

Fundamentals of Classical and Modern Error-Correcting Codes

Using easy-to-follow mathematics, this textbook provides comprehensive coverage of block codes and techniques for reliable communications and data storage. It covers major code designs and constructions from geometric, algebraic, and graph-theoretic points of view, decoding algorithms, error-control additive white Gaussian noise (AWGN) and erasure, and reliable data recovery. It simplifies a highly mathematical subject to a level that can be understood and applied with a minimum background in mathematics, provides step-by-step explanation of all covered topics, both fundamental and advanced, and includes plenty of practical illustrative examples to assist understanding. Numerous homework problems are included to strengthen student comprehension of new and abstract concepts, and a solution manual is available online for instructors. Modern developments, including polar codes, are also covered.

This is an essential textbook for senior undergraduates and graduates taking introductory coding courses, students taking advanced full-year graduate coding courses, and professionals working on coding for communications and data storage.

Shu Lin is Adjunct Professor in the Department of Electrical and Computer Engineering at the University of California, Davis, and an IEEE Life Fellow. He has authored and coauthored several books, including *LDPC Code Designs, Constructions, and Unification* (Cambridge University Press, 2016) and *Channel Codes: Classical and Modern* (Cambridge University Press, 2009).

Juane Li is Staff Systems Architect at Micron Technology Inc., San Jose, having previously completed her PhD at the University of California, Davis. She is also a coauthor of *LDPC Code Designs, Constructions, and Unification* (Cambridge University Press, 2016).

Fundamentals of Classical and Modern Error-Correcting Codes

SHU LIN

University of California, Davis

JUANE LI

Micron Technology Inc., San Jose



CAMBRIDGE
UNIVERSITY PRESS

Cambridge University Press
978-1-316-51262-3 — Fundamentals of Classical and Modern Error-Correcting Codes
Shu Lin , Juane Li
Frontmatter
[More Information](#)

CAMBRIDGE
UNIVERSITY PRESS

University Printing House, Cambridge CB2 8BS, United Kingdom
One Liberty Plaza, 20th Floor, New York, NY 10006, USA
477 Williamstown Road, Port Melbourne, VIC 3207, Australia
314–321, 3rd Floor, Plot 3, Splendor Forum, Jasola District Centre, New Delhi – 110025, India
103 Penang Road, #05–06/07, Visioncrest Commercial, Singapore 238467

Cambridge University Press is part of the University of Cambridge.
It furthers the University’s mission by disseminating knowledge in the pursuit of education, learning, and research at the highest international levels of excellence.

www.cambridge.org
Information on this title: www.cambridge.org/highereducation/isbn/9781316512623
DOI: 10.1017/9781009067928

© Cambridge University Press 2022

This publication is in copyright. Subject to statutory exception and to the provisions of relevant collective licensing agreements, no reproduction of any part may take place without the written permission of Cambridge University Press.

First published 2022

Printed in the United Kingdom by TJ Books Limited, Padstow, Cornwall, 2022

A catalogue record for this publication is available from the British Library.

Library of Congress Cataloging-in-Publication Data

Names: Lin, Shu, 1937– author. | Li, Juane, author.

Title: Fundamentals of classical and modern error-correcting codes / Shu Lin, University of California, Davis, Juane Li, Micron Technology, San Jose.

Description: Cambridge, United Kingdom ; New York, NY, USA : Cambridge University Press, 2021. | Includes bibliographical references and index.

Identifiers: LCCN 2021025406 (print) | LCCN 2021025407 (ebook) | ISBN 9781316512623 (hardback) | ISBN 9781009067928 (epub)

Subjects: LCSH: Error-correcting codes (Information theory)

Classification: LCC TK5102.96 .L53 2021 (print) | LCC TK5102.96 (ebook) | DDC 005.7/2–dc23

LC record available at <https://lcn.loc.gov/2021025406>

LC ebook record available at <https://lcn.loc.gov/2021025407>

ISBN 978-1-316-51262-3 Hardback

Additional resources for this publication at www.cambridge.org/lin-li

Cambridge University Press has no responsibility for the persistence or accuracy of URLs for external or third-party internet websites referred to in this publication and does not guarantee that any content on such websites is, or will remain, accurate or appropriate.

Contents

<i>List of Figures</i>	page xiii
<i>List of Tables</i>	xxi
<i>Preface</i>	xxv
<i>Acknowledgments</i>	xxviii
1 Coding for Reliable Digital Information Transmission and Storage	1
1.1 Introduction	2
1.2 Categories of Error-Correcting Codes	4
1.3 Modulation and Demodulation	6
1.4 Hard-Decision and Soft-Decision Decodings	8
1.5 Maximum A Posteriori and Maximum Likelihood Decodings	9
1.6 Channel Capacity on Transmission Rate	12
1.7 Classification of Channel Errors	13
1.8 Error-Control Strategies	14
1.9 Measures of Performance	16
1.10 Contents of the Book	18
References	23
2 Some Elements of Modern Algebra and Graphs	25
2.1 Groups	25
2.1.1 Basic Definitions and Concepts	25
2.1.2 Finite Groups	26
2.1.3 Subgroups and Cosets	29
2.2 Finite Fields	31
2.2.1 Basic Definitions and Concepts	32
2.2.2 Prime Fields	35
2.2.3 Finite Fields with Orders of Prime Powers	36
2.3 Polynomials over Galois Field $GF(2)$	39
2.4 Construction of Galois Field $GF(2^m)$	43
2.5 Basic Properties and Structures of Galois Field $GF(2^m)$	51
2.6 Computations over Galois Field $GF(2^m)$	58
2.7 A General Construction of Finite Fields	59

<i>Contents</i>	vi
2.8 Vector Spaces over Finite Fields	60
2.8.1 Basic Definitions and Concepts	60
2.8.2 Vector Spaces over Binary Field $GF(2)$	62
2.8.3 Vector Spaces over Nonbinary Field $GF(q)$	67
2.9 Matrices over Finite Fields	67
2.9.1 Concepts of Matrices over $GF(2)$	67
2.9.2 Operations of Matrices over $GF(2)$	69
2.9.3 Matrices over Nonbinary Field $GF(q)$	73
2.9.4 Determinants	74
2.10 Graphs	78
2.10.1 Basic Definitions and Concepts	78
2.10.2 Bipartite Graphs	82
Problems	83
References	86
3 Linear Block Codes	89
3.1 Definitions	89
3.2 Generator and Parity-Check Matrices	90
3.3 Systematic Linear Block Codes	96
3.4 Error Detection with Linear Block Codes	99
3.5 Syndrome and Error Patterns	102
3.6 Weight Distribution and Probability of Undetected Error	103
3.7 Minimum Distance of Linear Block Codes	105
3.8 Decoding of Linear Block Codes	109
3.9 Standard Array for Decoding Linear Block Codes	110
3.9.1 A Standard Array Decoding	110
3.9.2 Syndrome Decoding	116
3.10 Shortened and Extended Codes	118
3.11 Nonbinary Linear Block Codes	120
Problems	120
References	123
4 Binary Cyclic Codes	125
4.1 Characterization of Cyclic Codes	125
4.2 Structural Properties of Cyclic Codes	127
4.3 Existence of Cyclic Codes	131
4.4 Generator and Parity-Check Matrices of Cyclic Codes	133
4.5 Encoding of Cyclic Codes in Systematic Form	136
4.6 Syndrome Calculation and Error Detection	142
4.7 General Decoding of Cyclic Codes	145
4.8 Error-Trapping Decoding for Cyclic Codes	150
4.9 Shortened Cyclic Codes	153
4.10 Hamming Codes	154
4.11 Cyclic Redundancy Check Codes	158
4.12 Quadratic Residue Codes	159

<i>Contents</i>	vii
4.13 Quasi-cyclic Codes	161
4.13.1 Definitions and Fundamental Structures	161
4.13.2 Generator and Parity-Check Matrices in Systematic Circulant Form	163
4.13.3 Encoding of QC Codes	164
4.13.4 Generator and Parity-Check Matrices in Semi-systematic Circulant Form	168
4.13.5 Shortened QC Codes	176
4.14 Nonbinary Cyclic Codes	176
4.15 Remarks	177
Problems	177
References	182
5 BCH Codes	185
5.1 Primitive Binary BCH Codes	185
5.2 Structural Properties of BCH Codes	190
5.3 Minimum Distance of BCH Codes	192
5.4 Syndrome Computation and Error Detection	196
5.5 Syndromes and Error Patterns	198
5.6 Error-Location Polynomials of BCH Codes	199
5.7 A Procedure for Decoding BCH Codes	200
5.8 Berlekamp–Massey Iterative Algorithm	201
5.9 Simplification of Decoding Binary BCH Codes	205
5.10 Finding Error Locations and Error Correction	211
5.11 Nonprimitive Binary BCH Codes	212
5.12 Remarks	216
Problems	216
References	218
6 Nonbinary BCH Codes and Reed–Solomon Codes	220
6.1 Nonbinary Primitive BCH Codes	221
6.2 Decoding Steps of Nonbinary BCH Codes	224
6.3 Syndrome and Error Pattern of Nonbinary BCH Codes	225
6.4 Error-Location Polynomial of Nonbinary BCH Codes	226
6.5 Error-Value Evaluator	231
6.6 Decoding of Nonbinary BCH Codes	232
6.7 Key-Equation	235
6.8 Reed–Solomon Codes	235
6.8.1 Primitive Reed–Solomon Codes	236
6.8.2 Nonprimitive Reed–Solomon Codes	237
6.9 Decoding Reed–Solomon Codes with Berlekamp–Massey Iterative Algorithm	238
6.10 Euclidean Algorithm for Finding GCD of Two Polynomials	243
6.11 Solving the Key-Equation with Euclidean Algorithm	247
6.12 Weight Distribution and Probability of Undetected Error of Reed–Solomon Codes	251

<i>Contents</i>	viii
6.13 Remarks	252
Problems	252
References	255
7 Finite Geometries, Cyclic Finite-Geometry Codes, and Majority-Logic Decoding	258
7.1 Fundamental Concepts of Finite Geometries	259
7.2 Majority-Logic Decoding of Finite-Geometry Codes	261
7.3 Euclidean Geometries over Finite Fields	266
7.3.1 Basic Concepts and Properties	266
7.3.2 A Realization of Euclidean Geometries	269
7.3.3 Subgeometries of Euclidean Geometries	275
7.4 Cyclic Codes Constructed Based on Euclidean Geometries	277
7.4.1 Cyclic Codes on Two-Dimensional Euclidean Geometries	278
7.4.2 Cyclic Codes on Multi-Dimensional Euclidean Geometries	284
7.5 Projective Geometries	288
7.6 Cyclic Codes Constructed Based on Projective Geometries	292
7.7 Remarks	296
Problems	297
References	300
8 Reed–Muller Codes	303
8.1 A Review of Euclidean Geometries over GF(2)	304
8.2 Constructing RM Codes from Euclidean Geometries over GF(2)	305
8.3 Encoding of RM Codes	311
8.4 Successive Retrieval of Information Symbols	314
8.5 Majority-Logic Decoding through Successive Cancellations	319
8.6 Cyclic RM Codes	324
8.7 Remarks	325
Problems	326
References	327
9 Some Coding Techniques	331
9.1 Interleaving	332
9.2 Direct Product	334
9.3 Concatenation	341
9.3.1 Type-1 Serial Concatenation	342
9.3.2 Type-2 Serial Concatenation	344
9.3.3 Parallel Concatenation	346
9.4 $ \mathbf{u} \mathbf{u} + \mathbf{v} $ -Construction	347
9.5 Kronecker Product	348
9.6 Automatic-Repeat-Request Schemes	353
9.6.1 Basic ARQ Schemes	353
9.6.2 Mixed-Mode SR-ARQ Schemes	358
9.6.3 Hybrid ARQ Schemes	359
Problems	362
References	363

<i>Contents</i>	ix
10 Correction of Error-Bursts and Erasures	367
10.1 Definitions and Structures of Burst-Error-Correcting Codes	368
10.2 Decoding of Single Burst-Error-Correcting Cyclic Codes	370
10.3 Fire Codes	373
10.4 Short Optimal and Nearly Optimal Single Burst-Error-Correcting Cyclic Codes	376
10.5 Interleaved Codes for Correcting Long Error-Bursts	377
10.6 Product Codes for Correcting Error-Bursts	379
10.7 Phased-Burst-Error-Correcting Codes	380
10.7.1 Interleaved and Product Codes	380
10.7.2 Codes Derived from RS Codes	380
10.7.3 Burton Codes	381
10.8 Characterization and Correction of Erasures	382
10.8.1 Correction of Errors and Erasures over BSECs	383
10.8.2 Correction of Erasures over BECs	385
10.8.3 RM Codes for Correcting Random Erasures	388
10.9 Correcting Erasure-Bursts over BBECs	390
10.9.1 Cyclic Codes for Correcting Single Erasure-Burst	391
10.9.2 Correction of Multiple Random Erasure-Bursts	394
10.10 RS Codes for Correcting Random Errors and Erasures	394
Problems	402
References	403
11 Introduction to Low-Density Parity-Check Codes	406
11.1 Definitions and Basic Concepts	407
11.2 Graphical Representation of LDPC Codes	410
11.3 Original Construction of LDPC Codes	414
11.3.1 Gallager Codes	415
11.3.2 MacKay Codes	416
11.4 Decoding of LDPC Codes	416
11.4.1 One-Step Majority-Logic Decoding	418
11.4.2 Bit-Flipping Decoding	422
11.4.3 Weighted One-Step Majority-Logic and Bit-Flipping Decodings	425
11.5 Iterative Decoding Based on Belief-Propagation	427
11.5.1 Message Passing	428
11.5.2 Sum-Product Algorithm	429
11.5.3 Min-Sum Algorithm	436
11.6 Error Performance of LDPC Codes with Iterative Decoding	439
11.6.1 Error-Floor	439
11.6.2 Decoding Threshold	441
11.6.3 Overall Performance and Its Determinating Factors	442
11.7 Iterative Decoding of LDPC Codes over BECs	446
11.8 Categories of LDPC Code Constructions	449
11.9 Nonbinary LDPC Codes	450
Problems	453
References	456

<i>Contents</i>	x
12 Cyclic and Quasi-cyclic LDPC Codes on Finite Geometries	464
12.1 Cyclic-FG-LDPC Codes	465
12.2 A Complexity-Reduced Iterative Algorithm for Decoding Cyclic-FG-LDPC Codes	472
12.3 QC-EG-LDPC Codes	480
12.4 QC-PG-LDPC Codes	487
12.5 Construction of QC-EG-LDPC Codes by CPM-Dispersion	489
12.6 Masking Techniques	493
12.7 Construction of QC-FG-LDPC Codes by Circulant-Decomposition	496
12.8 A Complexity-Reduced Iterative Algorithm for Decoding QC-FG-LDPC Codes	503
12.9 Remarks	509
Problems	511
References	513
13 Partial Geometries and Their Associated QC-LDPC Codes	518
13.1 CPM-Dispersions of Finite-Field Elements	518
13.2 Matrices with RC-Constrained Structure	520
13.3 Definitions and Structural Properties of Partial Geometries	522
13.4 Partial Geometries Based on Prime-Order Cyclic Subgroups of Finite Fields and Their Associated QC-LDPC Codes	524
13.5 Partial Geometries Based on Prime Fields and Their Associated QC-LDPC Codes	531
13.6 Partial Geometries Based on Balanced Incomplete Block Designs and Their Associated QC-LDPC Codes	538
13.6.1 BIBDs and Partial Geometries	538
13.6.2 Class-1 Bose $(N, M, t, r, 1)$ -BIBDs	541
13.6.3 Class-2 Bose $(N, M, t, r, 1)$ -BIBDs	549
13.7 Remarks	556
Problems	556
References	559
14 Quasi-cyclic LDPC Codes Based on Finite Fields	562
14.1 Construction of QC-LDPC Codes Based on CPM-Dispersion	563
14.2 Construction of Type-I QC-LDPC Codes Based on Two Subsets of a Finite Field	564
14.3 Construction of Type-II QC-LDPC Codes Based on Two Subsets of a Finite Field	576
14.4 Masking-Matrix Design	580
14.4.1 Type-1 Design	580
14.4.2 Type-2 Design	582
14.4.3 Type-3 Design	584
14.5 A Search Algorithm for $2 \times 2/3 \times 3$ SM-Constrained Base Matrices for Constructing Rate-1/2 QC-LDPC Codes	586
14.6 Designs of 2×2 SM-Constrained RS Base Matrices	590

<i>Contents</i>	xi
14.7 Construction of Type-III QC-LDPC Codes Based on RS Codes	592
14.8 Construction of QC-RS-LDPC Codes with Girths at Least 8	598
14.9 A Special Class of QC-RS-LDPC Codes with Girth 8	603
14.10 Optimal Codes for Correcting Two Random CPM-Phased Erasure-Bursts	608
14.11 Globally Coupled LDPC Codes	612
14.12 Remarks	619
Problems	619
References	623
15 Graph-Theoretic LDPC Codes	628
15.1 Protograph-Based LDPC Codes	629
15.2 A Matrix-Theoretic Method for Constructing Protograph-Based LDPC Codes	640
15.3 Masking Matrices as Protomatrices	655
15.4 LDPC Codes on Progressive Edge-Growth Algorithms	670
15.5 Remarks	676
Problems	677
References	679
16 Collective Encoding and Soft-Decision Decoding of Cyclic Codes of Prime Lengths in Galois Fourier Transform Domain	684
16.1 Cyclic Codes of Prime Lengths and Their Hadamard Equivalents	685
16.2 Composing, Cascading, and Interleaving a Cyclic Code of Prime Length and Its Hadamard Equivalents	688
16.2.1 Composing	688
16.2.2 Cascading and Interleaving	689
16.3 Galois Fourier Transform of ICC Codes	692
16.4 Structural Properties of GFT-ICC Codes	694
16.5 Collective Encoding of GFT-ICC-LDPC Codes	696
16.6 Collective Iterative Soft-Decision Decoding of GFT-ICC-LDPC Codes	698
16.7 Analysis of the GFT-ISDD Scheme	700
16.7.1 Performance Measurements	700
16.7.2 Complexity	701
16.8 Joint Decoding of RS Codes with GFT-ISDD Scheme	701
16.9 Joint Decoding of BCH Codes with GFT-ISDD Scheme	706
16.10 Joint Decoding of QR Codes with GFT-ISDD Scheme	709
16.11 Code Shortening and Rate Reduction	710
16.11.1 Shortened GFT-ICC Codes	710
16.11.2 Reductions of Code Rate	713
16.12 Erasure Correction of GFT-ICC-RS-LDPC Codes	715
16.13 Remarks	717
Problems	717
References	719

<i>Contents</i>	xii
17 Polar Codes	721
17.1 Kronecker Matrices and Their Structural Properties	721
17.2 Kronecker Mappings and Their Logical Implementations	724
17.3 Kronecker Vector Spaces and Codes	735
17.4 Definition and Polarized Encoding of Polar Codes	738
17.5 Successive Information Retrieval from a Polarized Codeword	741
17.6 Channel Polarization	744
17.6.1 Some Elements of Information Theory	745
17.6.2 Polarization Process	747
17.6.3 Channel Polarization Theorem	760
17.7 Construction of Polar Codes	766
17.8 Successive Cancellation Decoding	768
17.8.1 SC Decoding of Polar Codes of Length $N = 2$	770
17.8.2 SC Decoding of Polar Codes of Length $N = 4$	773
17.8.3 SC Decoding of Polar Codes of Length $N = 2^\ell$	778
17.9 Remarks	779
Problems	780
References	781
Appendix A Factorization of $X^n + 1$ over $\text{GF}(2)$	784
Appendix B A $2 \times 2/3 \times 3$ SM-Constrained Masked Matrix Search Algorithm	785
Appendix C Proof of Theorem 14.4	786
Appendix D The 2×2 CPM-Array Cycle Structure of the Tanner Graph of $C_{\text{RS},n}(2, n)$	791
Appendix E Iterative Decoding Algorithm for Nonbinary LDPC Codes	793
E.1 Introduction	793
E.2 Algorithm Derivation	794
E.2.1 VN Update	795
E.2.2 CN Update: Complex Version	795
E.2.3 CN Update: Fast Hadamard Transform Version	796
E.3 The Nonbinary LDPC Decoding Algorithm	800
<i>Index</i>	802

Figures

1.1	Block diagram of a typical data-transmission (or data-storage) system	page 2
1.2	A simplified model of a coded system	4
1.3	A binary convolutional encoder with $k = 1$, $n = 2$, and $m = 2$	6
1.4	Transition probability diagrams: (a) BSC and (b) BI-DMC	8
1.5	A coded communication system with binary-input and N -ary-output discrete memoryless AWGN channel	10
1.6	The two-state Gilbert–Elliott model	14
1.7	Models for (a) BEC and (b) BSEC, where p_e represents the erasure probability and p_t represents the error probability	15
1.8	The BER performances of a coded communication using a $(127, 113)$ binary block code	17
1.9	Shannon limit E_b/N_0 (dB) as a function of code rate R	18
2.1	A graph with seven nodes and eight edges	79
2.2	Simple graphs	80
2.3	A bipartite graph	82
3.1	Systematic format of codewords in an (n, k) linear block code	96
3.2	Decoding regions for linear block codes	110
4.1	An encoding circuit for an (n, k) cyclic code with generator polynomial $\mathbf{g}(X) = 1 + g_1X + \cdots + g_{n-k-1}X^{n-k-1} + X^{n-k}$	139
4.2	An encoding circuit for the $(7, 4)$ cyclic code given in Example 4.5	140
4.3	Syndrome calculation circuit for an (n, k) cyclic code	143
4.4	Syndrome calculation circuit for the $(7, 4)$ cyclic code in Example 4.7	143
4.5	A general cyclic code decoder	147
4.6	A Meggitt decoder for the $(7, 4)$ cyclic code in Example 4.8	149
4.7	An error-trapping decoder for cyclic codes	151
4.8	A Hamming code decoder	156
4.9	A CSRAA encoder circuit	166
4.10	A CSRAA-based QC code encoder	167
5.1	An error-location search and correction circuit	212
6.1	A decoding block diagram of a q -ary BCH decoder	233
6.2	Error performances of the $(255, 239)$ and $(255, 223)$ RS codes over $\text{GF}(2^8)$	243

<i>List of Figures</i>	xiv
7.1 A five-point finite geometry	260
7.2 Error performances of the (1057, 813) cyclic PG code $C_{PG}(2, 2^5)$ in Example 7.12 decoded with the OSMLD	296
7.3 A six-point finite geometry	297
9.1 A product codeword array $\mathbf{v}_{1 \times 2}$	335
9.2 Diagonal transmission of a codeword array in a cyclic product code	340
9.3 A turbo codeword array	341
9.4 A type-1 serial concatenated coding system	342
9.5 A type-2 serial concatenated coding system	344
9.6 A parallel concatenated coding system	346
9.7 Stop-and-wait ARQ	354
9.8 Go-back- N ARQ with $N = 4$	355
9.9 Selective-repeat ARQ	355
9.10 An SR/GBN-ARQ scheme with $\lambda = 1$ and $N = 4$	359
10.1 An error-trapping decoder for l -burst-error-correcting cyclic codes	372
10.2 An error-trapping decoder for the (5, 1, 1)-Fire code in Example 10.1	375
10.3 Mathematical models: (a) BEC and (b) BSEC	383
11.1 The Tanner graph of the (10, 6) LDPC code given in Example 11.1	413
11.2 The Tanner graph of the (15, 7) LDPC code given in Example 11.2	413
11.3 Message passing from VN v_j to its neighbor (or adjacent) CNs	429
11.4 Message passing from CN c_i to its neighbor (or adjacent) VNs	429
11.5 A VN decoder in an SPA decoder	430
11.6 A CN decoder in an SPA decoder	431
11.7 A Tanner graph with a cycle of length 4 and message passing	432
11.8 A plot of the $\phi(x)$ function together with its approximates $2e^{-x}$ and $\log(x/2)$	435
11.9 The BER performances of the (4095, 3367) LDPC code given in Example 11.12 decoded with the OSML, BF, weighted BF, MSA, and SPA decodings	438
11.10 The error-floor phenomenon	439
11.11 The BER and BLER performances of the (3934, 3653) LDPC code given in Example 11.13 decoded with SPA and MSA	440
11.12 (a) The Tanner graph of the (10, 6) LDPC code given in Example 11.1, (b) a (4, 2) trapping set, and (c) a (3, 2) elementary trapping set	443
11.13 The BER and BLER performances of the (4095, 3367) LDPC code given in Example 11.12 decoded with 5, 10, 50, and 100 iterations of the MSA	447
11.14 The UEBR and UEBLR performances of the (4095, 3367) cyclic finite-geometry LDPC code given in Example 11.12 over a BEC	448
11.15 Two stopping sets of the (10, 6) LDPC code given in Example 11.1	448

<i>List of Figures</i>	xv
11.16 The Tanner graph of the 8-ary (10, 5) LDPC code given in Example 11.14	452
12.1 The BER and BLER performances of the (1023, 781) cyclic-EG-LDPC code $C_{EG,cyc}(2, 2^5)$ given in Example 12.1 decoded with 5, 10, and 50 iterations: (a) SPA and (b) MSA	467
12.2 The UEBR and UEBLR performances of the (1023, 781) cyclic-EG-LDPC code $C_{EG,cyc}(2, 2^5)$ given in Example 12.1 over the BEC	468
12.3 The error performances of the (4095, 3367) cyclic-EG-LDPC code given in Example 12.3 over: (a) AWGN channel and (b) BEC	470
12.4 The BER and BLER performances of the (1057, 813) cyclic-PG-LDPC code given in Example 12.4	471
12.5 The BER and BLER performances of the (4095, 3367) cyclic-EG-LDPC code in Example 12.5 using the RMSA with $\ell = 819$ and $f = 5$	478
12.6 The BER and BLER performances of the (4095, 3367) cyclic-EG-LDPC code decoded using the RMSA with $\ell = 1$ and $f = 16380$ given in Example 12.6	479
12.7 The BER and BLER performances of the (1057, 813) cyclic-PG-LDPC code given in Example 12.7 decoded with the RMSA using two grouping sizes: (a) $\ell = 151$, $f = 16$, and (b) $\ell = 244$, $f = 8$	481
12.8 The BER and BLER performances of the (1023, 909) QC-EG-LDPC code given in Example 12.9	486
12.9 The BER and BLER performances of the (3780, 3543) QC-EG-LDPC code given in Example 12.10	487
12.10 The BER and BLER performances of the (906, 662) QC-PG-LDPC code given in Example 12.11	488
12.11 The BER and BLER performances of the (2016, 1779) QC-EG-LDPC code given in Example 12.13	492
12.12 The performances of the (16384, 15363) QC-EG-LDPC code given in Example 12.14 over: (a) AWGN channel and (b) BEC	494
12.13 The BER and BLER performances of the unmasked (2048, 1027) and the masked (2048, 1024) QC-EG-LDPC codes given in Example 12.15	496
12.14 The BER and BLER performances of the (8176, 7156) QC-EG-LDPC code given in Example 12.16: (a) MSA and (b) hardware MSA decoder	501
12.15 The BER and BLER performances of the (4088, 2044) QC-EG-LDPC code given in Example 12.17	502
12.16 The BER and BLER performances of the (4599, 3068) QC-EG-LDPC code given in Example 12.18	503
12.17 The BER and BLER performances of the (2016, 1779) QC-EG-LDPC code $C_{EG,qc}(4, 32)$ given in Example 12.20 decoded with the CPM-RMSA of different sizes of decoding matrices: (a) $\ell = 1$ and (b) $\ell = 3$	510

<i>List of Figures</i>	xvi
13.1 The performances of the (961, 840) QC-RS-PaG-LDPC code given in Example 13.4 over: (a) AWGN channel and (b) BEC	529
13.2 The BER and BLER performances of the (7921, 7568) QC-RS-PaG-LDPC code $C_{RS,PaG,c}(4, 89)$ and the (7921, 7566) PEG code C_{peg} given in Example 13.5	531
13.3 The BER and BLER performances of the two QC-RS-PaG-LDPC codes given in Example 13.6: (a) (5696, 5343) and (b) (2848, 2495)	532
13.4 (a) The BER and BLER performances of the (11 584, 10 863) QC-PaG-LDPC code $C_{PaG,p}(4, 64)$ in Example 13.8 and (b) the BER performances of the four codes in Example 13.8	537
13.5 The performances of the (1016, 508) QC-PaG-LDPC code given in Example 13.9 over: (a) AWGN channel and (b) BEC	539
13.6 The BER and BLER performances of the (776, 680) QC-BIBD-PaG-LDPC code $C_{PaG,BIBD,1}(4, 32)$ given in Example 13.12	545
13.7 The performances of the (44 713, 41 781) and (23 456, 20 524) QC-BIBD-PaG-LDPC codes given in Example 13.13 over: (a) AWGN channel and (b) BEC	548
13.8 The BER and BLER performances of the four QC-BIBD-PaG-LDPC codes given in Example 13.14	549
13.9 The BER and BLER performances of the (3934, 3653) QC-BIBD-PaG-LDPC code given in Example 13.15	552
13.10 The BER and BLER performances of the two QC-BIBD-PaG-LDPC codes given in Example 13.16 over the AWGN channel: (a) (20 512, 17 951) and (b) (5128, 2564)	554
13.11 The UEBR and UEBLR performances of the two QC-BIBD-PaG-LDPC codes given in Example 13.16 over the BEC: (a) (20 512, 17 951) and (b) (5128, 2564)	555
14.1 The BER and BLER performances of the (180, 128) QC-LDPC code given in Example 14.2	567
14.2 The performances of the (5080, 4589) QC-LDPC code $C_{s,qc}(4, 40)$ given in Example 14.3 over: (a) AWGN channel and (b) BEC	569
14.3 The performances of the unmasked (1016, 525) and masked (1016, 508) QC-LDPC codes given in Example 14.3 over: (a) AWGN channel and (b) BEC	570
14.4 The BER performances of nine QC-LDPC codes in Example 14.4	572
14.5 The BER and BLER performances of the (16 120, 15 345) QC-LDPC code given in Example 14.5	573
14.6 The BER and BLER performances of the three QC-LDPC codes given in Example 14.6	575
14.7 The BER and BLER performances of the (4064, 3572) QC-LDPC code $C_{s,qc}(4, 32)$ given in Example 14.7	577

<i>List of Figures</i>	xvii
14.8 The BER and BLER performances of the (8192, 7171) QC-LDPC code given in Example 14.8	578
14.9 The BER and BLER performances of the (3440, 2755) and (6880, 6195) QC-LDPC codes given in Example 14.9	580
14.10 The BER performances of the nine QC-LDPC codes in Example 14.10	583
14.11 The BER and BLER performances of the unmasked (3960, 2643) and the masked (3960, 2640) QC-LDPC codes in Example 14.11	584
14.12 The BER and BLER performances of the unmasked (5280, 3305) and the masked (5280, 3302) QC-LDPC codes in Example 14.12	586
14.13 The BER and BLER performances of the (504, 252) QC-LDPC code $C_{s,qc,mask}(4, 8)$ and two other (504, 252) LDPC codes given in Example 14.13	590
14.14 The performances of the (32 704, 30 153) QC-RS-LDPC code given in Example 14.14 over: (a) AWGN channel and (b) BEC	595
14.15 The BER and BLER performances of the two QC-RS-LDPC codes given in Example 14.15	596
14.16 The BER and BLER performances of the (5696, 4985) QC-RS-LDPC code given in Example 14.16	598
14.17 The BER and BLER performances of the unmasked (4088, 2047) and the masked (4088, 2044) QC-RS-LDPC codes given in Example 14.17	601
14.18 The BER and BLER performances of the unmasked (680, 343) and the masked (680, 340) QC-RS-LDPC codes given in Example 14.18	604
14.19 The BER and BLER performances of the unmasked (2040, 1025) and masked (2040, 1020) QC-RS-LDPC codes given in Example 14.19	606
14.20 The BER and BLER performances of the four QC-RS-LDPC codes given in Example 14.20	608
14.21 The performances of the (4672, 4383) QC-RS-LDPC code in Example 14.22 over: (a) AWGN channel and (b) BEC	613
14.22 The performances of the (15 876, 14 871) and (15 876, 13 494) CN-QC-GC-LDPC codes given in Examples 14.23 and 14.24, respectively, over: (a) AWGN channel and (b) BEC	617
15.1 (a) The protograph \mathcal{G}_{ptg} , (b) three copies of the protograph \mathcal{G}_{ptg} and the grouping of their VNs and CNs, and (c) the connected bipartite graph $\mathcal{G}_{ptg}(3, 3)$ given in Example 15.1	634
15.2 (a) The protograph \mathcal{G}_{ptg} , (b) the bipartite graph $\mathcal{G}_{ptg,2}$, (c) the performances of the (680, 340) QC-PTG-LDPC code, and (d) the performances of the (4088, 2044) QC-PTG-LDPC code given in Example 15.2	638

<i>List of Figures</i>	xviii
15.3 The UEBR and UEBLR performances of the (680, 340) and (4088, 2044) QC-PTG-LDPC codes given in Example 15.2 over BEC	639
15.4 (a) The protograph \mathcal{G}_{ptg} and (b) the BER and BLER performances of the (5792, 2896) QC-PTG-LDPC code given in Example 15.3	641
15.5 The performances of the (2640, 1320) QC-PTG-LDPC code given in Example 15.6 over: (a) AWGN channel and (b) BEC	652
15.6 The protograph \mathcal{G}_0 specified by the protomatrix \mathbf{B}_0 given by (15.29)	653
15.7 (a) The protograph \mathcal{G}_{ptg} and (b) the BER and BLER performances of the (3060, 2040) QC-PTG-LDPC code given in Example 15.7	654
15.8 The BER and BLER performances of the (8176, 7156) QC-PTG-LDPC code given in Example 15.8	657
15.9 The BER and BLER performances of the (4080, 3060) QC-PTG-LDPC code given in Example 15.10	660
15.10 (a) The BER and BLER performances of the (3969, 3213) QC-PTG-LDPC code $C_{\text{ptg,qc}}$ given in Example 15.11 and the (3969, 3213)* QC-LDPC code C_{qc}^* in [33] and (b) the BER and BLER performances of the (8001, 6477) QC-PTG-LDPC code given in Example 15.11	671
15.11 Tree representation of the neighborhood $N_{v_i}^{(l)}$ within depth l of a VN v_i	673
15.12 The BER and BLER performances of the (4088, 2044) LDPC codes constructed by the PEG and ACE-PEG algorithms in Example 15.12	676
15.13 The Tanner graph of an SC-LDPC code	676
16.1 A collective encoding scheme for a GFT-ICC-LDPC code C_{LDPC}	696
16.2 A collective iterative soft-decision decoding scheme for a GFT-ICC-LDPC code C_{LDPC}	699
16.3 The FER and BLER performances of the (31, 25) RS code given in Example 16.5 decoded by the GFT-ISDD/MSA and other decoding algorithms	704
16.4 (a) The FER and BLER performances of the (127, 119) RS code given in Example 16.6 decoded by the GFT-ISDD/MSA and other decoding algorithms and (b) the average number of iterations required to decode the (127, 119) RS code in Example 16.6 vs. E_b/N_0 (dB)	705
16.5 The BLER performances of the (127, 113) BCH code given in Example 16.7 decoded by the GFT-ISDD/MSA, BM-HDDA, and MLD	707
16.6 The BLER performances of the (127, 120) Hamming code given in Example 16.8 decoded by the GFT-ISDD/MSA, the BM-HDDA, and MLD	708

<i>List of Figures</i>	xix
16.7 The BLER performances of the (23, 12) Golay code given in Example 16.9 decoded by the GFT-ISDD/MSA, HDDA, and MLD	710
16.8 (a) The BLER performances of the shortened (64, 58) RS code over GF(2 ⁷) and the (127, 121) RS code over GF(2 ⁷) and (b) the BLER performances of the shortened (32, 26) RS code over GF(2 ⁷) and the (127, 121) RS code over GF(2 ⁷) given in Example 16.10 decoded by the GFT-ISDD/MSA and the BM-HDDA	714
16.9 The BLER performances of the (16 129, 11 970) QC-LDPC code C _{BCH,LDPC} (6, 6, . . . , 6) in Example 16.11 decoded by the GFT-ISDD/MSA	715
17.1 The 1-fold Kronecker mapping circuit	725
17.2 The 2-fold Kronecker mapping circuit	728
17.3 The 3-fold Kronecker mapping circuit	730
17.4 The ℓ -fold Kronecker mapping circuit	733
17.5 (a) Two identical and independent channels W and (b) a combined 1-fold vector channel W^2	747
17.6 (a) A combined vector channel with W^+ as the base channel and (b) a combined vector channel with W^- as the base channel	751
17.7 (a) A combined vector channel W^4 and (b) the combined vector channel W^4 after rewire	752
17.8 A combined vector channel W^8	755
17.9 An ℓ -level channel polarization tree	759
17.10 The information transmission using the (8, 4) polar code C _p (4, 3) given in Example 17.8 over the BEC vector channel W^8 with the base BEC channel BEC(0.5)	761
17.11 The bit-coordinate channel capacities for BEC channel polarization with (a) $N = 16$ and (b) $N = 64$ for BEC(0.5)	762
17.12 The bit-coordinate channel capacities for BEC channel polarization with (a) $N = 256$ and (b) $N = 1024$ for BEC(0.5)	763
17.13 The bit-coordinate channel capacities for BEC channel polarization after sorting with (a) $N = 16$ and (b) $N = 64$ for BEC(0.5)	764
17.14 The bit-coordinate channel capacities for BEC channel polarization after sorting with (a) $N = 256$ and (b) $N = 1024$ for BEC(0.5)	765
17.15 Block diagram of a polar coded system	769
17.16 A block diagram of the SC decoding process for a polar code of length $N = 2^\ell$	770
17.17 An SC decoder for a polar code of length $N = 2$	771
17.18 A tree structure of an SC decoder of size $N = 2$	773
17.19 An SC decoder with size $N = 4$	774
17.20 The SC decoding process of a polar code of length $N = 4$	774
17.21 Message-passing and decision trees in decoding a polar code of length 4 with SC decoding	776

<i>List of Figures</i>	xx
17.22 The UEBR and UEBLR of the (256, 128) polar code over BEC decoded with the SC decoding in Example 17.20	779
C.1 Location patterns of six configurations of a cycle-6 C_6	789
E.1 Diagram of implementation of $\mathbf{P} = \mathbf{pH}_{16}$	799
E.2 Diagram of implementation of $\mathbf{P} = \mathbf{p}_0^7\mathbf{H}_8$	799
E.3 Diagram of the fast Hadamard transform	800

Tables

1.1	A binary block code with $k = 4$ and $n = 7$	<i>page 5</i>
1.2	Shannon limits, E_b/N_0 (dB), of a binary-input continuous-output AWGN channel with BPSK signaling for various code rates	19
2.1	The additive group $G = \{0, 1\}$ with modulo-2 addition	27
2.2	The additive group $G = \{0, 1, 2, 3, 4, 5, 6\}$ with modulo-7 addition	27
2.3	The multiplicative group $G = \{1, 2, 3, 4, 5, 6\}$ with modulo-7 multiplication	28
2.4	The additive group $G = \{0, 1, 2, 3, 4, 5, 6, 7\}$ under modulo-8 addition	30
2.5	A subgroup $S = \{0, 2, 4, 6\}$ of G given in Table 2.4 under modulo-8 addition	30
2.6	The prime field $\text{GF}(2)$ under modulo-2 addition and multiplication	35
2.7	The field $\text{GF}(7)$ under modulo-7 addition and multiplication	36
2.8	A list of primitive polynomials over $\text{GF}(2)$	43
2.9	$\text{GF}(2^4)$ generated by $p(X) = 1 + X + X^4$ over $\text{GF}(2)$	50
2.10	$\text{GF}(2^3)$ generated by $p(X) = 1 + X + X^3$ over $\text{GF}(2)$	56
2.11	$\text{GF}(2^6)$ generated by $p_1(X) = 1 + X + X^6$ over $\text{GF}(2)$	56
2.12	The prime field $\text{GF}(3)$ under modulo-3 addition and multiplication	60
2.13	$\text{GF}(3^2)$ generated by $p(X) = 2 + X + X^2$ over $\text{GF}(3)$	60
2.14	The vector space \mathbf{V}_5 over $\text{GF}(2)$ given in Example 2.18	66
2.15	A subspace \mathbf{S} and its dual space \mathbf{S}_d in \mathbf{V}_5 given in Example 2.18	66
3.1	A codebook for a $(6, 3)$ linear block code over $\text{GF}(2)$	91
3.2	The dual code C_d of the $(6, 3)$ linear block code C given by Table 3.1	94
3.3	The $(7, 4)$ linear block code generated by the matrix \mathbf{G} given by (3.13)	95
3.4	The $(7, 3)$ linear block code generated by the matrix \mathbf{H} (as a generator matrix) given by (3.14)	95
3.5	A standard array for an (n, k) linear block code	111
3.6	A standard array for the $(6, 3)$ code given in Example 3.11	111
3.7	A look-up table for syndrome decoding	116
3.8	A syndrome look-up decoding table for the $(6, 3)$ linear block code	117
4.1	A $(7, 4)$ cyclic code generated by $\mathbf{g}(X) = 1 + X + X^3$	133

<i>List of Tables</i>	xxii
4.2 The (7, 4) systematic cyclic code generated by $\mathbf{g}(X) = 1 + X + X^3$ in Example 4.4	138
4.3 The register contents of syndrome calculation circuit of the (7, 4) cyclic code with $\mathbf{r} = (0\ 0\ 0\ 1\ 0\ 1\ 1)$	143
4.4 A syndrome look-up decoding table for the (7, 4) cyclic code given in Example 4.8	148
4.5 Decoding steps for the (7, 4) cyclic code given in Example 4.8	149
4.6 A list of standardized CRC codes	158
4.7 A list of QR codes	160
4.8 The eight codewords of the (6, 3) QC code C_{qc} given in Example 4.12	162
4.9 The eight codewords of the (6, 3) QC code C_{qc} given in Example 4.13	163
5.1 Binary primitive BCH codes of lengths less than 2^{10}	187
5.2 Weight distribution of the dual code of a double-error-correcting binary primitive BCH code of length $n = 2^m - 1$, where $m \geq 3$ and m is odd	195
5.3 Weight distribution of the dual code of a double-error-correcting binary primitive BCH code of length $n = 2^m - 1$, where $m \geq 4$ and m is even	195
5.4 Weight distribution of the dual code of a triple-error-correcting binary primitive BCH code of length $n = 2^m - 1$, where $m \geq 5$ and m is odd	196
5.5 Weight distribution of the dual code of a triple-error-correcting binary primitive BCH code of length $n = 2^m - 1$, where $m \geq 6$ and m is even	196
5.6 Berlekamp–Massey iterative procedure for finding the error-location polynomial of a BCH code	203
5.7 Steps for finding the error-location polynomial of $\mathbf{r}(X) = X^3 + X^5 + X^{12}$ for the (15, 5) BCH code given in Example 5.3	206
5.8 Steps for finding the error-location polynomial of $\mathbf{r}(X) = X + X^3 + X^5 + X^7$ for the (15, 5) BCH code given in Example 5.3	206
5.9 A simplified Berlekamp–Massey iterative procedure for finding the error-location polynomial of a binary BCH code	207
5.10 Steps for finding the error-location polynomial of the binary (15, 5) BCH code given in Example 5.4	207
5.11 $GF(2^5)$ generated by the primitive polynomial $\mathbf{p}(X) = 1 + X^2 + X^5$	209
5.12 Steps for finding the error-location polynomial of $\mathbf{r}(X) = 1 + X^{12} + X^{20}$ for the binary (31, 16) BCH code in Example 5.5	210
5.13 Steps for finding the error-location polynomial of $\mathbf{r}(X) = X + X^3 + X^5 + X^7$ for the binary (31, 16) BCH code in Example 5.5	210

<i>List of Tables</i>	xxiii
5.14 GF(2 ⁶) generated by $\mathbf{p}(X) = 1 + X + X^6$ over GF(2)	214
6.1 Berlekamp–Massey iterative procedure for finding the error-location polynomial of a nonbinary BCH code	230
6.2 Steps for finding the error-location polynomial of the received polynomial $\mathbf{r}(X)$ for the 4-ary (15, 9) BCH code in Example 6.3	230
6.3 Steps for finding the error-location polynomial of the received polynomial $\mathbf{r}^*(X)$ for the 4-ary (15, 9) BCH code in Example 6.3	231
6.4 GF(2 ³) generated by $\mathbf{p}(X) = 1 + X + X^3$ over GF(2)	237
6.5 Steps for finding the error-location polynomial of $\mathbf{r}(X)$ for the 16-ary (15, 9) RS code over GF(2 ⁴) in Example 6.7	239
6.6 Steps for finding the error-location polynomial of $\mathbf{r}(X)$ for the 32-ary (31, 25) RS code in Example 6.8	241
6.7 Euclidean algorithm for finding the GCD of two polynomials $\mathbf{a}(X)$ and $\mathbf{b}(X)$ over GF(q)	246
6.8 Euclidean iterative algorithm for finding the GCD of two polynomials $\mathbf{a}(X)$ and $\mathbf{b}(X)$ over GF (2 ⁴) in Example 6.9	246
6.9 Euclidean algorithm for finding error-location polynomial $\boldsymbol{\sigma}(X)$ and error-value evaluator $\mathbf{Z}_0(X)$	248
6.10 Euclidean iterative algorithm for decoding the 16-ary (15, 9) RS code given in Example 6.10	249
6.11 Euclidean iterative algorithm for decoding the 32-ary (31, 25) RS code given in Example 6.11	250
7.1 GF(2 ⁴) generated by $\mathbf{p}(X) = 1 + X + X^4$ over GF(2)	272
7.2 GF(2 ⁴) as an extension field of GF(2 ²) = {0, 1, β , β^2 } with $\beta = \alpha^5$	272
7.3 GF(2 ⁶) as an extension field of GF(2 ²) = {0, 1, β , β^2 } with $\beta = \alpha^{21}$	273
7.4 Four parallel bundles of lines of EG(3, 2 ²) over GF(2 ²)	274
7.5 Two cyclic classes of lines of EG*(3, 2 ²) over GF(2 ²)	276
7.6 A list of two-dimensional EG codes	281
7.7 Lines of the projective geometry PG(2, 2 ²) over GF(2 ²)	292
7.8 A list of two-dimensional PG codes	294
9.1 The 16 codewords of the (7, 4) Hamming code	337
10.1 A list of Fire codes which have true burst-error-correcting capabilities larger than their designed values [27]	375
10.2 A list of optimal and nearly optimal cyclic and shortened cyclic l -burst-error-correcting codes [25]	376
10.3 Euclidean steps to find the solution ($\boldsymbol{\sigma}(X)$, $\mathbf{Z}_0(X)$) of the key-equation for decoding the (15, 9) RS code in Example 10.15	399
11.1 Decoding thresholds under IDBP for LDPC codes over AWGN channels	441
12.1 GF(2 ⁴) as an extension field of GF(2 ²) = {0, 1, β , β^2 } with $\beta = \alpha^5$	484
13.1 A list of bs for which $12b + 1$ is a prime and the field GF(12 b + 1) satisfies the condition given by (13.11)	542
13.2 A list of bs for which $20b + 1$ is a prime and the field GF(20 b + 1) satisfies the condition given by (13.19)	550
14.1 The nine QC-LDPC codes given in Example 14.4	571
14.2 The nine QC-LDPC codes constructed in Example 14.10	582

<i>List of Tables</i>	xxiv
14.3 Numbers of 4×8 masked base matrices over various fields $\text{GF}(q)$ that give rate-1/2 QC-LDPC codes with girths 8 or 10 for $\eta = 1$	588
14.4 Numbers of 4×8 masked matrices over $\text{GF}(331)$ that give rate-1/2 QC-LDPC codes with girths 8 and 10 for different choices of η s	589
15.1 The nonzero constituent matrices of the protomatrix $\mathbf{B}_{\text{ptg},2}$ and the locations of their 1-entries with decomposition factor $k = 85$ given in Example 15.5	648
15.2 The nonzero constituent matrices of the protomatrix $\mathbf{B}_{\text{ptg},2}$ and the locations of their 1-entries with decomposition factor $k = 511$ given in Example 15.5	649
15.3 The nonzero constituent matrices of the protomatrix $\mathbf{B}_{\text{ptg},2}$ and the locations of their 1-entries with decomposition factor $k = 330$ given in Example 15.6	650
15.4 The nonzero constituent matrices of the protomatrix $\mathbf{B}_{\text{ptg},2}$ and the locations of their 1-entries with decomposition factor $k = 255$ given in Example 15.7	653
15.5 The nonzero constituent matrices of the protomatrix \mathbf{B}_{ptg} and the locations of their 1-entries with decomposition factor $k = 511$ given in Example 15.8	656
15.6 The generators of the circulants in the array $\mathbf{H}_{\text{ptg,qc}}(511, 511)$ given in Example 15.8 and the locations of their 1-entries	656
15.7 The nonzero constituent matrices of the protomatrix \mathbf{B}_{ptg} and the locations of their 1-entries with decomposition factor $k = 330$ given in Example 15.9	658
15.8 The nonzero constituent matrices of the protomatrix \mathbf{B}_{ptg} and the locations of their 1-entries with decomposition factor $k = 255$ given in Example 15.10	659
15.9 Column and row weight distributions of the 12×63 protomatrix \mathbf{B}_{ptg} used in Example 15.11	661
15.10 The nonzero constituent matrices of the protomatrix \mathbf{B}_{ptg} and the locations of their 1-entries with decomposition factor $k = 63$ given in Example 15.11	662
15.11 The entries of $\mathbf{H}_{\text{ptg,qc}}(63, 63)$ given in Example 15.11	665
15.12 The nonzero constituent matrices of the protomatrix \mathbf{B}_{ptg} and the locations of their 1-entries with decomposition factor $k = 127$ given in Example 15.11	666
15.13 The entries of $\mathbf{H}_{\text{ptg,qc}}(127, 127)$ given in Example 15.11	669
16.1 $\text{GF}(2^3)$ generated by $\mathbf{p}(X) = 1 + X + X^3$ over $\text{GF}(2)$	688
17.1 The 2-fold Kronecker mappings of the 16 4-tuples over $\text{GF}(2)$	727
A.1 Factorization of $X^n + 1$ over $\text{GF}(2)$ with $1 \leq n \leq 31$	784

Preface

One of the serious problems in a digital data communication or storage system is the occurrence of errors caused by noise and interference in communication channels or imperfections in storage mediums. A major concern to the communication or storage-system designers is the control of these errors such that reliable transmission or storage of data can be achieved. In 1948, Shannon demonstrated in a landmark paper that by proper encoding and decoding of the data, errors induced by a noisy channel or imperfect storage medium can be reduced to any desired level without sacrificing the rate of information transmission or storage, as long as the information rate is less than the capacity of the channel or the storage medium. Since Shannon's work, a tremendous amount of research effort has been expended on the problems of devising efficient encoding and decoding methods and techniques for error control on noisy channels or imperfect storage mediums. As a result of this research effort, various efficient encoding and decoding methods and techniques have been developed to achieve the reliability required by today's explosive high-speed and large-volume digital communication and storage systems.

Much of the work on error-correcting codes (or error-control codes) developed since 1948 is highly mathematical in nature, and a thorough understanding requires an extensive background in modern algebra, combinatorial mathematics, and graph theory. This requirement may impede senior and first-year graduate students in electrical and computer engineering who are interested in learning and pursuing research in coding theory, and practicing engineers in industry who are interested in applying error-control coding techniques to practical systems.

One of the objectives of this book is to bring this highly complex material down to a reasonably simple level such that it can be understood and applied with a minimum background in mathematics. To achieve this objective, we take a middle ground between mathematical rigor and heuristic reasoning as the first author did in his first book on the introduction to error-correcting codes published in 1970. Because of the extensive developments in error-correcting codes over the past 50 years, it is not possible to include certain categories of error-correcting codes in this book. The main coverage of this book is the fundamental and essential aspects of codes with block structure, called block codes, and their up-to-date developments in construction, encoding, and decoding techniques. The presentation of these subjects is intended to be comprehensive. In

presenting every step of each topic in the book, illustrative examples are given to assist the readers to follow and fully understand the topic with a minimum barrier. Furthermore, derivations and long proofs that are not helpful in illustration of a topic are avoided. Long essential derivations or proofs of any topic are put in the appendices or referred to in published article(s).

In the following, a brief description of major coverage in each chapter is presented. Chapter 1 gives a brief overview of coding for error control in information transmission and data storage. Chapter 2 provides the readers with an elementary knowledge of modern algebra and graph theory that will aid in understanding the fundamental and essential aspects of error-correcting codes to be developed in the other chapters of the book. Chapter 3 gives an introduction to block codes with linear structure, called linear block codes, their structural properties, and general decoding methods. Chapter 4 introduces two special categories of linear block codes with cyclic and quasi-cyclic structures, called cyclic codes and quasi-cyclic codes, respectively. Also presented in this chapter are two small classes of cyclic codes, known as Hamming and quadratic-residue (QR) codes.

Chapters 5 and 6 present two well-known classes of cyclic codes constructed based on finite fields, called the Bose–Chaudhuri–Hocquenghem (BCH) and the Reed–Solomon (RS) codes. These two classes of cyclic codes have been widely used in digital-communication and data-storage systems. Major topics covered in these two chapters include code constructions, characterizations, and decoding algorithms. Presented in Chapter 7 are two classes of cyclic codes constructed based on two categories of finite geometries, named Euclidean and projective geometries. Finite-geometry codes in these two classes are low-density parity-check (LDPC) codes, which can be decoded with iterative soft-decision algorithms based on belief-propagation to achieve good error performance with practical implementation complexity.

Chapter 8 presents another well-known class of linear block codes, called Reed–Muller (RM) codes, for correcting multiple random errors. RM codes can be decoded with a simple majority-logic decoding algorithm using a successive-cancellation process. Chapter 9 presents several coding techniques that are commonly used in communication and storage systems for reliable information transmission and data storage. These coding techniques include: (1) interleaving; (2) direct product; (3) concatenation; (4) turbo coding; (5) $|\mathbf{u}| \mathbf{u} + \mathbf{v}|$ -construction; (6) Kronecker (or tensor) product; and (7) automatic-request-retransmission (ARQ) schemes. Chapter 10 presents various types of codes and coding techniques for correcting bursts of errors, random erasures, bursts of erasures, and combinations of random errors and erasures. Also presented in this chapter is a simple successive-peeling algorithm for correcting random erasures.

Chapter 11 introduces LDPC codes which can achieve near-capacity (or close to Shannon-limit) performance with iterative soft-decision decoding based on belief-propagation over various communication and data-storage channels. Many LDPC codes have been adopted as standard codes for various current and next-generation communication systems. Major aspects covered in this chapter include: (1) basic concepts and characteristics; (2) matrix and graphical representations; (3) various iterative decoding algorithms based on

belief-propagation; and (4) error performances over binary-input additive white Gaussian noise (AWGN) and erasure channels.

Chapters 12–14 present three classes of cyclic and quasi-cyclic LDPC codes that are constructed based on finite and partial geometries, finite fields, and experimental designs. These LDPC codes achieve good error performance on both binary-input AWGN and binary-erasure channels. Also included in these chapters are two reduced-complexity iterative decoding algorithms and a technique, called masking, for performance enhancement of LDPC codes. Chapter 15 presents two graphical methods for constructing LDPC codes, known as protograph and progressive-edge-growth methods. LDPC codes constructed based on protographs form a class of channel capacity approaching codes.

Chapter 16 presents a universal coding scheme for collective encoding and collective iterative soft-decision decoding of cyclic codes of prime lengths in the frequency domain. Collective encoding and decoding allows for reliability in information sharing among the received codewords during the decoding process. This collective decoding and information sharing can achieve a decoding gain over the maximum-likelihood decoding of individually received codewords. Collective encoding and decoding of BCH, RS, and QR codes are covered in this chapter.

Chapter 17 presents a class of channel-capacity-approaching codes, called polar codes. Essential aspects covered in this chapter include: (1) Kronecker matrices, mappings, and vector spaces; (2) polar codes from Kronecker mapping point of view; (3) multilevel encoding of polar codes; (4) construction of polar codes based on channel polarization; and (5) successive-cancellation decoding of polar codes.

Except for the first chapter, all other chapters contain a good number of problems. The problems are of various types, including those that require routine calculations, those that require computer solution or simulation, those that require derivations and proofs, and those that require designs and performance analysis. The problems are selected to strengthen students' or engineers' knowledge of the materials in each chapter.

This book can be used as a text for an introductory course on coding at the senior or beginning-graduate level or a more-comprehensive full-year graduate course. It can also be used as a self-guide for practicing engineers and computer scientists in industry who desire to learn the fundamentals and essentials of coding aspects and how they can be applied to the design of error-control systems. For a one-semester introductory course, the fundamentals presented in Chapters 1–6 and some selected topics in Chapters 7–11 can be used. For a two-semester sequence in coding theory, the first 10 chapters in fundamentals of coding can be used for the first semester and remaining chapters on advanced coding topics in the second semester. The book can also be used as a text for one-semester advanced-graduate course focused on LDPC and polar codes and collective encoding and decoding of cyclic codes of prime lengths. In this case, the instructor could use Chapters 11–17 or selected topics from Chapters 4–8. Furthermore, the materials covered in Chapters 1–4 can be used as supplementary subjects for an undergraduate course in information theory or digital-communication systems.

Acknowledgments

The first author (Shu Lin) would like to take this opportunity to acknowledge the late Professor Paul E. Pfeiffer, the late Professor Wesley W. Peterson, the late Professor Tadao Kasami, and Professor Franklin F. Kuo, who motivated, inspired, and guided him into the wonderful, elegant, and practically useful domain of coding theory at the beginning of his research career. Professor Pfeiffer was his thesis advisor at Rice University. Professor Peterson and Professor Kasami were two pioneers in coding theory, and Professor Peterson wrote the first and most influential book on error-correcting codes. Professor Kou is one of the founders of the Aloha system. Next, the first author would like to thank Professor Ian F. Blake, Professor Dan J. Costello, Jr., Professor Khaled A. S. Abdel-Ghaffar, and Professor William E. Ryan who closely worked with him over many years and helped him to broaden his knowledge in coding theory and its applications in digital communication and data-storage systems. Professor Costello is also his coauthor of two other books.

The most important person behind the first author during the past 57 years, which include the final two years of his graduate study and his entire teaching and research career, is his wife, Ivy. Without her love, support, tolerance, and comfort, as well as raising their three children, his teaching and research career would not have reached as many students or lasted as long as it has. She even took a course in modern algebra to understand what the author was working on at the beginning of his teaching career and to improve their communication. Any success the first author has, he owes to her. She is the one who encouraged him to write this book with his coauthor. So, it is here the first author says to his wife, I love you. Also, the first author would like to give his love and special thanks to his children, their spouses, and grandchildren for their continuing love and affection through the years.

The second author (Juane Li) expresses her sincere gratitude to her advisors, Professor Shu Lin and Professor Khaled A. S. Abdel-Ghaffar, for their valuable guidance, support, and encouragement during her graduate study at the University of California at Davis and during days after her graduation. Professor Lin has rich knowledge, profound ideas, and endless enthusiasm in the error-correcting coding research area. Professor Abdel-Ghaffar is knowledgeable, mathematically rigorous, precise, and patient with students. These two professors' critical and constructive suggestions have pushed her work to a higher

level. She also owes a special debt of gratitude to her family for their affection, encouragement, and support over the years.

Both authors are very grateful to Professor Ian F. Blake, Professor Harry Tan, Professor Shih-Chun Chang, Professor William E. Ryan, and Professor Khaled A. S. Abdel-Ghaffar who expended tremendous effort in reading through this book in detail and provided critical comments and numerous valuable suggestions. They would also like to express their appreciation to Professor Qin Huang and Professor Mona Nasser, Dr. Keke Liu, and Dr. Xin Xiao for their contributions to several topics in this book. They also acknowledge the talented Dr. Zijian Wu who provided such a beautiful photograph of a lotus for the cover image of their book. Last but not least, the authors would like to thank Dr. Julie Lancashire at Cambridge University Press. Without her strong encouragement, warm support, and patience, we would not have been able to bring this book to fruition. Julie, you are a top and thoughtful editor and a good friend.

Cambridge University Press
978-1-316-51262-3 — Fundamentals of Classical and Modern Error-Correcting Codes
Shu Lin , Juane Li
Frontmatter
[More Information](#)
