \{r_n\}$:

$$g(\mathbf{x}) = p(\mathbf{x}|\mathbf{r}) \propto p(\mathbf{x})f(\mathbf{r}|\mathbf{x}), \quad (4.8)$$

where we have used Bayes' rule. If the *a priori* distribution $p(\mathbf{x})$ for the transmitted vectors is uniform, then

$$g(\mathbf{x}) \propto f(\mathbf{r}|\mathbf{x}). \quad (4.9)$$

The corresponding marginal functions have the form

$$g_n(x_n) = p(x_n|\mathbf{r}) \propto \sum_{\sim\{x_n\}} f(\mathbf{r}|\mathbf{x}) \quad (4.10)$$

which, for each possible value of x_n , is separate summation over all variables except x_n . It follows that selecting the value of x_n for which $g_n(x_n)$ is largest gives the most likely value of x_n given the entire vector \mathbf{r} , thereby minimizing the probability of error for each individual symbol.

4.2.1 Trellis Processing

The typical application of the sum-product algorithm to the detection of information sent via an ISI channel is usually the forward/backward algorithm [4], also known as the BCJR algorithm (after inventors Bahl, Cocke, Jelinik and Raviv), *a posteriori* probability (APP) algorithm, or *maximum a posteriori* (MAP) algorithm.

This algorithm applies the sum-product algorithm to a cycle-free trellis that represents a modified form of the global function $p(\mathbf{x}|\mathbf{r})$ through a hidden Markov model. If we define $y_n \triangleq \sum_k h_{n-k}x_k$, then we have a Markov model with input variables $\mathbf{x} = \{x_n\}$, state

¹For notational convenience in this chapter, we use subscripts rather than arguments to denote time.

variables $\mathbf{s} = \{s_n\}$, and output variables $\mathbf{y} = \{y_n\}$. The Markov model is hidden because \mathbf{y} is not observable; rather, we are able to observe only \mathbf{r} , a noisy version of \mathbf{y} :

$$r_n = \sum_k h_{n-k} x_k + w_n = y_n + w_n. \quad (4.11)$$

The *a posteriori* joint probability mass function for \mathbf{x} , \mathbf{s} , and \mathbf{y} given the fixed observation \mathbf{r} is

$$\begin{aligned} p(\mathbf{x}, \mathbf{s}, \mathbf{y} | \mathbf{r}) &\propto p(\mathbf{x}, \mathbf{s}, \mathbf{y}) f(\mathbf{r} | \mathbf{x}, \mathbf{s}, \mathbf{y}) \\ &= p(\mathbf{x}) p(\mathbf{s}, \mathbf{y} | \mathbf{x}) f(\mathbf{r} | \mathbf{y}) \\ &\propto p(\mathbf{s}, \mathbf{y} | \mathbf{x}) f(\mathbf{r} | \mathbf{y}) \\ &= I(\mathbf{s}, \mathbf{y}, \mathbf{x}) \prod_{n=0}^{N-1} f(r_n | y_n) \\ &= \prod_{n=0}^{N-1} I_n(s_n, y_n, x_n, s_{n+1}) \prod_{n=0}^{N-1} f(r_n | y_n). \end{aligned} \quad (4.12)$$

where the *a priori* distribution $p(\mathbf{x})$ for the transmitted vectors is uniform, N is the length of \mathbf{x} , and $I_n(s_n, y_n, x_n, s_{n+1})$ for $n = 0, 1, \dots, N-1$ are the component indicator functions of the overall indicator function $I(\mathbf{s}, \mathbf{y}, \mathbf{x})$ for valid configurations of \mathbf{s} , \mathbf{y} , and \mathbf{x} . The factor graph for this particular factorization of $p(\mathbf{x}, \mathbf{s}, \mathbf{y} | \mathbf{r})$ is depicted in Fig. 4.6. Since this factor

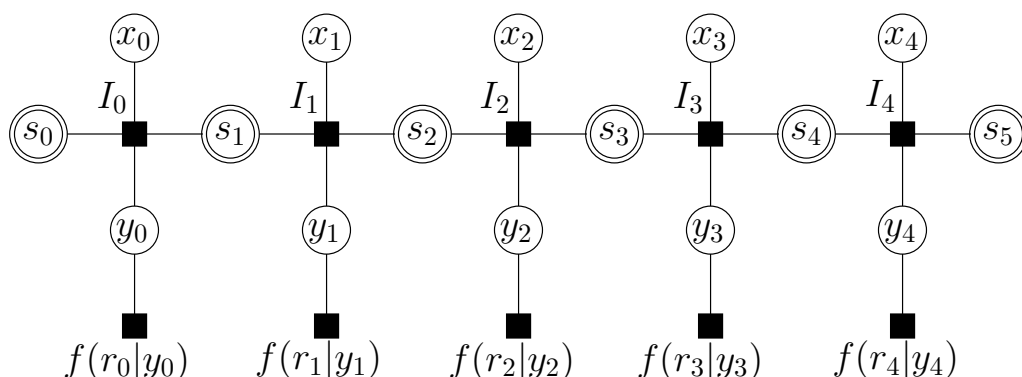


Figure 4.6: A cycle-free factor graph for the global function $p(\mathbf{x}, \mathbf{s}, \mathbf{y} | \mathbf{r})$: the x_n are input variables, the s_n are state variables, the y_n are output variables, and each r_n is the noisy observation of y_n .

graph is cycle-free, the *a posteriori* probabilities for each transmitted symbol x_n , i.e.,

$$g_n(x_n) = p(x_n|\mathbf{r}) \propto \sum_{\sim\{x_n\}} p(\mathbf{x}, \mathbf{s}, \mathbf{y}|\mathbf{r}), \quad (4.13)$$

can be computed using the sum-product algorithm.

4.2.2 Iterative Processing

A trellis is not the only way in which we can represent the global function $p(\mathbf{x}|\mathbf{r})$. Let us now consider an alternative factor graph that does not require hidden state variables. If we factor the global function as

$$\begin{aligned} p(\mathbf{x}|\mathbf{r}) &\propto p(\mathbf{x})f(\mathbf{r}|\mathbf{x}) \\ &\propto f(\mathbf{r}|\mathbf{x}) \\ &= \prod_{n=0}^{N-1} f(r_n|\mathbf{x}) \\ &\propto \prod_{n=0}^{N-1} f(r_n|x_{n-J}, \dots, x_{n+K}) \end{aligned} \quad (4.14)$$

where we have used a uniform *a priori* distribution for $p(\mathbf{x})$, the independence of the r_n 's conditioned on \mathbf{x} , and a channel impulse response h_n with J strictly causal taps and K strictly anticausal taps, then we obtain the factor graph depicted in Fig. 4.7. It is this particular graph that, as we shall demonstrate later in Section 4.4, is closely connected to the iterated-decision equalizer.

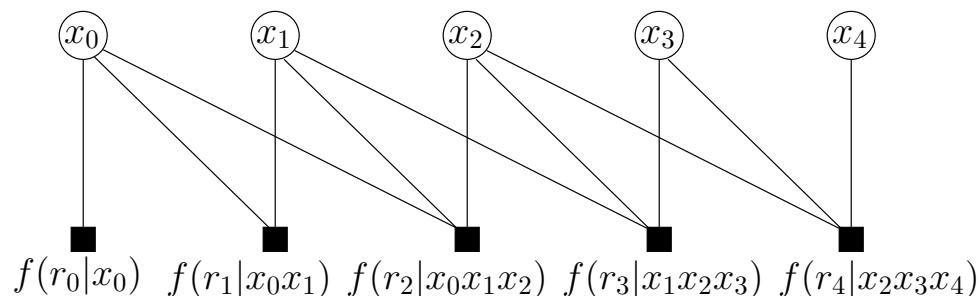


Figure 4.7: A factor graph with cycles for the global function $p(\mathbf{x}|\mathbf{r})$: the x_n are input variables and the r_n are ISI channel output variables. In this example, the channel taps h_n are nonzero for $n = 0, 1, 2$ and the block length is $N = 5$.

Unlike the previous factor graph, this graph is not cycle-free, so the marginal functions for each transmitted symbol x_n cannot be computed exactly by the sum-product algorithm. However, the same message propagation rules can still be applied to a graph with cycles. Since simulations have shown that applying sum-product based decoding algorithms with very long codes can achieve near-capacity results on factor graph with cycles [12, 17], we can reasonably expect that the sum-product algorithm still gives good approximations to the marginal functions.

With the presence of cycles in this factor graph, the sum-product algorithm has no obvious beginning and no natural termination. Furthermore, there are plenty of ways to schedule the order in which messages are passed over all of the edges. An edge can now be traversed in the same direction by updated messages at different times during the duration of the algorithm. Every time a message is passed along an edge, it replaces the message that was previously sent in the same direction along that edge.

A natural message-passing schedule is suggested by the bipartite property of the factor graph. Messages can be passed simultaneously over all edges in one direction, followed by the other direction, and then back again in the first direction, i.e., messages are repeatedly sent back and forth in parallel between the variable and factor nodes. We refer to an iteration as the successive passing of messages in both directions. We choose to begin the algorithm with the convention that the variable nodes all send constant unity functions to the factor nodes, and we terminate the algorithm after a pre-determined number of iterations.

Let us focus on how the approximation to the marginal function $g_2(x_2)$ is updated during each iteration. At the beginning of each iteration, there is an approximation to $g_n(x_n)$ at each variable node. These approximate marginal distributions can be interpreted as *a priori* distributions with notation $p(x_n)$. Messages from the variable nodes other than x_2 are sent to the factor nodes that are neighbors of x_2 . In this case, the factor nodes receiving the messages are $f(r_2|x_0x_1x_2)$, $f(r_3|x_1x_2x_3)$, and $f(r_4|x_2x_3x_4)$. At the node $f(r_2|x_0x_1x_2)$, the product of the function is taken with the incoming approximate functions $p(x_0)$ and $p(x_1)$, and a not-sum operation is performed for x_2 . Thus, a message is sent from the node $f(r_2|x_0x_1x_2)$ to the node x_2 that approximately represents the function

$$\sum_{\sim\{x_2\}} f(r_2|x_0x_1x_2)p(x_0)p(x_1) \approx \sum_{\sim\{x_2\}} f(r_2x_0x_1|x_2) = f(r_2|x_2). \quad (4.15)$$

Similarly, the message sent from nodes $f(r_3|x_1x_2x_3)$ and $f(r_4|x_2x_3x_4)$ to x_2 approximately represent $f(r_3|x_2)$ and $f(r_4|x_2)$ respectively. The product $f(r_2|x_2)f(r_3|x_2)f(r_4|x_2)$, taken at the node x_2 is approximately equal to $f(\mathbf{r}|x_2)$, which is proportional to $g_2(x_2) = p(x_2|\mathbf{r})$.

4.3 The Max-Sum Algorithm

Although the BCJR algorithm was proposed three decades ago [4], it initially received little attention because of numeric problems in representing the probabilities and the computational complexity of the mixed multiplications and additions of these values.

The max-log approximation for the sum-product algorithm addresses these issues. First, because the noise is Gaussian, each not-sum of conditional probability distribution functions is a summation of exponentials over many variables. Since the sum is typically dominated by the exponential with the largest exponent, only the largest term of the summation is kept to reduce the number of computations. Hence, the not-sum operator is replaced by a “not-max” operator. Second, processing of probabilities occurs in the logarithmic domain; the products of probability functions now become the *sum* of log probability functions. The range of values handled in this way becomes more manageable, and also additions are simpler than products.

With these heuristic modifications designed to simplify implementation, the sum-product algorithm on a graph without cycles ends up providing the exact solution to a problem that is slightly different from the minimization of symbol error probability. The modified algorithm is in fact the sum-product algorithm in the “max-sum” semiring, which computes the ML vector [2]. To understand this connection, we briefly discuss the max-sum semiring.

4.3.1 Max-Sum Semiring

A commutative semiring is a set \mathcal{S} , together with two binary operations called “+” and “·”, which satisfy the following axioms:

1. The operation “+” is associative and commutative, and there exists an identity element 0 such that $s + 0 = s$ for all $s \in \mathcal{S}$.
2. The operation “·” is associative and commutative, and there exists an identity element 1 such that $s \cdot 1 = s$ for all $s \in \mathcal{S}$.

3. All $a, b, c \in \mathcal{S}$ satisfy the distributive law

$$(a \cdot b) + (a \cdot c) = a \cdot (b + c). \quad (4.16)$$

Because the range $\mathcal{S} = [0, \infty)$ of the functions in the sum-product algorithm is a commutative semiring with ordinary addition and multiplication, the sum-product algorithm operates in what is called the “sum-product” semiring.

The semiring that we will focus on in the rest of this chapter is the “max-sum” semiring, consisting of the set $\mathcal{S} = [-\infty, \infty)$ with the ordinary addition operator replaced with the “max” operator and the ordinary product operator replaced with summation. The marginal functions of a global function $g(x_1, \dots, x_n)$ are now defined as

$$g_n(x_n) = \max_{\sim\{x_n\}} g(x_1, \dots, x_n) \quad (4.17)$$

where the not-max operator $\max_{\sim\{x_n\}}$ is the max operator over all arguments of the function except for x_n .

For example, suppose g is a function of x_1, x_2, x_3, x_4, x_5 and can be “factored” into the sum

$$g(x_1, x_2, x_3, x_4, x_5) = g_A(x_1, x_3) + g_B(x_2) + g_C(x_2, x_3, x_4) + g_D(x_4, x_5), \quad (4.18)$$

represented by the same graph in Fig. 4.2. Analogous to (4.5) and (4.6), the marginal function $g_1(x_1)$ can be expressed as

$$g_1(x_1) = \max_{\sim\{x_1\}} \left(g_A(x_1, x_3) + \max_{\sim\{x_3\}} \left(g_B(x_2) + g_C(x_2, x_3, x_4) + \max_{\sim\{x_4\}} g_D(x_4, x_5) \right) \right), \quad (4.19)$$

and $g_2(x_2)$ can be expressed as

$$g_2(x_2) = g_B(x_2) + \max_{\sim\{x_2\}} \left(g_C(x_2, x_3, x_4) + \max_{\sim\{x_3\}} g_A(x_1, x_3) + \max_{\sim\{x_4\}} g_D(x_4, x_5) \right). \quad (4.20)$$

The local substitutions made in the factor graph are now depicted in Fig. 4.8.

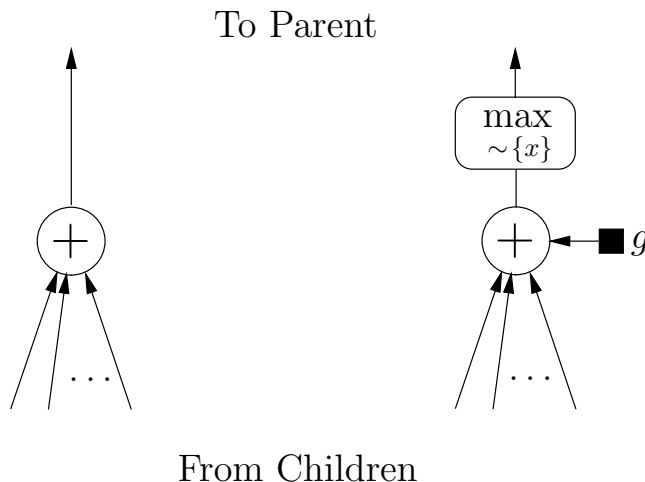


Figure 4.8: Local substitutions that transform a rooted cycle-free graph to an expression tree for a marginal function in the max-sum semiring. On the left is the substitution at a variable node; on the right is the substitution at a factor node g with parent x .

4.3.2 ML Detection

Applying the max-sum algorithm to the ISI channel case, the global function we are interested in is the logarithm of the *a posteriori* joint probability mass function for \mathbf{x} given the fixed observation \mathbf{r} :

$$g(\mathbf{x}) = \log p(\mathbf{x}|\mathbf{r}). \quad (4.21)$$

The corresponding marginal functions have the form

$$g_n(x_n) = \max_{\sim\{x_n\}} \log p(\mathbf{x}|\mathbf{r}) \quad (4.22)$$

which, for each possible value of x_n , is separate maximization over all variables except x_n . It follows that the value of x_n for which $g_n(x_n)$ is largest is the n th component of the maximum-likelihood vector for \mathbf{x} , i.e. the vector for which $p(\mathbf{x}|\mathbf{r})$ is largest. In other words, each symbol of the maximum likelihood sequence, i.e. $\arg \max_{x_n} g_n(x_n)$, can be computed using the max-sum algorithm. So while the sum-product algorithm minimizes the probability of symbol error, the max-sum algorithm minimizes the probability of sequence error.

If the max-sum algorithm is performed in the ISI channel case on the trellis in Fig. 4.9, each symbol of the ML sequence is determined by messages passed in both the forward and backward directions along the trellis. In comparison, the well-known Viterbi algorithm [27],

which also computes the ML sequence, is equivalent to passing messages only in the forward direction of the trellis in Fig. 4.9 but maintaining memory of survivor paths. In general, only after all possible forward messages have been passed can decisions on any of the symbols occur by tracing back along the path corresponding to the ML sequence. Thus, with the Viterbi algorithm, there is also some kind of implicit “backward” mechanism. Nevertheless, because of the difference in the backward mechanism, the max-sum algorithm on the trellis can compute each symbol of the ML sequence individually, while the Viterbi algorithm generally computes all symbols only after message passing in the forward direction is completed.

The max-sum algorithm can also be performed on the graph with cycles depicted in Fig. 4.10, thereby providing an approximate, rather than exact, solution to the ML detection problem. As pointed out in Section 4.2.2 for the corresponding sum-product algorithm, a natural message-passing schedule is suggested by the bipartite property of the factor graph. We begin the algorithm with the convention that the variable nodes all send zero functions to the factor nodes, and we refer to an iteration as the passing of messages in parallel from all variable to factor nodes and then back from all factor to variable nodes. As we discuss in the next section, it is this version of the max-sum algorithm that is strongly related to the iterated-decision detector.

4.4 Relating the Max-Sum Algorithm to Iterated-Decision Detection

In this section, we demonstrate that the iterated-decision equalizer can be interpreted as an approximation to the max-sum algorithm on the factor graph of Fig. 4.10. In doing so, we gain a new perspective on the convergence results of Theorems 3.1 and 3.2. A concentration result was proved in [53] that the decoder performance on random graphs converges to its expected value as the length of the code increases (i.e. the size of the random graph gets large) for a variety of channels, generalizing the result for low-density parity-check codes and binary symmetric channels (BSCs) in [45]. With mode-interleaved precoding, the iterated-decision detector can be viewed as a message-passing algorithm on a large random graph, so the concentration results suggests that performance of the iterated-decision detector asymptotically converges.

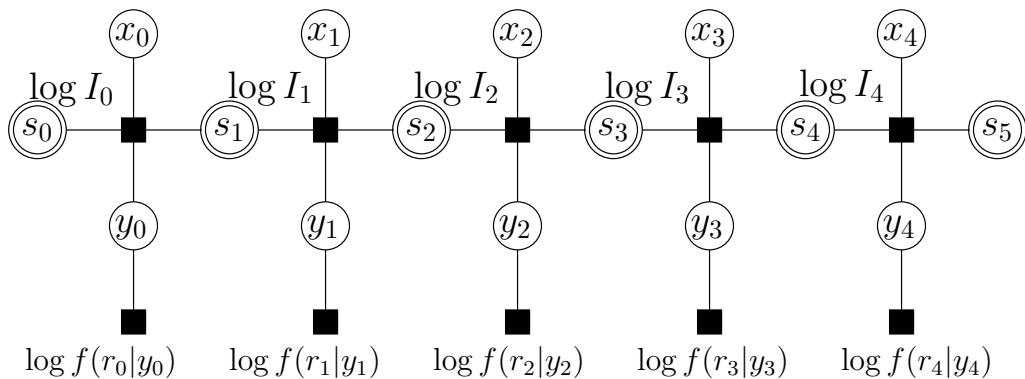


Figure 4.9: A cycle-free factor graph for the global function $\log p(\mathbf{x}, \mathbf{s}, \mathbf{y}|\mathbf{r})$: the x_n are input variables, the s_n are state variables, the y_n are output variables, and each r_n is the noisy observation of y_n .

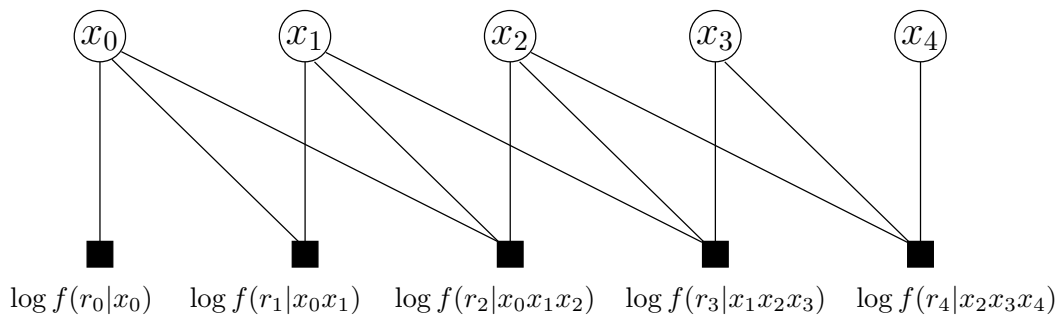


Figure 4.10: A factor graph with cycles for the global function $\log p(\mathbf{x}|\mathbf{r})$: the x_n are input variables and the r_n are ISI channel output variables. In this example, the channel taps h_n are nonzero for $n = 0, 1, 2$ and the block length is $N = 5$.

The first step in making the connection between the max-sum algorithm and the iterated-decision equalizer is to compare the structure of the two algorithms. We decompose each input \tilde{x}_n^l to the slicer of the iterated-decision equalizer into a sum of terms corresponding to information gained about x_n from each received symbol in the sequence r_n . From (3.83), (3.80), and (3.81), the decomposition proceeds as

$$\begin{aligned}
\tilde{x}_n^l &= \sum_k r_k b_{n-k}^l - \sum_j \hat{x}_j^{l-1} d_{n-j}^l \\
&= \sum_k r_k b_{n-k}^l - \sum_{j \neq n} \rho^{l-1} \hat{x}_j^{l-1} \sum_k h_k b_{n-j-k}^l \\
&= \sum_k r_k b_{n-k}^l - \sum_{j \neq n} \rho^{l-1} \hat{x}_j^{l-1} \sum_k h_{k-j} b_{n-k}^l \\
&= \sum_k r_k b_{n-k}^l - \sum_k b_{n-k}^l \sum_{j \neq n} \rho^{l-1} \hat{x}_j^{l-1} h_{k-j} \\
&= \sum_k b_{n-k}^l \left(r_k - \sum_{j \neq n} \rho^{l-1} \hat{x}_j^{l-1} h_{k-j} \right). \tag{4.23}
\end{aligned}$$

The second equality of (4.23) comes from the fact that d_n^l is equal to the convolution of h_n and b_n^l scaled by ρ^{l-1} except with the zeroth tap of the convolution set to zero, the third equality is a change of summation variable, and the fourth equality is a change of summation order. The block diagram for computing \tilde{x}_n^l as in (4.23) is shown in Fig. 4.11 for the case in which $h_n \neq 0$ for $n = 0, 1, 2$ and $b_n \neq 0$ for $n = -2, -1, 0$. The coefficients next to each edge are multiplicative factors. We note that the pattern of the signal flow is similar to the message flow in Fig. 4.12, which shows the flow of messages on the factor graph of Fig. 4.10 used to update the variable node x_n during each iteration of the max-sum algorithm. From the figures, it is clear that both the iterated-decision equalizer and the max-sum algorithm update a variable node using information obtained from *other* variables and received data. By overlaying the block diagrams like the one in Fig. 4.11 for all the \tilde{x}_n^l 's, we obtain the complete iterated-decision equalizer. Similarly, by overlaying the expression trees like the one in Fig. 4.12 for all the x_n 's, we obtain the max-sum algorithm.

In the remainder of this section, we compare the content of the flows on these two very similar structures, and show that even the content is also very similar. For simplicity, we focus our analysis on binary variables, i.e., $x \in \{+1, -1\}$, and defer a discussion of nonbinary variables to the end of this section.

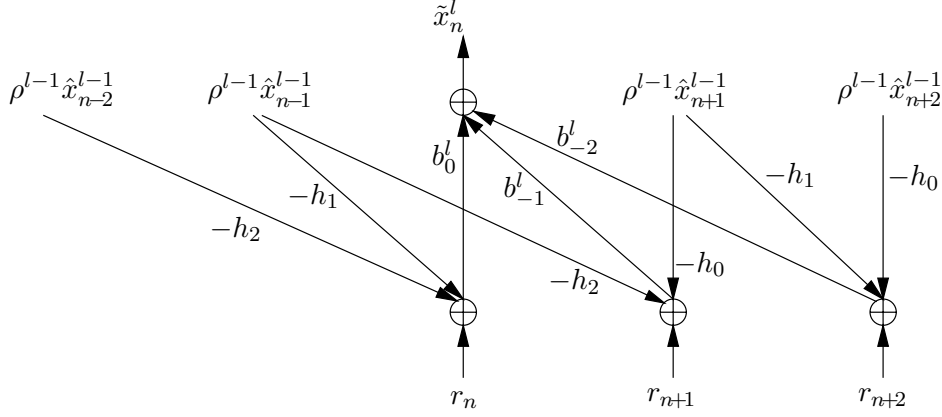


Figure 4.11: A block diagram for the computation of \tilde{x}_n^l in iterated-decision equalization. In this example, $h_n \neq 0$ for $n = 0, 1, 2$ and $b_n \neq 0$ for $n = -2, -1, 0$. The coefficients next to the edges are scaling factors.

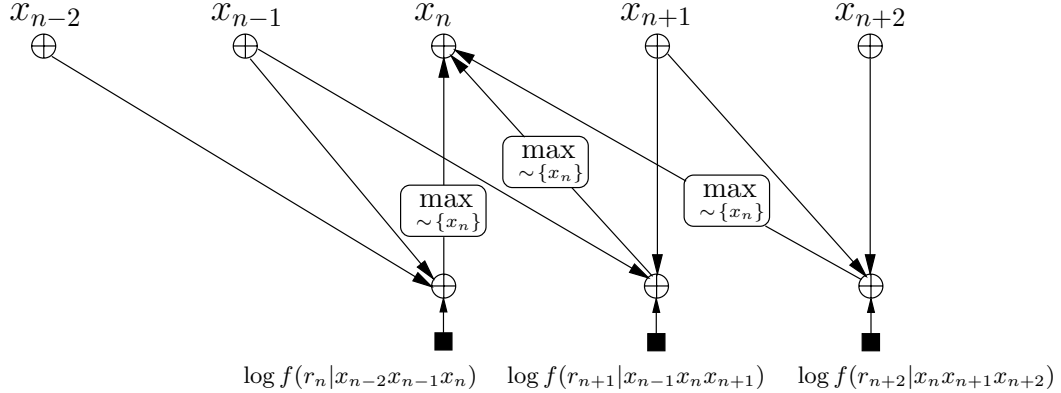


Figure 4.12: The expression tree for the computation of \tilde{x}_n^l during an iteration of the max-sum algorithm on the factor graph in Fig. 4.10. In this example, $h_n \neq 0$ for $n = 0, 1, 2$.

4.4.1 Binary Signalling Over AWGN Channels

We begin by developing a couple of tools associated with binary signalling over AWGN channels that will prove useful in the sequel. The received symbol over such an AWGN channel is

$$\tilde{x} = x + w, \quad (4.24)$$

where $x \in \{-1, +1\}$ and w is independent noise of distribution $\mathcal{N}(0, \sigma^2)$. Given the value of x , the conditional probability distribution function of \tilde{x} is

$$f(\tilde{x}|x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-(\tilde{x}-x)^2/2\sigma^2}, \quad (4.25)$$

which can be used to compute log-likelihood ratios (L -values) and Bayes' least-squares estimators.

Log-Likelihood Ratios

The log-likelihood ratio (L -value) is commonly used to describe the probability mass function of a binary random variable such as x given an observation. The L -value for x given \tilde{x} is calculated as

$$L(x|\tilde{x}) = \log \frac{\Pr(x = +1|\tilde{x})}{\Pr(x = -1|\tilde{x})}. \quad (4.26)$$

Using Bayes' rule, equal *a priori* probabilities, and (4.25),

$$\begin{aligned} L(x|\tilde{x}) &= \log \frac{f(\tilde{x}|x = +1) \Pr(x = +1)}{f(\tilde{x}|x = -1) \Pr(x = -1)} \\ &= \log \frac{f(\tilde{x}|x = +1)}{f(\tilde{x}|x = -1)} \\ &= \log \frac{1}{\sqrt{2\pi}\sigma} - \frac{(\tilde{x} - 1)^2}{2\sigma^2} - \log \frac{1}{\sqrt{2\pi}\sigma} + \frac{(\tilde{x} + 1)^2}{2\sigma^2} \\ &= \frac{2\tilde{x}}{2\sigma^2} + \frac{2\tilde{x}}{2\sigma^2} \\ &= \frac{2\tilde{x}}{\sigma^2}. \end{aligned} \quad (4.27)$$

Thus $L(x|\tilde{x})$ is directly proportional to \tilde{x} , scaled by the variance of the additive noise. The usefulness of this fact is not limited only to AWGN channels. In situations where an information bit is affected by approximately Gaussian additive interference and noise, the sum of the bit, interference, and noise can be interpreted as the L -value of the bit scaled by the total variance of the interference and noise. We can also rearrange (4.26) to get the expressions

$$\Pr(x = +1|\tilde{x}) = \frac{e^{L(x|\tilde{x})}}{1 + e^{L(x|\tilde{x})}} \quad (4.28)$$

$$\Pr(x = -1|\tilde{x}) = \frac{1}{1 + e^{L(x|\tilde{x})}} \quad (4.29)$$

and, in light of (4.27),

$$\Pr(x = +1|\tilde{x}) = \frac{e^{2\tilde{x}/\sigma^2}}{1 + e^{2\tilde{x}/\sigma^2}} \quad (4.30)$$

$$\Pr(x = -1|\tilde{x}) = \frac{1}{1 + e^{2\tilde{x}/\sigma^2}}. \quad (4.31)$$

Bayes' Least-Squares Estimation

The Bayes' least-squares estimate \hat{x}_{BLS} minimizes the mean-square error between itself and x , where \hat{x}_{BLS} is a function of \tilde{x} , i.e.,

$$\hat{x}_{\text{BLS}}(\tilde{x}) = \arg \min_a \sum_x (x - a)^2 \Pr(x|\tilde{x}). \quad (4.32)$$

Since this estimator minimizes the mean-square estimation error, it is alternatively referred to as an MMSE estimator. The minimization of (4.32) can be performed by differentiating with respect to a and setting the result to zero. In doing so, we obtain

$$\hat{x}_{\text{BLS}}(\tilde{x}) = E[x|\tilde{x}], \quad (4.33)$$

which is the mean of the posterior density $f(x|\tilde{x})$.

In our case, we evaluate $E[x|\tilde{x}]$ using (4.30) and (4.31) to get

$$\begin{aligned} E[x|\tilde{x}] &= \Pr(x = +1|\tilde{x}) - \Pr(x = -1|\tilde{x}) \\ &= \frac{e^{2\tilde{x}/\sigma^2}}{1 + e^{2\tilde{x}/\sigma^2}} - \frac{1}{1 + e^{2\tilde{x}/\sigma^2}} \\ &= \tanh\left(\frac{\tilde{x}}{\sigma^2}\right) \\ &= \left| \tanh\left(\frac{\tilde{x}}{\sigma^2}\right) \right| \text{sgn}(\tilde{x}), \end{aligned} \quad (4.34)$$

where the last equality is because the hyperbolic tangent is an odd-symmetric function that is positive when its argument is positive. An alternate expression for $E[x|\tilde{x}]$ is given by

$$\begin{aligned} E[x|\tilde{x}] &= \Pr(x = \text{sgn}(\tilde{x})|\tilde{x}) \cdot \text{sgn}(\tilde{x}) - \Pr(x \neq \text{sgn}(\tilde{x})|\tilde{x}) \cdot \text{sgn}(\tilde{x}) \\ &= [1 - 2 \Pr(x \neq \text{sgn}(\tilde{x})|\tilde{x})] \text{sgn}(\tilde{x}) \\ &= \rho(\tilde{x}) \text{sgn}(\tilde{x}), \end{aligned} \quad (4.35)$$

where

$$\rho(\tilde{x}) \triangleq |E[x|\tilde{x}]| = 1 - 2 \Pr(x \neq \text{sgn}(\tilde{x})|\tilde{x}) \geq 0. \quad (4.36)$$

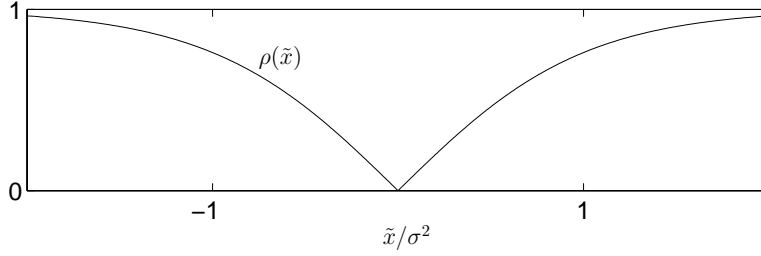


Figure 4.13: The function $\rho(\tilde{x}) = |\tanh(\tilde{x}/\sigma^2)|$.

Comparing (4.35) to (4.34) and incorporating (4.27), we see that

$$\rho(\tilde{x}) = \left| \tanh\left(\frac{\tilde{x}}{\sigma^2}\right) \right| = \left| \tanh\left(\frac{L(x|\tilde{x})}{2}\right) \right|. \quad (4.37)$$

The quantity $\rho(\tilde{x})$ can be interpreted as a measure of reliability of the observation \tilde{x} . As illustrated in Fig. 4.13, in this context a value of \tilde{x} close to zero is interpreted as unreliable, and a value of \tilde{x} far away from zero is very reliable.

4.4.2 Low-Bandwidth Message Passing from Variable to Factor Nodes

With the concepts of L -values and Bayes' least-squares estimators established, we return back to the max-sum algorithm. In the binary case, a different *a priori* probability mass function is associated with each x_n at the beginning of the l th iteration (Fig. 4.14(a)). For the time being, let us assume that the *a priori* probability mass function for x_n was derived during the previous iteration from a noisy observation \tilde{x}_n^{l-1} , defined in (4.24). Thus each symbol x_n can be associated with a Bayes' least squares estimate $E[x_n|\tilde{x}_n^{l-1}]$ which can be considered as an alternative form of likelihood information because of (4.27). As determined in (4.35), $E[x_n|\tilde{x}_n^{l-1}]$ consists of a reliability $\rho(\tilde{x}_n^{l-1})$ and the most significant bit $\hat{x}_n^{l-1} \triangleq \text{sgn}(\tilde{x}_n^{l-1})$. As shown in Fig. 4.14(b), we propose to approximate the *a priori* probability mass function for x_n as

$$p(x_n) \approx \delta(x_n - \rho(\tilde{x}_n^{l-1})\hat{x}_n^{l-1}). \quad (4.38)$$

In essence, we are making a scalar approximation to a binary probability mass function.

Furthermore, rather than keep a different reliability for each symbol x_n , we propose to combine all reliabilities into a single parameter. We next define ρ^{l-1} as the expectation of

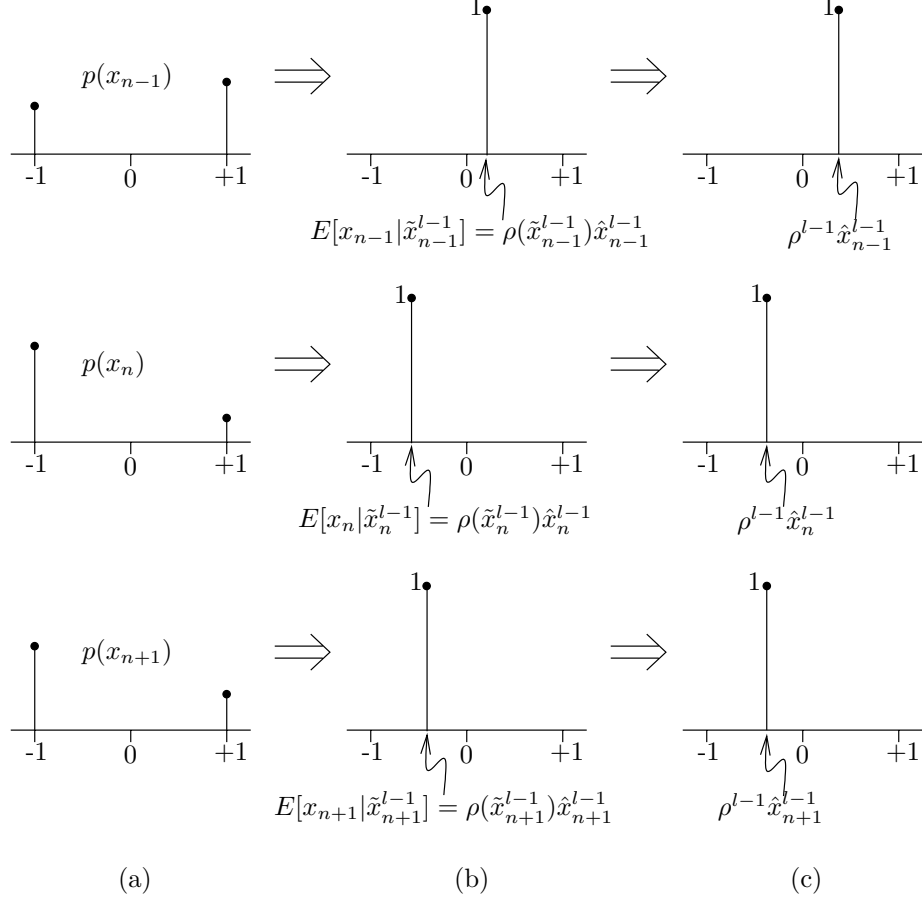


Figure 4.14: (a) A different *a priori* probability mass function is associated with each binary variable at the beginning of each iteration. (b) The probability mass functions are approximated by impulse functions located at the associated Bayes' least squares estimates. (c) The expected absolute value of the Bayes' least squares estimate replaces the magnitude of the Bayes' least squares estimates for each binary variable.

$\rho(\tilde{x}_n^{l-1})$ with respect to \tilde{x}_n^{l-1} , so we have from (4.36)

$$\rho^{l-1} \triangleq E[\rho(\tilde{x}_n^{l-1})] = E[|E[x_n | \tilde{x}_n^{l-1}]|] = 1 - 2E[\Pr(x_n \neq \hat{x}_n^{l-1})], \quad (4.39)$$

which can be alternatively verified by integrating (4.37) with respect to the *a priori* probability distribution function

$$f(\tilde{x}) = \frac{1}{2} \cdot \frac{1}{\sqrt{2\pi}\sigma} e^{-(\tilde{x}-1)^2/2\sigma^2} + \frac{1}{2} \cdot \frac{1}{\sqrt{2\pi}\sigma} e^{-(\tilde{x}+1)^2/2\sigma^2}. \quad (4.40)$$

Since \tilde{x}_n^{l-1} is directly proportional to $L(x_n | \tilde{x}_n^{l-1})$, the single parameter ρ^{l-1} conveniently summarizes information about all possible L -values at a given SNR. The quantity $E[\Pr(x_n \neq$

\hat{x}_n^{l-1}] in (4.39) can be interpreted as the average probability of bit error, which is consistent with the expression for ρ^{l-1} in the context of the iterated-decision detector in (3.32).

We call this simplification “low-bandwidth message passing” because the messages sent from variable to function nodes describe functions parametrized only by a single parameter ρ^{l-1} that summarizes reliability, and the most significant bit of each \hat{x}_n^{l-1} . We are essentially approximating the *a priori* distributions as impulse functions at $\rho^{l-1}\hat{x}_n^{l-1}$ (Fig. 4.14(c)); i.e.,

$$p(x_n) \approx \delta(x_n - \rho^{l-1}\hat{x}_n^{l-1}). \quad (4.41)$$

4.4.3 Message Passing from Factor to Variable Nodes

In the expression tree of Fig. 4.12, which describes how the max-sum algorithm updates information about x_n , the logarithms of the approximate *a priori* distributions arriving at a factor node are summed with the logarithm of the associated factor, and a not-max operation is performed. For example, the message sent to the variable node x_n from the function node $\log f(r_n|x_{n-2}x_{n-1}x_n)$ describes the function

$$\begin{aligned} & \max_{\sim\{x_n\}} \left(\log f(r_n|x_{n-2}x_{n-1}x_n) + \log \delta(x_{n-2} - \rho^{l-1}\hat{x}_{n-2}^{l-1}) + \log \delta(x_{n-1} - \rho^{l-1}\hat{x}_{n-1}^{l-1}) \right) \\ &= \max_{\sim\{x_n\}} \left(\log f(r_n|x_{n-2}x_{n-1}x_n) \delta(x_{n-2} - \rho^{l-1}\hat{x}_{n-2}^{l-1}) \delta(x_{n-1} - \rho^{l-1}\hat{x}_{n-1}^{l-1}) \right) \\ &= \max_{\sim\{x_n\}} \log f(r_n|x_{n-2} = \rho^{l-1}\hat{x}_{n-2}^{l-1}, x_{n-1} = \rho^{l-1}\hat{x}_{n-1}^{l-1}, x_n) \\ &= \log f(r_n|x_{n-2} = \rho^{l-1}\hat{x}_{n-2}^{l-1}, x_{n-1} = \rho^{l-1}\hat{x}_{n-1}^{l-1}, x_n). \end{aligned} \quad (4.42)$$

This function passed from the node $\log f(r_n|x_{n-2}x_{n-1}x_n)$ to node x_n is a function of the binary variable x_n and, as such, can be parametrized by a single value. One particular parametrization is the *L*-value

$$\log \frac{f(r_n|x_{n-2} = \rho^{l-1}\hat{x}_{n-2}^{l-1}, x_{n-1} = \rho^{l-1}\hat{x}_{n-1}^{l-1}, x_n = +1)}{f(r_n|x_{n-2} = \rho^{l-1}\hat{x}_{n-2}^{l-1}, x_{n-1} = \rho^{l-1}\hat{x}_{n-1}^{l-1}, x_n = -1)}. \quad (4.43)$$

Since

$$f(r_n|x_{n-2}x_{n-1}x_n) = \frac{1}{\sqrt{2\pi N_0}} e^{-(r_n - h_2x_{n-2} - h_1x_{n-1} - h_0x_n)^2/2N_0}, \quad (4.44)$$

the L -value can be expressed as

$$2h_{n-2}(r_n - h_n\rho^{l-1}\hat{x}_{n-2}^{l-1} - h_{n-1}\rho^{l-1}\hat{x}_{n-1}^{l-1})/N_0 \quad (4.45)$$

Similarly, the function passed from the node $f(r_{n+1}|x_{n-1}x_nx_{n+1})$ to node x_n can be parametrized as

$$2h_{n-1}(r_{n+1} - h_n\rho^{l-1}\hat{x}_{n-1}^{l-1} - h_{n-2}\rho^{l-1}\hat{x}_{n+1}^{l-1})/N_0 \quad (4.46)$$

and the function from $f(r_{n+2}|x_nx_{n+1}x_{n+2})$ to node x_n can be parametrized as

$$2h_n(r_{n+2} - h_{n-1}\rho^{l-1}\hat{x}_{n+1}^{l-1} - h_{n-2}\rho^{l-1}\hat{x}_{n+2}^{l-1})/N_0. \quad (4.47)$$

4.4.4 Combining Messages at Variable Nodes

At the variable node x_n , the sum of the functions the L -values (4.45), (4.46), and (4.47) represent also can be parametrized by the literal sum of the L -values themselves:

$$\frac{2}{N_0} \sum_k h_{k-n} \left(r_k - \sum_{j \neq n} h_{k-j} \rho^{l-1} \hat{x}_j^{l-1} \right). \quad (4.48)$$

Let us develop some intuition as to the interpretation of this L -value. Suppose the L -value originated from a noisy observation of a binary variable in AWGN, as described by (4.24), (4.25), and (4.26). What values of the observation \tilde{x}_n^l and the noise variance σ^2 would be consistent with the L -value in (4.48)?

To answer this question, let us return to the iterated-decision equalizer. As shown in (4.23), the input to the slicer \tilde{x}_n^l can be thought of as the sequence $r_n - \sum_{j \neq n} \rho^{l-1} \hat{x}_j^{l-1} h_{n-j}$ processed by the linear MMSE filter b_n^l with frequency response

$$B^l(\omega) = \frac{(1 - (\rho^{l-1})^2)H^*(\omega)}{(1 - (\rho^{l-1})^2)|H(\omega)|^2 + N_0}. \quad (4.49)$$

From (4.23), the slicer input can be expressed as

$$\begin{aligned}
\tilde{x}_n^l &= \sum_k \left(\frac{1}{2\pi} \int_{-\pi}^{\pi} B^l(\omega) e^{j\omega(n-k)} d\omega \right) \left(r_k - \sum_{j \neq n} \rho^{l-1} \hat{x}_j^{l-1} h_{k-j} \right) \\
&= \sum_k \left(\frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{(1 - (\rho^{l-1})^2) H^*(\omega)}{(1 - (\rho^{l-1})^2) |H(\omega)|^2 + N_0} e^{j\omega(n-k)} d\omega \right) \left(r_k - \sum_{j \neq n} \rho^{l-1} \hat{x}_j^{l-1} h_{k-j} \right).
\end{aligned} \tag{4.50}$$

The corresponding mean-square error between \tilde{x}_n^l and x_n is

$$\sigma^2 = \frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{(1 - (\rho^{l-1})^2) N_0}{(1 - (\rho^{l-1})^2) |H(\omega)|^2 + N_0} d\omega. \tag{4.51}$$

If the mean-square error is approximated as independent and Gaussian, then from (4.27) the L -value corresponding to the slicer input \tilde{x}_n^l is

$$\begin{aligned}
L(x_n | \tilde{x}_n^l) &= \frac{2 \sum_k \left(\frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{(1 - (\rho^{l-1})^2) H^*(\omega)}{(1 - (\rho^{l-1})^2) |H(\omega)|^2 + N_0} e^{j\omega(n-k)} d\omega \right) \left(r_k - \sum_{j \neq n} \rho^{l-1} \hat{x}_j^{l-1} h_{k-j} \right)}{\frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{(1 - (\rho^{l-1})^2) N_0}{(1 - (\rho^{l-1})^2) |H(\omega)|^2 + N_0} d\omega} \\
&\approx \frac{2 \sum_k \left(\frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{(1 - (\rho^{l-1})^2)}{(1 - (\rho^{l-1})^2) |H(\omega)|^2 + N_0} d\omega \right) \left(\frac{1}{2\pi} \int_{-\pi}^{\pi} H^*(\omega) e^{j\omega(n-k)} d\omega \right) \left(r_k - \sum_{j \neq n} \rho^{l-1} \hat{x}_j^{l-1} h_{k-j} \right)}{N_0 \left(\frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{(1 - (\rho^{l-1})^2)}{(1 - (\rho^{l-1})^2) |H(\omega)|^2 + N_0} d\omega \right)} \\
&= \frac{2}{N_0} \sum_k h_{k-n} \left(r_k - \sum_{j \neq n} \rho^{l-1} \hat{x}_j^{l-1} h_{k-j} \right).
\end{aligned} \tag{4.52}$$

Comparing to (4.48), we see that the slicer input of the iterated-decision equalizer leads approximately to the same L -value as the max-sum algorithm, thereby providing an intuitive explanation as to why the iterated-decision equalizer approximates the ML solution with mode-interleaved precoding. Without mode-interleaved precoding, the sequence \tilde{x}_n^l is not guaranteed to resemble the output of an AWGN channel, and (4.52) is a less reliable result.

This L -value for x_n , common to both the iterated-decision equalizer and the max-sum algorithm, can be used to determine a new *a priori* distribution for x_n via (4.30) and (4.31). A new iteration begins and, regarding the assumption in Section 4.4.2, it is clear now that the *a priori* distributions can indeed be interpreted as being derived from noisy observations in light of (4.50) and (4.51).

4.4.5 Nonbinary Variables

In this section, we discuss the generalization of the preceding analysis from binary variables to nonbinary variables. We make the mild assumptions that the equiprobable signal constellation has zero mean and symmetry about zero.

We begin with a generalization of the L -value given in (4.26). We define the L -value for a variable $x \in \mathcal{X}$ given \tilde{x} as

$$L(x|\tilde{x}) = \sum_{\alpha \in \mathcal{X}} \alpha \log \Pr(x = \alpha|\tilde{x}). \quad (4.53)$$

For a symbol x received in independent $\mathcal{N}(0, \sigma^2)$ noise, the L -value is

$$\begin{aligned} L(x|\tilde{x}) &= \sum_{\alpha \in \mathcal{X}} \alpha \log \frac{f(\tilde{x}|x = \alpha) \Pr(x = \alpha)}{f(\tilde{x})} \\ &= \sum_{\alpha \in \mathcal{X}} \alpha \log f(\tilde{x}|x = \alpha) + \frac{1}{|\mathcal{X}|} \sum_{\alpha \in \mathcal{X}} \alpha - f(\tilde{x}) \sum_{\alpha \in \mathcal{X}} \alpha \\ &= \sum_{\alpha \in \mathcal{X}} \alpha \log f(\tilde{x}|x = \alpha) \\ &= \log \frac{1}{\sqrt{2\pi}\sigma} \sum_{\alpha \in \mathcal{X}} \alpha - \sum_{\alpha \in \mathcal{X}} \alpha \frac{(\tilde{x} - \alpha)^2}{2\sigma^2} \\ &= -\frac{\tilde{x}^2}{2\sigma^2} \sum_{\alpha \in \mathcal{X}} \alpha + \frac{\tilde{x}}{\sigma^2} \sum_{\alpha \in \mathcal{X}} \alpha^2 - \frac{1}{2\sigma^2} \sum_{\alpha \in \mathcal{X}} \alpha^3 \\ &= \frac{\tilde{x}}{\sigma^2} \sum_{\alpha \in \mathcal{X}} \alpha^2 \end{aligned} \quad (4.54)$$

where we have used the zero mean of the constellation \mathcal{X} in the third, fifth, and sixth equalities, and the symmetry of \mathcal{X} in the sixth equality. Thus, given the noise variance σ^2 , there is a one-to-one mapping between \tilde{x} and $L(x|\tilde{x})$, so $L(x|\tilde{x})$ can parametrize the distribution of x .

We can also generalize the notion of the symbol reliability $\rho(\tilde{x})$. For an observation \tilde{x} of x in independent $\mathcal{N}(0, \sigma^2)$, we can use the Bayes' least squares estimate

$$E[x|\tilde{x}] = \sum_{\alpha \in \mathcal{X}} \alpha p(x = \alpha|\tilde{x}) \quad (4.55)$$

to define $\rho(\tilde{x})$ as

$$\rho(\tilde{x}) \triangleq \frac{E[x|\tilde{x}]\hat{x}}{\mathcal{E}_s} \quad (4.56)$$

with \hat{x} as the minimum distance symbol decision for \tilde{x} .

With these generalized definitions for the L -values and symbol reliabilities, the analysis of Sections 4.4.2–4.4.4 still holds. In particular, the parameter ρ^{l-1} that summarizes the probability mass functions for each symbol x_n is defined as the expectation of $\rho(\tilde{x}_n^{l-1})$ with respect to \tilde{x}_n^{l-1} ; i.e.,

$$\rho^{l-1} \triangleq E[\rho(\tilde{x}_n^{l-1})] \quad (4.57)$$

where the expectation is with respect to the distribution

$$f(\tilde{x}) = \frac{1}{M} \sum_{x \in \mathcal{X}} \frac{1}{\sqrt{2\pi}\sigma} e^{-(\tilde{x}-x)^2/2\sigma^2}. \quad (4.58)$$

This definition is consistent with the expression for ρ^{l-1} in the context of the iterated-decision detector in (3.32), since

$$\rho^{l-1} = E[\rho(\tilde{x}_n^{l-1})] = E\left[\frac{E[x_n|\tilde{x}_n^{l-1}]\hat{x}_n^{l-1}}{\mathcal{E}_s}\right] = E\left[\frac{E[x_n\hat{x}_n^{l-1}|\tilde{x}_n^{l-1}]}{\mathcal{E}_s}\right] = \frac{E[x_n\hat{x}_n^{l-1}]}{\mathcal{E}_s}. \quad (4.59)$$

Since \tilde{x}_n^{l-1} is directly proportional to $L(x_n|\tilde{x}_n^{l-1})$, the single parameter ρ^{l-1} conveniently summarizes information about all possible L -values at a given SNR. The quantities $E[x|\tilde{x}]$, $\rho(\tilde{x})$, and $f(\tilde{x})$ as a function of \tilde{x} are shown in Fig. 4.15 for $\mathcal{X} = \{-3, -1, +1, +3\}$.

4.5 Summary

In this chapter, we have compared the iterated-decision equalizer with the max-sum algorithm on a particular factor graph with cycles. We have demonstrated that both algorithms compute similar L -values for the symbols after each iteration provided that mode-interleaved precoding is used. Thus, since the max-sum algorithm approximates ML detection, so does the iterated-decision equalizer. Let us conclude with some comments on the general detection case when \mathbf{H} is not Toeplitz.

For the general detection case in which \mathbf{H} is an arbitrary $Q \times N$ matrix, trellis processing as described in Section 4.3.2 is generally not possible, but iterative processing is. The factor graph associated with iterative processing for the general detection case is based on the

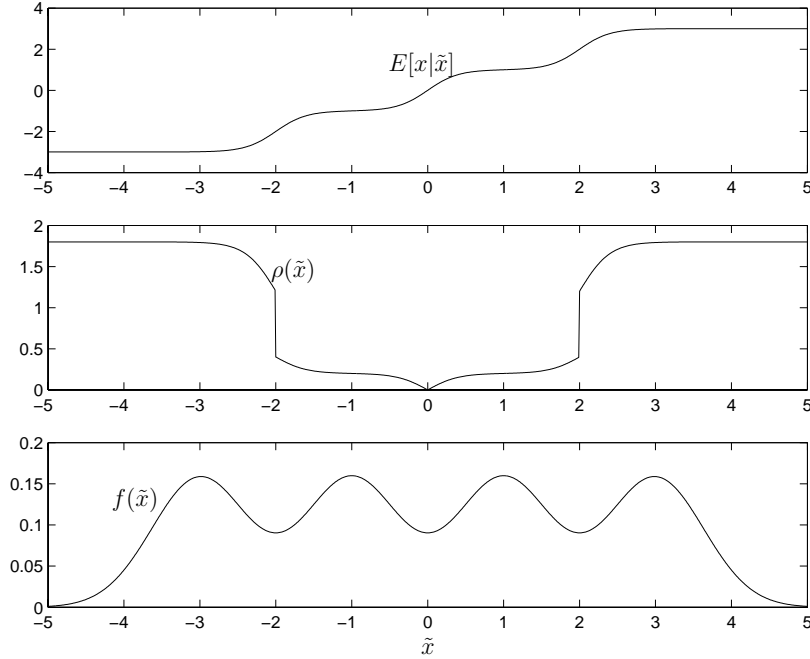


Figure 4.15: The quantities $E[x|\tilde{x}]$, $\rho(\tilde{x})$, and $f(\tilde{x})$ as a function of \tilde{x} , for 4-PAM where $\mathcal{X} = \{-3, -1, +1, +3\}$.

global function $\log p(\mathbf{x}|\mathbf{r})$, where

$$\begin{aligned}
 p(\mathbf{x}|\mathbf{r}) &\propto p(\mathbf{x})f(\mathbf{r}|\mathbf{x}) \\
 &\propto f(\mathbf{r}|\mathbf{x}) \\
 &= \prod_{n=0}^{Q-1} f(r_n|\mathbf{x})
 \end{aligned} \tag{4.60}$$

because of the independence of the r_n 's conditioned on \mathbf{x} . Thus, the factor graph has N variable nodes, Q factor nodes, and edges connecting every variable node to every factor node. The analysis in Section 4.4 is extendable in a straightforward manner to the general detection case, so we can conclude that the iterated-decision detector approximates the ML solution provided by the max-sum algorithm.

Chapter 5

Iterative Detection And Decoding Algorithms

In Section 1.2.3, we discussed the broader objective of combining detection and channel coding to approach the mutual information of an interference channel without having channel state information at the transmitter. In this chapter, we discuss how iterated-decision detection can be integrated into such a communication system.

The optimal receiver in a probability of error sense uses an exact maximum-likelihood (ML) or maximum *a posteriori* (MAP) algorithm that treats the encoder and the interference channel as a single product code and performs joint detection and decoding, depicted in Fig. 1.7. However, as discussed in Section 1.2.3, the complexity of such a receiver is usually determined by the product of the complexities of the optimal detector for the corresponding uncoded system and the optimal decoder for the corresponding AWGN channel, thus rendering such a system impractical.

The iterated-decision detector can readily be used at the receiver in cascade with a decoder as depicted in Fig. 1.8. As in the uncoded case, the iterated-decision detector progressively removes interference from the coded data until convergence occurs. At this point, the decoder can process either the “soft” decisions $\tilde{\mathbf{x}}^l$ or the “hard” decisions $\hat{\mathbf{x}}^l$ to give an estimate of the original uncoded data. However, using the iterated-decision detector in the receiver design of Fig. 1.8 does not give the best level of performance for this order of complexity.

Instead of just one pass through the detector and decoder, the two blocks can send information back and forth between them as shown in Fig. 5.1. With its excellent performance and its ability to take into account *a priori* information, the iterated-decision detector is a

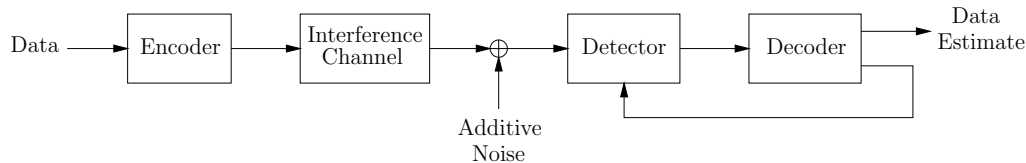


Figure 5.1: Iterative processing using separate detection and decoding blocks.

natural candidate for the detector block. With this iterative receiver structure, complexity is still on the same order as before because detection and decoding are still dealt with separately, yet optimal joint detection and decoding can be approximated.

In this chapter, we introduce iterative detection and decoder schemes for interference channels inspired by the message-passing interpretation of the iterated-decision detector in Chapter 4. In Section 5.1, we discuss the use of binary codes with iterated-decision detection for power-limited interference channels, which support low information rates. We view the receiver as implementing message passing algorithms on a factor graph partitioned into separate coding and interference subgraphs, where message passing on the interference subgraph is efficiently implemented by the iterated-decision detector. Tools to analyze the performance of such systems are introduced and, as with capacity-approaching codes over AWGN channels, we find that capacity-approaching codes over interference channels also exhibit the phenomenon that an arbitrarily small bit error probability can be achieved if the noise level is smaller than a certain threshold. However, binary codes do not suffice to approach the high information rates supported by bandwidth-limited interference channels. Typically, information bits need to be mapped to symbols in a larger constellation, which are then transmitted over the channel. In Section 5.2 we investigate the use of multilevel coding [68] with iterated-decision detection. We observe that multistage decoding for multilevel codes is very similar to iterated-decision detection, and thus propose a combined iterative detector and decoder. Although this combined receiver is not quite optimal, it inspires a class of iterative detection/decoding algorithms based on the iterated-decision detector that can potentially enable reliable transmission at rates close to the mutual information of the channel. Since uncoded square QAM constellations as special cases of lattice codes, the iterated-decision detector in Chapter 3 for uncoded systems is generalized in a straightforward way to lattice-coded systems in Section 5.3.

We emphasize that the schemes presented in this chapter are not comprehensive; rather,

they give a flavor of the kinds of low-complexity receivers that are possible in communication systems with channel coding and channel interference. As in Chapter 4, we focus on the ISI channel case with the understanding that these results apply more generally.

5.1 Binary Codes

In Chapter 4, it was demonstrated that the iterated-decision detector can be interpreted as an efficient message-passing algorithm on a graph. We can extend this concept further by creating a graph that incorporates the effects of both interference and coding.

As an example, let us consider a system in which low-density parity-check (LDPC) codewords are sent over an ISI channel. The LDPC code \mathcal{C} with parity-check matrix

$$\begin{bmatrix} 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix} \quad (5.1)$$

is a regular binary (2,3)-LDPC code, meaning that each column of the matrix has 2 nonzero entries and each row has 3 nonzero entries. If the codewords are sent over an ISI channel with impulse response h_n that is nonzero for $n = 0, 1, 2$, then the *a posteriori* joint probability mass function for \mathbf{x} given the fixed observation \mathbf{r} is

$$\begin{aligned} p(\mathbf{x}|\mathbf{r}) &\propto p(\mathbf{x})f(\mathbf{r}|\mathbf{x}) \\ &\propto [\mathbf{x} \in \mathcal{C}] \prod_{n=0}^5 f(r_n|\mathbf{x}) \\ &\propto \prod_{k=0}^3 I_k(\mathbf{x}) \prod_{n=0}^5 f(r_n|x_{n-2}, \dots, x_n) \end{aligned} \quad (5.2)$$

where

$$I_0(\mathbf{x}) = [x_1 \oplus x_2 \oplus x_4 = 0] \quad (5.3)$$

$$I_1(\mathbf{x}) = [x_0 \oplus x_2 \oplus x_3 = 0] \quad (5.4)$$

$$I_2(\mathbf{x}) = [x_0 \oplus x_4 \oplus x_5 = 0] \quad (5.5)$$

$$I_3(\mathbf{x}) = [x_1 \oplus x_3 \oplus x_5 = 0] \quad (5.6)$$

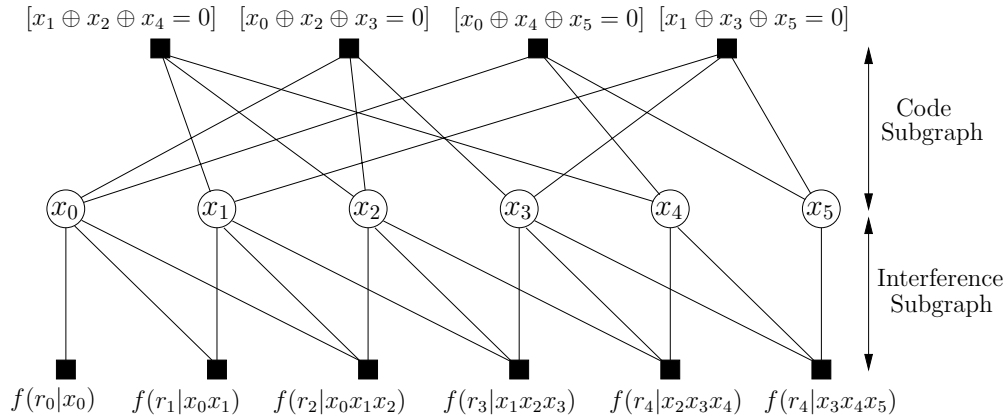


Figure 5.2: A factor graph representing the global function $p(\mathbf{x}|\mathbf{r})$ for an LDPC-coded system with interference.

are indicator functions corresponding to the local parity constraints. The factor graph representing $p(\mathbf{x}|\mathbf{r})$ is depicted in Fig. 5.2, consisting of two bipartite graphs superimposed on one another. In fact, the graph is tripartite with two of the three vertex sets having no edges between them.

As discussed in Section 4.2, since the factor graph representing $p(\mathbf{x}|\mathbf{r})$ has cycles the class of message-passing algorithms with different message-passing schedules on the graph approximate the performance of the optimal receiver in Fig. 1.7. In order to keep the receiver at a manageable complexity, however, we restrict our attention to message-passing schedules that rely on the fact that the overall graph consists of an interference subgraph and a coding subgraph.

The detector/decoder in Fig. 1.8 falls within this restricted class of algorithms. The detector block in Fig. 1.8 passes messages back and forth in parallel on the bipartite interference subgraph, which is approximately but efficiently implementable using the iterated-decision detector. The converged L -values for \mathbf{x} are then passed onto the decoder block, which passes messages back and forth in parallel on the bipartite LDPC subgraph until the L -values for \mathbf{x} converge yet again. Hard decisions are then made on the converged L -values.

However, as depicted in Fig. 5.1, hard decisions need not be made immediately. Rather, the converged L -values for \mathbf{x} can be sent back to the iterative detector for further rounds of processing. Now, supplied with *a priori* information from the decoder block in addition to the original data received over the channel, the detector can output more reliable L -values to the decoder. Thus, a hierarchy of iterations emerges—iterations within the detector

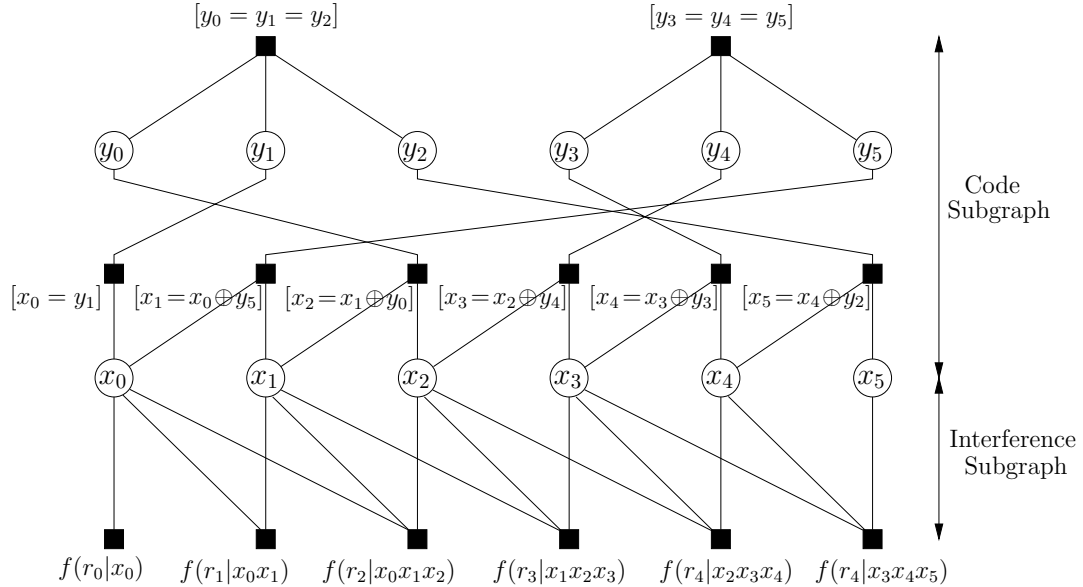


Figure 5.3: A factor graph representing the global function $p(\mathbf{x}|\mathbf{r})$ for an RA-coded system with interference.

block followed by iterations within the decoder block together form a single iteration of the overall receiver structure. Note that convergence of the L -values for \mathbf{x} within either the detector block or the decoder block is not a necessary condition for passing the L -values to the next block.

The concepts in our LDPC example apply more generally. Essentially, the effects of the interference and the code can be combined into a single factor graph. The detector block corresponds to message passing on the interference subgraph, and the decoding block corresponds to message passing on the code subgraph. Iterations within the detector block followed by iterations within the decoding block form an iteration in the larger system. Figure 5.3 shows the factor graph for an interference channel with repeat-accumulate (RA) codes [19]. An encoder for an RA code repeats each input bit a certain number of times, permutes the result, and then forms an output sequence via an accumulator. The corresponding graph for turbo codes [9] is similar in concept but a bit messier to illustrate, so is omitted here.

5.1.1 Implementation with Iterated-Decision Detection

To implement iterative detection and decoding, we can use the iterated-decision detector to implement message passing on the interference subgraph and any standard message-

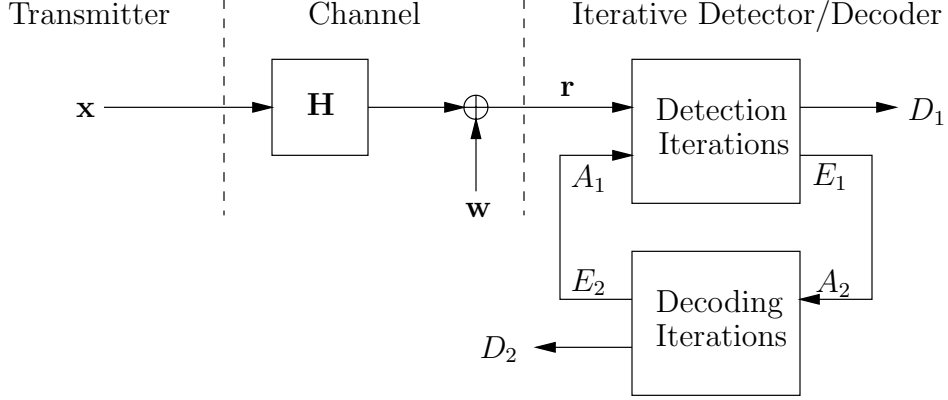


Figure 5.4: The flow of L -values between the detector and decoder blocks.

passing decoder to implement message-passing on the code subgraph. Thus, the iterated-decision detector sends information to the decoder and vice-versa, as shown in Fig. 5.4. The variables A_1 , E_1 , D_1 , A_2 , D_2 , E_2 denote log-likelihood ratios (L -values [37]). The detector block combines channel observations with *a priori* knowledge A_1 and outputs extrinsic information E_1 . This extrinsic information E_1 is accepted as *a priori* knowledge A_2 to the decoder, which outputs the extrinsic information E_2 that becomes the *a priori* knowledge A_1 of the detector block. The variables D_1 and D_2 , defined as $D_2 = E_2 + A_2$, can be used to make decisions on the symbols. Note that this model includes the special case of an uncoded system, for which $E_2 = A_2$. Let us now discuss how the exchanged information is interfaced between the two blocks.

In this coded scenario, the slicer of the iterated-decision detector is eliminated, and the vector $\tilde{\mathbf{x}}$, which would otherwise have been the slicer input, is passed directly to the decoder. As described in Theorem 3.1, the vector $\tilde{\mathbf{x}}$ is the codeword \mathbf{x} in marginally Gaussian noise, so the decoder can interpret $\tilde{\mathbf{x}}$ as the output of an AWGN channel. A message-passing decoder requires knowledge of the AWGN variance to compute conditional probabilities involving the received information, so (3.37) can be used to compute the variance of the effective noise. If explicitly required by the particular decoding algorithm, the *a priori* L -values $A_2 = E_1$ can be computed using (4.27).

Depending on the particular message-passing decoder used, the information available at the output of the decoder may be either extrinsic L -values E_2 or the sum of both extrinsic and *a priori* L -values $D_2 = E_2 + A_2$. If the latter is available, the extrinsic information E_2 can be computed by subtracting off the *a priori* L -values A_2 given by (4.27). Once available,

the extrinsic values E_2 can be converted into the single parameter ρ by computing the mean of $|\tanh(E_2/2)|$, as suggested by (4.37) and (4.39). Furthermore, symbol-by-symbol decisions $\hat{\mathbf{x}}$ can be made based on the extrinsic values E_2 . Both ρ and $\hat{\mathbf{x}}$ are then passed to the iterated-decision detector for another iteration of processing. After a sufficient number of iterations, symbol decisions can be made.

In the next few sections, we analyze the behavior of this iterative system by characterizing the relationship between the inputs A^l and outputs E^l of the detection and decoding blocks during the l th iteration. Eventually, these relationships are used to visualize the iterative exchange of information between the two blocks on extrinsic information transfer (EXIT) charts [58].

5.1.2 Binary Signalling Over AWGN Channels

As in Section 4.4.1, we begin by studying some concepts associated with binary signalling over AWGN channels, which will prove useful in the sequel. The scenario we discuss in this section is modelled as

$$\tilde{x} = x + w \tag{5.7}$$

where $x \in \{-1, +1\}$, w is independent noise of distribution $\mathcal{N}(0, \sigma_w^2)$, and \tilde{x} is the received symbol.

Gaussian L -Value Distributions

We begin by studying the distribution for the L -value of \tilde{x} . From (4.27), the associated L -value for \tilde{x} is

$$\tilde{X} \triangleq L(x|\tilde{x}) = \frac{2}{\sigma_w^2} \cdot \tilde{x} = \frac{2}{\sigma_w^2}(x + w). \tag{5.8}$$

We can also express \tilde{X} as

$$\tilde{X} = \mu_{\tilde{X}}x + w_{\tilde{X}} \tag{5.9}$$

where

$$\mu_{\tilde{X}} = \frac{2}{\sigma_w^2} \tag{5.10}$$

and $w_{\tilde{X}}$ being Gaussian distributed with mean zero and variance

$$\sigma_{\tilde{X}}^2 = \frac{4}{\sigma_w^2}. \tag{5.11}$$

Thus, the mean and variance of \tilde{X} are connected by

$$\mu_{\tilde{X}} = \frac{\sigma_{\tilde{X}}^2}{2}, \quad (5.12)$$

and the conditional probability density function of the L -value \tilde{X} is

$$p_{\tilde{X}}(t|x) = \frac{1}{\sqrt{2\pi}\sigma_{\tilde{X}}} e^{-(t-(\sigma_{\tilde{X}}^2/2)x)^2/2\sigma_{\tilde{X}}^2}. \quad (5.13)$$

Essentially, (5.11) relates the variance σ_w^2 of w to the variance $\sigma_{\tilde{X}}^2$ of the corresponding L -value \tilde{X} for AWGN channels. As σ_w^2 decreases, both the mean $\mu_{\tilde{X}}$ and variance $\sigma_{\tilde{X}}^2$ of the L -value increase.

Mutual Information Between Channel Input and L -Values

We measure the quality of the L -values at the receiver using the mutual information between the L -values and the channel input x . Based on the channel model of (5.9), the mutual information between the discrete-valued channel input x and the continuous-valued L -value \tilde{X} is computed as [43]

$$I_{\tilde{X}} = I(x; \tilde{X}) = \frac{1}{2} \sum_{k=-1,+1} \int_{-\infty}^{\infty} p_{\tilde{X}}(t|x=k) \log_2 \frac{2p_{\tilde{X}}(t|x=k)}{p_{\tilde{X}}(t|x=-1) + p_{\tilde{X}}(t|x=+1)} dt. \quad (5.14)$$

Substituting (5.13) into (5.14), we obtain

$$I_{\tilde{X}} = \mathcal{J}(\sigma_{\tilde{X}}) \quad (5.15)$$

where the function $\mathcal{J}(\cdot)$ is defined as

$$\mathcal{J}(\sigma) = 1 - \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}\sigma} e^{-(t-\sigma^2/2)^2/2\sigma^2} \log_2(1 + e^{-t}) dt. \quad (5.16)$$

The function $\mathcal{J}(\cdot)$ is monotonically increasing in σ with

$$\lim_{\sigma \rightarrow 0} \mathcal{J}(\sigma) = 0 \quad (5.17)$$

$$\lim_{\sigma \rightarrow \infty} \mathcal{J}(\sigma) = 1, \quad (5.18)$$

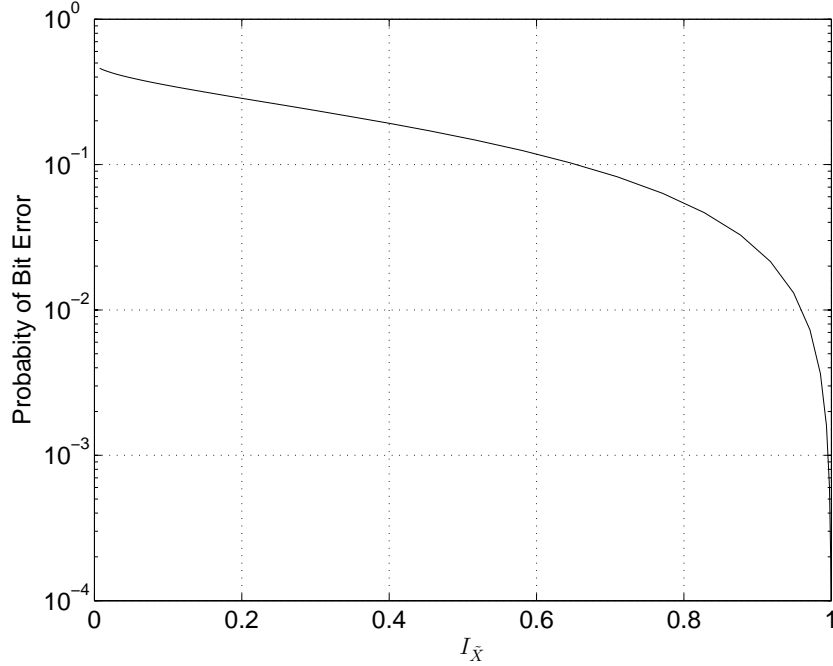


Figure 5.5: Relationship between mutual information $I_{\tilde{X}}$ and the probability of bit error $\Pr(\epsilon)$.

and thus the inverse function $\mathcal{J}^{-1}(\cdot)$ exists. Since we are dealing with binary variables, it makes sense that the maximum mutual information is equal to one.

The mutual information $I_{\tilde{X}}$, which measures the quality of the L -values at the receiver, can be explicitly translated into the more physically meaningful metric of bit-error rate. For an L -value variance of $\sigma_{\tilde{X}}^2$, the mutual information is $I_{\tilde{X}} = \mathcal{J}(\sigma_{\tilde{X}})$ and the probability of bit error is

$$\Pr(\epsilon) = \mathcal{Q}\left(\sqrt{\frac{\sigma_{\tilde{X}}^2}{4}}\right), \quad (5.19)$$

where we have used (5.11) and the error-rate formula for binary signalling over AWGN channels [51]. The relationship between the mutual information $I_{\tilde{X}}$ and the probability of bit error $\Pr(\epsilon)$ is plotted in Fig. 5.5.

5.1.3 Transfer Characteristic of the Detection Block

In this section, we characterize the input/output relationship between the L -values A_1 and E_1 of one iteration of the iterated-decision detector when mode-interleaved precoding is present.

We approximate the *a priori* L -values A_1 of the detection block as independent Gaussians conditioned on x , but this requires some justification. Since $A_1 = E_2$, we need to examine the validity of the independent Gaussian assumption for E_2 , the extrinsic information of the decoder. First, increasingly Gaussian-like L -value distributions for E_2 can be observed by successive simulated decoder iterations [69], a possible result of the Gaussian-like *a priori* distribution for A_2 and of the large number of variables involved in the computation of E_2 . Second, the independence assumption is reasonable because the extrinsic information E_2 about a variable x is not influenced by *a priori* knowledge A_2 of the same variable, and mode-interleaved precoding and codes of long block length further reduce any correlations.

With this approximation, A_1 can be modelled as

$$A_1 = \mu_{A_1} x + w_{A_1} \quad (5.20)$$

where w_{A_1} is an independent Gaussian random variable with variance $\sigma_{A_1}^2$ and mean zero, and

$$\mu_{A_1} = \frac{\sigma_{A_1}^2}{2} \quad (5.21)$$

because A_1 is an L -value modelled on Gaussian distributions, as in the case of (5.9) and (5.12). The relationship (5.21) can be alternatively derived using the symmetry condition $p_{A_1}(t|x) = p_{A_1}(-t|x)e^{tx}$ [53], where

$$p_{A_1}(t|x) = \frac{1}{\sqrt{2\pi}\sigma_{A_1}} e^{-(t - (\sigma_{A_1}^2/2)x)^2 / 2\sigma_{A_1}^2}. \quad (5.22)$$

The mutual information between the channel input x and the *a priori* information A_1 of the detector is thus $I_{A_1} = \mathcal{J}(\sigma_{A_1})$ where $\mathcal{J}(\cdot)$ is given by (5.16).

We now turn to the mutual information between the channel input x and the extrinsic L -values E_1 . As discussed in Section 3.1, the parameter ρ appears in the optimal filters, and the effect that A_1 has on the quality of E_1 is through ρ . From (4.37) and (4.27), the reliability corresponding to each L -value A_1 is

$$\rho \left(\frac{2A_1}{\sigma_{A_1}^2} \right) = \left| \tanh \left(\frac{A_1}{2} \right) \right|. \quad (5.23)$$

The constant value of ρ that is used in the detection block is then the expected value of $\rho(2A_1/\sigma_{A_1}^2)$ with respect to the distribution of A_1 :

$$\begin{aligned}\rho &= \int_{-\infty}^{\infty} \left| \tanh\left(\frac{A}{2}\right) \right| \left(\frac{1}{2} \cdot \frac{1}{\sqrt{2\pi}\sigma_{A_1}} e^{-(A-\sigma_{A_1}^2/2)^2/2\sigma_{A_1}^2} + \frac{1}{2} \cdot \frac{1}{\sqrt{2\pi}\sigma_{A_1}} e^{-(A+\sigma_{A_1}^2/2)^2/2\sigma_{A_1}^2} \right) dA \\ &= 1 - 2Q\left(\sqrt{\frac{\sigma_{A_1}^2}{4}}\right).\end{aligned}\tag{5.24}$$

From Theorems 3.1 and 3.2, we know that with mode-interleaved precoding, the parameter ρ^{l-1} leads to the soft output

$$\tilde{x}^l = x + u^l,\tag{5.25}$$

where u^l is a marginally Gaussian uncorrelated noise process with variance

$$\sigma_{u^l}^2 = \frac{1}{\gamma^l}.\tag{5.26}$$

The associated L -values for \tilde{x}^{l-1} can be taken as extrinsic information E^{l-1} ,

The L -values associated with \tilde{x}^l are taken as the extrinsic information E_1 , since the filtering that produces \tilde{x}^l does not take into account the *a priori* information A_1 via the corresponding bit \hat{x}^{l-1} . and the mutual information between the channel input x and E_1 is thus $I_{E_1} = \mathcal{J}(\sigma_{E_1})$ where

$$\sigma_{E_1}^2 = 4/\sigma_{u^l}^2 = 4\gamma^l.\tag{5.27}$$

Thus the relationship between I_{A_1} and I_{E_1} is characterized by (5.24), the channel-specific relationship between γ^l and ρ^{l-1} , (5.27), and (5.16). For the iterated-decision equalizer and the three-tap ISI channel of (3.112) with frequency-interleaved precoding, the relationship between γ^l and ρ^{l-1} is given by (3.88), (3.87), and (3.34), and the resulting transfer characteristic is plotted in Fig. 5.6 at different SNRs. For the iterated-decision equalizer and the asymptotic random ISI channel, the relationship between γ^l and ρ^{l-1} is given by (3.103) and (3.34), and the resulting transfer characteristic is plotted in Fig. 5.7.

There are several things to note about Figs. 5.6 and 5.7. First, I_{E_1} is a monotonically increasing function of I_{A_1} , meaning that the more information there is in A_1 , the more information there will be in E_1 . Mathematically, this is due to the monotonicity of (5.24), (5.27), (5.16), and the relationship between γ^l and ρ^{l-1} . Second, for a fixed I_{A_1} , the value of I_{E_1} increases with the SNR of the channel. Third, when $I_{A_1} = 1$, it is interesting that

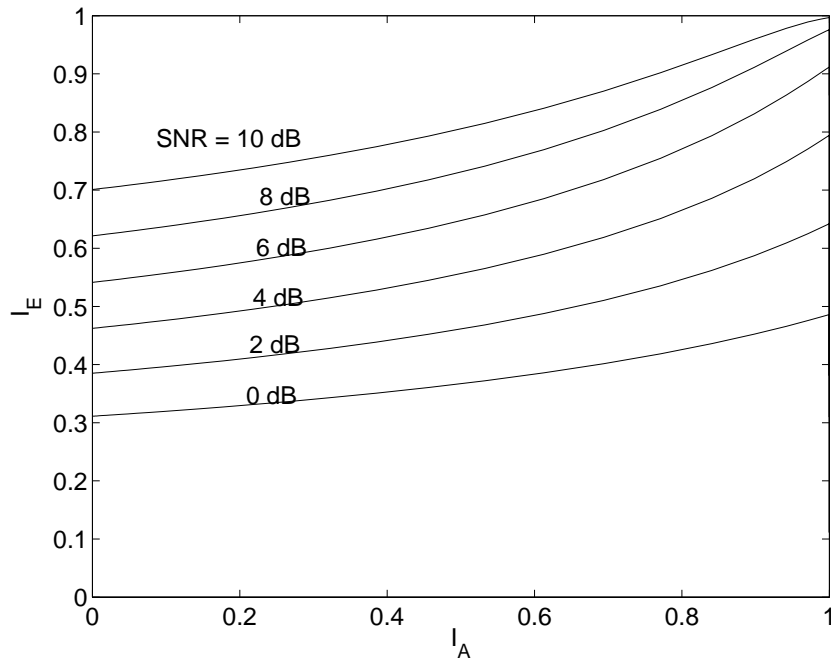


Figure 5.6: Transfer characteristic for the iterated-decision equalizer at various SNRs, with frequency interleaving and the three-tap ISI channel of (3.112).

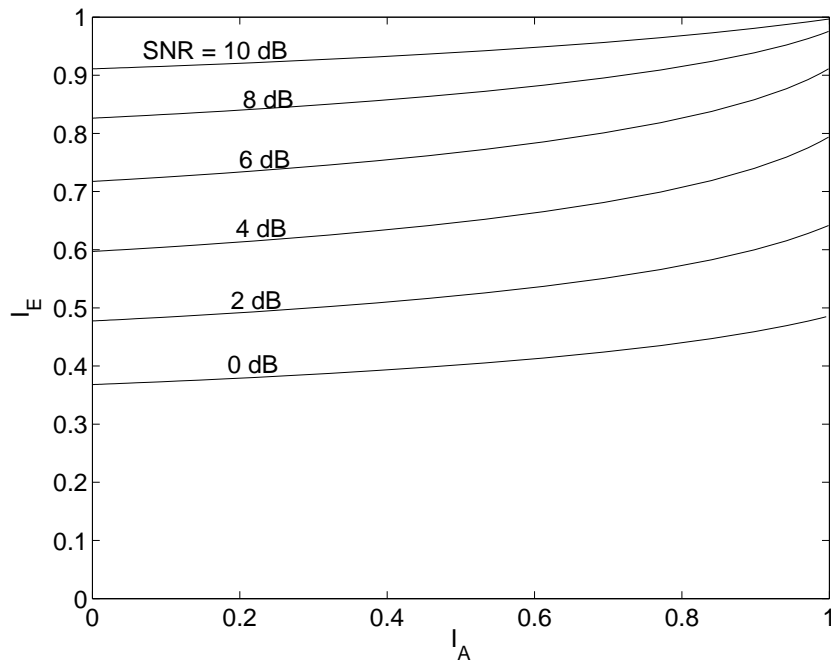


Figure 5.7: Transfer characteristic for the iterated-decision equalizer at various SNRs, with the asymptotic random ISI channel.

$I_{E_1} < 1$ for all finite SNRs. In other words, the extrinsic information E_1 contains less knowledge of x than the perfect *a priori* knowledge of x that is input to the detection block. The reason for this phenomenon is that the perfect *a priori* information is used only to cancel out the interference. The extrinsic information E_1 , which is then based on the received signal with all the interference removed, cannot be perfect because of noise originating from the channel. Fourth, when $I_{A_1} = 1$ all interference is cancelled, so I_{E_1} is equal in both Figs. 5.6 and 5.7 for the same SNR, emphasizing the fact that channels with the same Frobenius norm have the potential to perform the same. However, the slopes of the transfer characteristics of different channels at the same SNR indicate the degree of “difficulty” with which the iterated-decision detector removes interference. The steeper slopes in Fig. 5.6 show that the iterated-decision detector cancels more interference for the channel corresponding to Fig. 5.7.

5.1.4 Transfer Characteristic of the Decoding Block

This extrinsic information E_1 of the detector is taken to be the *a priori* information A_2 for the message-passing decoder. The transfer characteristic of the decoding block describes the relationship between A_2 and the decoder extrinsic output E_2 in terms of mutual information. In this section, we focus on LDPC codes because not only do they give fairly high coding gains, but the transfer characteristic for sum-product decoding (described in Section 4.1) is easily computed using density evolution with Gaussian approximations [18].

Let us consider a regular binary (d_v, d_c) -LDPC code, where d_v is the number of neighbors a variable node has and d_c is the number of neighbors a check node has. Under sum-product decoding, messages are sent back and forth between variable and check nodes. Typically, messages are sent alternately in parallel from variable to check nodes and back from check to variable nodes. A variable node receives messages from its d_v neighbors, processes them, and sends messages back to all the neighbors. Similarly, a check node receives d_c from its neighbors, processes them, and sends messages back. Each output message that is sent from a node is a function of all incoming messages to that node, except for the one that arrived on the edge over which the output message is sent. This restriction is necessary to produce the correct marginal *a posteriori* probabilities on cycle-free graphs. This two-step procedure is performed repeatedly. After n iterations, the information at each variable node summarizes information collected from its depth- $2n$ subgraph.

We use the “local tree assumption” that the girth of the graph is large enough so that the subgraph associated with a variable node is cycle-free. The incoming messages to every node can then be taken to be independent. The assumption is valid because of the concentration theorem of [53], which states that for almost all randomly constructed long codes and for almost all inputs, with high probability the decoder performance will be close to the performance one can observe on the corresponding cycle-free graph.

When all variables in the code subgraph are binary and all functions (except single-variable functions) are parity checks, the node processing in Fig. 4.3 can be greatly simplified. LDPC codes and RA codes both fall within this category, as depicted in Figs. 5.2 and 5.3 respectively. The simplification comes from being able to parametrize binary probability mass functions using a single parameter, typically the log-likelihood ratio (L -value). If v is the L -value message sent from a variable node to a check node, then

$$v = \sum_{i=0}^{d_v-1} u_i \quad (5.28)$$

where u_i , $i = 1, 2, \dots, d_v - 1$ are the incoming L -values from all of the neighbors of the variable node except for the neighbor to receive the message, and u_0 is the L -value received from outside of the code subgraph. (For coding over AWGN channels, u_0 is the L -value associated with the noisy channel observation. For coding over interference channels, u_0 is the L -value associated with the output of a detection iteration.) The corresponding rule for computing the messages sent from a check node to a variable node, known as the “tanh rule,” is

$$\tanh \frac{u}{2} = \prod_{j=1}^{d_c-1} \tanh \frac{v_j}{2} \quad (5.29)$$

where v_j , $j = 1, 2, \dots, d_c - 1$ are the incoming L -values and u is sent on the remaining edge.

Density evolution [53] is a tool that analytically determines the convergence behavior of ensembles of asymptotically long codes. First, without loss of generality, the all-zero codeword is assumed to be sent. The observed L -value u_0 is considered a random variable whose density is determined by fixed channel parameters (e.g. noise variance). Then the densities of subsequent messages are iteratively computed from the relationships between the random variables given in (5.28) and (5.29). Eventually, the density of the L -value v converges to a density with a finite probability of error, or to a “point mass at infinity,” with

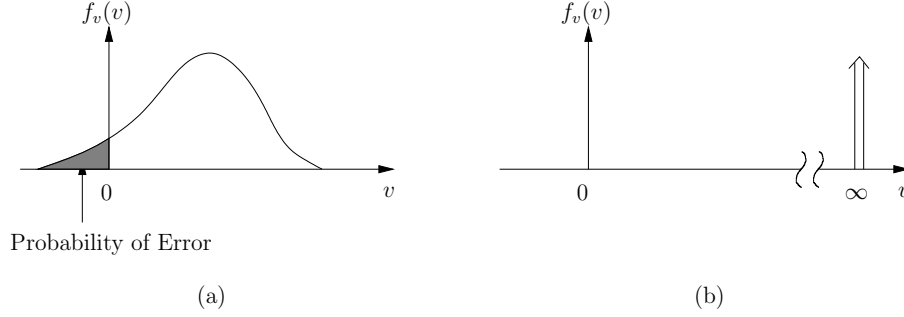


Figure 5.8: The density of the message from a variable node converges either to (a) a density with a finite probability of error or (b) a “point mass at infinity.”

zero probability of error, as illustrated in Fig. 5.8. For binary erasure channels (BECs) the message densities are one-dimensional and analysis is tractable, but for most other channels including the AWGN channel, the message densities are infinite dimensional and density evolution is too complicated to perform.

Density evolution is made tractable in [18] by approximating the message densities as Gaussians for regular LDPC codes or Gaussian mixtures for irregular LDPC codes using reasoning similar to that in Section 5.1.3. A Gaussian is specified by its mean and variance, so those are the only two parameters that need to be tracked. Furthermore, we can enforce the symmetry condition that $f(x) = f(-x)e^x$ where $f(x)$ is a message density for an L -value [52], which is preserved under density evolution for all messages. This condition translates to $\sigma^2 = 2m$ for an $\mathcal{N}(m, \sigma^2)$ Gaussian variable. Thus, only the mean needs to be tracked. From (5.28), the mean of v during the l th iteration is

$$m_v^l = m_{u_0} + (d_v - 1)m_u^{l-1} \quad (5.30)$$

where m_{u_0} is the mean of u_0 and m_u is the mean of the i.i.d. random variables u_i for $1 \leq i < d_v$. The initial value m_u^0 is set to zero. From (5.29), the update rule for m_u is [18]

$$m_u^l = \phi^{-1} \left(1 - [1 - \phi(m_v^l)]^{d_c - 1} \right) \quad (5.31)$$

where

$$\phi(x) = \begin{cases} 1 - \frac{1}{\sqrt{4\pi x}} \int_{-\infty}^{\infty} \tanh\left(\frac{u}{2}\right) e^{-\frac{(u-x)^2}{4x}} du & \text{for } x > 0 \\ 1 & \text{for } x = 0 \end{cases} \quad (5.32)$$

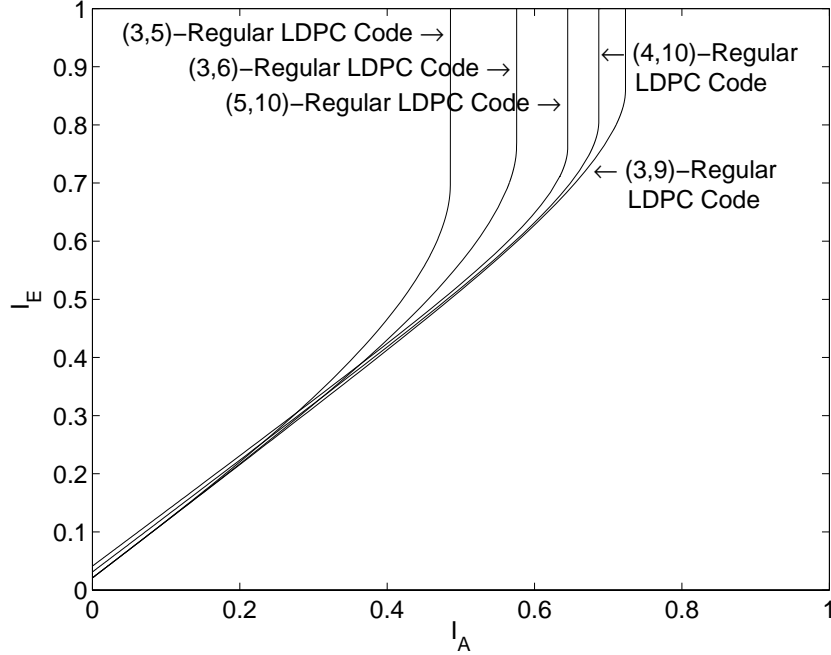


Figure 5.9: Transfer characteristic for sum-product decoding of various regular LDPC codes. An infinite number of decoding iterations is assumed.

is a continuous and monotonically decreasing function on $[0, \infty)$, with $\phi(0) = 1$ and $\phi(\infty) = 0$. A good approximation of $\phi(x)$ that we use is [18]

$$\phi(x) \approx \begin{cases} e^{-0.4527x^{0.86}+0.0218} & \text{for } x < 10 \\ \sqrt{\frac{\pi}{x}} e^{-\frac{x}{4}(1-\frac{10}{7x})} & \text{for } x \geq 10. \end{cases} \quad (5.33)$$

Figure 5.9 shows the transfer characteristic for various regular LDPC codes. Each point of a curve is generated by fixing the variance $2/m_{u_0}$ of the noise in which x is observed. By (5.11), this translates to a variance of $2m_{u_0}$ for the initial L -value density of u_0 . Thus, the mutual information between the channel input x and the L -values A_2 is $I_{A_2} = \mathcal{J}(\sqrt{2m_{u_0}})$, where $\mathcal{J}(\cdot)$ is given in (5.16). After using (5.30) and (5.31) iteratively until convergence, the mutual information between the transmitted bits x and the L -values E_2 is $I_{E_2} = \mathcal{J}(\sqrt{2m_v^\infty})$. Figure 5.9 clearly shows the well-documented threshold phenomenon, in which an arbitrarily small error probability can be achieved if I_{A_2} is above (i.e. the noise level is below) a certain threshold as the block length goes to infinity [30, 31, 45, 53].

5.1.5 EXIT Charts

Since information is iteratively passed from the output of the detector to the input of the decoder and then from the output of the decoder to the input of the detector, it is helpful to plot the transfer characteristic of the detection block on the same graph as the transfer characteristic for the decoder, except with the axes swapped. The presence of both curves on the same graph allows visualization of the exchange of extrinsic information. This kind of graph is called an extrinsic information transfer (EXIT) chart [58].

To compute the trajectory that represents the exchange of information, we initialize to $I_{A_1^0} = 0$ and $l = 1$ and repeatedly perform the following sequence of computations:

$$I_{E_1^l} = T_1(I_{A_1^{l-1}}) \quad (5.34)$$

$$I_{A_2^l} = I_{E_1^l} \quad (5.35)$$

$$I_{E_2^l} = T_2(I_{A_2^l}) \quad (5.36)$$

$$I_{A_1^l} = I_{E_2^l} \quad (5.37)$$

where l is increased to $l + 1$ after each iteration and $T_1(\cdot)$ and $T_2(\cdot)$ are the transfer characteristics of the detector and decoder respectively. Thus, the relationship between $I_{E_2^{l+1}}$ and $I_{E_2^l}$ is

$$I_{E_2^{l+1}} = T_2(T_1(I_{E_2^l})). \quad (5.38)$$

On the EXIT chart, this relationship corresponds to successively moving upwards and then to the right between the two curves characterizing the detection and decoding blocks, with each “step” of the trajectory being one iteration of the iterated-decision detector followed by decoder iterations until convergence of densities. The mutual information continues to increase until $I_{E_2^{l+1}} = I_{E_2^l}$ or, equivalently, $T_2^{-1}(I_{E_2}) = T_1(I_{E_2})$.

Figure 5.10 shows an EXIT chart for the iterated-decision equalizer combined with a sum-product decoder, for a (5, 10)-regular LDPC code and the asymptotic random ISI channel at a rate-normalized SNR of $\text{SNR}_{\text{norm}} = 3$ dB. Rate-normalized SNR allows us to compare systems with different modulations schemes; the definition we use here is the one for systems in which the transmitter does not do water pouring, discussed in Appendix D. The EXIT chart predicts that codewords are decoded with only two iterations. In comparison, ten simulated trajectories for coded block lengths of $N = 10000$ and random channels

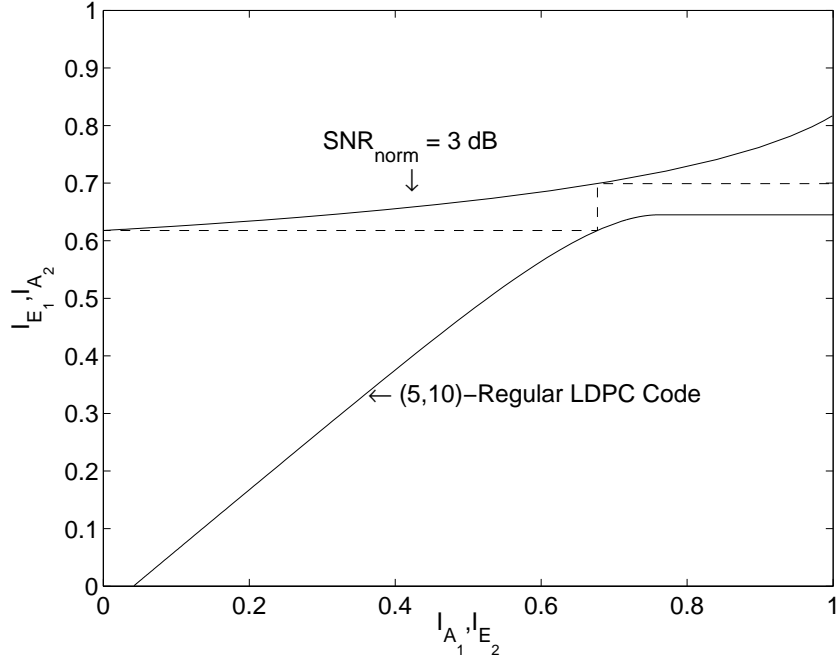


Figure 5.10: EXIT chart for the iterated-decision equalizer and an iterative decoder, for a (5, 10)-regular LDPC code and the asymptotic random ISI channel at an SNR_{norm} of 3 dB.

of length 1024 are shown in Fig. 5.11, superimposed on the theoretical transfer characteristic curves. These simulated trajectories are plotted by translating simulation results into mutual information values. We compute $I_{A_2^l} = I_{E_1^l}$ from $\mathcal{J}(\sigma_{E_1^l})$, where $\mathcal{J}(\cdot)$ is given by (5.16) and $\sigma_{E_1^l}$ is computed by (5.27) and (3.26). The values $I_{A_1^l} = I_{E_2^l}$ are computed by taking the simulated probability of bit error $\Pr(\epsilon^l)$ at the decoder output, numerically solving for $\sigma_{E_2^l}^2$ in the equation (5.19), and then computing $\mathcal{J}(\sigma_{E_2^l})$. Note that it is not practically feasible to alternatively compute $\sigma_{E_2^l}^2$ by taking the sample mean or variance of E_2^l because of the possibility of L -values at infinity. Looking at Fig. 5.11, we note that there is some variance amongst the trajectories—a few of the trajectories show that the codeword was decoded on the first iteration, while the rest show that the codeword was decoded in two, as predicted by Fig. 5.10. Nevertheless, the theoretical trajectory is a fairly good indicator of what to expect experimentally.

Figure 5.12 plots the transfer characteristic for the iterated-decision equalizer at different SNR for the asymptotic random channel, along with the characteristic for the decoding of a (5, 10)-regular LDPC code. These curves indicate that an arbitrarily small error probability can be achieved if the SNR is larger than a certain threshold—in this example, the threshold

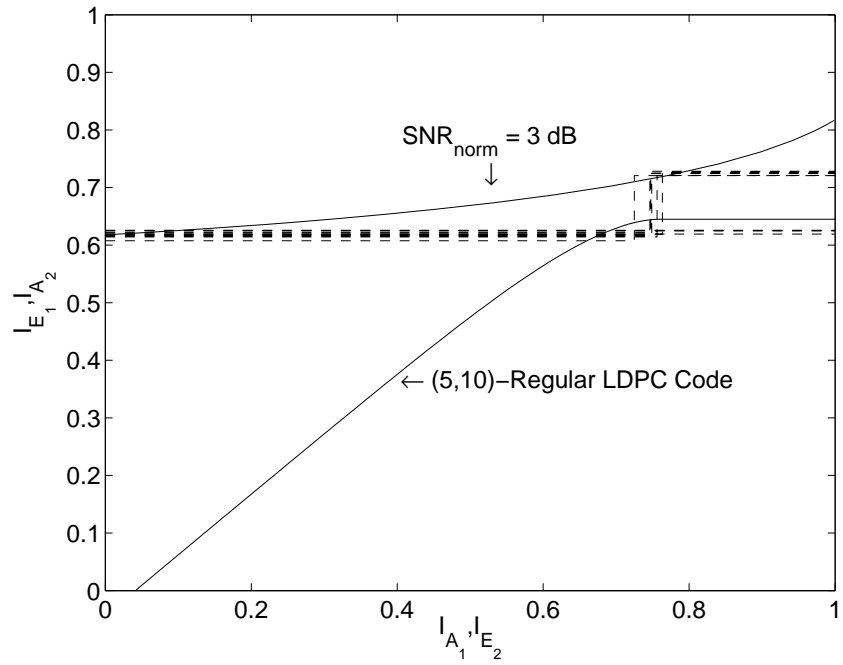


Figure 5.11: Ten simulated trajectories for the iterated-decision equalizer and an iterative decoder, for a (5,10)-regular LDPC code and a random ISI channel of length 1024 at an SNR_{norm} of 3 dB.

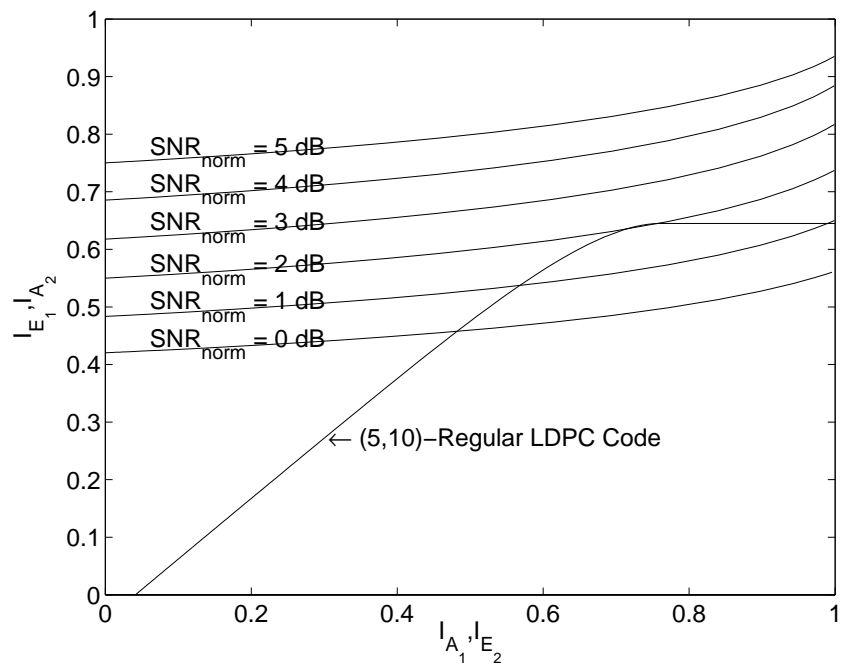


Figure 5.12: EXIT chart for the iterated-decision equalizer and an iterative decoder, for a (5,10)-regular LDPC code and the asymptotic random ISI channel.

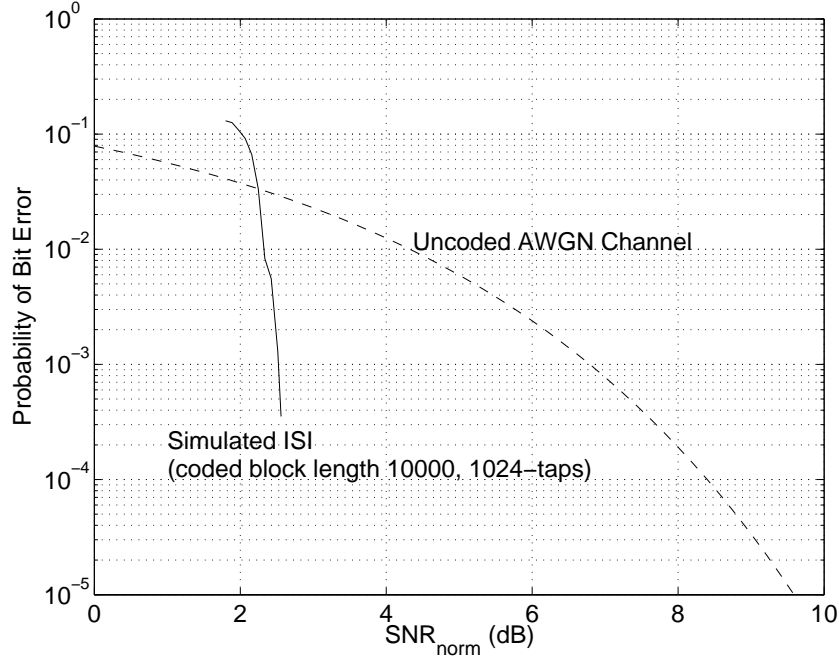


Figure 5.13: Simulated performance of the iterated-decision equalizer and an iterative decoder, for a (5, 10)-regular LDPC code and a random ISI channel of length 1024.

is slightly larger than an SNR_{norm} of 2 dB. By numerical simulation, the exact threshold is determined to be at an SNR_{norm} of 2.03 dB. At SNRs lower than this threshold, the curves intersect or “pinch-off,” corresponding to a final error probability that is bounded away from zero. The simulated performance of the iterated-decision equalizer and a decoder for a (5, 10)-regular LDPC code for coded block lengths of $N = 10000$ and random channel lengths of 1024 is shown in Fig. 5.13 as a function of SNR_{norm} . The simulations show that the bit-error rate begins to drop significantly at an SNR_{norm} slightly greater than 2 dB. With block lengths longer than $N = 10000$, we would expect the curve to drop even more steeply. Interestingly, the corresponding pinch-off for a (5, 10)-regular LDPC code over an AWGN channel is 2.01 dB when normalized to the capacity of the AWGN channel [18].

The iterated-decision detector for uncoded systems, discussed in Chapter 3, corresponds to the degenerate case in which the decoder is the identity operator $E_2 = A_2$. Thus, $I_{E_2} = I_{A_2}$ and the *a priori* information A_1 used in the next iteration is simply equal to the extrinsic information E_1 from the previous iteration. Figure 5.14 shows the EXIT chart for the iterated-decision equalizer operating at an SNR of 6 dB in an uncoded system. We have already noted that with perfect *a priori* information A_1 , the extrinsic mutual information

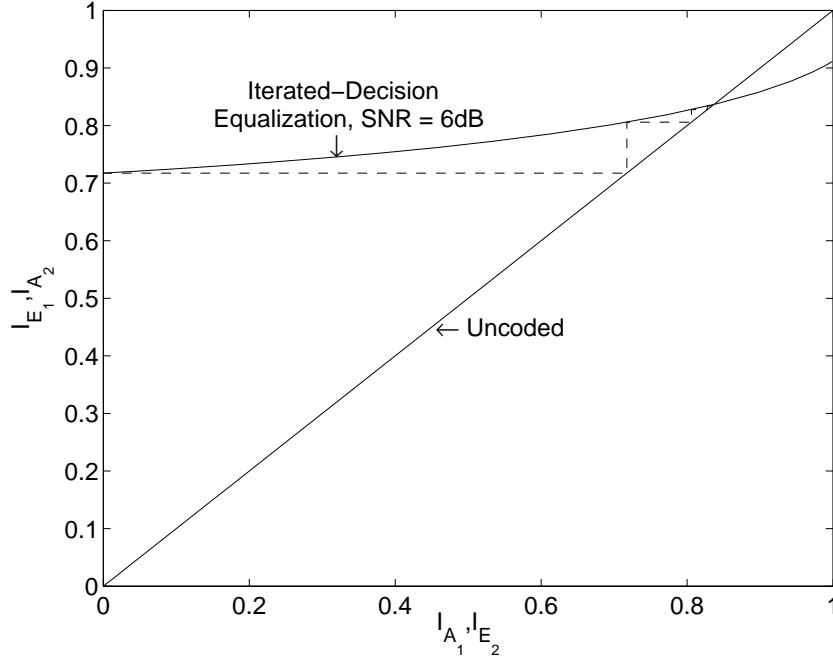


Figure 5.14: EXIT chart for the iterated-decision equalizer an SNR of 6 dB, for an uncoded system with the asymptotic random ISI channel.

$I_{E_1} < 1$. Thus in this uncoded case, the mutual information value at which both curves intersect is always less than unity for finite SNR.

5.1.6 Comparison to Turbo Equalization

The combination of iterated-decision detection and decoding is similar to turbo equalization [20], but there are some intriguing differences. First, interleaving is almost always a component of turbo equalization to decorrelate error events introduced by the equalizer so that the decoder effectively sees an AWGN channel. The counterpart to interleaving in the combined iterated-decision detector and decoder is mode-interleaved precoding, which creates an effective AWGN channel for the decoder as shown in Chapter 3. Second, separate likelihoods are kept for different symbols in the equalization portion of turbo equalization. However, for the iterated-decision detector, the same likelihood is kept for all symbols because each symbol effectively sees the same statistical AWGN channel. While it would be interesting in future to investigate the performance loss of the iterated-decision detector in keeping only hard decisions and an average symbol reliability instead of individual symbol reliabilities, we suspect that with mode-interleaved precoding the loss is asymptotically

negligible. This would make the iterated-decision detector attractive not only in applications where low complexity is required, but also in applications where memory to store symbol reliabilities is limited, or where the message-passing algorithm is distributed and communication between nodes is costly.

The turbo equalization method in [33] contains a soft ISI canceller that has similarities to the combined iterated-decision detector and decoder, but is not optimized to perform as well. First, while soft information is indeed supplied to the filter in [33] corresponding to $d^l[n]$, the soft information is not incorporated into the design of the filters $b^l[n]$ and $d^l[n]$. Like the multistage detection of Fig. 1.15, the filter $b^l[n]$ is simply a matched filter and the noncausal filter $d^l[n]$ is the cascade of the channel and matched filter, the implicit assumption being that perfect hard decisions, not soft information, are supplied to $d^l[n]$. Second, a scaling factor is chosen in [33] to minimize the mean-square error between the transmitted symbols $x[n]$ and the equalizer outputs $\tilde{x}^l[n]$. The resulting soft equalizer output $\tilde{x}^l[n]$ is taken directly in [33] as the *a priori* L -value that is input to the decoder component. As shown in Section 4.4.1, the L -value should be related to $\tilde{x}^l[n]$ by a scaling factor that is inversely proportional to the effective noise variance. Moreover, the bias of the MMSE scaling factor needs to be taken into account. This second difference was corrected in [60].

The authors in [60] additionally propose a turbo equalization method that uses a soft ISI canceller whose filter is time-varying, because the coefficients are based on separate reliabilities for each symbol.

5.2 Multilevel Codes

For channels with high SNR, high rate codes are required to approach fundamental information theoretic limits. Furthermore, for interference channels, detection and decoding have to be integrated at the receiver. In this section, we discuss the combination of iterated-decision detection with multilevel codes [68]. In Sections 5.2.1 and 5.2.2, we review multilevel coding and decoding for AWGN channels. For interference channels, the similarity between iterated-decision detection and multistage decoding inspires us to merge the two together in Section 5.2.3, and the message passing interpretation in Section 5.1 is extended to iterated-decision detection with multilevel codes in Section 5.2.4.

5.2.1 Multilevel Encoding

Multilevel codes are high-rate, nonbinary codes that result from the combination of multiple binary codes. The multilevel encoder is depicted in Fig. 5.15. A vector of information bits \mathbf{p} is partitioned into L smaller vectors $\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_L$. These vectors are input to a corresponding set of encoders, producing a set of coded vectors $\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_L$. The encoders may be for different types of codes (block, convolutional, turbo, etc.) with different rates, but the length of all the \mathbf{q}_l 's must be equal for $l = 1, 2, \dots, L$. To satisfy this constraint, \mathbf{p} must obviously be partitioned into the \mathbf{p}_l 's according to the different encoder rates. The code rate R of the overall scheme is equal to the sum of the individual code rates R_l , i.e.,

$$R = \sum_{l=1}^L R_l. \quad (5.39)$$

The final stage of the encoder maps $\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_L$ to a vector of 2^L -ary symbols. Each set of L bits that are located in the same position within the \mathbf{q}_l 's forms a binary address (b_1, b_2, \dots, b_L) that is uniquely mapped to a symbol x in the 2^L -ary alphabet \mathcal{X} . Many different mappings are possible, and Fig. 5.16 shows six different mappings for the 8-PAM constellation. Usually, the mapping is derived by successively partitioning the signal set \mathcal{X} into subsets. Block and Ungerboeck partitioning of the 8-PAM signal constellation are depicted in Figs. 5.17 and 5.18 respectively. Let us define the subset $\mathcal{X}(b_1, \dots, b_{l-1}) \triangleq \{x | b_1, \dots, b_{l-1}\}$. Then in the first step, at partition level 1, the signal set \mathcal{X} is divided into $\mathcal{X}(b_1 = 0)$ and $\mathcal{X}(b_1 = 1)$. At each subsequent level $i \leq L$, the signal set $\mathcal{X}(b_1, \dots, b_{l-1})$ is further partitioned into $\mathcal{X}(b_1, \dots, b_{l-1}, 0)$ and $\mathcal{X}(b_1, \dots, b_{l-1}, 1)$. After the L th partition,

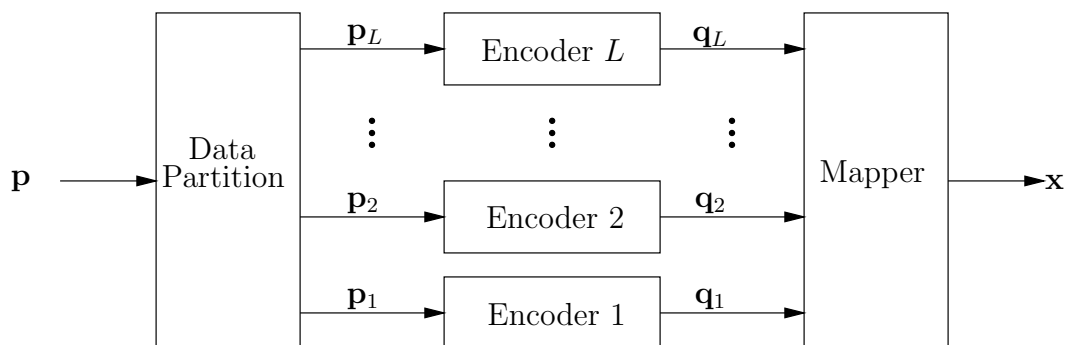


Figure 5.15: A multilevel encoder.

Gray	000	001	011	010	110	111	101	100
Block	000	001	010	011	100	101	110	111
d21	000	011	101	110	111	001	010	100
d23	000	011	101	110	001	010	100	111
Anti-Gray	000	111	001	110	011	100	010	101
Ungerboeck	000	100	010	110	001	101	011	111

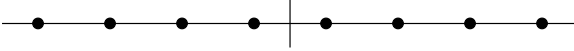


Figure 5.16: Different mapping schemes for 8-PAM.

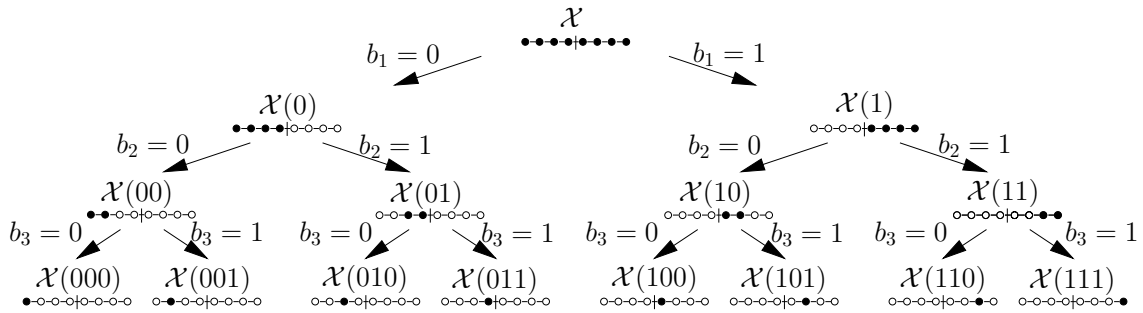


Figure 5.17: Block partitioning for 8-PAM.

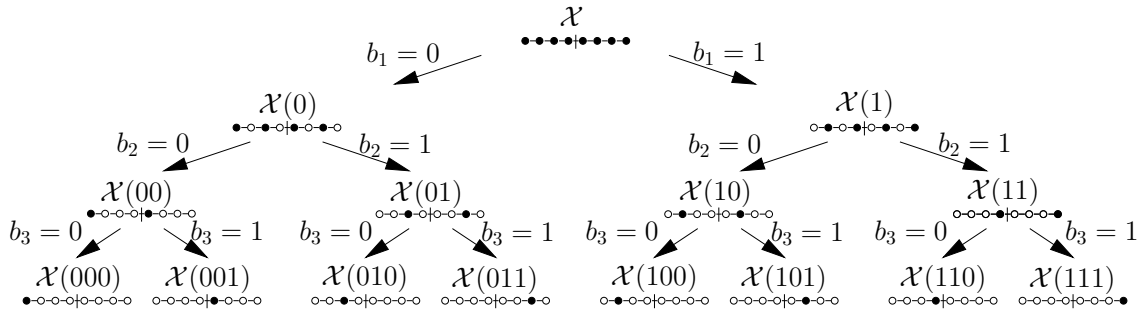


Figure 5.18: Ungerboeck partitioning for 8-PAM.

all subsets contain only one signal point. As we shall discuss later, the choice of partitioning scheme affects the properties of the coded modulation system.

5.2.2 Decoding for AWGN Channels

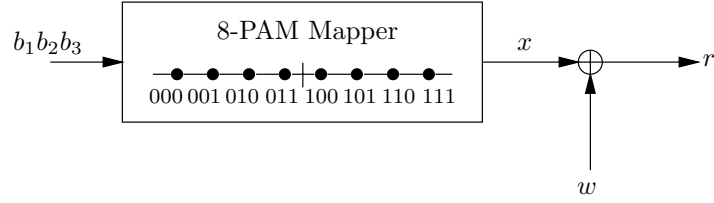
If we turn to information theory, not only do we discover a convenient decoding algorithm, but we also gain insight on how to design multilevel codes. The physical channel is characterized by the set $\{f(r|x)|x \in \mathcal{X}\}$ of conditional probability density functions of the received point r given the transmitted symbol x . Since the mapping from binary addresses (b_1, b_2, \dots, b_L) to symbols x is reversible, the mutual information between (b_1, b_2, \dots, b_L) and the AWGN channel output r is equal to the mutual information between x and r . Furthermore, using the chain rule of mutual information, we have that

$$\begin{aligned} I(R; X) &= I(R; B_1, B_2, \dots, B_L) \\ &= I(R; B_1) + I(R; B_2|B_1) + \dots + I(R; B_L|B_1, B_2, \dots, B_{L-1}). \end{aligned} \quad (5.40)$$

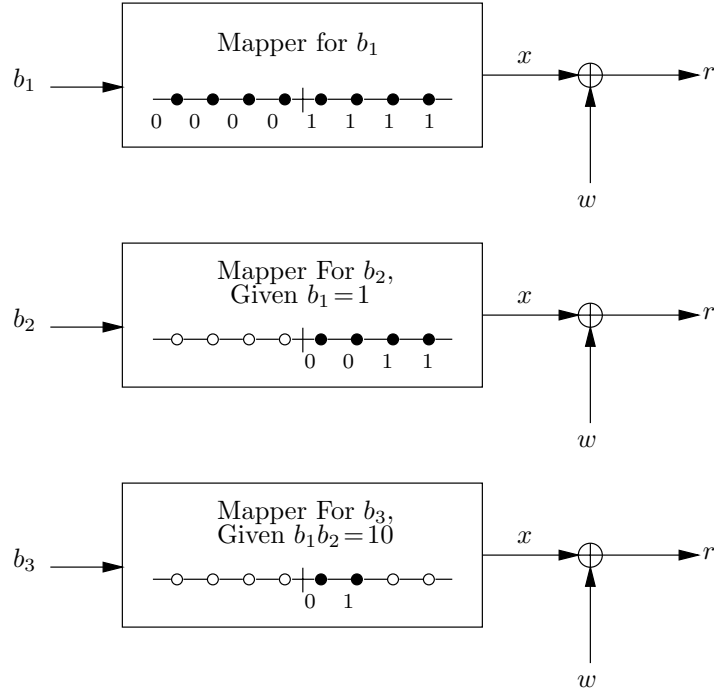
The terms of the equation give the interpretation that the physical channel can be decomposed into L parallel theoretical subchannels. The l th theoretical subchannel is responsible for the transmission of b_l , provided that b_1, \dots, b_{l-1} are known. Figure 5.19 compares the physical channel to the set of theoretical subchannels for block labelling. By the chain rule of mutual information, both scenarios have the same mutual information. However, the physical channel in Fig. 5.19(a) maps the entire binary address to a signal point x that is transmitted over the noisy channel, whereas the l th theoretical subchannel in Fig. 5.19(b) maps the bit b_l to one of the signal points in $\mathcal{X}(b_1, \dots, b_l)$. Thus, the mapper for the l th theoretical subchannel depends upon the bits b_1, \dots, b_{l-1} of lower levels. For example, the mapper for b_3 in Fig. 5.19(b) is determined by $b_1 b_2 = 10$.

Assuming all symbols in \mathcal{X} are equiprobable, the mutual information of the l th theoretical subchannel can be computed as

$$\begin{aligned} I(R; B_l|B_1, B_2, \dots, B_{l-1}) &= E_{b_1 \dots b_{l-1}} [I(R; B_l|b_1, b_2, \dots, b_{l-1})] \\ &= \frac{1}{2^{l-1}} \sum_{b_1 \dots b_{l-1}} I(R; B_l|b_1, b_2, \dots, b_{l-1}). \end{aligned} \quad (5.41)$$



(a)



(b)

Figure 5.19: Channels for block partitioning of 8-PAM. (a) Physical channel. (b) Parallel theoretical subchannels.

Each term in the sum can in turn be computed as [43]

$$\begin{aligned}
 & I(R; B_l | b_1, b_2, \dots, b_{l-1}) \\
 &= \frac{1}{2} \int_{-\infty}^{\infty} f(r | b_1 \dots b_{l-1} 0) \log \frac{f(r | b_1 \dots b_{l-1} 0)}{\frac{1}{2} f(r | b_1 \dots b_{l-1} 0) + \frac{1}{2} f(r | b_1 \dots b_{l-1} 1)} dr \\
 &+ \frac{1}{2} \int_{-\infty}^{\infty} f(r | b_1 \dots b_{l-1} 1) \log \frac{f(r | b_1 \dots b_{l-1} 1)}{\frac{1}{2} f(r | b_1 \dots b_{l-1} 0) + \frac{1}{2} f(r | b_1 \dots b_{l-1} 1)} dr \quad (5.42)
 \end{aligned}$$

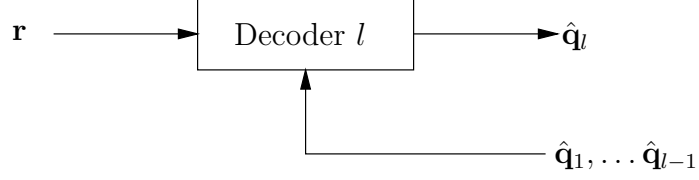


Figure 5.20: A multistage decoder.

where

$$f(r|b_1 \cdots b_l) = \frac{1}{2^{L-l}} \sum_{x \in \mathcal{X}(b_1 \cdots b_l)} f(r|x). \quad (5.43)$$

The chain rule of mutual information in (5.40) suggests a method of decoding multilevel codes. Multistage decoders, depicted in Fig. 5.20, decode the component codes one at a time, generating decisions in the order of $\hat{\mathbf{q}}_1, \hat{\mathbf{q}}_2, \dots, \hat{\mathbf{q}}_L$. The l th decoder processes not only the received vector \mathbf{r} but also the decisions $\hat{\mathbf{q}}_1, \hat{\mathbf{q}}_2, \dots, \hat{\mathbf{q}}_{l-1}$ of previous stages. Since the mutual information $I(R; X)$ of a 2^L -ary modulation scheme is equal to the sum of the mutual information of the theoretical parallel subchannels, the rate $I(R; X)$ is achievable via multilevel encoding and multistage decoding if and only if the individual rates are chosen to be equal to the mutual information of the corresponding theoretical subchannels, i.e.,

$$R_l = I(R; B_l | B_1, B_2, \dots, B_{l-1}). \quad (5.44)$$

Although multistage decoding does not take into account higher levels when decoding lower levels, it suffices to achieve $I(R; X)$ as long as the rates are appropriately chosen. Note that both the value $I(R; X)$ and the achievability of $I(R; X)$ are not affected by the mapping scheme selected; the choice of mapping scheme affects only the rates R_l of the theoretical subchannels. This can be observed in Figs. 5.21 and 5.22, which plot the various rates for block and Ungerboeck partitioning respectively. To achieve an overall rate of 2.5 bits/dim, the optimum individual rates for block partitioning are $R_1 = 0.71$, $R_2 = 0.86$, and $R_3 = 0.93$, and the optimum individual rates for Ungerboeck partitioning are $R_1 = 0.52$, $R_2 = 0.98$, and $R_3 = 1$. Figure 5.23 shows $I(R; X)$ using PAM constellations of different sizes, corresponding to codes with different numbers of levels. This plot, which is similar to one appearing in [62], shows that the SNR gap between the achievable rate $I(R; X)$ and capacity asymptotically approaches 1.53 dB.

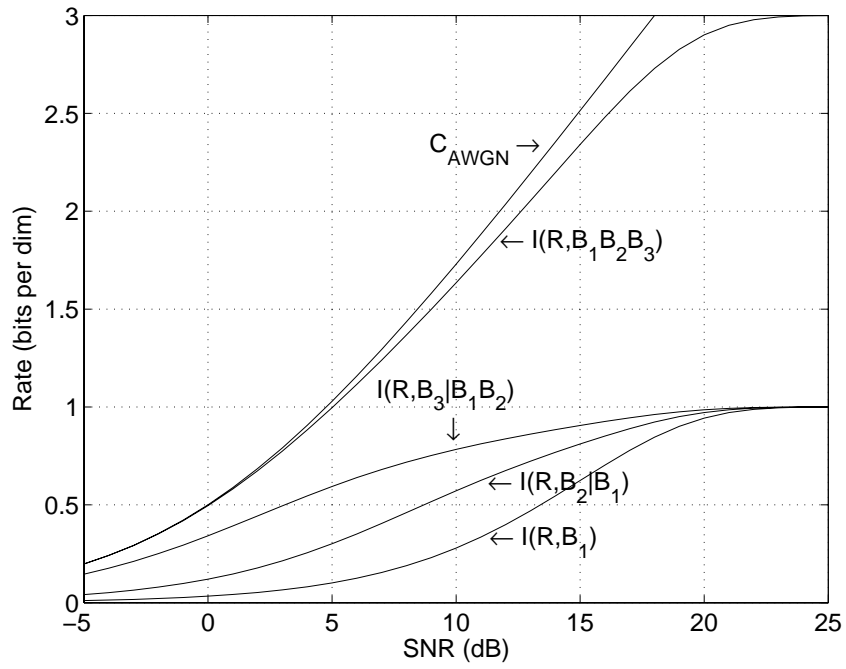


Figure 5.21: Mutual information of the physical AWGN channel and the theoretical sub-channels for a three-level code (8-PAM) with block partitioning.

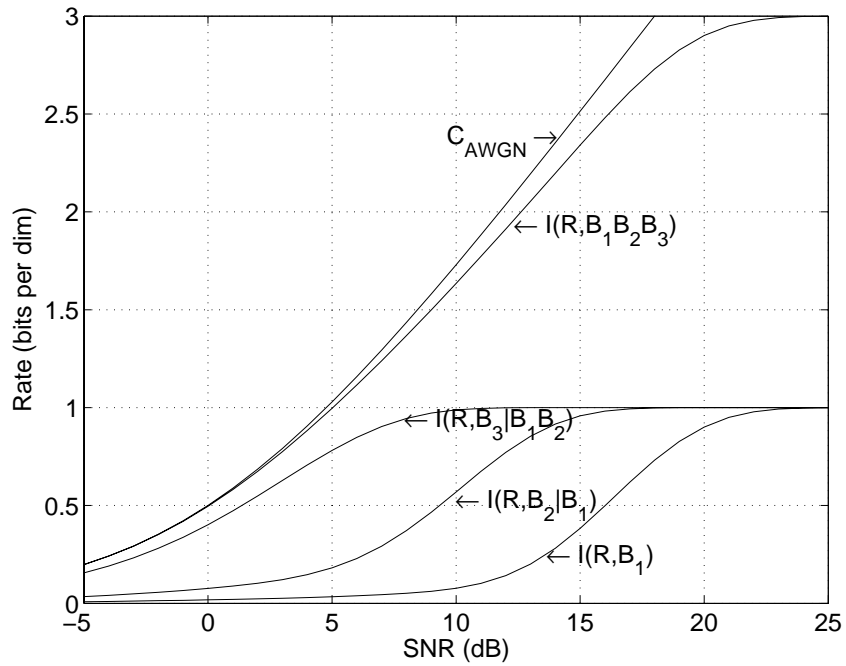


Figure 5.22: Mutual information of the physical AWGN channel and the theoretical sub-channels a for three-level code (8-PAM) with Ungerboeck partitioning.

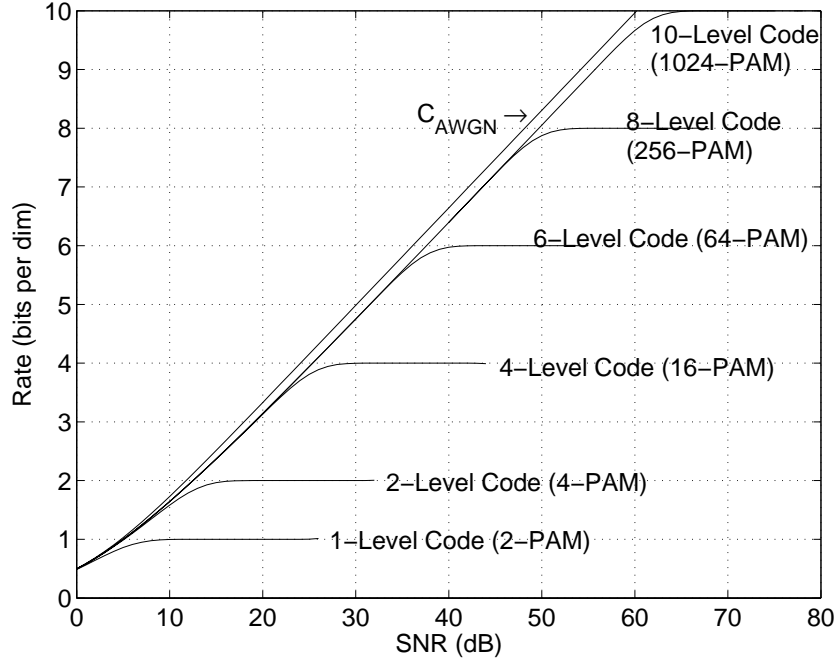


Figure 5.23: Mutual information of an AWGN channel with PAM constellations.

A much wider variety of rate combinations can be used to achieve the rate $I(R; X)$ provided that joint maximum-likelihood decoding is used instead of low-complexity multistage decoding. From well known results in multiuser information theory, the rate $I(R; X)$ is achievable via multilevel encoding and joint maximum-likelihood decoding if and only if the rates R_l satisfy the following conditions:

$$\sum_{l=1}^L R_l = I(R; B_1, B_2, \dots, B_L) \quad (5.45)$$

$$\sum_{l \in \mathcal{S}} R_l \leq I(R; \{B_l | l \in \mathcal{S}\} | \{B_j | j \in \bar{\mathcal{S}}\}) \quad (5.46)$$

for all $\mathcal{S} \subset \{1, 2, \dots, L\}$ of indices, where $\bar{\mathcal{S}}$ is the complementary set of \mathcal{S} . The achievable rate region for two-level coding is shown in Fig. 5.24. The dot corresponding to the rate pair $(R_1, R_2) = (I(R; B_1), I(R; B_2 | B_1))$ corresponds to multistage decoding in which the first component code is decoded first, while the dot corresponding to the rate pair $(R_1, R_2) = (I(R; B_1 | B_2), I(R; B_2))$ corresponds to multistage decoding in which the other component code is decoded first. The diagram clearly shows that while multistage decoding can achieve two rate pairs that add up to the maximum rate $I(R; X)$, an infinite number of alternative

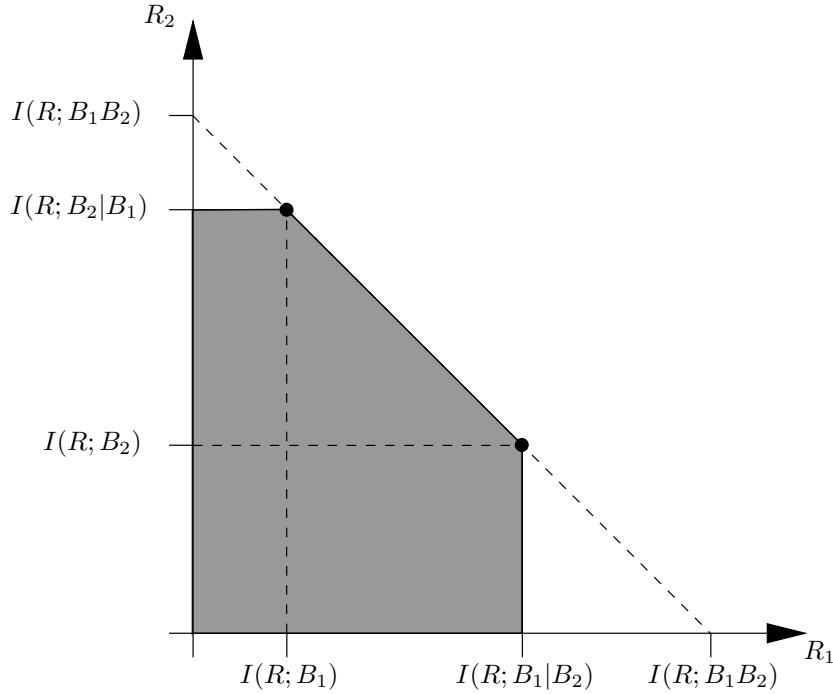


Figure 5.24: Achievable rate region for two-level coding.

rate pairs in between the two dots can also sum up to $I(R; X)$. These other rate pairs can be achieved with multistage decoding via *time sharing*, in which part of the time $(R_1, R_2) = (I(R; B_1), I(R; B_2|B_1))$ and part of the time $(R_1, R_2) = (I(R; B_1|B_2), I(R; B_2))$. Without time sharing though, the first stage of a multistage decoder is always unsuccessful because $R_1 > I(R; B_1)$ and $R_2 > I(R; B_2)$, and these other rate pairs are only achievable with joint ML decoding.

5.2.3 Iterated-Decision Detection and Multistage Decoding

A comparison of Fig. 5.20 with Fig. 3.2 suggests a natural way to combine multistage decoding with iterated-decision detection when dealing with an interference channel rather than an AWGN channel. Figure 5.25 shows a version of the iterated-decision detection in which the symbol-by-symbol slicer has been replaced by the decoder for the l th component code. The decisions from the previous stages or iterations of the receiver, $\hat{\mathbf{q}}_1, \dots, \hat{\mathbf{q}}_{l-1}$, are assumed to be correct and are used to form $\hat{\mathbf{x}}^{l-1}(\hat{\mathbf{q}}_1, \dots, \hat{\mathbf{q}}_{l-1})$, an estimate of \mathbf{x} . In turn, $\hat{\mathbf{x}}^{l-1}(\hat{\mathbf{q}}_1, \dots, \hat{\mathbf{q}}_{l-1})$ is used to form an estimate of the interference to be subtracted from the l th decoder input. For example, if the block partitioning of Fig. 5.17 is used to map the

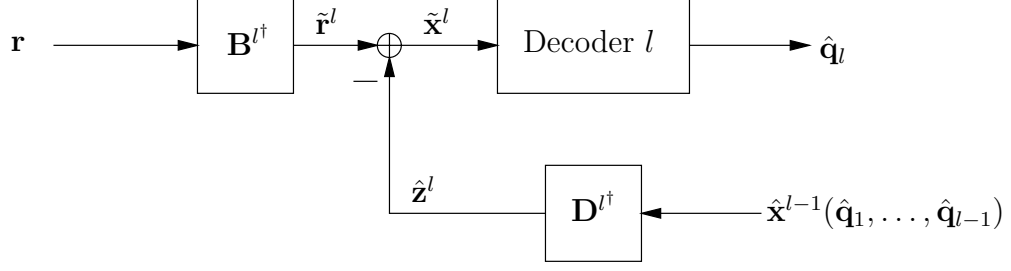


Figure 5.25: Multistage decoding combined with iterated-decision detection.

bits in $\mathbf{q}_1, \dots, \mathbf{q}_L$ to 2^L -PAM symbols in \mathbf{x} , then

$$\hat{\mathbf{x}}^{l-1}(\hat{\mathbf{q}}_1, \dots, \hat{\mathbf{q}}_{l-1}) \propto \sum_{k=1}^{l-1} 2^{L-k} (2\hat{\mathbf{q}}_k - 1). \quad (5.47)$$

Similarly, if the Ungerboeck partitioning of Fig. 5.18 is used, then

$$\hat{\mathbf{x}}^{l-1}(\hat{\mathbf{q}}_1, \dots, \hat{\mathbf{q}}_{l-1}) \propto \sum_{k=1}^{l-1} 2^{k-1} (2\hat{\mathbf{q}}_k - 1). \quad (5.48)$$

We focus on block and Ungerboeck partitioning in this section because the mappers can be expressed as linear combinations of $\mathbf{q}_1, \dots, \mathbf{q}_L$, which is naturally compatible with this version of a combined receiver.

With this definition for $\hat{\mathbf{x}}^{l-1}$, the optimal filters \mathbf{B}^l and \mathbf{D}^l have to be modified slightly. We define the normalized correlation between \mathbf{x} and $\hat{\mathbf{x}}^{l-1}$ as

$$\frac{E[\mathbf{x} \cdot \hat{\mathbf{x}}^{l-1\dagger}]}{\sqrt{\mathcal{E}_s} \sqrt{\mathcal{E}_{s^{l-1}}}} = \boldsymbol{\rho}^{l-1} \triangleq \text{diag}\{\rho_1^{l-1}, \rho_2^{l-1}, \dots, \rho_P^{l-1}\} \quad (5.49)$$

where the energy of the symbols in \mathbf{x} and $\hat{\mathbf{x}}^{l-1}$ are \mathcal{E}_s and $\mathcal{E}_{s^{l-1}}$ respectively. The variance of the effective noise at the slicer input in (3.7) becomes

$$\begin{aligned} & \text{var } u_i^l \\ &= \mathcal{N}_0 \mathbf{b}_i^{l\dagger} \mathbf{b}_i^l + \mathcal{E}_s \left(\mathbf{d}_i^l - \boldsymbol{\rho}^{l-1\dagger} \sqrt{\frac{\mathcal{E}_s}{\mathcal{E}_{s^{l-1}}}} (\mathbf{b}_i^{l\dagger} \mathbf{H} - \mathbf{e}_i^\dagger)^\dagger \right)^\dagger \left(\mathbf{d}_i^l - \boldsymbol{\rho}^{l-1\dagger} \sqrt{\frac{\mathcal{E}_s}{\mathcal{E}_{s^{l-1}}}} (\mathbf{b}_i^{l\dagger} \mathbf{H} - \mathbf{e}_i^\dagger)^\dagger \right) \\ & \quad + \mathcal{E}_s (\mathbf{b}_i^{l\dagger} \mathbf{H} - \mathbf{e}_i^\dagger) (\mathbf{I} - \boldsymbol{\rho}^{l-1} \boldsymbol{\rho}^{l-1\dagger}) (\mathbf{b}_i^{l\dagger} \mathbf{H} - \mathbf{e}_i^\dagger)^\dagger. \end{aligned} \quad (5.50)$$

The expression for the optimal filter \mathbf{D}^l is thus

$$\begin{aligned}
\mathbf{D}^l &= \boldsymbol{\rho}^{l-1\dagger} \sqrt{\frac{\mathcal{E}_s}{\mathcal{E}_{\hat{s}^{l-1}}}} \left(\mathbf{B}^{l\dagger} \mathbf{H} - \mathbf{I} \right)^\dagger \\
&= \frac{E[\mathbf{x} \cdot \hat{\mathbf{x}}^{l-1\dagger}]}{\mathcal{E}_{\hat{s}^{l-1}}} \left(\mathbf{B}^{l\dagger} \mathbf{H} - \mathbf{I} \right)^\dagger \\
&= \left(\frac{E[(\mathbf{x} - \hat{\mathbf{x}}^{l-1}) \cdot \hat{\mathbf{x}}^{l-1\dagger}]}{\mathcal{E}_{\hat{s}^{l-1}}} + \mathbf{I} \right) \left(\mathbf{B}^{l\dagger} \mathbf{H} - \mathbf{I} \right)^\dagger \\
&= \left(\mathbf{B}^{l\dagger} \mathbf{H} - \mathbf{I} \right)^\dagger
\end{aligned} \tag{5.51}$$

where we have used (5.49) and the fact that the correlation between the error $\mathbf{x} - \hat{\mathbf{x}}^{l-1}$ and $\hat{\mathbf{x}}^{l-1}$ is zero for both block and Ungerboeck partitioning, the reason being that $\mathbf{x} - \hat{\mathbf{x}}^{l-1}$ is a function of $\mathbf{q}_l, \dots, \mathbf{q}_L$ and $\hat{\mathbf{x}}^{l-1}$ is a function of $\mathbf{q}_1, \dots, \mathbf{q}_{l-1}$. The expression (3.23) for the optimal filter \mathbf{B}^l remains the same, but the expression $\mathcal{E}_s(\mathbf{I} - \boldsymbol{\rho}^{l-1}\boldsymbol{\rho}^{l-1\dagger})$ can be simplified to

$$\begin{aligned}
\mathcal{E}_s(\mathbf{I} - \boldsymbol{\rho}^{l-1}\boldsymbol{\rho}^{l-1\dagger}) &= \mathcal{E}_s \left(\mathbf{I} - \frac{E[\mathbf{x} \cdot \hat{\mathbf{x}}^{l-1\dagger}](E[\mathbf{x} \cdot \hat{\mathbf{x}}^{l-1\dagger}])^\dagger}{\mathcal{E}_s \mathcal{E}_{\hat{s}^{l-1}}} \right) \\
&= \mathcal{E}_s \mathbf{I} - \frac{(E[(\mathbf{x} - \hat{\mathbf{x}}^{l-1}) \cdot \hat{\mathbf{x}}^{l-1\dagger}] + \mathcal{E}_{\hat{s}^{l-1}} \mathbf{I})(E[(\mathbf{x} - \hat{\mathbf{x}}^{l-1}) \cdot \hat{\mathbf{x}}^{l-1\dagger}] + \mathcal{E}_{\hat{s}^{l-1}} \mathbf{I})^\dagger}{\mathcal{E}_{\hat{s}^{l-1}}} \\
&= (\mathcal{E}_s - \mathcal{E}_{\hat{s}^{l-1}}) \mathbf{I}
\end{aligned} \tag{5.52}$$

where again we have used (5.49) and the fact that the correlation between the error $\mathbf{x} - \hat{\mathbf{x}}^{l-1}$ and $\hat{\mathbf{x}}^{l-1}$ is zero for both block and Ungerboeck partitioning.

These modifications make intuitive sense under the assumption that decoder l perfectly decodes \mathbf{q}_l . With perfect knowledge of $\mathbf{q}_1, \dots, \mathbf{q}_{l-1}$ at the receiver, \mathbf{D}^l creates a perfect estimate of the interference caused by $\mathbf{q}_1, \dots, \mathbf{q}_{l-1}$. With the interference subtracted off, it is then as though the transmit vector \mathbf{x} is a function only of $\mathbf{q}_l, \dots, \mathbf{q}_L$; i.e., for block partitioning

$$\mathbf{x} \propto \sum_{k=l}^L 2^{L-k} (2\mathbf{q}_k - 1) \tag{5.53}$$

and for Ungerboeck partitioning

$$\mathbf{x} \propto \sum_{k=l}^L 2^{k-1} (2\mathbf{q}_k - 1). \tag{5.54}$$

The energy of the symbols in \mathbf{x} is then $\mathcal{E}_s - \mathcal{E}_{\hat{s}^{l-1}}$, where the value of $\mathcal{E}_{\hat{s}^{l-1}}$ depends on which partitioning scheme is used. An iteration of the combined multistage decoder and iterated-decision detector in Fig. 5.25 is thus equivalent to the unbiased MMSE linear detector \mathbf{B}^\dagger in cascade with the l th decoder.

We now investigate the rate achievable with this detection/decoding scheme. From Theorem 3.1, we know that the effective noise process at the input to the l th decoder is marginally Gaussian. Assuming all symbols in \mathcal{X} are equiprobable, the achievable rate of the l th iteration can be computed as

$$\begin{aligned} I(\tilde{X}; B_l | B_1, B_2, \dots, B_{l-1}) &= E_{b_1 \dots b_{l-1}} \left[I(\tilde{X}; B_l | b_1, b_2, \dots, b_{l-1}) \right] \\ &= \frac{1}{2^{l-1}} \sum_{b_1 \dots b_{l-1}} I(\tilde{X}; B_l | b_1, b_2, \dots, b_{l-1}). \end{aligned} \quad (5.55)$$

Each term in the sum can in turn be computed as [43]

$$\begin{aligned} &I(\tilde{X}; B_l | b_1, b_2, \dots, b_{l-1}) \\ &= \frac{1}{2} \int_{-\infty}^{\infty} f(\tilde{x} | b_1 \dots b_{l-1} 0) \log \frac{f(\tilde{x} | b_1 \dots b_{l-1} 0)}{\frac{1}{2}f(\tilde{x} | b_1 \dots b_{l-1} 0) + \frac{1}{2}f(\tilde{x} | b_1 \dots b_{l-1} 1)} d\tilde{x} \\ &\quad + \frac{1}{2} \int_{-\infty}^{\infty} f(\tilde{x} | b_1 \dots b_{l-1} 1) \log \frac{f(\tilde{x} | b_1 \dots b_{l-1} 1)}{\frac{1}{2}f(\tilde{x} | b_1 \dots b_{l-1} 0) + \frac{1}{2}f(\tilde{x} | b_1 \dots b_{l-1} 1)} d\tilde{x} \end{aligned} \quad (5.56)$$

where

$$f(\tilde{x} | b_1 \dots b_l) = \frac{1}{2^{L-l}} \sum_{x \in \mathcal{X}(b_1 \dots b_l)} f(\tilde{x} | x). \quad (5.57)$$

To gain insight into the rates achievable using the combined multistage decoder and iterated-decision detector, we look at a couple of representative channels. These examples demonstrate that the combination of multistage decoding and iterated-decision detection at the receiver does not necessarily achieve the maximum possible information rate for all channels and, in fact, may be quite far from the maximum rate.

We first study the case of the random ISI channel, for which the variance of the effective noise process at the decoder input is given by (3.102). In Fig. 5.26, we plot the theoretical rates supported by each iteration of the combined multistage decoder and iterated-decision equalizer for an 8-PAM constellation addressed using block partitioning. Figure 5.27 shows the rates achievable using PAM constellations of different sizes, corresponding to codes with different numbers of levels. From the figure, the SNR gap between the achievable rate and

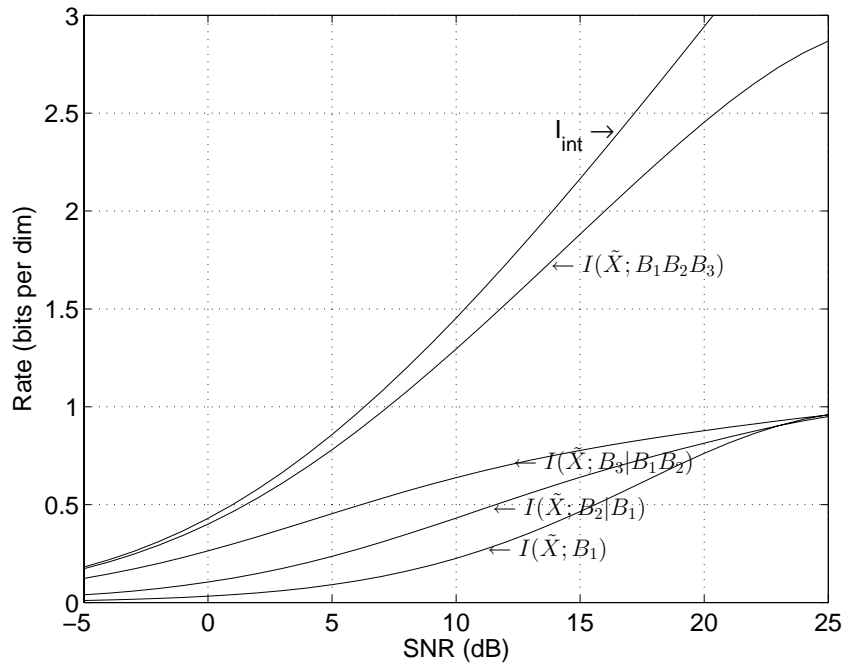


Figure 5.26: Information rates supported by each iteration of the combined multistage decoder and iterated-decision equalizer for a three-level code (8-PAM) with block partitioning over the limiting random ISI channel.

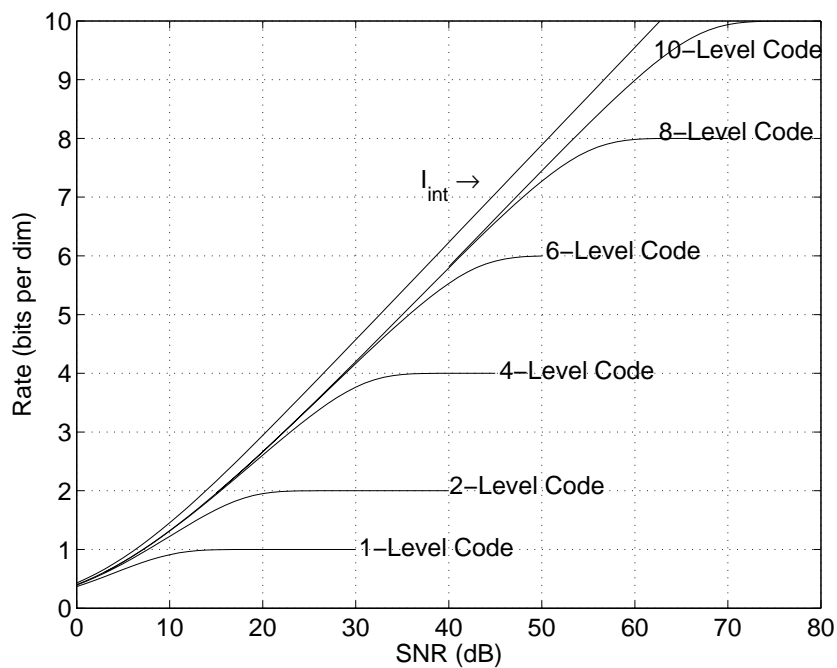


Figure 5.27: Information rates supported by the combined multistage decoder and iterated-decision equalizer for block partitioning over the limiting random ISI channel.

the mutual information of the ISI channel (with no water pouring at the transmitter) is approximately 2.8 dB. Although 1.53 dB of the gap can be attributed to shaping loss due to the use of a discrete constellation rather than a continuous Gaussian distribution [26], the remaining 1.3 dB indicates that the combined multistage decoder and iterated-decision equalizer with block partitioning does not achieve the maximum information rate possible over the random ISI channel. The reason why the combination of multistage decoding and iterated-decision detection in Fig. 5.25 does not asymptotically get to within 1.53 dB of I_{int} is as follows. As mentioned earlier, an iteration of the combined receiver is equivalent to an unbiased MMSE linear detector in cascade with the l th decoder. However, with an MMSE linear detector, an excessive amount of residual interference is treated as noise by the decoder, so information in the structured interference is lost.

We also study the corresponding rates for the random ISI channel with Ungerboeck partitioning in Figs. 5.28 and 5.29, and an interesting observation can be made. Unlike the AWGN channel case in which $I(R; B_1 B_2 B_3)$ is equal for both block partitioning and Ungerboeck partitioning, in the ISI channel case a comparison of Figs. 5.27 and 5.29 reveals that $I(\tilde{X}; B_1 B_2 B_3)$ is not equal. In fact, the SNR gap between the achievable rate of the proposed scheme and the mutual information of the random ISI channel is at least 10 dB for Ungerboeck partitioning, making it significantly worse than block partitioning. Ungerboeck and block partitioning differ in the order the component codes are decoded, which affects the achievable rate. Block partitioning treats the less significant bits as interference and decodes the most significant bits first, while Ungerboeck partitioning treats the most significant bits as interference and decodes the least significant bits first. Thus, Ungerboeck partitioning loses more information since more interference is treated as unstructured noise by the decoder than with block partitioning.

We next look at the case of the square i.i.d. Gaussian matrix channel, for which the variance of the effective noise process at the decoder input is given by (3.58). Figure 5.30 shows the rates achievable using block partitioning with PAM constellations of different sizes, corresponding to codes with different numbers of levels. The SNR gap between the achievable rate and the mutual information of the channel is approximately 7–8 dB, which reiterates the fact that the combined multistage decoder and iterated-decision detector generally does not achieve the maximum possible rate for a given channel.

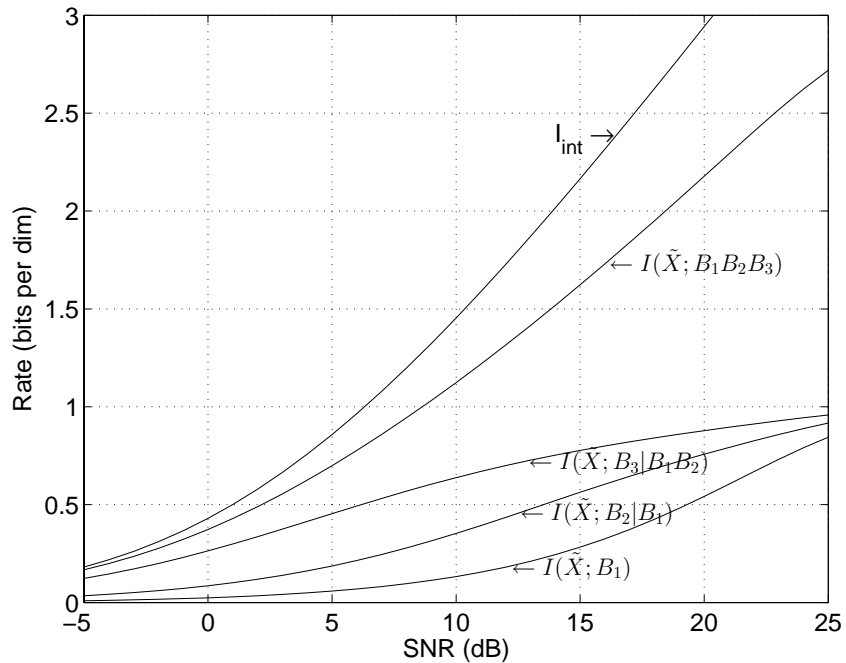


Figure 5.28: Information rates supported by each iteration of the combined multistage decoder and iterated-decision equalizer for a three-level code (8-PAM) with Ungerboeck partitioning over the limiting random ISI channel.

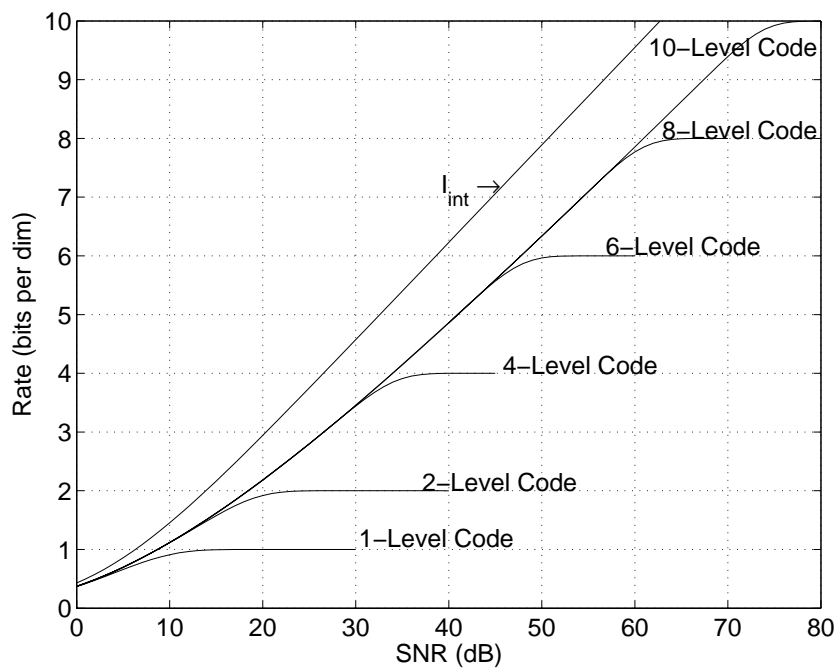


Figure 5.29: Information rates supported by the combined multistage decoder and iterated-decision equalizer for Ungerboeck partitioning over the limiting random ISI channel.

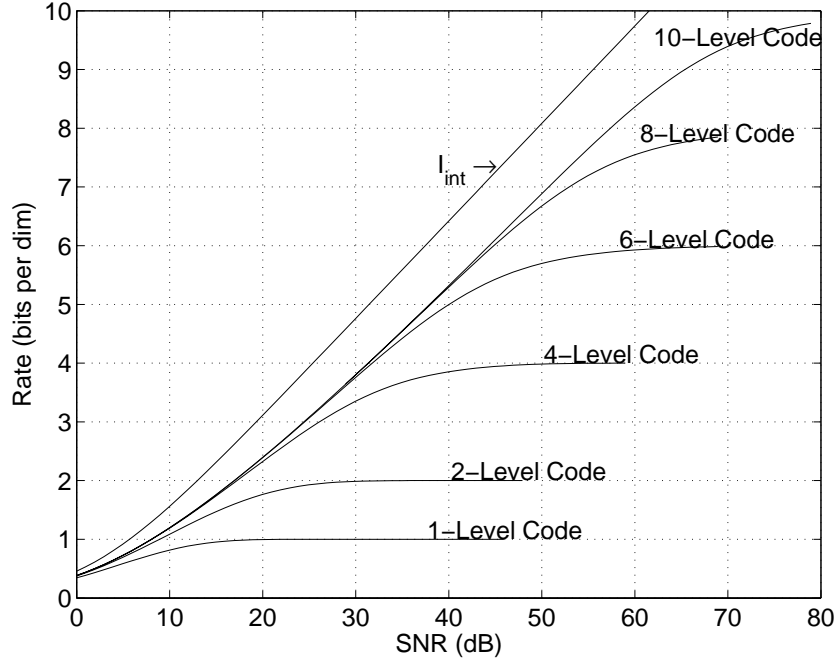


Figure 5.30: Information rates supported by the combined multistage decoder and iterated-decision detector for block partitioning over the limiting i.i.d. Gaussian matrix channel with $\beta = N/Q = 1$.

5.2.4 A Class of Iterative Detection/Decoding Algorithms

The previous section showed that rates close to the mutual information of an interference channel can be achieved with multilevel codes and the combined multistage decoder and iterated-decision detector. Since this combined receiver achieves high rates by simply decoding each component code exactly once and in sequence, even higher rates may be attainable with joint ML detection and decoding.

As an extension to Section 5.1, consider the *a posteriori* joint probability mass function for \mathbf{x} given the fixed observation \mathbf{r} :

$$\begin{aligned}
 p(\mathbf{x}|\mathbf{r}) &\propto p(\mathbf{x})f(\mathbf{r}|\mathbf{x}) \\
 &\propto [\mathbf{x} \in \mathcal{C}]f(\mathbf{r}|\mathbf{x}) \\
 &= [\mathbf{q}_1 \in \mathcal{C}_1][\mathbf{q}_2 \in \mathcal{C}_2] \cdots [\mathbf{q}_L \in \mathcal{C}_L]f(\mathbf{r}|\mathbf{q}_1\mathbf{q}_2 \cdots \mathbf{q}_L)
 \end{aligned} \tag{5.58}$$

where \mathcal{C} is the multilevel code and \mathcal{C}_l is the l th component code. Thus, the corresponding factor graph consists of L component code subgraphs and an interference subgraph, con-

nected together via L sets of binary variable nodes, one for each component code. This factor graph defines a class of message-passing algorithms that improve upon the combined multistage decoder and iterated-decision detector of the previous section by allowing bits in all coding levels to be iteratively decoded, rather than decoded only once in sequence. Since the graph has cycles, the class of algorithms approximate joint optimal detection and decoding. Although an infinite number of message-passing schedules are possible, we restrict attention to ones in which message-passing is essentially limited to within the subgraphs, with information exchanged periodically between the interference subgraph and the L code subgraphs.

To implement this message-passing detector and decoder, we can use the iterated-decision detector to implement message passing on the interference subgraph and L standard message passing decoders to implement message passing on the code subgraphs.

When extrinsic L -values for all the bits are passed from the decoders to the iterated-decision detector, the L -values can be summarized into a single parameter ρ and a set of symbol decisions $\hat{\mathbf{x}}$. The symbol decisions can be made by making hard decisions $\hat{b}_1, \hat{b}_2, \dots, \hat{b}_L$ on the bits b_1, b_2, \dots, b_L of each symbol in the multilevel code. The computation of ρ , however, is less straightforward. To compute ρ , we compute the individual symbol reliability for every symbol and then take the average over all symbols. A variation of the individual symbol reliability expression (4.56) is

$$\rho(L(b_1), L(b_2), \dots, L(b_L)) = \frac{E[\mathcal{X}(b_1, b_2, \dots, b_L) | L(b_1), L(b_2), \dots, L(b_L)] \mathcal{X}(\hat{b}_1, \hat{b}_2, \dots, \hat{b}_L)}{\mathcal{E}_s}, \quad (5.59)$$

where $L(b_l)$ denotes the extrinsic L -value of the bit b_l from the l th decoder, and \mathcal{E}_s is the average symbol energy of the multilevel symbols. For the special case of block partitioning, the Bayes' least squares estimate of $\mathcal{X}(b_1, b_2, \dots, b_L)$ given $L(b_1), L(b_2), \dots, L(b_L)$ is

$$\begin{aligned} & E[\mathcal{X}(b_1, b_2, \dots, b_L) | L(b_1), L(b_2), \dots, L(b_L)] \\ &= E \left[\sum_{l=1}^L 2^{L-l} b_l | L(b_1), L(b_2), \dots, L(b_L) \right] \\ &= \sum_{l=1}^L 2^{L-l} E[b_l | L(b_1), L(b_2), \dots, L(b_L)] \\ &= \sum_{l=1}^L 2^{L-l} E[b_l | L(b_l)]. \end{aligned} \quad (5.60)$$

If the L -value distributions are approximated as Gaussian as in Section 5.1.3, then we can use (4.35) and (4.37) to write

$$\begin{aligned} & E [\mathcal{X}(b_1, b_2, \dots, b_L) | L(b_1), L(b_2), \dots, L(b_L)] \\ &= \sum_{l=1}^L 2^{L-l} \left| \tanh \left(\frac{L(b_l)}{2} \right) \right| \text{sgn}(L(b_l)). \end{aligned} \quad (5.61)$$

Substituting back into (5.59), we have

$$\begin{aligned} & \rho(L(b_1), L(b_2), \dots, L(b_L)) \\ &= \frac{\left(\sum_{l=1}^L 2^{L-l} \left| \tanh \left(\frac{L(b_l)}{2} \right) \right| \text{sgn}(L(b_l)) \right) \mathcal{X}(\hat{b}_1, \hat{b}_2, \dots, \hat{b}_L)}{\mathcal{E}_s} \\ &= \frac{\left(\sum_{l=1}^L 2^{L-l} \left| \tanh \left(\frac{L(b_l)}{2} \right) \right| \text{sgn}(L(b_l)) \right) \left(\sum_{l=1}^L 2^{L-l} \text{sgn}(L(b_l)) \right)}{\mathcal{E}_s}. \end{aligned} \quad (5.62)$$

Similarly, for Ungerboeck partitioning

$$\begin{aligned} & \rho(L(b_1), L(b_2), \dots, L(b_L)) \\ &= \frac{\left(\sum_{l=1}^L 2^{l-1} \left| \tanh \left(\frac{L(b_l)}{2} \right) \right| \text{sgn}(L(b_l)) \right) \left(\sum_{l=1}^L 2^{l-1} \text{sgn}(L(b_l)) \right)}{\mathcal{E}_s}. \end{aligned} \quad (5.63)$$

By taking the average of all the symbol reliabilities $\rho(L(b_1), L(b_2), \dots, L(b_L))$, we obtain the parameter ρ used by the iterated-decision detector.

The slicer of the iterated-decision detector is eliminated as in Section 5.1, and extrinsic information for all the bits in the form of the vector $\tilde{\mathbf{x}}$, which would otherwise have been the slicer input, is passed from the iterated-decision detector to the decoders. There are a number of different ways in which the decoders can use the extrinsic information $\tilde{\mathbf{x}}$, and it is currently unclear without further research the best strategy for the decoders to exploit knowledge of $\tilde{\mathbf{x}}$. We conclude this section by outlining some of the possible strategies.

From the Theorems 3.1 and 3.2, the vector $\tilde{\mathbf{x}}$ can be modelled as the output of an AWGN channel with mode-interleaved precoding. Thus, one simple strategy is to treat $\tilde{\mathbf{x}}$ as if it were the output of an AWGN channel and to perform multistage decoding as described in Section 5.2.2. While the l th decoder depends only on the hard decisions $\hat{b}_1 = \text{sgn}(L(b_1)), \dots, \hat{b}_{l-1} = \text{sgn}(L(b_{l-1}))$ of the lower-level decoders, the decoders pass to the iterated-decision detector the soft L -values $L(b_1), \dots, L(b_L)$ for each symbol. While

a simple strategy, the fact that the decoders exchange only hard information amongst themselves suggests that performance can be improved by exchanging soft information instead. Furthermore, incorporating incorrect hard decision from lower-level decoders can cause errors to propagate in higher-level decoders, possibly leading to divergence of the overall detection and decoding scheme. A more sophisticated decoding strategy would have the l th decoder depend on the soft decisions $L(b_1), \dots, L(b_{l-1})$ of the lower-level decoders rather than hard decisions, but it is not clear how such a decoder would incorporate the soft decisions of other decoders into its own message-passing algorithm.

Rather than have the decoders depend sequentially on each other, each decoder can process the vector $\tilde{\mathbf{x}}$ from the iterated-decision detector independently. For example, modelling each symbol \tilde{x} as the output of an AWGN channel, we can compute the L -value for b_l based on \tilde{x} as

$$\begin{aligned}
L(b_l|\tilde{x}) &= \log \frac{\Pr(b_l = +1|\tilde{x})}{\Pr(b_l = -1|\tilde{x})} \\
&= \log \frac{f(\tilde{x}|b_l = +1) \Pr(b_l = +1)}{f(\tilde{x}|b_l = -1) \Pr(b_l = -1)} \\
&= \log \frac{f(\tilde{x}|b_l = +1)}{f(\tilde{x}|b_l = -1)} \\
&= \log \frac{\sum_{x \in \mathcal{X}(b_1 \dots b_{l-1} 1 b_{l+1} \dots b_L)} f(\tilde{x}|x)}{\sum_{x \in \mathcal{X}(b_1 \dots b_{l-1} 0 b_{l+1} \dots b_L)} f(\tilde{x}|x)}. \tag{5.64}
\end{aligned}$$

Computing these L -values directly from $\tilde{\mathbf{x}}$, each decoder can operate independently. Extrinsic information from each decoder can then be passed back to the iterated-decision detector for another round of processing. Since each decoder receives information implicitly through $\tilde{\mathbf{x}}$ rather than directly from other decoders, performance can be improved by exchanging soft information directly amongst the decoders.

Ideally, decoders that can incorporate knowledge of the vector $\tilde{\mathbf{x}}$ from the iterated-decision detector and soft information from *all* the other decoders would be very useful. Message passing on the interference and component code subgraphs of the joint factor graph could then be implemented by the iterated-decision detector and such decoders, and optimal joint detection and decoding would be well-approximated. Moreover, the detection and decoding schedule would be very flexible, since different components could operate in parallel or sequentially in any order. The challenge remains, though, of developing such decoders that operate with good performance and low complexity.

5.3 Lattice Codes

An alternate coding strategy for channels that support high rates is to use lattice codes. Unlike the random codes discussed in Section 1.2, lattice codes are highly structured and are thus of great practical interest. Moreover, it has been shown that lattice codes can achieve the capacity of the AWGN channel [23, 63].

An N -dimensional lattice Λ is a discrete infinite subset of \mathbb{R}^N that satisfying the group property. If the lattice spans \mathbb{R}^N , then it can be described by a set of linearly independent generators such that the lattice consists of the set of all integer linear combinations of the generators:

$$\Lambda = \{\mathbf{G}\mathbf{y} = \sum_i y_i \mathbf{g}_i | \mathbf{y} \in \mathbb{Z}^n\}. \quad (5.65)$$

Because of the group property, the lattice is “geometrically uniform,” meaning that each point of the lattice has the same number of neighbors at each distance, and all decision regions of a minimum-distance decoder (Voronoi regions) are congruent and tessellate \mathbb{R}^N . In fact, any lattice translate $\Lambda + \mathbf{t}$ is also geometrically uniform.

A lattice code \mathcal{C} is the finite set of points of the lattice translate $\Lambda + \mathbf{t}$ that lie within a compact bounding region $\mathcal{R} \in \mathbb{R}^N$, i.e.,

$$\mathcal{C} = (\Lambda + \mathbf{t}) \cap \mathcal{R}. \quad (5.66)$$

5.3.1 Decoding for AWGN Channels

There are two main kinds of decoders for lattice codes over AWGN channels. *Minimum-distance decoding* minimizes the probability of error by finding the nearest point of \mathcal{C} to the received signal. In [63], it was shown that lattice codes can be used to achieve the capacity of the AWGN channel with minimum-distance decoding. *Lattice decoding* takes full advantage of the underlying lattice structure and decodes the received signal to the nearest point of $\Lambda + \mathbf{t}$, whether or not it lies within the bounding region \mathcal{R} . Though considerably simpler than minimum-distance decoding, it was shown in [23] that lattice decoding could also be used to achieve the capacity of the AWGN channel.

5.3.2 Generalized Iterated-Decision Detection

The iterated-decision detector in Section 3.1 for uncoded systems can be generalized in a natural way to accommodate lattice-coded systems. The modified system, depicted in Fig. 5.31, uses either a minimum-distance decoder or lattice decoder instead of a symbol-by-symbol slicer. The optimal filters (3.78) and (3.82) remain the same, but the algorithm in Section 3.1.4 to compute the set of correlation matrices ρ^l changes. With mode-interleaved precoding, the SINRs γ_i^l for $i = 1, 2, \dots, N$ are asymptotically equal, so the algorithm becomes:

1. Set $\rho^0 = 0$ and let $l = 1$.
2. Compute the SINR γ^l at the slicer input on the l th decoding pass from $\rho^{l-1} = \rho^{l-1}\mathbf{I}$ via (3.26) and (3.27).
3. Approximate the lattice codeword error probability $\Pr(\epsilon^l)$ at the slicer output from γ^l using the union bound estimate for the codeword error rate of either a minimum-distance decoder or lattice decoder for AWGN channels [28]:

$$\Pr(\epsilon^l) \approx K_{\min}(\Lambda) \mathcal{Q} \left(\sqrt{\frac{d_{\min}^2(\Lambda)}{4(\sigma^l)^2}} \right) \quad (5.67)$$

where $d_{\min}^2(\Lambda)$ is the squared minimum distance in the lattice Λ , $K_{\min}(\Lambda)$ is the number of neighbors of each point at the minimum distance, $(\sigma^l)^2$ is the effective noise variance in one dimension computed as $\mathcal{E}_s/2\gamma^l$, and $\mathcal{Q}(v)$ is defined in (2.5).

4. Approximate ρ^l from $\Pr(\epsilon^l)$. From Appendix E,

$$\rho^l \approx 1 - \frac{d_{\min}^2(\Lambda)}{2\mathcal{E}_c} \Pr(\epsilon^l), \quad (5.68)$$

where \mathcal{E}_c is the average codeword energy.

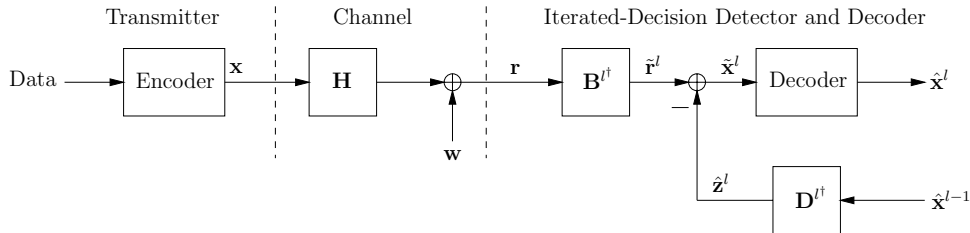


Figure 5.31: Iterated-decision detection and decoding for lattice codes.

5. Increment l and go to step 2.

We make some remarks on the algorithm. First, the approximation for ρ^l in (5.68) is valid whether the coding is done across components of the same transmit vector, or across components of successive transmit vectors. Second, the union bound estimate in (5.67) is valid at high SNR. For lattice codes, the required SNR may be unrealistically high. Third, the iterated-decision detector in Section 3.1 achieves the AWGN channel bound at high SNR for uncoded systems with mode-interleaved precoding. An extension of this uncoded result to lattice codes suggests that the modified iterated-decision detector can potentially achieve the AWGN channel bound at high SNR (high compared to the Shannon limit) for lattice-coded systems with mode-interleaved precoding. Of course, it is impossible for the modified iterated-decision detector to achieve the AWGN channel bound at *all* SNRs because the mutual information of an interference channel without water pouring is always less than the capacity of the AWGN channel at the same SNR.

More research is required at this point to investigate both the fundamental limits and practical implementation of this promising generalization of the iterated-decision detector for lattice codes.

Chapter 6

Conclusions

This final chapter summarizes the contributions of this thesis and outlines some directions for future work.

6.1 Contributions

In this thesis, we have proposed and analyzed low-complexity strategies to approach the information-theoretic limits of an interference channel when channel state information is unavailable at the transmitter. These strategies have centered around the iterated-decision detector, which cancels channel interference from received signals using tentative decisions whose reliability improves with each successive iteration.

The iterated-decision detector works particularly well when used in conjunction with mode-interleaved precoding at the transmitter. Mode-interleaved precoding, discussed in Chapter 2, is channel-independent transmitter precoding that randomizes the channel in such a way that there are two distinct benefits. First, the pairwise error probability of two transmit vectors over an interference channel becomes asymptotically the same as the pairwise error probability of the same two vectors over an AWGN channel at the same SNR. While this result does not necessarily imply that the overall probability of error of ML detection for an interference channel is asymptotically the same as the probability of error for the corresponding AWGN channel, there are some cases in which the performance of ML detection for an interference channel does indeed resemble performance for an AWGN channel. For example, ML detection with mode-interleaved precoding can achieve the AWGN channel bound at high SNR. Second, the use of mode-interleaved precoding

justifies Gaussian approximations for the effective noise made at the slicer input of the iterated-decision detector and enables the asymptotic performance analyses in Chapter 3. These analyses show that with mode-interleaved precoding, the iterated-decision detector well-approximates ML detection for uncoded systems at high SNR, and thus achieves the AWGN bound at high SNR. The excellent performance of iterated-decision detection with mode-interleaved precoding is not surprising in light of the analysis in Chapter 4, which suggests that iterated-decision detection is a simplified version of an approximate message-passing ML detection algorithm. We also discussed in Chapter 2 the implementation of mode-interleaved precoding for a variety of communication channels that include:

- MIMO channels with many transmit and receive antennas with a rich scattering assumption
- Other MIMO channels with many transmit and receive antennas
- MIMO channels with few transmit and receive antennas
- CDMA systems with many users and long user signatures that are or can be modelled as i.i.d. Gaussian
- Other CDMA systems with many users and long user signatures
- CDMA systems with few users and/or short signature sequences
- ISI channels with impulse responses whose taps can be modelled as i.i.d. random variables
- ISI channels with short and/or correlated impulse responses

To achieve reliable communication at rates up to the mutual information of an interference channel, channel coding is required. In Chapter 5, we discussed ways in which iterated-decision detection and mode-interleaved precoding could be incorporated into coded systems. The main challenge is the design of a low-complexity receiver that approximates optimal joint detection and decoding as closely as possible. Although binary codes, multi-level codes, and lattice codes were discussed separately, the common feature of the proposed receivers was the exchange of information between a separate detection block and one or more decoding blocks. The iterative exchange of information between blocks can be viewed as an approach to approximate optimal joint detection and decoding. The iterated-decision detector was modified to implement the detection block because of its low complexity, excellent performance, and ability to accept reliability information from external sources.

6.2 Future Work

Although in earlier chapters we have touched upon possibilities for more research, we now discuss some of the potentially more fruitful research directions inspired by this thesis.

6.2.1 Low-Bandwidth Message Passing

It is intriguing that the iterated-decision detector, interpreted as a low-bandwidth message-passing algorithm in Section 4.4, retains only the most significant bit plus the average of the individual symbol reliabilities and yet approximates ML detector performance. This observation raises a number of interesting questions about message-passing algorithms more generally. For example, standard message-passing decoding algorithms on graphs with large cycles may be operating with an unnecessary amount of precision. If low-bandwidth message passing suffices to achieve essentially the same level of performance as high-precision message passing, then low-bandwidth message passing is an attractive option. This is especially true in applications where memory to store symbol reliabilities is limited, or where the message-passing algorithm is distributed and communication between nodes is costly.

6.2.2 Analogy to Price's Result

In [50], Price shows that if the ideal¹ zero-forcing decision-feedback equalizer (ZF-DFE) is theoretically used at high SNR, then the SNR gap to capacity is independent of the interference. In particular, the SNR gap to capacity is the same for channels with *and* without interference. In [16], Price's result is generalized to the ideal minimum mean-square error decision-feedback equalizer (MMSE-DFE) at any SNR. For example, this generalization is consistent with Figs. 3.14 and C.2 for random ISI channels. Figure 3.14 shows that the SNR of the random ISI channel with the MMSE-DFE must asymptotically be 2.507 dB higher than the SNR of an AWGN channel for the uncoded bit-error rate to be the same, while Fig. C.2 shows that minimum SNR of the random ISI channel required for reliable transmission is also asymptotically 2.507 dB higher than the minimum SNR of the AWGN channel. Thus, the asymptotic SNR gap to capacity is the same for both the random ISI channel with the MMSE-DFE and the AWGN channel. Furthermore, there is a claim in [16] that if a coding method can achieve a certain coding gain over an AWGN

¹An ideal DFE always feeds back correct decisions.

channel, then that same gain can be achieved using an adaptation of that method with an ideal MMSE-DFE over an ISI channel.

These results are meaningful only if the performance of an ideal MMSE-DFE can be obtained. If the channel is known at the transmitter, then the feedback filter of the DFE can be implemented using Tomlinson-Harashima precoding [59, 38] at the transmitter, leading to ideal MMSE-DFE performance. If the channel is not known at the transmitter, the ideal assumption is invalid and the results of [50, 16] are not applicable.

In Section 5.1.5, we showed that the pinch-off SNR_{norm} for the asymptotic random ISI channel along with a (5, 10)-regular LDPC code is 2.03 dB when the iterated-decision detector is used with a standard sum-product decoder, with the SNR normalized to the mutual information of the interference channel without water pouring. In [18], the pinch-off SNR_{norm} for an AWGN channel with a (5, 10)-regular LDPC code is 2.01 dB when a standard sum-product decoder is used, with the SNR normalized to the capacity of the AWGN channel. The fact that these two quantities are very close numerically suggests that a result analogous to Price's result may be true for the iterated-decision detector. Specifically, it is possible that codes designed to get close to AWGN channel capacity can also be used with iterated-decision detection to get equally as close to the mutual information of an interference channel without water pouring. Such a result would suggest that the iterated-decision detector could be considered a canonical detector in systems with channel interference and coding.

6.2.3 Iterated-Decision Transmitter Precoding

As mentioned in the last section, if the channel is known at the transmitter then the feedback filter of the decision-feedback detector can be implemented at the transmitter [14, 38, 59], leading to ideal MMSE decision-feedback performance. However, even the ideal MMSE decision-feedback detector does not necessarily give the best performance, as evidenced by Fig. 3.14 where the SNR gap between the curves for the theoretical (ideal) MMSE-DFE and the iterated-decision detector is asymptotically 2.507 dB. This observation motivates the development of an iterated-decision transmitter precoder that implements part or all of the iterated-decision detector at the transmitter. It is not clear at this point what the correct structure would be for the iterated-decision transmitter precoder. One possibility is to block-iterate the symbols to be transmitted over the channel until the average energy

of the symbols falls within some power constraint. Each iteration would involve some kind of noncausal filtering of the “tentative” transmit symbols from the previous iteration, and possibly a modulo function as in [38, 59].

6.2.4 Iterated-Decision Lattice Decoding for AWGN Channels

The decoding of arbitrary lattice codes can be very complex [67], but it may be possible to modify the iterated-decision detector as a low-complexity lattice decoder. The transmission of a lattice codeword over an AWGN channel can be modelled with the equation $\mathbf{r} = \mathbf{H}\mathbf{x} + \mathbf{w}$, where \mathbf{r} is the received vector, the columns of \mathbf{H} are the lattice generators, \mathbf{x} is the coordinate vector, and \mathbf{w} is the noise vector. Given \mathbf{r} and \mathbf{H} , the objective is to determine the coordinate vector $\hat{\mathbf{x}}$ that minimizes $\|\mathbf{r} - \mathbf{H}\hat{\mathbf{x}}\|$. This problem appears very similar to the detection problem, but the use of a coordinate-by-coordinate decision slicer to determine $\hat{\mathbf{x}}$ may lead to lattice points that lie far outside the bounding region of the lattice code. To be useful, the iterated-decision detector must be modified somehow to take into account arbitrary bounding regions without significantly increasing the complexity.

6.2.5 Iterated-Decision Quantizer

The problem of quantization [36] is also closely related to the problem of detection—finding the nearest point to the source signal in quantization is similar to finding the nearest point to the received signal in detection. In particular, if the reconstruction points of the quantizer form a lattice, then we have a situation that can be modelled with the equation $\mathbf{r} = \mathbf{H}\mathbf{x} + \mathbf{w}$, where \mathbf{r} is the source data, the columns of \mathbf{H} are the lattice generators of the quantizer, \mathbf{x} is the coordinate vector, and \mathbf{w} is the error between the source data \mathbf{r} and the reconstruction point $\mathbf{H}\mathbf{x}$. Given \mathbf{r} and \mathbf{H} , the objective is to determine a coordinate vector \mathbf{x} that minimizes some metric involving the error. If the metric is the squared norm of \mathbf{w} , then we indeed get a problem that is very similar to a detection problem in Gaussian noise.

However, there are some key differences that make the application of the iterated-decision detector to the quantization problem not so straightforward. First, the source data is not necessarily concentrated around the reconstruction points. In some applications, the source data may be uniformly distributed over some region in \mathbb{R}^N , making \mathbf{w} uniformly distributed within the Voronoi region of the zero lattice point. The iterated-decision detector, which performs well at high SNR (i.e., when \mathbf{w} is concentrated near the

zero lattice point), may not lead to the same quality of performance when applied to the quantization problem. In general, the distribution of the source must also be taken into account. Second, as with iterated-decision lattice decoding for AWGN channels, the use of a coordinate-by-coordinate decision slicer may lead to lattice points that lie far outside the bounding region.

6.2.6 Approximate Solutions to Complex Problems

Although exact ML detection with mode-interleaved precoding is intractably complex, an approximation to the solution can be obtained using the iterated-decision detector, as shown in Chapter 4. This approximation, though not the exact solution, is asymptotically as useful as the exact solution if bit-error rate is the primary concern. With other problems such as lattice decoding for AWGN channels or quantization whose exact solutions may also be very difficult to obtain, there may also exist approximate solutions that are much easier to come by and are still meaningful in some sense.

It would be interesting to try and formalize the notion of an approximate solution in terms of, for example, optimization theory or complexity theory, and to define more precisely what it means for an approximate solution to be “meaningful in some sense.” One possibility would be to define an approximate solution as the exact solution to a slightly perturbed problem, and then to identify the perturbed problem. Understanding such ideas could have some significant implications for complexity theory.

Appendix A

Isotropically Distributed Vectors and Matrices

In this appendix, we define the concept of isotropically distributed vectors and matrices and highlight the key properties that are used throughout the thesis. A more detailed discussion can be found in [48].

Definition A.1 *An m -dimensional complex random vector ϕ is isotropically distributed if its probability density is invariant to all unitary transformations; i.e., $f(\phi) = f(\Theta^* \phi)$ for all Θ such that $\Theta^* \Theta = \mathbf{I}_m$.*

Intuitively, an isotropically distributed complex vector is equally likely to point in any direction in complex space. Thus, the probability density of ϕ is a function of its magnitude but not its direction. If, in addition, ϕ is constrained to be a unit vector, then the probability density is

$$f(\phi) = \frac{\Gamma(m)}{\pi^m} \delta(\phi^* \phi - 1), \quad (\text{A.1})$$

and ϕ is conveniently generated by $\phi = \mathbf{z} / \sqrt{\mathbf{z}^* \mathbf{z}}$, where \mathbf{z} is an m -dimensional vector of independent $\mathcal{CN}(0, 1)$ random variables.

Definition A.2 *An $n \times m$ complex random matrix Φ is isotropically distributed if its probability density is unchanged when premultiplied by an $n \times n$ unitary matrix; i.e., $f(\Phi) = f(\Theta^* \Phi)$ for all Θ such that $\Theta^* \Theta = \mathbf{I}_n$.*

From the definition of an isotropically distributed matrix, it can be shown that the probability density is also unchanged when the matrix is postmultiplied by an $m \times m$ unitary

matrix; i.e., $f(\Phi) = f(\Phi\Theta)$ for all Θ such that $\Theta^*\Theta = \mathbf{I}_m$. Furthermore, by combining Definitions A.1 and A.2, we can readily see that the column vectors of Φ are themselves isotropically distributed vectors.

Appendix B

Proof of Theorem 3.2

The proof requires the following pair of lemmas.

Lemma B.1 *Let $H(\omega)$ be a complex-valued 2π -periodic function that is continuous almost everywhere on $[-\pi, \pi]$. Then the functions*

$$\tilde{S}_1^l(\omega) = \left(\frac{\mu^l}{\mathcal{E}_s(1 - (\rho^{l-1})^2)} \right) \left[\frac{|H(\omega)|^2/\xi^l}{(1 + |H(\omega)|^2/\xi^l)^2} - \frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{|H(\omega)|^2/\xi^l}{(1 + |H(\omega)|^2/\xi^l)^2} d\omega \right] \quad (\text{B.1})$$

$$\begin{aligned} \tilde{S}_2^l(\omega) &= \left(\frac{\mu^l}{\mathcal{E}_s(1 - (\rho^{l-1})^2)} \cdot \frac{|H(\omega)|^2/\xi^l}{1 + |H(\omega)|^2/\xi^l} - 1 \right)^2 \\ &\quad - \frac{1}{2\pi} \int_{-\pi}^{\pi} \left(\frac{\mu^l}{\mathcal{E}_s(1 - (\rho^{l-1})^2)} \cdot \frac{|H(\omega)|^2/\xi^l}{1 + |H(\omega)|^2/\xi^l} - 1 \right)^2 d\omega \end{aligned} \quad (\text{B.2})$$

$$\tilde{S}_3^l(\omega) = \frac{\mu^l}{\mathcal{E}_s(1 - (\rho^{l-1})^2)} \cdot \frac{|H(\omega)|^2/\xi^l}{1 + |H(\omega)|^2/\xi^l} - 1 \quad (\text{B.3})$$

are continuous almost everywhere on $[-\pi, \pi]$ and satisfy

$$\frac{1}{2\pi} \int_{-\pi}^{\pi} |\tilde{S}_i^l(\omega)|^2 d\omega < \infty \quad \text{for } i = 1, 2, 3. \quad (\text{B.4})$$

Proof: The proof is straightforward and follows from observing that

$$\tilde{S}_i^l(\omega) = \tilde{f}_i(|H(\omega)|^2) \quad \text{for } i = 1, 2 \quad (\text{B.5})$$

where $\tilde{f}_i(z)$ for $i = 1, 2, 3$ are real-valued continuous functions, bounded between $-(\mu^l/\mathcal{E}_s(1-(\rho^{l-1})^2))^2$ and $(\mu^l/\mathcal{E}_s(1-(\rho^{l-1})^2))^2$ on the interval $z \in [0, \infty)$. \square

Lemma B.2 *Let $\tilde{S}(\omega)$ be a complex-valued 2π -periodic function that is continuous almost everywhere on $[-\pi, \pi]$ and satisfies $\frac{1}{2\pi} \int_{-\pi}^{\pi} \tilde{S}(\omega) d\omega = 0$ and $\frac{1}{2\pi} \int_{-\pi}^{\pi} |\tilde{S}(\omega)|^2 d\omega < \infty$. Furthermore, let $F_N(\cdot)$ be the frequency interleaver defined by (3.94) that permutes the N $2\pi/N$ bands of the frequency spectrum. Then, if $N \rightarrow \infty$, we have that*

$$\mathcal{I}[n] = \frac{1}{4\pi^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} E[\tilde{S}(F_N^{-1}(\omega))S^*(F_N^{-1}(\nu))]e^{j(\omega-\nu)n} d\omega d\nu \rightarrow 0 \quad (\text{B.6})$$

for each n .

Proof: In the case that $n = 0$,

$$\begin{aligned} \mathcal{I}[n] &= \frac{1}{4\pi^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} E[\tilde{S}(F_N^{-1}(\omega))S^*(F_N^{-1}(\nu))] d\omega d\nu \\ &= \frac{1}{4\pi^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \tilde{S}(\omega)S^*(\nu) d\omega d\nu \\ &= \left| \frac{1}{2\pi} \int_{-\pi}^{\pi} \tilde{S}(\omega) d\omega \right|^2 \\ &= 0. \end{aligned} \quad (\text{B.7})$$

In the case that $0 < n < N$,

$$\mathcal{I}[n] = \frac{1}{4\pi^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \tilde{S}(\omega)\tilde{S}^*(\nu)E[e^{j(F_N(\omega)-F_N(\nu))n}]d\omega d\nu. \quad (\text{B.8})$$

Evaluating the expectation, we have that

$$\begin{aligned} &E[e^{j(F_N(\omega)-F_N(\nu))n}] \\ &= \begin{cases} -\frac{1}{N-1}e^{j[(\omega \bmod 2\pi/N)-(\nu \bmod 2\pi/N)]n} & \omega, \nu \text{ not in same frequency band} \\ e^{j(\omega-\nu)n} & \omega, \nu \text{ in same frequency band.} \end{cases} \end{aligned} \quad (\text{B.9})$$

Thus

$$\begin{aligned}
\mathcal{I}[n] &= \sum_{k=0}^{N-1} \frac{1}{4\pi^2} \int_{-\pi+\frac{2\pi k}{N}}^{-\pi+\frac{2\pi(k+1)}{N}} \int_{-\pi+\frac{2\pi k}{N}}^{-\pi+\frac{2\pi(k+1)}{N}} \tilde{S}(\omega) \tilde{S}^*(\nu) e^{j(\omega-\nu)n} d\omega d\nu \\
&+ \sum_{k=0}^{N-1} \sum_{\substack{l=0 \\ l \neq k}}^{N-1} \frac{1}{4\pi^2} \int_{-\pi+\frac{2\pi k}{N}}^{-\pi+\frac{2\pi(k+1)}{N}} \int_{-\pi+\frac{2\pi l}{N}}^{-\pi+\frac{2\pi(l+1)}{N}} \tilde{S}(\omega) \tilde{S}^*(\nu) \\
&\quad \times \left(-\frac{1}{N-1} \right) e^{j[(\omega \bmod 2\pi/N) - (\nu \bmod 2\pi/N)]n} d\omega d\nu \\
&= \frac{N}{N-1} \sum_{k=0}^{N-1} \frac{1}{4\pi^2} \int_{-\pi+\frac{2\pi k}{N}}^{-\pi+\frac{2\pi(k+1)}{N}} \int_{-\pi+\frac{2\pi k}{N}}^{-\pi+\frac{2\pi(k+1)}{N}} \tilde{S}(\omega) \tilde{S}^*(\nu) e^{j(\omega-\nu)n} d\omega d\nu \\
&- \frac{1}{N-1} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} \frac{1}{4\pi^2} \int_{-\pi+\frac{2\pi k}{N}}^{-\pi+\frac{2\pi(k+1)}{N}} \int_{-\pi+\frac{2\pi l}{N}}^{-\pi+\frac{2\pi(l+1)}{N}} \tilde{S}(\omega) \tilde{S}^*(\nu) \\
&\quad \times e^{j[(\omega \bmod 2\pi/N) - (\nu \bmod 2\pi/N)]n} d\omega d\nu \\
&= \frac{N}{N-1} \sum_{k=0}^{N-1} \left| \frac{1}{2\pi} \int_{-\pi+\frac{2\pi k}{N}}^{-\pi+\frac{2\pi(k+1)}{N}} \tilde{S}(\omega) e^{j\omega n} d\omega \right|^2 - \frac{1}{N-1} \left| \frac{1}{2\pi} \int_{-\pi}^{\pi} \tilde{S}(\omega) e^{j(\omega \bmod 2\pi/N)n} d\omega \right|^2.
\end{aligned} \tag{B.10}$$

In the limit as $N \rightarrow \infty$, we have

$$\begin{aligned}
\lim_{N \rightarrow \infty} \mathcal{I}[n] &= \lim_{N \rightarrow \infty} \frac{N}{N-1} \sum_{k=0}^{N-1} \left| \frac{1}{2\pi} \int_{-\pi+\frac{2\pi k}{N}}^{-\pi+\frac{2\pi(k+1)}{N}} \tilde{S}(\omega) e^{j\omega n} d\omega \right|^2 \\
&\quad - \lim_{N \rightarrow \infty} \frac{1}{N-1} \left| \frac{1}{2\pi} \int_{-\pi}^{\pi} \tilde{S}(\omega) e^{j(\omega \bmod 2\pi/N)n} d\omega \right|^2 \\
&= \lim_{N \rightarrow \infty} \sum_{k=0}^{N-1} \left| \frac{1}{2\pi} \tilde{S} \left(\frac{2\pi k}{N} \right) \frac{2\pi}{N} \right|^2 - \lim_{N \rightarrow \infty} \frac{1}{N-1} \left| \frac{1}{2\pi} \int_{-\pi}^{\pi} \tilde{S}(\omega) d\omega \right|^2 \\
&= \lim_{N \rightarrow \infty} \frac{1}{N} \left(\frac{1}{2\pi} \int_{-\pi}^{\pi} |\tilde{S}(\omega)|^2 d\omega \right) \\
&= 0.
\end{aligned} \tag{B.11}$$

Thus as $N \rightarrow \infty$, $\mathcal{I}[n] \rightarrow 0$ for each n . \square

We now proceed to a proof of our main result.

The term $u^l[n]$ can be decomposed into

$$u^l[n] = u_1^l[n] + u_2^l[n] \tag{B.12}$$

with

$$u_1^l[n] = b_{F_N^{-1}}^l[n] * w[n] \quad (\text{B.13})$$

$$u_2^l[n] = \tilde{c}_{F_N^{-1}}^l[n] * x[n] - d_{F_N^{-1}}^l[n] * \hat{x}^{l-1}[n] \quad (\text{B.14})$$

$$c_{F_N^{-1}}^l[n] = b_{F_N^{-1}}^l[n] * h_{F_N^{-1}}[n] \quad (\text{B.15})$$

$$\tilde{c}_{F_N^{-1}}^l[n] = c_{F_N^{-1}}^l[n] - \delta[n]. \quad (\text{B.16})$$

Let us consider $u_1^l[n]$ as defined in (B.13) first. We obtain the mean and covariance, respectively, of $u_1^l[n]$ given a frequency interleaver $F_N(\cdot)$ as

$$E[u_1^l[n]|F_N] = b_{F_N^{-1}}^l[n] * E[w[n]] = 0 \quad (\text{B.17})$$

and

$$\begin{aligned} R_{u_1^l u_1^l|F_N}[n] &= \frac{\mathcal{N}_0}{2\pi} \int_{-\pi}^{\pi} |B_{F_N^{-1}}^l(\omega)|^2 e^{j\omega n} d\omega \\ &= \frac{(\mu^l)^2}{\mathcal{E}_s(1 - (\rho^{l-1})^2)} \cdot \frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{|H_{F_N^{-1}}(\omega)|^2 / \xi^l}{(1 + |H_{F_N^{-1}}(\omega)|^2 / \xi^l)^2} e^{j\omega n} d\omega, \end{aligned} \quad (\text{B.18})$$

where we have used (3.78) and (3.34). In turn, averaging over the possible frequency interleavers $F_N(\cdot)$, we obtain

$$\begin{aligned} E[R_{u_1^l u_1^l|F_N}[n]] &= \frac{(\mu^l)^2}{\mathcal{E}_s(1 - (\rho^{l-1})^2)} \cdot \frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{|H(\omega)|^2 / \xi^l}{(1 + |H(\omega)|^2 / \xi^l)^2} E[e^{jF_N(\omega)n}] d\omega \\ &= \left(\frac{(\mu^l)^2}{\mathcal{E}_s(1 - (\rho^{l-1})^2)} \cdot \frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{|H(\omega)|^2 / \xi^l}{(1 + |H(\omega)|^2 / \xi^l)^2} d\omega \right) \delta[n] \\ &= \left(\frac{(\mu^l)^2}{\mathcal{E}_s(1 - (\rho^{l-1})^2)} \cdot \frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{|H_{F_N^{-1}}(\omega)|^2 / \xi^l}{(1 + |H_{F_N^{-1}}(\omega)|^2 / \xi^l)^2} d\omega \right) \delta[n]. \end{aligned} \quad (\text{B.19})$$

Next, we define

$$\tilde{R}_{u_1^l u_1^l|F_N}[n] \triangleq R_{u_1^l u_1^l|F_N}[n] - E[R_{u_1^l u_1^l|F_N}[n]] = \mathcal{E}_s(1 - (\rho^{l-1})^2) \cdot \frac{1}{2\pi} \int_{-\pi}^{\pi} \tilde{S}_1^l(F_N^{-1}(\omega)) e^{j\omega n} d\omega \quad (\text{B.20})$$

where $\tilde{S}_1^l(\omega)$ is given in (B.1). Then,

$$\begin{aligned}
& \text{var } R_{u_1^l u_1^l | F_N}[n] \\
&= E[|\tilde{R}_{u_1^l u_1^l | F_N}[n]|^2] \\
&= \left(\mathcal{E}_s (1 - (\rho^{l-1})^2) \right)^2 \left(\frac{1}{2\pi} \right)^2 \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} E \left[\tilde{S}_1^l(F_N^{-1}(\omega)) (\tilde{S}_1^l(F_N^{-1}(\nu)))^* \right] e^{j(\omega-\nu)n} d\omega d\nu. \quad (\text{B.21})
\end{aligned}$$

Applying, Lemma B.1 followed by Lemma B.2 to (B.22), we then obtain, for each n ,

$$\text{var } R_{u_1^l u_1^l | F_N}[n] \rightarrow 0. \quad (\text{B.22})$$

Hence, combining (B.19) with (B.22), we have, for any particular frequency interleaver $F_N(\cdot)$,

$$R_{u_1^l u_1^l | F_N}[n] \xrightarrow{\text{m.s.}} \left(\frac{(\mu^l)^2}{\mathcal{E}_s (1 - (\rho^{l-1})^2)} \cdot \frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{|H(\omega)|^2 / \xi^l}{(1 + |H(\omega)|^2 / \xi^l)^2} d\omega \right) \delta[n] \quad (\text{B.23})$$

for each n .

We now consider $u_2^l[n]$ as defined in (B.14). Again, for a fixed realization of $F_N(\cdot)$ (and hence $c_{F_N^{-1}}^l[n]$), we have

$$E[u_2^l[n] | F_N] = \tilde{c}_{F_N^{-1}}^l[n] * E[x[n]] - \rho^{l-1} \tilde{c}_{F_N^{-1}}^l[n] * E[\hat{x}^{l-1}[n]] = 0 \quad (\text{B.24})$$

and

$$\begin{aligned}
& R_{u_2^l u_2^l | F_N}[n] \\
&= \mathcal{E}_s (1 - (\rho^{l-1})^2) \cdot \frac{1}{2\pi} \int_{-\pi}^{\pi} |\tilde{C}_{F_N^{-1}}^l(\omega)|^2 e^{j\omega n} d\omega \\
&= \mathcal{E}_s (1 - (\rho^{l-1})^2) \cdot \frac{1}{2\pi} \int_{-\pi}^{\pi} \left(\frac{\mu^l}{\mathcal{E}_s (1 - (\rho^{l-1})^2)} \cdot \frac{|H_{F_N^{-1}}(\omega)|^2 / \xi^l}{1 + |H_{F_N^{-1}}(\omega)|^2 / \xi^l} - 1 \right)^2 e^{j\omega n} d\omega, \quad (\text{B.25})
\end{aligned}$$

where we have used (B.16), (B.15), (3.78), and (3.34). However, (B.25) is asymptotically independent of $F_N(\cdot)$. To see this, first note that

$$\begin{aligned}
& E[R_{u_2^l u_2^l | F_N}[n]] \\
&= \mathcal{E}_s(1 - (\rho^{l-1})^2) \cdot \frac{1}{2\pi} \int_{-\pi}^{\pi} \left(\frac{\mu^l}{\mathcal{E}_s(1 - (\rho^{l-1})^2)} \cdot \frac{|H(\omega)|^2 / \xi^l}{1 + |H(\omega)|^2 / \xi^l} - 1 \right)^2 E[e^{jF_N(\omega)n}] d\omega \\
&= \left(\mathcal{E}_s(1 - (\rho^{l-1})^2) \cdot \frac{1}{2\pi} \int_{-\pi}^{\pi} \left(\frac{\mu^l}{\mathcal{E}_s(1 - (\rho^{l-1})^2)} \cdot \frac{|H(\omega)|^2 / \xi^l}{1 + |H(\omega)|^2 / \xi^l} - 1 \right)^2 d\omega \right) \delta[n] \\
&= \left(\mathcal{E}_s(1 - (\rho^{l-1})^2) \cdot \frac{1}{2\pi} \int_{-\pi}^{\pi} \left(\frac{\mu^l}{\mathcal{E}_s(1 - (\rho^{l-1})^2)} \cdot \frac{|H_{F_N^{-1}}(\omega)|^2 / \xi^l}{1 + |H_{F_N^{-1}}(\omega)|^2 / \xi^l} - 1 \right)^2 d\omega \right) \delta[n].
\end{aligned} \tag{B.26}$$

Then, since

$$\tilde{R}_{u_2^l u_2^l | F_N}[n] \triangleq R_{u_2^l u_2^l | F_N}[n] - E[R_{u_2^l u_2^l | F_N}[n]] = \mathcal{E}_s(1 - (\rho^{l-1})^2) \cdot \frac{1}{2\pi} \int_{-\pi}^{\pi} \tilde{S}_2^l(F_N^{-1}(\omega)) e^{j\omega n} d\omega \tag{B.27}$$

where $\tilde{S}_2^l(\omega)$ is as defined in (B.2), we have

$$\begin{aligned}
& \text{var } R_{u_2^l u_2^l | F_N}[n] \\
&= E[|\tilde{R}_{u_2^l u_2^l | F_N}[n]|^2] \\
&= \left(\mathcal{E}_s(1 - (\rho^{l-1})^2) \right)^2 \left(\frac{1}{2\pi} \right)^2 \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} E \left[\tilde{S}_2^l(F_N^{-1}(\omega)) (\tilde{S}_2^l(F_N^{-1}(\nu)))^* \right] e^{j(\omega - \nu)n} d\omega d\nu.
\end{aligned} \tag{B.28}$$

Hence applying, in turn, Lemmas B.1 and B.2 to (B.28), we then obtain, for each n ,

$$\text{var } R_{u_2^l u_2^l | F_N}[n] \rightarrow 0. \tag{B.29}$$

Hence, combining (B.26) with (B.29), we have, for any particular frequency interleaver $F_N(\cdot)$,

$$R_{u_2^l u_2^l | F_N}[n] \xrightarrow{\text{m.s.}} \left(\mathcal{E}_s(1 - (\rho^{l-1})^2) \cdot \frac{1}{2\pi} \int_{-\pi}^{\pi} \left(\frac{\mu^l}{\mathcal{E}_s(1 - (\rho^{l-1})^2)} \cdot \frac{|H(\omega)|^2 / \xi^l}{1 + |H(\omega)|^2 / \xi^l} - 1 \right)^2 d\omega \right) \delta[n] \tag{B.30}$$

for each n .

Since $w[n]$ is statistically independent of $x[n]$ and can be assumed to be independent of $\hat{x}^{l-1}[n]$,

$$E[u_1^l[n](u_2^l[k])^*|F_N] = 0, \quad \text{for all } n \text{ and } k \quad (\text{B.31})$$

and hence

$$R_{u^l u^l|F_N}[n] = R_{u_1^l u_1^l|F_N}[n] + R_{u_2^l u_2^l|F_N}[n]. \quad (\text{B.32})$$

Finally, we need to show that for a given realization of the frequency interleaver $F_N(\cdot)$ that $x[n]$ and $u^l[n]$ are asymptotically uncorrelated. Due to (B.32), it suffices to show that $x[n]$ is asymptotically uncorrelated with $u_1^l[n]$ and $u_2^l[n]$ individually.

First, using (B.13), we have

$$E[u_1^l[n]x^*[k]] = \sum_m E[w[m]x^*[k]]E[b_{F_N^{-1}}^l[n-m]] = 0 \quad (\text{B.33})$$

where the last equality follows from the fact that the processes $w[n]$ and $x[n]$ are statistically independent.

Next, using (B.14), we have

$$\begin{aligned} E[u_2^l[n]x^*[k]] &= \sum_m E[x[m]x^*[k]]\tilde{c}_{F_N^{-1}}^l[n-m] - \sum_m E[\hat{x}^{l-1}[m]x^*[k]]d_{F_N^{-1}}^l[n-m] \\ &= \mathcal{E}_s \tilde{c}_F^l[n-k] - \rho^{l-1} \mathcal{E}_s d_{F_N^{-1}}^l[n-k] \end{aligned} \quad (\text{B.34})$$

where the last equality follows from (3.35) and the fact that the symbol stream $x[n]$ is white.

Thus, it remains only to show that $\tilde{c}_{F_N^{-1}}^l[n] \xrightarrow{\text{m.s.}} 0$ and $d_{F_N^{-1}}^l[n] \xrightarrow{\text{m.s.}} 0$ for all n .

To see this, we first note that

$$E[\tilde{c}_{F_N^{-1}}^l[n]] = 0 \quad (\text{B.35})$$

$$E[d_{F_N^{-1}}^l[n]] = 0, \quad (\text{B.36})$$

where (B.36) follows from the fact that $E[d_{F_N^{-1}}^l[n]] = \left(\frac{1}{2\pi} \int_{-\pi}^{\pi} D^l(\omega) d\omega\right) \delta[n]$ and the fact that $d^l[0] = 0$. Next,

$$\text{var } \tilde{c}_{F_N^{-1}}^l[n] = \left(\frac{1}{2\pi}\right)^2 \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} E[\tilde{S}_3^l(F_N^{-1}(\omega))(\tilde{S}_3^l(F_N^{-1}(\nu)))^*] e^{j\omega n} e^{-j\nu n} d\omega d\nu \quad (\text{B.37})$$

$$\text{var } d_{F_N^{-1}}^l[n] = \left(\frac{1}{2\pi}\right)^2 \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} (\rho^{l-1})^2 E[\tilde{S}_3^l(F_N^{-1}(\omega))(\tilde{S}_3^l(F_N^{-1}(\nu)))^*] e^{j\omega n} e^{-j\nu n} d\omega d\nu \quad (\text{B.38})$$

Hence applying again, in turn, Lemmas B.1 and B.2 to each of (B.37) and (B.38), we then obtain, for each n ,

$$\text{var } \tilde{c}_{F_N^{-1}}^l[n] \rightarrow 0 \tag{B.39}$$

$$\text{var } d_{F_N^{-1}}^l[n] \rightarrow 0. \tag{B.40}$$

Hence, combining (B.35) and (B.36) with (B.39) and (B.40), we obtain the desired results.

Appendix C

Information Limits of Random ISI Channels

Let us consider the quantities I_{int} and C_{int} for the limiting random ISI channel introduced in Section 3.2.1. Recall that arbitrarily close samples of the asymptotic random process $H(\omega)$ are independent, complex-valued, circularly symmetric Gaussian random variables with zero mean and unity variance. Thus, arbitrarily close samples of $|H(\omega)|^2$ are independent, real-valued, exponentially distributed random variables with unit mean.

Thus the mutual information of the channel in bits per two dimensions is [43]

$$\begin{aligned}
 I_{\text{int}} &= \frac{1}{2\pi} \int_{-\pi}^{\pi} \log_2 \left(1 + \frac{\mathcal{E}_s |H(\omega)|^2}{\mathcal{N}_0} \right) d\omega \\
 &= \frac{1}{\ln 2} \int_0^{\infty} \ln(1 + \alpha) \cdot \frac{\mathcal{N}_0}{\mathcal{E}_s} e^{-\frac{\mathcal{N}_0}{\mathcal{E}_s} \alpha} d\alpha \\
 &= \frac{1}{\ln 2} \left\{ - \left[e^{-\frac{\mathcal{N}_0}{\mathcal{E}_s} \alpha} \ln(\alpha + 1) \right]_{\alpha=0}^{\alpha=\infty} + \int_0^{\infty} \frac{e^{-\frac{\mathcal{N}_0}{\mathcal{E}_s} \alpha}}{\alpha + 1} d\alpha \right\} \\
 &= \frac{1}{\ln 2} e^{\frac{\mathcal{N}_0}{\mathcal{E}_s}} E_1 \left(\frac{\mathcal{N}_0}{\mathcal{E}_s} \right)
 \end{aligned} \tag{C.1}$$

where we have used integrating by parts followed by (3.106).

For the purposes of determining C_{int} , without loss of generality we fix $|H(\omega)|^2$ deterministically to be the shifted and normalized version of the exponential cumulative distribution function,

$$\frac{\omega}{2\pi} = \frac{1}{2} - e^{-|H(\omega)|^2}, \tag{C.2}$$

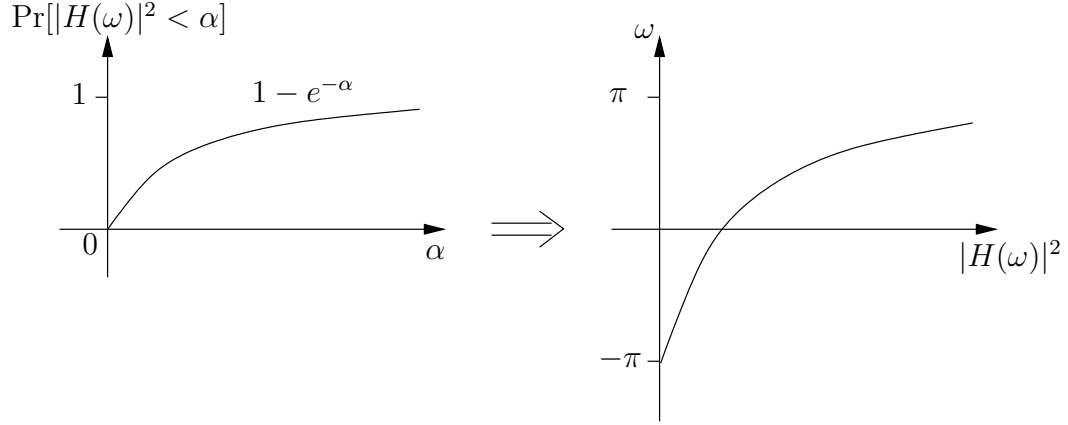


Figure C.1: For the purposes of computing the capacity of the random ISI channel, the magnitude squared of the channel frequency response can be derived from an exponential distribution.

illustrated in Fig. C.1. Rearranging, we get

$$|H(\omega)|^2 = -\ln\left(\frac{1}{2} - \frac{\omega}{2\pi}\right). \quad (\text{C.3})$$

The transmit spectrum is given by the water-pouring formula [43]

$$S_X(\omega) = \max\left(\mathcal{L} - \frac{N_0}{|H(\omega)|^2}, 0\right) \quad (\text{C.4})$$

where \mathcal{L} is chosen such that the average transmit energy is

$$\mathcal{E}_s = \frac{1}{2\pi} \int_{-\pi}^{\pi} S_X(\omega) d\omega. \quad (\text{C.5})$$

If Ω is the band over which $S_X(\omega)$ is nonzero (i.e., the water-pouring band), then the capacity of the channel is [43]

$$C_{\text{int}} = \frac{\Omega}{2\pi} \log\left(\frac{\frac{\mathcal{E}_s}{N_0} \cdot \frac{2\pi}{\Omega} + \langle |H(\omega)|^{-2} \rangle_{A,\Omega}}{\langle |H(\omega)|^{-2} \rangle_{G,\Omega}}\right) \quad (\text{C.6})$$

where $\langle |H(\omega)|^{-2} \rangle_{A,\Omega}$ and $\langle |H(\omega)|^{-2} \rangle_{G,\Omega}$ are, respectively, the arithmetic and geomet-

ric means of $|H(\omega)|^{-2}$ over the water-pouring band Ω , given by

$$\langle |H(\omega)|^{-2} \rangle_{A,\Omega} = \frac{1}{|\Omega|} \int_{\Omega} \frac{1}{|H(\omega)|^2} d\omega \quad (\text{C.7})$$

$$\log \langle |H(\omega)|^{-2} \rangle_{G,\Omega} = \frac{1}{|\Omega|} \int_{\Omega} \log \frac{1}{|H(\omega)|^2} d\omega. \quad (\text{C.8})$$

Note that $\mathcal{E}_s/\mathcal{N}_0$ in (C.6) is scaled by a factor of $2\pi/\Omega$ because using only the water-pouring band filters out noise power.

Figure C.2 compares C_{int} and I_{int} to C_{AWGN} . As noted in Section 1.2.4, the capacity of the asymptotic random ISI channel exceeds that of the corresponding AWGN channel at low SNR, since transmit power can be loaded onto favorable frequencies. If water pouring is not possible, then at low SNR the mutual information of the random ISI channel approaches the capacity of the corresponding AWGN channel. At high SNR, the capacity of the ISI channel becomes less than the AWGN channel capacity, and also the effect of water pouring becomes negligible. The asymptotic slopes of all three curves are equal, implying that the penalty of the random ISI channel is only a fixed rate loss.

This asymptotic rate loss of the random ISI channel with respect to the AWGN channel

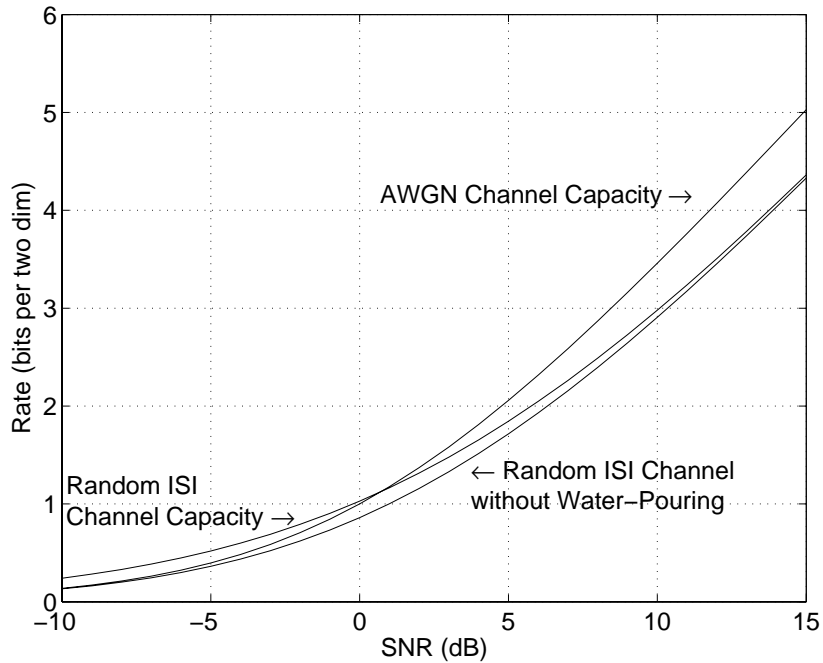


Figure C.2: Information rates for the asymptotic random ISI channel.

can be quantified. In the high-SNR regime, we can use the series expansion [1]

$$E_1(t) = -\Gamma_0 - \ln t - \sum_{k=1}^{\infty} \frac{(-1)^k t^k}{tt!} \quad (\text{C.9})$$

where $\Gamma_0 = 0.57721 \dots$ denotes Euler's constant, to show that I_{int} in (C.1) becomes

$$I_{\text{int}} \rightarrow -\Gamma_0 - \ln \frac{\mathcal{E}_s}{\mathcal{N}_0} = \ln \left(\frac{\mathcal{E}_s}{\mathcal{N}_0 e^{\Gamma_0}} \right). \quad (\text{C.10})$$

Comparing to C_{AWGN} in (1.14), we see that the SNR gap is e^{Γ_0} or 2.507 dB. Since the asymptotic slope of the curves is 1 bit per 3 dB of SNR, the asymptotic rate loss is $\log_2 e^{\Gamma_0} \approx 0.8327$ bits per two dimensions.

Appendix D

Rate-Normalized SNR

Given a communication system operating at a particular SNR and a particular information rate, it is possible to normalize the SNR with respect to the maximum theoretical information rate, allowing one to compare systems with different modulations schemes. Since there are different information limits for different scenarios as discussed in Section 1.2, for each scenario there is a different notion of rate-normalized SNR. In this appendix, we review some of the definitions for rate-normalized SNR and also give a new definition for the scenario in which the transmitter has no knowledge of the channel.

D.1 AWGN Channels

Consider a 4-QAM constellation and a 16-QAM constellation. At the same SNR, the signal points of 16-QAM are denser than those of 4-QAM, and hence the probability of symbol error is higher. In order to make the probability of symbol error approximately equal, the symbol energy of the 16-QAM constellation needs to be increased so that the minimum distances of both constellations are equal. Since the average symbol energy of M -QAM is

$$\mathcal{E}_s = \frac{(M-1)d_{\min}^2}{6} \quad (\text{D.1})$$

where d_{\min} is the minimum distance, the transmit energy must be increased by a factor of $(16-1)/(4-1) = 5$, or approximately 7 dB.

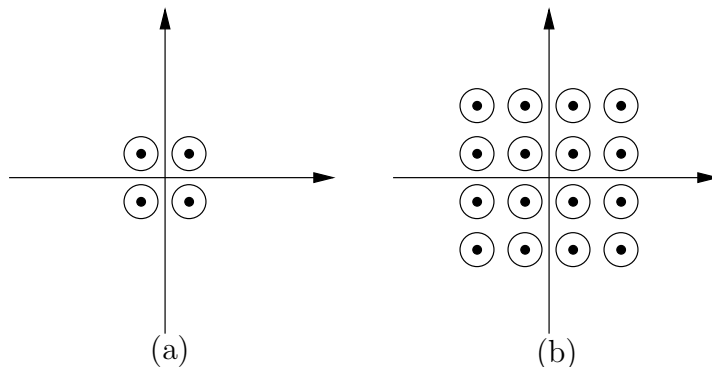


Figure D.1: The noise balls for (a) 4-QAM and (b) 16-QAM when the normalized SNR is equal. Normalization makes both constellations essentially the same, since they both come from the same infinite lattice.

Rate-normalized SNR is defined as

$$\text{SNR}_{\text{norm}} \triangleq \frac{\text{SNR}}{2^R - 1} \quad (\text{D.2})$$

where R is the rate per two dimensions. Intuitively, 4-QAM and 16-QAM originate by choosing different constellation boundaries from the same (shifted) infinite lattice. Rate normalization ignores the particular constellation boundary chosen, and focuses attention instead on the ratio of the noise energy and the squared minimum distance in the lattice. The 4-QAM and 16-QAM constellations have the same symbol-error probability (ignoring edge effects) when SNR_{norm} is the same. Figure D.1 illustrates the “noise balls” for 4-QAM and 16-QAM when the SNR_{norm} is the same.

In addition to being useful for comparing the error probability of various modulation schemes, rate-normalized SNR is convenient for comparing the gap to capacity. From the AWGN capacity formula in (1.7), the rate R in bits per two dimensions that can be supported with arbitrarily small error probability satisfies

$$R < \log_2(1 + \text{SNR}), \quad (\text{D.3})$$

so the SNR in turn satisfies

$$\text{SNR} > 2^R - 1. \quad (\text{D.4})$$

Thus, the Shannon SNR limit varies with rate R , and it becomes difficult to compare the gap to capacity for modulation schemes with differing rates. However, using SNR_{norm} , (D.4)

can be equivalently expressed as

$$\text{SNR}_{\text{norm}} = \frac{\text{SNR}}{2^R - 1} > 1. \quad (\text{D.5})$$

Thus when $\text{SNR}_{\text{norm}} > 1$, the probability of error can be driven to zero, but when $\text{SNR}_{\text{norm}} < 1$ the Shannon limits are violated and the error probability is bounded away from zero. For any modulation scheme, the dB gap between SNR_{norm} and unity indicates how far the modulation scheme is operating from fundamental limits. Moreover, since SNR_{norm} is a linear function of SNR, the dB gap measures how much the SNR must be increased to achieve a given rate R . Thus, for an uncoded system the dB gap quantifies the potential benefit of adding error-correction coding to the system.

D.2 Interference Channels with Water Pouring

The concept of normalized SNR can be generalized to channels with interference. The way in which normalized SNR is generalized depends upon how SNR is defined. Two natural ways to define the SNR are at the input to the channel and at the output.

At the input to the channel, the average transmit energy is \mathcal{E}_s . If the noise spectrum at the channel output is \mathcal{N}_0 , then the equivalent noise spectrum at the channel input is \mathcal{N}_0/λ_k . The input SNR is then

$$\text{SNR}_{\text{in}} = \frac{\mathcal{E}_s}{\mathcal{N}_0 \langle \frac{1}{\lambda_k} \rangle_{A,K}} \cdot \frac{N}{K} \quad (\text{D.6})$$

where K is the set of modes allotted transmit power via water pouring and $\langle 1/\lambda_k \rangle_{A,K}$ is the arithmetic mean of $1/\lambda_k$ over K , given by (1.11). Using the capacity formula (1.10), the rate R that can be supported by the channel satisfies

$$R < \frac{K}{N} \log_2 \left(\frac{\langle \frac{1}{\lambda_k} \rangle_{A,K}}{\langle \frac{1}{\lambda_k} \rangle_{G,K}} (1 + \text{SNR}_{\text{in}}) \right) \quad (\text{D.7})$$

where $\langle 1/\lambda_k \rangle_{G,K}$ is the geometric mean of $1/\lambda_k$ over K as given by (1.12), so the SNR_{in}

required to support a rate R is thus

$$\text{SNR}_{\text{in}} > 2^{RN/K} \frac{\left\langle \frac{1}{\lambda_k} \right\rangle_{G,K}}{\left\langle \frac{1}{\lambda_k} \right\rangle_{A,K}} - 1. \quad (\text{D.8})$$

Thus, we define the normalized SNR as

$$\begin{aligned} \text{SNR}_{\text{norm}} &\triangleq \frac{\text{SNR}_{\text{in}}}{2^{RN/K} \frac{\left\langle \frac{1}{\lambda_k} \right\rangle_{G,K}}{\left\langle \frac{1}{\lambda_k} \right\rangle_{A,K}} - 1} \\ &= \frac{\frac{\mathcal{E}_s}{\mathcal{N}_0} \cdot \frac{N}{K}}{2^{RN/K} \left\langle \frac{1}{\lambda_k} \right\rangle_{G,K} - \left\langle \frac{1}{\lambda_k} \right\rangle_{A,K}} \\ &> 1. \end{aligned} \quad (\text{D.9})$$

In this interference case, SNR_{norm} is not linear with \mathcal{E}_s because the geometric and arithmetic means of $1/\lambda_k$ over the water-pouring modes depend implicitly on \mathcal{E}_s . Therefore, SNR_{norm} in the interference case no longer represents gap to capacity. At high SNR, however, virtually all modes are used and SNR_{norm} is a linear function of \mathcal{E}_s .

Alternatively, using the water-pouring formula (1.8), the output SNR is

$$\text{SNR}_{\text{out}} = \frac{\langle \mathcal{E}_{s,k} \lambda_k \rangle_{A,K}}{\mathcal{N}_0} = \frac{\langle (\mathcal{L} - \mathcal{N}_0/\lambda_k) \lambda_k \rangle_{A,K}}{\mathcal{N}_0} \quad (\text{D.10})$$

where $\mathcal{E}_{s,k}$ is the power allotted to the k th mode. Combining (1.8) and (1.9), we obtain the expression

$$\mathcal{L} = \mathcal{E}_s \cdot \frac{N}{K} + \mathcal{N}_0 \left\langle \frac{1}{\lambda_k} \right\rangle_{A,K}, \quad (\text{D.11})$$

which makes the output SNR

$$\text{SNR}_{\text{out}} = \left(\frac{\mathcal{E}_s}{\mathcal{N}_0} \cdot \frac{N}{K} + \left\langle \frac{1}{\lambda_k} \right\rangle_{A,K} \right) \langle \lambda_k \rangle_{A,K} - 1. \quad (\text{D.12})$$

Using the capacity formula (1.10), the rate R that can be supported by the channel satisfies

$$R < \frac{K}{N} \log_2 \left(\frac{1 + \text{SNR}_{\text{out}}}{\left\langle \frac{1}{\lambda_k} \right\rangle_{G,K} \langle \lambda_k \rangle_{A,K}} \right), \quad (\text{D.13})$$

so the SNR_{out} required to support a rate R is thus

$$\text{SNR}_{\text{out}} > 2^{RN/K} \left\langle \frac{1}{\lambda_k} \right\rangle_{G,K} \langle \lambda_k \rangle_{A,K} - 1. \quad (\text{D.14})$$

The normalized SNR is then defined as

$$\begin{aligned} \text{SNR}_{\text{norm}} &\triangleq \frac{\text{SNR}_{\text{out}}}{2^{RN/K} \left\langle \frac{1}{\lambda_k} \right\rangle_{G,K} \langle \lambda_k \rangle_{A,K} - 1} \\ &= \frac{\left(\frac{\mathcal{E}_s}{\mathcal{N}_0} \cdot \frac{N}{K} + \left\langle \frac{1}{\lambda_k} \right\rangle_{A,K} \right) \langle \lambda_k \rangle_{A,K} - 1}{2^{RN/K} \left\langle \frac{1}{\lambda_k} \right\rangle_{G,K} \langle \lambda_k \rangle_{A,K} - 1} \\ &= \frac{\frac{\mathcal{E}_s}{\mathcal{N}_0} \cdot \frac{N}{K} + \left\langle \frac{1}{\lambda_k} \right\rangle_{A,K} - \frac{1}{\langle \lambda_k \rangle_{A,K}}}{2^{RN/K} \left\langle \frac{1}{\lambda_k} \right\rangle_{G,K} - \frac{1}{\langle \lambda_k \rangle_{A,K}}} \\ &> 1. \end{aligned} \quad (\text{D.15})$$

Again, SNR_{norm} represents gap to capacity only at high SNR.

D.3 Interference Channels without Water Pouring

As discussed in Section 1.2, the difference in achievable rates for an interference channel with and without water pouring can be notable, especially at low SNR. It would seem unfair, therefore, to use the versions of normalized SNR defined in either (D.9) or (D.15).

The mutual information of an interference channel without water pouring,

$$I_{\text{int}} = \frac{1}{N} \sum_{k=1}^N \log \left(1 + \frac{\mathcal{E}_s \lambda_k}{\mathcal{N}_0} \right) = \log \left\langle 1 + \frac{\mathcal{E}_s \lambda_k}{\mathcal{N}_0} \right\rangle_G, \quad (\text{D.16})$$

suggests a new definition of SNR that is neither the channel input SNR nor the channel output SNR. If we define the new SNR as

$$\text{SNR}_G \triangleq \left\langle 1 + \frac{\mathcal{E}_s \lambda_k}{\mathcal{N}_0} \right\rangle_G - 1, \quad (\text{D.17})$$

then the rate R that can be supported by the channel satisfies

$$R < \log_2 (1 + \text{SNR}_G). \quad (\text{D.18})$$

The SNR_G required to support a rate R is thus

$$\text{SNR}_G > 2^R - 1 \tag{D.19}$$

and the normalized SNR is defined as

$$\begin{aligned} \text{SNR}_{\text{norm}} &\triangleq \frac{\text{SNR}_G}{2^R - 1} \\ &= \frac{\left\langle 1 + \frac{\mathcal{E}_s \lambda_k}{N_0} \right\rangle_G - 1}{2^R - 1} \\ &> 1. \end{aligned} \tag{D.20}$$

This version of SNR_{norm} is approximately a linear function of \mathcal{E}_s at high SNR, so in that regime SNR_{norm} represents the SNR gap to the mutual information. This is the definition of rate-normalized SNR that is used in Chapter 5 to evaluate the performance of the iterated-decision detector with coding.

Appendix E

The Correlation Coefficient for Lattice Codes

With mode-interleaved precoding, the SINRs γ_i^l for $i = 1, 2, \dots, N$ are asymptotically equal, so the normalized correlation matrix defined in (3.5) simplifies to $\boldsymbol{\rho}^{l-1} = \rho^{l-1} \mathbf{I}$. The scalar ρ^{l-1} can then be expressed as

$$\rho^{l-1} = \frac{E[x\hat{x}^{l-1*}]}{\mathcal{E}_s} \quad (\text{E.1})$$

or more generally as

$$\rho^{l-1} = \frac{E[\mathbf{x}^\dagger \hat{\mathbf{x}}]}{\mathcal{E}_c} \quad (\text{E.2})$$

where \mathcal{E}_c is the average codeword energy. The following proposition approximates ρ^{l-1} for lattice codes over AWGN channels, which can be used to estimate ρ^{l-1} for lattice codes over interference channels with mode-interleaved precoding and iterated-decision detection.

Proposition E.1 *For any lattice code \mathcal{C} whose shaping region is an N -sphere, the normalized correlation ρ between a codeword \mathbf{x} transmitted over an AWGN channel and the decoded codeword $\hat{\mathbf{x}}$ is related to the probability of codeword error $\Pr(\epsilon)$ by the expression*

$$\rho \approx 1 - \frac{d_{\min}^2}{2\mathcal{E}_c} \Pr(\epsilon), \quad (\text{E.3})$$

where d_{\min} is the minimum Euclidean distance and \mathcal{E}_c is the average energy.

Proof: By defining $\mathbf{e} = \mathbf{x} - \hat{\mathbf{x}}$ and using the fact that $E[\mathbf{x}^\dagger \mathbf{x}] = \mathcal{E}_c$, we have that

$$\rho \triangleq \frac{E[\mathbf{x}^\dagger \hat{\mathbf{x}}]}{\mathcal{E}_c} = \frac{E[\mathbf{x}^\dagger \mathbf{x}]}{\mathcal{E}_c} - \frac{E[\mathbf{x}^\dagger \mathbf{e}]}{\mathcal{E}_c} = 1 - \frac{E[\mathbf{x}^\dagger \mathbf{e}]}{\mathcal{E}_c}. \quad (\text{E.4})$$

It remains to determine $E[\mathbf{x}^\dagger \mathbf{e}]$.

Using the theorem on total probability, we have

$$E[\mathbf{x}^\dagger \mathbf{e}] = E[\mathbf{x}^\dagger \mathbf{e} | \mathbf{e} \neq 0] \Pr(\mathbf{e} \neq 0) + E[\mathbf{x}^\dagger \mathbf{e} | \mathbf{e} = 0] \Pr(\mathbf{e} = 0) = E[\mathbf{x}^\dagger \mathbf{e} | \mathbf{e} \neq 0] \Pr(\epsilon), \quad (\text{E.5})$$

where $\Pr(\epsilon)$ is the probability of codeword error. Let K be the number of nearest neighbors of a lattice point at the minimum distance. Then there are K possible minimum-distance error events, which we label as $\pm \mathbf{e}_1, \pm \mathbf{e}_2, \dots, \pm \mathbf{e}_{K/2}$. We make the simplifying approximation that any decoding error that occurs is a minimum-distance error event. Under this assumption, $\Pr(\mathbf{e} = +\mathbf{e}_k) = \Pr(\mathbf{e} = -\mathbf{e}_k) \approx \Pr(\epsilon)/K$. Using this approximation and another application of the theorem on total probability, (E.5) becomes

$$\begin{aligned} E[\mathbf{x}^\dagger \mathbf{e}] &\approx \sum_{k=1}^{K/2} \left(E[\mathbf{x}^\dagger \mathbf{e} | \mathbf{e} = +\mathbf{e}_k] \Pr(\mathbf{e} = +\mathbf{e}_k) + E[\mathbf{x}^\dagger \mathbf{e} | \mathbf{e} = -\mathbf{e}_k] \Pr(\mathbf{e} = -\mathbf{e}_k) \right) \\ &\approx \frac{\Pr(\epsilon)}{K} \sum_{k=1}^{K/2} \left(E[\mathbf{x}^\dagger \mathbf{e} | \mathbf{e} = +\mathbf{e}_k] + E[\mathbf{x}^\dagger \mathbf{e} | \mathbf{e} = -\mathbf{e}_k] \right). \end{aligned} \quad (\text{E.6})$$

In order to simplify (E.6) further, we introduce some additional notation. Let $\mathcal{S}_{\mathbf{e}_k^+}$ be the set of all codewords in \mathcal{C} that, when transmitted and decoded, cannot produce the error event $+\mathbf{e}_k$. As Fig. E.1 suggests, $\mathcal{S}_{\mathbf{e}_k^+}$ roughly consists of all points in half of the outer shell of the n -sphere, where the sphere is divided in half by an $(N-1)$ -dimensional plane perpendicular to $+\mathbf{e}_k$. Similarly, let $\mathcal{S}_{\mathbf{e}_k^-}$ be the set of all codewords in \mathcal{C} that, when transmitted and decoded, cannot produce the error event $-\mathbf{e}_k$. With this notation, we can expand (E.6) as

$$E[\mathbf{x}^\dagger \mathbf{e}] \approx \frac{\Pr(\epsilon)}{K} \sum_{k=1}^{K/2} \left(\frac{1}{|\mathcal{C} \setminus \mathcal{S}_{\mathbf{e}_k^+}|} \sum_{\mathbf{x} \in \mathcal{C} \setminus \mathcal{S}_{\mathbf{e}_k^+}} \mathbf{x}^\dagger \mathbf{e}_k - \frac{1}{|\mathcal{C} \setminus \mathcal{S}_{\mathbf{e}_k^-}|} \sum_{\mathbf{x} \in \mathcal{C} \setminus \mathcal{S}_{\mathbf{e}_k^-}} \mathbf{x}^\dagger \mathbf{e}_k \right). \quad (\text{E.7})$$

Since $\mathcal{C} \setminus \mathcal{S}_{\mathbf{e}_k^+}$ is approximately the union of $\mathcal{S}_{\mathbf{e}_k^-}$ and $\mathcal{C} \setminus (\mathcal{S}_{\mathbf{e}_k^+} \cup \mathcal{S}_{\mathbf{e}_k^-})$ (approximately, because $\mathcal{S}_{\mathbf{e}_k^+}$ and $\mathcal{S}_{\mathbf{e}_k^-}$ are not necessarily mutually exclusive), and $\mathcal{C} \setminus \mathcal{S}_{\mathbf{e}_k^-}$ is approximately the union

of $\mathcal{S}_{\mathbf{e}_k^+}$ and $\mathcal{C} \setminus (\mathcal{S}_{\mathbf{e}_k^+} \cup \mathcal{S}_{\mathbf{e}_k^-})$, (E.7) becomes

$$\begin{aligned}
E[\mathbf{x}^\dagger \mathbf{e}] &\approx \frac{\Pr(\epsilon)}{K} \sum_{k=1}^{K/2} \left(\frac{1}{|\mathcal{C} \setminus \mathcal{S}_{\mathbf{e}_k^+}|} \sum_{\mathbf{x} \in \mathcal{S}_{\mathbf{e}_k^-}} \mathbf{x}^\dagger \mathbf{e}_k + \frac{1}{|\mathcal{C} \setminus \mathcal{S}_{\mathbf{e}_k^+}|} \sum_{\mathbf{x} \in \mathcal{C} \setminus (\mathcal{S}_{\mathbf{e}_k^+} \cup \mathcal{S}_{\mathbf{e}_k^-})} \mathbf{x}^\dagger \mathbf{e}_k \right. \\
&\quad \left. - \frac{1}{|\mathcal{C} \setminus \mathcal{S}_{\mathbf{e}_k^-}|} \sum_{\mathbf{x} \in \mathcal{S}_{\mathbf{e}_k^+} } \mathbf{x}^\dagger \mathbf{e}_k - \frac{1}{|\mathcal{C} \setminus \mathcal{S}_{\mathbf{e}_k^-}|} \sum_{\mathbf{x} \in \mathcal{C} \setminus (\mathcal{S}_{\mathbf{e}_k^+} \cup \mathcal{S}_{\mathbf{e}_k^-})} \mathbf{x}^\dagger \mathbf{e}_k \right) \\
&= \frac{\Pr(\epsilon)}{K |\mathcal{C} \setminus \mathcal{S}_{\mathbf{e}_k^+}|} \sum_{k=1}^{K/2} \left(\sum_{\mathbf{x} \in \mathcal{S}_{\mathbf{e}_k^-} } \mathbf{x}^\dagger \mathbf{e}_k - \sum_{\mathbf{x} \in \mathcal{S}_{\mathbf{e}_k^+} } \mathbf{x}^\dagger \mathbf{e}_k \right), \tag{E.8}
\end{aligned}$$

where we have used the fact that $|\mathcal{S}_{\mathbf{e}_k^+}| = |\mathcal{S}_{\mathbf{e}_k^-}|$, and thus $|\mathcal{C} \setminus \mathcal{S}_{\mathbf{e}_k^+}| = |\mathcal{C} \setminus \mathcal{S}_{\mathbf{e}_k^-}|$.

As a last step, we determine expressions for $\sum_{\mathbf{x} \in \mathcal{S}_{\mathbf{e}_k^-}} \mathbf{x}$ and $\sum_{\mathbf{x} \in \mathcal{S}_{\mathbf{e}_k^+}} \mathbf{x}$. Due to the roughly symmetrical distribution of $\mathcal{S}_{\mathbf{e}_k^-}$ in its hemisphere, $\sum_{\mathbf{x} \in \mathcal{S}_{\mathbf{e}_k^-}} \mathbf{x}$ can be approximated as a scalar multiple of \mathbf{e}_k (see Fig. E.1):

$$\sum_{\mathbf{x} \in \mathcal{S}_{\mathbf{e}_k^-}} \mathbf{x} \approx \alpha_k \mathbf{e}_k, \tag{E.9}$$

where α_k is a positive real number. Similarly, we can make the approximation

$$\sum_{\mathbf{x} \in \mathcal{S}_{\mathbf{e}_k^+}} \mathbf{x} \approx -\alpha_k \mathbf{e}_k. \tag{E.10}$$

From (E.9), α_k can be expressed as

$$\alpha_k = \sum_{\mathbf{x} \in \mathcal{S}_{\mathbf{e}_k^-}} \frac{\mathbf{x}^\dagger \mathbf{e}_k}{\mathbf{e}_k^\dagger \mathbf{e}_k}. \tag{E.11}$$

Note that the points in $\mathcal{S}_{\mathbf{e}_k^-}$ define $|\mathcal{S}_{\mathbf{e}_k^-}|$ exhaustive and mutually exclusive sets in \mathcal{C} , where each set consists of all points which differ by an integer multiple of \mathbf{e}_k . Since $\frac{\mathbf{x}^\dagger \mathbf{e}_k}{\mathbf{e}_k^\dagger \mathbf{e}_k}$ is the projection of $\mathbf{x} \in \mathcal{S}_{\mathbf{e}_k^-}$ onto \mathbf{e}_k in terms of multiples of \mathbf{e}_k , there are approximately

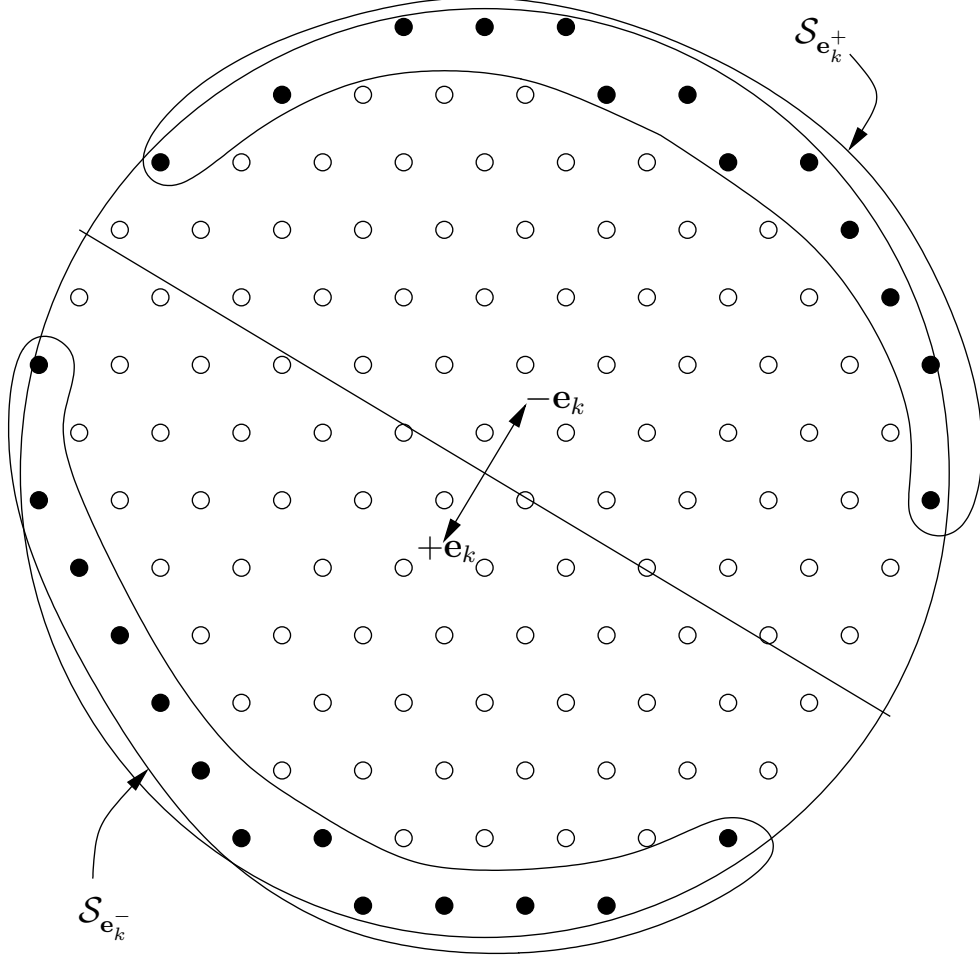


Figure E.1: The sets $\mathcal{S}_{e_k^+}$ and $\mathcal{S}_{e_k^-}$ are defined as the codewords that, when transmitted and decoded, cannot produce the error events $+\mathbf{e}_k$ and $-\mathbf{e}_k$ respectively. The plane perpendicular to the minimum-distance error event \mathbf{e}_k divides the spherical bounding region of the lattice into two hemispheres; the black dots of the two hemispheres constitute $\mathcal{S}_{e_k^+}$ and $\mathcal{S}_{e_k^-}$.

$2 \left(\frac{\mathbf{x}^\dagger \mathbf{e}_k}{\mathbf{e}_k^\dagger \mathbf{e}_k} \right) + 1$ points in the set associated with $\mathbf{x} \in \mathcal{S}_{e_k^-}$. Thus,

$$|\mathcal{C}| \approx \sum_{\mathbf{x} \in \mathcal{S}_{e_k^-}} \left[2 \left(\frac{\mathbf{x}^\dagger \mathbf{e}_k}{\mathbf{e}_k^\dagger \mathbf{e}_k} \right) + 1 \right] \quad (\text{E.12})$$

or, equivalently,

$$|\mathcal{C} \setminus \mathcal{S}_{e_k^+}| \approx \sum_{\mathbf{x} \in \mathcal{S}_{e_k^-}} 2 \left(\frac{\mathbf{x}^\dagger \mathbf{e}_k}{\mathbf{e}_k^\dagger \mathbf{e}_k} \right). \quad (\text{E.13})$$

Comparing this equation to (E.11), we conclude that

$$\alpha_k \approx \frac{|\mathcal{C} \setminus \mathcal{S}_{\mathbf{e}_k^+}|}{2}. \quad (\text{E.14})$$

Substituting (E.9), (E.10) and (E.14) into (E.8), we obtain

$$\begin{aligned} E[\mathbf{x}^\dagger \mathbf{e}] &\approx \frac{\Pr(\epsilon)}{K |\mathcal{C} \setminus \mathcal{S}_{\mathbf{e}_k^+}|} \sum_{k=1}^{K/2} \left(\frac{|\mathcal{C} \setminus \mathcal{S}_{\mathbf{e}_k^+}|}{2} \mathbf{e}_k^\dagger \mathbf{e}_k + \frac{|\mathcal{C} \setminus \mathcal{S}_{\mathbf{e}_k^+}|}{2} \mathbf{e}_k^\dagger \mathbf{e}_k \right) \\ &= \frac{d_{\min}^2}{2} \Pr(\epsilon), \end{aligned} \quad (\text{E.15})$$

where we have used the fact that $\mathbf{e}_k^\dagger \mathbf{e}_k = d_{\min}^2$. Combining this expression for $E[\mathbf{x}^\dagger \mathbf{e}]$ with (E.4), we obtain the desired result. \square

Bibliography

- [1] M. Abramowitz and I. A. Stegun, Eds., *Handbook of Mathematical Functions*. New York: Dover, 1965.
- [2] S. M. Aji and R. J. McEliece, “The generalized distributive law,” *IEEE Trans. Inform. Theory*, vol. 46, pp. 325–343, Mar. 2000.
- [3] M. E. Austin, “Decision-feedback equalization for digital communication over dispersive channels,” M.I.T. Res. Lab Electron., Tech. Rep. 461, Aug. 1967.
- [4] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, “Optimal decoding of linear codes for minimizing symbol error rate,” *IEEE Trans. Inform. Theory*, vol. 20, pp. 284–287, Mar. 1974.
- [5] Z. D. Bai and J. W. Silverstein, “No eigenvalues outside the support of the limiting spectral distribution of large dimensional sample covariance matrices,” *Ann. Probab.*, vol. 26, no. 1, pp. 316–345, 1998.
- [6] Z. D. Bai and Y. Q. Yin, “Limit of the smallest eigenvalue of a large dimensional sample covariance matrix,” *Annals of Probability*, vol. 21, pp. 1275–1294, 1993.
- [7] S. Beheshti, S. H. Isabelle, and G. W. Wornell, “Joint intersymbol and multiple-access interference suppression algorithms for CDMA systems,” *European Trans. Telecomm. & Related Technol.*, vol. 9, pp. 403–18, Sep.–Oct. 1998.
- [8] C. A. Belfiore and J. H. Park, Jr., “Decision-feedback equalization,” *Proc. IEEE*, vol. 67, pp. 1143–1156, Aug. 1979.
- [9] C. Berrou, A. Glavieux, and P. Thitimajshima, “Near Shannon limit error-correcting coding and decoding: Turbo-codes,” in *Proc. 1993 Int. Conf. Commun.*, vol. 2, pp. 1064–1070, May 1993.
- [10] A. T. Bharucha-Reid and M. Sambandham, *Random Polynomials*. Orlando, FL: Academic Press, 1986.
- [11] E. Biglieri, “Ungerboeck codes do not shape the signal power spectrum,” *IEEE Trans. Inform. Theory*, vol. IT-32, pp. 595–596, July 1986.
- [12] J. Boutros, G. Caire, E. Viterbo, H. Sawaya, and S. Vialle, “Turbo code at 0.03 dB from capacity limit,” in *Proc. IEEE Int. Symp. Inform. Theory*, Lausanne, Switzerland, June–July 2002, p. 56.

- [13] J. Boutros and E. Viterbo, "Signal space diversity: A power- and bandwidth-efficient diversity technique for the Rayleigh fading channel," *IEEE Trans. Inform. Theory*, vol. 44, pp. 1453–1467, July 1998.
- [14] G. Caire and S. Shamai, "On the achievable throughput of a multiantenna Gaussian broadcast channel," *IEEE Trans. Inform. Theory*, vol. 49, pp. 1691–1706, July 2003.
- [15] A. M. Chan and G. W. Wornell, "A class of block-iterative equalizers for intersymbol interference channels: Fixed channel results," *IEEE Trans. Commun.*, vol. 49, pp. 1966–1976, Nov. 2001.
- [16] J. M. Cioffi, G. P. Dudevoir, M. V. Eyuboglu, and G. D. Forney, Jr. "MMSE decision-feedback equalizers and coding—Part II: Coding results," *IEEE Trans. Commun.*, vol. 43, pp. 2595–2604, Oct. 1995.
- [17] S.-Y. Chung, G. D. Forney, Jr., T. J. Richardson, and R. Urbanke, "On the design of low-density parity-check codes within 0.0045 dB of the Shannon limit," *IEEE Commun. Letters*, vol. 5, pp. 58–60, Feb. 2001.
- [18] S.-Y. Chung, T. J. Richardson, and R. L. Urbanke, "Analysis of sum-product decoding of low-density parity-check codes using a Gaussian approximation," *IEEE Trans. Inform. Theory*, vol. 47, pp. 657–670, Feb. 2001.
- [19] D. Divsalar, H. Jin and R. J. McEliece, "Coding theorems for 'turbo-like' codes," in *Proc. 1998 Allerton Conf.*, Allerton, IL, pp. 201–210, Sept. 1998.
- [20] C. Douillard, M. Jézéquel, C. Berrou, A. Picart, P. Didier and A. Glavieux, "Iterative correction of intersymbol interference: Turbo equalization," *Eur. Trans. Telecomm.*, vol. 6, pp. 507–511, Sept.–Oct. 1995.
- [21] A. Duel-Hallen, "Decorrelating decision-feedback multiuser detector for synchronous CDMA," *IEEE Trans. Commun.*, vol. 41, pp. 285–290, Feb. 1993.
- [22] Y. C. Eldar and A. M. Chan, "On the Asymptotic Performance of the Decorrelator," *IEEE Trans. Inform. Theory*, vol. 49, pp. 2309–2313, Sept. 2003
- [23] U. Erez and R. Zamir, "Lattice decoding can achieve $\frac{1}{2} \log(1 + \text{SNR})$ on the AWGN channel using nested codes," in *Proc. Int. Symp. Inform. Theory*, Washington, DC, p. 125, June 2001.
- [24] J. E. Evans and D. N. C. Tse, "Large system performance of linear multiuser receivers in multipath fading channels," *IEEE Trans. Inform. Theory*, vol. 46, pp. 2059–2078, Sep. 2000.
- [25] W. Feller, *An Introduction to Probability Theory and Its Applications*, vol. I, 3rd ed. New York: Wiley, 1968.
- [26] R. F. H. Fischer, *Precoding and Signal Shaping for Digital Transmission*. New York: Wiley, 2002.
- [27] G. D. Forney, Jr., "Maximum likelihood sequence estimation of digital sequences in the presence of intersymbol interference," *IEEE Trans. Inform. Theory*, vol. IT-18, pp.363–378, May 1972.

- [28] G. D. Forney, Jr. and G. Ungerboeck, "Modulation and coding for linear Gaussian channels," *IEEE Trans. Inform. Theory*, vol. 44, pp. 2384–2415, Oct. 1998.
- [29] G. J. Foschini, "Layered space-time architecture for wireless communication in a fading environment when using multi-element antennas," *Bell Labs Tech. J.*, pp. 41–59, Autumn 1996.
- [30] R. G. Gallager, "Low-density parity-check codes," *IRE Trans. Inform. Theory*, vol. IT-8, pp. 21–28, Jan. 1962.
- [31] R. G. Gallager, *Low-Density Parity-Check Codes*. Cambridge, MA: MIT Press, 1963.
- [32] A. Gersho and T. L. Lim, "Adaptive cancellation of intersymbol interference for data transmission," *Bell Syst. Tech. J.*, vol. 60, pp. 1997–2021, Nov. 1981.
- [33] A. Glavieux, C. Laot, and J. Labat, "Turbo equalization over a frequency selective channel," in *Proc. Int. Symp. Turbo Codes*, (Brest, France), pp. 96–102, Sept. 1997.
- [34] I. S. Gradshteyn and I. M. Ryzhik, *Table of Integrals, Series, and Products*, 6th ed. New York: Academic Press, 2000.
- [35] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 3rd ed. Baltimore, MD: Johns Hopkins Univ. Press, 1996.
- [36] R. M. Gray and D. L. Neuhoff, "Quantization," *IEEE Trans. Inform. Theory*, vol. 44, pp. 2325–2383, Oct. 1998.
- [37] J. Hagenauer, E. Offer, and L. Papke, "Iterative decoding of binary block and convolutional codes," *IEEE Trans. Inform. Theory*, vol. 42, pp. 429–445, Mar. 1996.
- [38] H. Harashima and H. Miyakawa, "A method of code conversion for a digital communication channel with intersymbol interference," *IEEE Trans. Commun.*, vol. COM-20, pp. 774–780, Aug. 1972.
- [39] S. Haykin, *Adaptive Filter Theory*, 3rd ed. Englewood Cliffs, NJ: Prentice Hall, 1996.
- [40] M. L. Honig and H. V. Poor, "Adaptive interference suppression" in *Wireless Communications: Signal Processing Perspectives* (H. V. Poor and G. W. Wornell, eds.). Upper Saddle River, NJ: Prentice-Hall, 1998.
- [41] H. Imai and S. Hirakawa, "A new multilevel coding method using error correcting codes," *IEEE Trans. Inform. Theory*, vol. 23, pp. 371–377, May 1977.
- [42] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inform. Theory*, vol. 47, pp. 498–519, Feb. 2001.
- [43] E. A. Lee and D. G. Messerschmitt, *Digital Communication*, 2nd ed. Boston, MA: Kluwer, 1994.
- [44] J. Leech and N. J. A. Sloane, "Sphere packing and error-correcting codes," *Canad. J. Math.*, vol. 23, pp. 718–745, 1971.
- [45] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, and D. A. Spielman, "Improved low-density parity-check codes using irregular graphs," *IEEE Trans. Inform. Theory*, pp. 585–598, Feb. 2001.

- [46] H. Lütkepohl, *Handbook of Matrices*. Chichester, England: Wiley, 1996.
- [47] U. Madhow and M. L. Honig, “MMSE interference suppression for direct-sequence spread spectrum CDMA,” *IEEE Trans. Commun.*, vol. 42, pp. 3178–3188, Dec. 1994.
- [48] T. L. Marzetta and B. M. Hochwald, “Capacity of a mobile multiple-antenna communication link in Rayleigh flat fading,” *IEEE Trans. Inform. Theory*, vol. 45, pp. 139–157, Jan. 1999.
- [49] R. Muirhead, *Aspects of Multivariate Statistical Theory*. New York: Wiley, 1982.
- [50] R. Price, “Nonlinearly feedback-equalized PAM versus capacity for noisy filter channels,” in *Proc. 1972 IEEE Int. Conf. Commun.*, pp. 22-12 to 22-17, June 1972.
- [51] J. G. Proakis, *Digital Communications*, 3rd ed. New York: McGraw-Hill, 1995.
- [52] T. J. Richardson, M. A. Shokrollahi, and R. L. Urbanke, “Design of capacity-approaching irregular low-density parity-check codes,” *IEEE Trans. Inform. Theory*, vol. 47, pp. 619–637, Feb. 2001.
- [53] T. J. Richardson and R. L. Urbanke, “The capacity of low-density parity-check codes under message-passing decoding,” *IEEE Trans. Inform. Theory*, vol. 47, pp. 599–618, Feb. 2001.
- [54] K. H. Rosen, *Discrete Mathematics and Its Applications*, 5th ed. Boston: McGraw-Hill, 2003.
- [55] A. Ruiz, J. M. Cioffi, and S. Kasturia, “Discrete multiple tone modulation with coset coding for the spectrally shaped channel,” *IEEE Trans. Commun.* vol. 40, pp. 1012–1029, June 1992.
- [56] C. E. Shannon, “A mathematical theory of communication,” *Bell Syst. Tech. J.*, vol. 27, pp. 379–423 and pp. 623–656, July and Oct. 1948.
- [57] D. Shnidman, “A generalized Nyquist criterion and an optimum linear receiver for a pulse modulation system,” *Bell System Tech. J.*, vol. 46, pp. 2163–2177, Nov. 1967.
- [58] S. ten Brink, “Designing iterative decoding schemes with the extrinsic information transfer chart,” *AEÜ Int. J. Electron. Commun.*, vol. 54, pp. 389–398, Nov. 2000.
- [59] M. Tomlinson, “New automatic equalizer employing modulo arithmetic,” *Electron. Lett.*, vol. 7, pp. 138–139, Mar. 1971.
- [60] M. Tüchler, R. Koetter, and A. C. Singer, “Turbo equalization: Principles and new results,” *IEEE Trans. Commun.*, vol. 50, pp. 754–767, May 2002.
- [61] D. W. Tufts, “Nyquist’s problem—The joint optimization of transmitter and receiver in pulse amplitude modulation,” *Proc. IEEE*, vol. 53, pp. 248–259, Mar. 1965.
- [62] G. Ungerboeck, “Channel coding with multilevel/phase signals,” *IEEE Trans. Inform. Theory*, vol. IT-28, pp. 55–67, Jan. 1982.
- [63] R. Urbanke and B. Rimoldi, “Lattice codes can achieve capacity on the AWGN channel,” *IEEE Trans. Inform. Theory*, vol. 44, pp. 273–278, Jan. 1998.

- [64] M. K. Varanasi and B. Aazhang, "Near-optimum detection in synchronous code-division multiple-access systems," *IEEE Trans. Commun.*, vol. 39, pp. 725–736, May 1991.
- [65] S. Verdú, "Computational complexity of optimal multiuser detection," *Algorithmica*, vol. 4, no. 3, pp. 303–312, 1989.
- [66] S. Verdú, *Multiuser Detection*. Cambridge, U.K.: Cambridge Univ. Press, 1998.
- [67] E. Viterbo and E. Biglieri, "A universal lattice decoder," in *14^{eme} Colloq. GRETSI*, Juan-les-Pins, France, pp. 611-614, Sept. 1993.
- [68] U. Wachsmann, R. F. H. Fischer, and J. B. Huber, "Multilevel codes: Theoretical concepts and practical design rules," *IEEE Trans. Inform. Theory*, vol. 45, pp. 1361–1391, July 1999.
- [69] N. Wiberg, "Codes and decoding on general graphs," Ph.D. dissertation, Linköping Univ., Sweden, 1996.
- [70] Z. Xie, R. Short, and C. Rushforth, "A family of suboptimum detectors for coherent multiuser communications," *IEEE J. Selected Areas Commun.*, vol. 8, pp. 683–690, May 1990.