CAMBRIDGE

Cambridge University Press
978-0-521-84868-8 - Channel Codes: Classical and Modern
William E. Ryan and Shu Lin
Index
More information

Index

abelian group, 30
accumulator-based codes LDPC in standards, 285–7
  for CCSDS deep space applications, 286
  IEEE applications, 287
ACE (approximate cycle EMD), 260–1, 546
ACK (acknowledgement) signals, 3
additive groups, 33, 35, 506–10
APP (a posteriori probability), 180
  APP ratio, 213
ARA (accumulate-repeat-accumulate) LDPC codes, 279–85
  protographs with, 280–3
  protographs-based design, 283–5
area property for EXIT charts, 424–6
ARJA (accumulate-repeat-jagged-accumulate) LDPC codes, 284–6
  for CCSDS for deep space applications, 286
ARQ (automatic request for repeat) schemes, 3–4, 672–3
array dispersion construction
  nonbinary QC-LDPC codes, 618
  QC-LDPC codes, 580–6
    examples, 585, 585–6
    l-fold dispersion, 583–4
    masking parameters, 583–4
associative law/operations, 29, 50, 51
  with polynomials, 53
asymptotic ensemble weight enumerators, 371–8
  complexity issues, 376–8
  examples, 374–6, 377–8
  Stirling's approximation, 372
asymptotically optimal QC-LDPC codes, erasure burst correction, 575–80
  examples, 577, 578–9, 579–81
AWGN (additive white Gaussian noise) process, 14, 22–3

Bayes' rule, 180–2
  with PCCC iterative decoders, 318
BBEC (binary burst erasure channel), 561, 570
BCH (Bose–Chandhuri–Hocquenghem) codes, construction, 111–21
  about BCH codes, 111
  BCH bound, 113
  example, 113–14
  LCM (least common multiple), 111
  primitive BCH codes, 112
  smallest-degree polynomials, 111
  t-error-correcting binary BCH code, 111–13
    designed minimum distance, 113
  and TPCs, 328–9
  Vandermonde matrix, 113
  see also decoding BCH codes
BCJR-algorithm/trellis, 171–2, 180, 183, 185–7, 242–3, 644
BEC (binary erasure channels)
  BBEC (binary burst erasure channel), 561, 570
  capacity, 11–13
  erasures, 561
  good LDPC channels for, 565–70
    examples, 566, 567–8, 568–70
    UEBR (unresolved erasure bit rate), 565
  iterative decoding of LDPC codes for, 561–3
  iterative erasure filling for, 243–4
  ML decoder for, 244–6
  random-erasure-correction capability, 563–5
  see also erasure bursts/erasure burst correction
BER (bit-error rate/probability), 9–10
BF (bit-flipping) algorithm for LDPC codes over the BSC, 468–76
  decoding algorithm, 469
  LPCF-BF decoding algorithm, 469
  optimum threshold, 469
  preset threshold, 468
  reliability profile, 468
  weighted BF decoding; algorithm one, 469–72
BI-AWGN (binary-input AWGN) channel, 7–9
  capacity, 14–18
  E_b/N_o limits, 16–17
  Gallager error exponent, 18
  Gallager random coding bound, 18
  hard-decision BI-AWGN channel, 15
  Shannon capacity, 15–16
  soft-decision BI-AWGN channel, 15
  sphere-packing bounds, 18
  unconstrained-input AWGN channel, 15–16
bi-mode decoder and low error floors, 661–6
  bi-mode-decoder extension, 665–6
  Margulis solution, 663–5
    FER curves for, 665
  short-QC-code solution, 663
BIBDs (balanced incomplete block designs) and LDPC codes, 523–4
  see also Bose BIBDs ...
binary operations *see* sets and binary operations

© in this web service Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-0-521-84868-8 - Channel Codes: Classical and Modern
William E. Ryan and Shu Lin
Index
More information

binary-input memoryless channels capacity, 11–18
 BEC (binary erasure channels), 11–13
 BI-AWGN (binary-input AWGN) channel, 14–18
 BSC (binary symmetric channel), 12–13
 Z channel, 13–14
bipartite graphs, 86–8, 538–40
bit flipping algorithm for the BSC, 247–8
bit-pinning and concatenation, 666–8
bit-wise MAP criterion, 9
bit-wise MAP decoding, 180–7
 APP (a posteriori probability), 180
 backward metric, 184
 Bayes' rule, 180–2
 BCJR-algorithm, 180, 183
  log-domain BCJR-algorithm, 185–7
 branch metric, 184
 forward metric, 183
 LLR (log-likelihood ratio), 180
 termination bits, 183
Blahut–Arimoto algorithm, 11
BM (Berlekamp–Massey) algorithm, 115, 117–21
Bose BIBDs class-I and QC-LDPC codes, 524–30
 about class-I Bose BIBDs, 524–5
 class-I type-I BIBD-LDPC codes, 525–7
  examples, 526–7
 class-I type-II Bose BIBD-LDPC codes, 527–30
  examples, 528–9, 529–30
Bose BIBDs class-II and QC-LDPC codes, 530–6
 about class-II QC-Bose BIBDs, 530–1
 class-II type-I QC-BIBD-LDPC codes, 531–2
  example, 531–2
 class-II type-II QC-BIBD-LDPC codes, 533–6
  examples, 533, 535–6
Bose BIBDs type-II LDPC code construction by dispersion, 536–7
Bose–Chandhuri–Hocquenghem codes *see* BCH .....
bound-distance decoding, 106
box-plus SPA decoder, 222–5
 CN update, 224–5
 VN update, 225
BPSK (binary phase-shift keying), 19
BSC (binary symmetric channel), 7–9
 bit-flipping algorithm, 247–8
 capacity, 12–13
 Gallager's algorithm A and algorithm B, 246–7
 and the MLSD, 172–3
BTC (block turbo code) *see* TPC (turbo product code)

C-OWE (conditional output-WE), 341
C-PWE (conditional parity-WE), 341, 344, 345
cancelation law, 43
capacity formulas for common channel models, 10–24
 about channel-capacity, 10–11
 AWGN noise process, 22–4
 binary-input memoryless channels, 11–18
 Blahut-Arimoto algorithm, 11
 channels with memory, 21–4
  finite-state channels, 21
  GE (Gilbert-Elliot) channels, 21–2
  ISI (intersymbol interference) channel, 21–4
  i.u.d. (independent, uniformly distributed) capacity, 23, 24
 conditional entropy, 11
 entropy of channel output, 11
 inconsistency issues, 11
 LDPC codes, invention of, 10
 M-ary-input memoryless channels coding limits, 18–21
 mutual channel information, 10
 turbo codes, invention of, 10
catastrophic convolution encoders, 158–9
Cayley table, 31
CC (constituent code), with PCCC, 302
CCSDS (Consultative Committee for Space Data Systems), 286
channel-coding overview, 3–4
 ACK (acknowledgement) signals, 3
 ARQ (automatic request for repeat) schemes, 3
  go-back-N ARQ, 4
  incremental redundancy ARQ, 4
  selective-repeat ARQ, 4
 error-correction/control codes, 3
 error-detection codes, 3
 FEC (forward-error-correction) schemes, 3
 hybrid FEC/ARQ schemes, 3
 NAK (negative acknowledgement) signals, 3
 stop-and-wait ARQ schemes, 3
channels, 1–3
 channel capacity, 2–3
 encoders and decoders, 1–2
characteristics of fields, 40
Chase decoders, 334
CI-WE (cumulative information-WE), 342
circulant decomposition for construction of QC-EG-LDPC codes, 450–5
circulant permutation matrices, 262
 nonbinary, 615
circulant-based architecture, 264
CNPs (check-node processors), with EXIT charts, 412–17, 420, 641
code packets, 675
code rate, 2–3
 linear block codes, 95
codeword error, 8
 codeword-error probability, 10
codewords, in linear block codes, 94
combinatorial designs
 about combinatorial designs, 523
 see also BIBDs (balanced incomplete block designs) and LDPC codes; Bose BIBDs
common channel models *see* capacity formulas for common channel models
commutative groups, 30, 32
commutative law/operation, 29, 50
 with polynomials, 53
complete error-correction decoding, 102
compression, 1
 lossy/lossless, 2
computer-based design of LDPC codes, 257–95
 accumulator-based codes in standards, 285–7

© in this web service Cambridge University Press www.cambridge.org

Cambridge University Press
978-0-521-84868-8 - Channel Codes: Classical and Modern
William E. Ryan and Shu Lin
Index
More information

ACE (approximate cycle EMD) algorithm, 260–1
ARA (accumulate-repeat-accumulate) codes, 279–85
double-accumulator-based codes, 277–85
G-LDPC (generalized LDPC) codes, 287–91
IRA (irregular repeat-accumulate) codes, 267–76
IRAA (irregular repeat-accumulate-accumulate) codes, 278–9
MET (multi-edge-type) codes, 265–6
PEG (progressive-edge-growth) algorithm, 259–60
protograph LDPC codes, 261–5
QC-IRA (quasi-cyclic IRA) code design, 271–6
RA (repeat-accumulate) codes, 267
single-accumulator-based codes, 266–77
concatenated codes, 131–3
decoding, 132
inner decoding, 132
outer decoding, 132
encoding, 131–2
turbo coding, 132
concentration theorem, 388
confidence coefficient, WBF decoding, 473
constrained-input intersymbol interference (ISI) channel, 21
convolution codes, 147–200
algebraic description, 149–52
binary convolution code, 150–1
examples, 149–50, 151–2
finite field GF(2), 149
parity check matrix, 151
ring of polynomials, 149
systematic form, 151
archetype code description, 147–9
compact matrix equation for, 148
generator matrix/polynomials, 148–9
LDPC convolutional codes, 670–2
see also bit-wise MAP decoding; differential Viterbi decoding; MLSD (maximum-likelihood sequence decoder)
convolution codes, alternative representations, 163–71
graphical representations, 170–1
FSTD (finite-state transition-diagram), 170
tail-biting, 171
trellis diagrams, 171
as semi-infinite linear codes, 164–9
examples, 166–8, 169
generator matrix, 166
convolution codes, encoder realizations and classifications, 152–63
catastrophic encoders, 158–9
code design, 163
controller canonical form/transposed canonical form, 153
delay-free requirements, 153
encoder class choice, 157–8
example, 154–5
FIR (finite-impulse-response) filters, 152
IIR (infinite-impulse-response) filters, 152–3
minimal encoders, 159–62
examples, 159–60, 162
Type I, 159–61
Type II, 161–2
realizability issues, 153–6
RSC (recursive systematic convolutional) encoders, 156–7
class variations, 156–7
correlation metric, 176
cosets, 37–8, 72
coset leaders, 103–4
CPM (circulant permutation matrix), 485, 495–6
crossover probability ε, 7
CSA (compare–select–add) operation, 179
CSRAA (cyclic shift–register–adder–accumulator) circuit, 135–7
cumulative metric, 173
cyclic codes, 106–11
about cyclic codes, 106
bounded distance decoding, 110
code polynomials, 107–11
generator polynomials, 107
message polynomials, 107, 109
nonzero code polynomials, 107
parity-check polynomials, 108
received polynomials, 110
reciprocal polynomials, 108
definition, 106
for erasure bursts correction, 586–9
right cycle-shift, 106
shortened cyclic code, 110
cyclic finite-geometry LDPC codes
erasure burst correction, 573–4
see also EG (Euclidean geometries)
cyclic groups, 35
cyclic PG-LDPC code construction, 455–8
example, 456–7
incidence vectors, 457
quasi-cyclic PG-LDPC code construction, 458–9
example, 459

decoding BCH codes, 114–21
BM (Berlekamp-Massey) algorithm, 115, 117–21
simplifying, 120–1
correction term, 118
elementary-symmetric functions, 116
error-location numbers, 115
error-location polynomial, 116
example, 119–20
kth discrepancy, 118
Newton identities, 116–17
power-sum symmetric functions, 116
decoding linear block codes, 102–6
bound-distance decoding, 106
complete error-correction decoding, 102
coset leaders, 103–4
decoding failure, 106
decoding regions, 102
error-correction capability, 105–6
minimum-distance (nearest-neighbor) decoding, 102

© in this web service Cambridge University Press www.cambridge.org

Cambridge

Cambridge University Press
978-0-521-84868-8 - Channel Codes: Classical and Modern
William E. Ryan and Shu Lin
Index
More information

decoding linear block codes (*Cont.*)
- MLD (maximum-likelihood decoding), 102
- optimal standard array, 104
- standard arrays, 103
- syndrome/table-look-up decoding, 105

decoding regions, 102
decoding thresholds for LDPC and turbo code ensembles, 388–428
- density evolution for irregular LDPC codes, 394–9
- density evolution for regular LDPC codes, 388–94
- EXIT charts, area property for, 424–6
- EXIT charts for LDPC codes, 412–20
- EXIT charts for turbo codes, 420–3
- EXIT techniques for protograph-based codes, 417–20
- GA (Gaussian approximation), 402–7
- quantized density evolution, 399–401
- universality of LDPC codes, 407–12

decomposition of EG for construction of LDPC codes, 439–44
demodulators, 2
density evolution for irregular LDPC codes, 394–9
- differential-evolution algorithm, 396
- examples, 396–7, 398–9

density evolution for regular LDPC codes, 388–94
- algorithm for, 393
- concentration theorem, 388
- decoding thresholds, 389
- density-evolution algorithm, 389–90
- examples, 393, 394
- pdf (probability density function) derivation, 391–2

design criteria, 7–10
- BER (bit-error rate/probability), 9–10
- bit-wise MAP criterion, 9
- codeword error, 8
- codeword-error probability, 10
- crossover probability ε, 7
- Euclidean-distance, 9
- FER (frame-error rate), 10
- Hamming-distance, 9
- MAP (maximum-a posteriori) rule, 8–9
- minimum-probability-of-error, 8
- ML (maximum-likelihood) rule, 8–9
- SER (symbol-error rate), 10
- symbol-error probability, 10
- WER (word-error rate), 10

DFE (decision-feedback-equalizer), 644
differential Viterbi decoding, 177–9
- CSA (compare-select-add) operation, 179
- tentative cumulative metric difference, 178

differential-evolution algorithm, 396
digital data communications, applications, 1
dispersion construction
- for Bose BIBDs type-II LDPC codes, 536–7
- for QC-LDPC codes, 580–6

distributive law, 39, 50
- with polynomials, 53

double-accumulator-based LDPC codes, 277–85
- ARA (accumulate-repeat-accumulate) codes, 279–85
- IRAA (irregular repeat-accumulate-accumulate) codes, 278–9

EBC (equivalent block code), 346
EG (Euclidean geometries), 70–6
- μ-flats, 72–6
- LDPC code construction
  - by decomposition, 439–44
  - by masking/irregular masking, 444–50
  - with lines, 430–9
  - with lines for cyclic EG-LDPC codes, 432–4
  - with lines for QC EG-LDPC code, 434–6
  - with parallel bundles of lines, 436–9
- lines, 71
  - cyclic class of, 75–6
  - disjoint, 71
  - intersecting bundles, 71
  - parallel, 71
- m-dimensional, 72–3
- and OSMLG decoding, 465
- primitive properties, 75

elementary-symmetric functions, 116
EMD (extrinsic message degree) for a cycle, 261
ensemble decoding *see* decoding thresholds for LDPC and turbo code ensembles
ensemble enumerators, turbo and LDPC codes, 339–85
- about ensemble enumerators, 339
- for asymptotic ensemble weight enumerators, 371–8
- for ensemble trapping-set enumerators, 379–82
- for finite-length ensemble weight enumerators, 368–71
- for IRA code enumerators, 364–6
- notation, 340–3
- for PCCs (parallel-concatenated codes), 343–56
- protograph-based LDPC codes, 367–82
- for RA code enumerators, 362–4
- for SCCs (serial-concatenated codes), 356–62
- *see also* WE (weight enumerators)/enumerating functions

ensemble trapping-set enumerators, 379–82
- asymptotic enumerators, 381
- elementary enumerators, 380–1
- ensemble stopping-set enumerators, 381–2
- finite-size enumerators, 379–80

entropy of channel output, 11
- conditional entropy, 11

erasure bursts/erasure burst correction, 570–80, 586–9
- about erasure-bursts correction, 570–3
- with asymptotically optimal QC-LDPC codes, 575–80
- by array dispersion constructed QC-LDPC codes, 580–6
- cyclic codes for, 586–9
- with cyclic finite-geometry LDPC codes, 573–4
- error-burst-correction efficiency, 573
- recoverability of erasure bursts, 572–3
- with superposition LDPC codes, 574–5
- zero (covering) spans, 570–2, 576

erasures/erasure patterns, 561

© in this web service Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-0-521-84868-8 - Channel Codes: Classical and Modern
William E. Ryan and Shu Lin
Index
More information

error correction capability, 105–6
  OSMLG error correction, 463
error correction/control codes, 3
error detection
  capability for, 101–2
  codes for, 3
  with linear block codes, 98–9
    error patterns, 98
    syndrome, 99
    transmission errors, 98–9
    undetectable error patterns, 99
  performance, 438
  probability of an undetected error, 100
error floors and trapping sets for LDPC codes, 651–7
  about error-floors and trapping sets, 651–4
  aperiodic oscillating trapping sets, 652
  error floor estimation, 654–7
  periodically oscillating trapping sets, 652
  with QC-LDPC codes, 653–4
  stable trapping sets, 652
  VNs (variable nodes), 652–3
  see also low error floors, LDPC decoder design for
Euclidean geometries *see* EG (Euclidean geometries)
Euclidean metric, 173
  and MLSD, 173
Euclidean-distance, 9
Euclid's division algorithm, 54
Euclid's iterative division algorithm, 126–8
Euler's formula, 69
EXIT (extrinsic-information-transfer) based design for LDPC coded modulation, 638–44
  pdfs via Monte Carlo simulations, 640
  VN and CN processors, 641
EXIT (extrinsic-information-transfer) charts
  about EXIT charts, 412–14
  area property for, 424–6
    LDPC codes, 425–6
    serial-concatenated codes, 424–5
  for LDPC codes, 412–20
    and CNPs (check-node processors), 412–17
    for irregular LDPC codes, 416–17
    for ISI channels, 648–50
    for regular LDPC codes, 414–15
    and VNPs (variable-node processors), 412–17
  for turbo codes, 420–3
    CNs and VNs with, 420
    and conditional pdfs, 421
    production procedure, 423
EXIT (extrinsic-information-transfer) techniques for protograph-based codes, 417–20
  multidimensional EXIT algorithm, 419–20
extension of real-number fields, 39
extrinsic information concept, 211

FEC (forward-error-correction) schemes, 3–4, 672–3
FER (frame-error rate), 10
FFT-QSPA (fast-Fourier-transform QSAP) decoding of nonbinary LDPC codes, 598–600
  advantages, 600
FG (finite geometries), 70–80
  about FGs, 70
  EGs (Euclidean geometries), 70–6
  projective geometries, 76–80
FG (finite geometry) based construction of nonbinary LDPC codes, 600–14
  nonbinary QC EG-LDPC codes, 607–9
  nonbinary regular EG-LDPC codes, 610–11
  q^m-ary cyclic EG-LDPC codes, 601–7
  see also PG (projective geometries)
FG (finite-geometry) LDPC codes, 430–80
  OSMLG (one-step majority-logic) for, 460–8
fields, 38–44
  about fields, 38
  addition and multiplication operations, 38–9
  characteristics of fields, 40
  distributive law, 39
  extension of real-number fields, 39
  order of the field, 39
  prime fields, 39, 40
  rational number fields, 39–40
  unit elements (multiplicative identities), 38
  zero elements (additive identities) of, 38
  see also GF (Galois (finite) fields)
finite fields for construction
  for general construction of QC-LDPC codes, 485–6
  of LDPC codes, 484–522
  of nonbinary QC-LDPC codes, 614–19
  see also GF (Galois (finite) fields); QC (quasi-cyclic)-LDPC codes construction
finite groups, 32–5
  additive groups, 33, 35
  commutative groups, 32
  cyclic groups, 35
  identity elements, 32
  modulo-m addition, 32–3
  modulo-p multiplication, 33–4
  multiplicative groups, 34–5
finite-length ensemble weight enumerators, 368–71
  example, 371
  Kronecker delta functions, 369
  weight-vector enumerators, 368–71
finite-state channels, 21
fountain codes, 672–5
  about fountain codes, 672–3
  LT (Luby Transfer) codes, 674–5
  raptor codes, 675
  tornado codes, 673–4
FSTD (finite-state transition-diagram), 170

G-LDPC (generalized LDPC) codes, 287–91
  examples, 288–9, 289
  parity-check matrix, 290
  protograph design, 287–9
  rate-1/2 G-LDPC code, 290–1
GA (Gaussian approximation), 402–7
  about the GA, 402–3
  consistency condition, 402
  consistent normal density, 402
  for irregular LDPC codes, 404–7
    GA density-evolution recursion, 406
    pdf for, 405–6

© in this web service Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-0-521-84868-8 - Channel Codes: Classical and Modern
William E. Ryan and Shu Lin
Index
More information

GA (Gaussian approximation) (*Cont.*)
  for regular LDPC codes, 403–4
    example, 404
Gallager codes, 257–8
Gallager error exponent, 18
Gallager random coding bound, 18
Gallager SPA decoder, 218–22
  about the decoder, 218, 221–2
  algorithm, 220
  BEC special case, 219
  BI-AWGNC special case, 219
  BSC special case, 219
  Rayleigh special case, 219–20
Gallager's algorithm A and algorithm B, 246–7
GCP (generalized-constraint processor), 669
GE (Gilbert–Elliot) channels, 21–2
GF (Galois (finite) fields), 41–4
  cancelation law, 43
  examples, 41–2
  order of nonzero fields, 43–4
  and prime fields, 41
  primitive elements, 44
  see also finite fields for construction
GF (Galois (finite) fields), construction, 56–63
  examples, 60–2
  extension fields, 60
  ground fields, 60
  isomorphic fields, 63
  polynomial representations, 59–63
  power representations, 59–63
  subfields, 56
  vector representations, 61–3
GF (Galois (finite) fields), properties, 64–9
  additive subgroups, 69–70
  conjugates, 64–5
  cyclic subgroups, 70
  Euler's formula, 69
  examples, 68, 69, 70
  extension fields, 64, 69
  minimal polynomials, 66–8
  monic irreducible polynomials, 67
  primitive elements, 65
girth 8 Tanner graph LDPC codes, 554–7
  examples, 555–6, 557
  SPC (single parity-check code), 555
graphs, basic concepts, 80–3
  complete graphs, 81–2
  degree of a vertex, 81
  edges
    adjacent edges, 80
    edge sets, 80
    incident edges, 80
  finite/infinite graphs, 81
  matrices
    adjacency, 83
    incidence, 83
  pendants, 80
  regular graphs, 81–2
  self loops, 80
  subgraphs, 82–3
  vertex-sets, 80
  vertices
    end vertices, 80–1
    isolated vertices, 80–1
graphs, bipartite, 86–8
  adjacency matrix, 88
  cycles are even length, 87
  definition, 86
  girth, 87
graphs, paths and cycles, 84–6
  acyclic graphs, 86
  cycle/cycle length, 85
  diameter of a connected graph, 85
  eccentricity of a vertex, 84–5
  girth, 85–6
  paths
    connected vertices, 84
    definition, 84
    distance between vertices, 84
    length of, 84
    shortest path, 85
    starting and ending, 84
  pendants, 86
  radius of a connected graph, 85
  trees, 86
Gray labeling, 637–8
groups, 30–8
  additive groups, 33
  basic concept, 30–1
    abelian group, 30
    commutative group, 30
    definition, 30
    finite/infinite groups, 31
    identity elements, 30
    simple group example, 31
    unique groups, 30–1
  cosets, 37–8
    example, 37
  Lagrange's theorem, 38
  subgroups, 35–8
    examples, 36
  see also finite groups

Hamming-code (7,4), 4–7
Hamming-distance, 6, 9, 100
Hamming-metric, and the MLSD, 173
Hamming-weight, 99
hybrid FEC/ARQ schemes, 3

identity elements, 30, 32
incidence matrix, 524
incidence vectors, 457, 525–6, 601
independence assumption, 215
interleaved codes, 130–1
  block interleaving, 130–1
  interleaving depth, 131
  multiple bursts of errors, 131
interpolation complexity coefficient, 603
intersecting bundles of lines of Euclidean geometries, 512–16
intersecting flats in EGs and matrix dispersion construction of nonbinary QC-EG-LDPC codes, 624–8
  example, 627–8

© in this web service Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-0-521-84868-8 - Channel Codes: Classical and Modern
William E. Ryan and Shu Lin
Index
More information

intersections between sets, 30
IO-WE (input-output WE), 341, 362–3
IP-WE (information-parity WE), 340, 342
IRA (irregular repeat-accumulate) code enumerators, 364–6
  example, 366
IRA (irregular repeat-accumulate) LDPC codes, 267–76
  about IRA codes, 267
  QC-IRA (quasi-cyclic IRA) code design, 271–6
  tail-biting, 268, 270
  Tanner graph for, 268
IRAA (irregular repeat-accumulate-accumulate) LDPC codes, 278–9
  example, 278
  QC-IRAA/QC-IRA comparison, 279
IRGA (IRA codes generalized accumulator based), 277
ISI (intersymbol interference) channels
  LDPC coded with EXIT charts, 648–50
  limits/achievable rates, 21–4
  turbo equalization for, 644–8
    example, 646–7
iterative decoders
  of generalized LDPC codes, 241–3
    example, 242
  of LDPC codes for BEC, 561–3
    erasures, 561
    stopping sets, 563
iterative decoding performance *see* decoding thresholds for LDPC and turbo code ensembles
i.u.d. (independent, uniformly distributed) capacity C_{iud}, 23, 24

Kronecker delta functions, 369

Lagrange's theorem, 38
LCM (least common multiple), 111
LDPC (low-density parity-check) coded modulation, 636–44
  about LDPC coded modulation, 636–8
  EXIT based design, 638–44
  LLR output, 637
  M-ary modulation, 636
  MAP symbol-to-bit metric calculator demodulator, 636–7
LDPC (low-density parity-check) codes
  about LDPC codes, 201, 248
  BEC decoding algorithms, 243–7
  BSC bit-flipping algorithm, 247–8
  BSC, Gallager's algorithm A and algorithm B, 246–7
  classifications, 205–8
    cyclic codes, 206
    generalized codes, 207–8
    quasi-cyclic codes, 206
    random codes, 206
  convolution codes, 670–2
    construction, 671–2
    decoders for, 672
    encoders for, 672
  with fields, 38
  Gallager codes, 257–8
  graphical representation, 202–5
    example, 203
    Tanner graphs, 202–4
  invention of, 10
  iterative decoders for generalized LDPC codes, 241–3
  McKay codes, 258–9
  matrix representations, 201–2
  message passing and turbo principle, 208–13
  RC ((row-column) constraint, 202
  regular/irregular LDPC codes, 202
  universality of, 407–12
  see also computer-based design of LDPC codes; nonbinary LDPC codes; SPA (sum-product algorithm), LDPC codes
linear block codes, 94–106
  about linear block codes, 94–5
  BCH (Bose–Chandhuri–Hocquenghem) codes, 111–21
  code rate, 95
  codewords, 94
  concatenated codes, 131–3
  cyclic codes, 106–11
  decoding, 102–6
  definition, 95
  detecting/correcting transmission errors, 95
  error detection with, 98–9
  generator and parity-check matrices, 95–8
    codewords, 97
    combinatorially equivalent codes, 98
    linear systematic code, 97
    null space, 96
    parity check bits, 97
    parity check equations, 98
    redundant rows, 96
    row space, 96
  information sequences, 94
  interleaved codes, 130–1
  product codes, 129–30
  QC (quasi-cyclic) codes, 133–42
  redundant bits, 94–5
  repetition and single-parity-check codes, 142–3
  single-parity-check codes, 142–3
  weight distribution, 99–102
  see also nonbinary linear block codes
lines of EG for construction of LDPC codes, 430–6
  and parallel bundles of lines, 436–9
LLR (log-likelihood ratio), 180, 637, 667
LMS (least-mean-square) algorithm, 644
low error floors, LDPC decoder design for, 657–70
  about low error floors, 657–9
  bimode decoder, 661–6
  codes under study, 659–61
    Margulis code, 661
    short QC code, 660–1
  concatenation and bit-pinning, 666–8
    about concatenation and bit-pinning, 666–7
    Margulis-code solution, 668
  generalized-LDPC decoder, 668–70
    Margulis-code solution, 670
    short-QC-code solution, 669

© in this web service Cambridge University Press www.cambridge.org

Cambridge

Cambridge University Press
978-0-521-84868-8 - Channel Codes: Classical and Modern
William E. Ryan and Shu Lin
Index
More information

low error floors, LDPC decoder design for (*Cont.*)
  iterative decoders and trapping set problems, 658–9
LPCF-BF (largest parity-check failures) decoding algorithm, 469
LR (likelihood ratio), LDPC codes, 213
LT (Luby transfer) codes, 674–5
  code packets, 675
  encoding process, 675
  rate compatible codes, 674

M-ary input memoryless channels coding limits, 18–21
  M-ary AWGN channel, 19–21
  M-ary symmetric channel, 21
  Shannon capacity, 19, 21
  unconstrained-input AWGN channel, 18–19
M-ary modulation, 636
McKay codes, 258–9
MAP (maximum-a posteriori) rule/decoders, 8–9
  with iterative decoders, 241–2
  SPC decoder, 217–18
MAP (maximum-a posteriori) symbol-to-bit metric calculator demodulator, 636–7
Margulis code/solution, 661, 663–5, 668, 670
masking
  for EG-LDPC code construction, 444–50
  for non-binary QC-LDPC construction, 617
Mathematical Theory of Communication, C. Shannon, 1–3
matrix dispersions of elements of a finite field, 484–5
  CPM (circulant permutation matrix), 485
MDS (maximum-distance-separable) code, 123
Message passing and turbo principle, LDPC codes, 208–13
  belief propagation, 208
  constituent decoders, 210
  example, 209
  extrinsic information concept, 211
  intrinsic information concept, 211
  turbo principle, 212–13
MET (multi-edge-type) LDPC codes, 265–6
minimal convolution encoders, 159–62
minimum-distance (nearest-neighbor) decoding, 102
minimum-probability-of-error design criteria, 8
minimum-weight codewords of an RS code with two information symbols, 487–95
ML (maximum-likelihood) decoder performance
  for block codes, 187–9
  for convolutional codes, 193–5
    example, 194–5
  and weight enumerators, 339
ML (maximum-likelihood) rule, 8–9
MLD (maximum-likelihood decoding), 102
  for the BEC, 244–6
MLSD (maximum-likelihood sequence decoder/detector), 172–7
  AWGN branch metric, 176
  BSC and BI-AWGN channel, 172–3
  correlation metric, 176
  cumulative metric, 173
  decision stage, 174–5
    block-oriented 1 and 2 approaches, 175
    stream-oriented 1 and 2 approaches, 175
  Euclidean distance metric, 176
  example, 176–7
  Hamming metric, 173
  with ISI channels, 644
  survivor path, 173
  Viterbi algorithm, 174
MMSE (minimum-mean-squared-error) equalizer, 644
modulators, 2
modulo-m addition, 32–3
modulo-p multiplication, 33–4
monic polynomials, 52
Monte Carlo integration, 14
multiple bursts of errors, 131
multiplicative groups, 34–5
multiplied row constraints, 485
mutual channel information, 10

NAK (negative acknowledgement) signals, 3
Newton identities, 116–17
nonbinary LDPC codes, 592–633
  about nonbinary LDPC codes, 592–3
  definitions, 592–3
  see also FG (finite geometries); finite fields for construction; intersecting flats in EGs and matrix dispersion for construction of nonbinary QC-EG-LDPC codes; QSPA (q-ary SPA) decoding of nonbinary LDPC codes
nonbinary linear block codes, 121–8
  about q-ary block codes, 121–2
  error-location polynomial, 126
  error-value evaluator, 126
  Euclid's iterative division algorithm, 126–8
  example, 128
  MDS code, 123
  nonprimitive RS code, 125
  overall parity-check symbol, 124–5
  RS (Reed-Solomon) codes, 123–5
  syndrome polynomial, 126
nonbinary QC EG-LDPC codes, 607–9
  examples, 608–9
nonbinary regular EG-LDPC codes, 610–11
  example, 610–11

OFDM (orthogonal frequency-division multiplexing), 645
order of nonzero fields, 43–4
orthogonal syndrome sums, 462
OSMLG (one-step majority-logic) for FG OSMLG codes, 460–8
  EG-LDPC codes, 465
  examples, 465–6, 467
  and FG-LDPC codes, 466–7
  orthogonal syndrome sums, 462
  OSMLG decoding, 463–4
    algorithm, 464
  OSMLG error correction capability, 463
  over the BSC, 461–8

© in this web service Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-0-521-84868-8 - Channel Codes: Classical and Modern
William E. Ryan and Shu Lin
Index
More information

parity-check failure, 461
WBF (weighted BF) decoding algorithms, 460

pair-wise constraint, 547
parallel flats in Euclidean geometries, 620–4
parity checking
parity check bits, 97
parity check failure, 461
parity check matrix, 96, 98
parity-check equations, 98
parity vectors, 587
paths on graphs, 84–5
PCC (parallel-concatenated codes) ensemble enumerators, 343–56
about ensemble enumerator PCCs, 343–5
PCCC (parallel-concatenated convolution codes), 298–306
about PCCCs, 298–9
algorithm
decision after last iteration, 316
initialization, 314–15
nth iteration, 315
outline, 314
CC (constituent code), 302
example, 304–5
interleaver, critical properties, 300
interleaver gain, 304
lower complexity approximations, 316–20
Bayes' rule with, 318
bit-wise reliability, 319
path-wise reliability, 319
SOVA (soft-output Viterbi algorithm), 317, 319
Viterbi decoder/algorithm, 317
performance estimate on the BI-AWGNC, 301–6
pseudo-code, 313–14
puncturers, 301
RSC codes, critical properties, 299–300
spectral thinning, 302, 305–6
summary, 313–16
PCCC (parallel-concatenated convolution codes) ensemble enumerators, 345–56
about PCCC ensemble enumerators, 345–8
deriving C-PWEs, 345
design principles, 350–1
EBCs (equivalent block codes), 346
ensemble BER performance, 348
ensemble performance bounds and simulated performance curves, 355–6
example performance bounds, 351–6
with non-recursive constituent encoders, 348
with recursive constituent encoders, 348–50
PCCC (parallel-concatenated convolution codes) iterative decoder, 306–20
about PCCC iterative decoders, 306–8
BCJR algorithm, 310–11
decoder details, 309–13
overview, 308–9
SISO BCJR decoder, 309–12
pdf (probability density function), 388, 391–2, 405–6
conditional pdfs, 421
with EXIT charts, 640
PEG (progressive-edge-growth)
algorithm, 259–60
QC-IRA LDPC code design, 273
tanner graph construction of LDPC codes, 542–6
performance measures, 7–10
see also design criteria
Peterson decoding algorithm, 111
PG (projective geometries), 76–80
μ-flats, 78–80
based on construction of nonbinary LDPC codes, 611–14
example, 613–14
for construction of cyclic and QC-LDPC codes, 455–9
lines, 77–8
intersecting bundles, 78–9
m-dimensional, 76
nonzero element partitioning, 77
polynomials over finite fields, 51–6
about polynomials, 51
adding/multiplying polynomials, 52–3
associative/commutative/distributive laws, 53
Euclid's division algorithm, 54
examples, 53, 54
irreducible polynomials, 54–5
monic polynomials, 52
primitive polynomials, 55
rings, 53–4
zero polynomials, 52
power-sum symmetric functions, 116
PR (partial response) with the ISI format, 645
prime fields, 41
additive groups, 506–10
primitive elements, 44, 510–12
product codes, 129–30
construction, 129
decoding, 130
direct products, 129
encoding, 129–30
column encoding, 130
row encoding, 130
parity check symbols, 129
protograph LDPC codes, 261–5
with ARA codes, 280–5
circulant permutation matrices, 262
decoding architectures, 264–5
circulant-based architecture, 264
example, 263
lifting process, 262
protograph-based LDPC code enumerators, 367–82
about protograph enumerators, 367
asymptotic ensemble weight enumerators, 371–8
ensemble trapping-set enumerators, 379–82
finite-length ensemble weight enumerators, 368–71
quasi-cyclic code issues, 367
puncturing
with C-IRA LDPC code design, 275
with turbo-codes, 301

© in this web service Cambridge University Press www.cambridge.org

Cambridge
Cambridge University Press
978-0-521-84868-8 - Channel Codes: Classical and Modern
William E. Ryan and Shu Lin
Index
More information

QC (quasi-cyclic) codes, 133–42
  circulants, 133–5
    generators of, 133
    systematic circular form, 134–5
  codewords, 141–2
  CSRAA circuit, 135–7
  definition, 133
  and ensemble enumerators, 367
  example, 134–5
  full-rank matrix, 137
  parity-check matrix, 137–9
  semi-systematic circulant form, 139–41
  t-section QC code, 133
QC (quasi-cyclic)-LDPC codes construction
  about QC-LDPC codes from finite fields, 485–6
  from the additive group of a prime field, 506–10
    examples, 508–10
  from intersecting bundles of lines of Euclidean geometries, 512–16
    example, 515
  from minimum weight codewords of an RS code, 487–95
    examples, 489–95
    Tanner graph, 488
  from primitive elements of a field, 510–12
    example, 512
  from subgroups of a finite field, 501–6
    the additive group of the field, 501–3
    examples, 502–3, 505–6
    the multiplicative group of the field, 503–6
  from universal parity check matrices, 495–500
    examples, 497–500
    RC-constrained arrays of CPMs, 495–6
  PG-LDPC codes, 458–9
  see also EG (Euclidean geometries)
QC-EG-LDPC (quasi-cyclic-EG-LDPC) code construction
  by circulant decomposition, 450–5
    column/row decomposition, 450–2, 455
    column/row splitting, 450–2
    example, 452–4
  by parallel flats in EGs and matrix dispersion, 620–4
    example, 622–4
QC-IRA (quasi-cyclic IRA) LDPC code design, 271–6
  constant information word length designs, 274–5
  design algorithm, 273
  examples, 273, 275–6
  PEG conditioning, 273
  puncturing, 275
  QC-IRA/regular QC-LDPC comparison, 272
q^m-ary cyclic EG-LDPC codes, 601–6
  examples, 603–5, 605–6
  incidence vectors, 601
  interpolation complexity coefficient, 603
QPSK (quaternary phase-shift keying), 19
QSPA (q-ary SPA) decoding of nonbinary LDPC codes, 593–8
  FFT-QSPA, 598–600
    advantages, 600
  initialization, 596
  Tanner graphs, 595
quantized density evolution, 399–401
  algorithm for, 401
  example, 401

RA (repeat-accumulate) codes, 267
  RA code enumerators, 362–4
  see also IRA (irregular repeat-accumulate) LDPC codes
random-erasure-correction capability, stopping sets, 563–4
raptor codes, 675
RCBP (reduced-complexity box-plus) decoder, 236–41
  algorithm, 240
recursive constituent encoders, 348–50
redundant bits, in linear block codes, 94–5
reliability profile, and the BF (bit-flipping) algorithm, 468
REP (repetition) codes, 214–15
  and APP processor, 216
  and SPC codes, 142–3
replacement constraint, 547
Richardson/Novichkov decoder, 234–6
rings, and polynomials, 53–4
RS (Reed–Solomon) codes, 123–5, 495–500
  structured RS-based LDPC codes, 516–20
RSC (recursive systematic convolution) encoders, 156–7, 298–300
  class variations, 156–7
  critical properties, 299–300
  example, 300

scalars with vectors, 45
SCC (serial-concatenated code) ensemble enumerators, 356–62
  about SCC ensemble enumerators, 356–7
SCCC (serial-concatenated convolution code), 320–8
  about SCCC encoders, 320
  BI-AWGNC performance estimate, 320–3
  ensemble enumerators, 358–62
    high-SNR region, 360–1, 362
    low-SNR region, 361–2
  examples, 320, 321–2, 323
  inner/outer code rates, 320
  iterative decoder, 323–8
    algorithm, 325–8
    SISO decoding modules, 324
SER (symbol-error rate), 10
sets and binary operations, 28–30
  associative law/operations, 29
  binary operation on a set, 28–9
  cardinality of a set, 28
  commutative law/operation, 29
  elements of a set, 28
  finite/infinite sets, 28
  intersections between sets, 30
  sets, definition, 28
  subsets/proper subsets, 28
Shannon, C., *Mathematical Theory of Communication*, 1–3
  channel, 2

© in this web service Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-0-521-84868-8 - Channel Codes: Classical and Modern
William E. Ryan and Shu Lin
Index
More information

channel encoder and decoder, 1–2
code rate, 2–3
modulator and demodulator, 2
source encoder and decoder, 1
source and user (sink), 1
Shannon capacity, 15–16, 19, 21
Shannon limit, with fields, 38
Shannon–McMillan–Breiman theorem, 23
single-accumulator-based LDPC codes, 266–77
IRA (irregular repeat-accumulate) codes, 267–76
IRGA (IRA codes generalized accumulator based), 277
RA (repeat accumulate codes), 267
SISO (soft-in/soft-out) BCJR decoder, 307–12, 324–5
SNR (signal-noise ratio), and weight enumerators, 339
SOVA (soft-output Viterbi algorithm), 317, 319
SPA (sum-product algorithm), LDPC codes, 213–26
about SPA, 213–15
APP ratio, 213
box-plus SPA decoder, 222–5
Gallager SPA decoder, 218–22
independence assumption, 215
LR (likelihood ratio), 213
performance comments, 225–6
REP (repetition) codes, 214–15
and APP processor, 216
single parity-check code MAP decoder, 217–18
SPA (sum-product algorithm), LDPC codes, reduced complexity, 226–41
about reduced-complexity SPA, 226
approximate min[*] decoder, 233–4
attenuated and offset min-sum decoders, 229–31
min-sum decoder, 226–9
examples, 227–8
min-sum-with-correction decoder, 231–3
RCBP (reduced-complexity box-plus) decoder, 236–41
Richardson/Novichkov decoder, 234–6
SPC (single parity-check) code, 142–3, 555
MAP decoder, 217–18
Spectral thinning, with PCCC, 302
sphere-packing bounds, 18
SSF (stopping-set-free), 564
standards, accumulator-based LDPC codes in, 285–7
Stirling's approximation, 372
stop-and-wait ARQ schemes, 3
stopping sets/stopping set free, 563–4
structured RS-based LDPC codes, 516–20
example, 520
subgroups, 35–8
superposition construction of LDPC codes, 546–54
base and constituent matrices construction, 548–51
examples, 549, 550–1
construction of product LDPC codes, 552–4
example, 553–4
cyclic replacement, 548
pair-wise constraint, 547
replacement constraint, 547
superposition dispersion construction of nonbinary QC-LDPC codes, 628–30
column/row-replacement constraint, 629
example, 629–30
superposition LDPC codes, erasure burst correction, 574–5
survivor path, and MLSD, 173
symbol-error probability, 10

tail-biting, 171, 268, 270
Tanner graphs
with ACE algorithm, 260–1
girth 8 LDPC codes, 554–7
for IRA codes, 268
with LDPC codes, 202–3
PEG construction of LDPC codes, 542–6
ACE message degree, 546
example, 545
with QC-LDPC codes, 488
and QSAP, 595
tornado codes, 673–4
TPC (turbo product code), 328–34
about TPCs, 328–30
and BCH codes, 328–9
example, 329
turbo decoding of product codes, 330–4
about the decoding, 330
codeword decisions, 331
computing soft outputs and extrinsic information, 331–3
decoder process, 333–4
obtaining codeword lists, 330–1
transmission errors, 98–9
trapping loops, WBF decoding, 474
trapping sets *see* error floors and trapping sets for LDPC codes
trellis diagrams, 171
trellis-based construction of LDPC codes, 537–42
code construction, 540–2
example, 541–2
removing short cycles from a bipartite graph, 538–40
trellis-based decoders
about trellis-based decoders, 172–3
BCJR algorithm, 180, 183, 185–7
bit-wise MAP decoding, 180–7
differential Viterbi decoding, 177–9
MLSD and Viterbi algorithm, 172–7
performance estimates, 187–95
bit error probability, 189
ML decoder for block codes, 187–9
ML decoder for convolutional codes, 193–5
two-codeword error probability, 188
weight enumerators, 189–93
see also bit-wise MAP decoding; differential Viterbi decoding; MLSD (maximum-likelihood sequence decoder)

© in this web service Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-0-521-84868-8 - Channel Codes: Classical and Modern
William E. Ryan and Shu Lin
Index
More information

turbo codes, 132, 298–337
  invention of, 10
  PCCC, 298–320
    iterative decoder, 306–20
  SCCC, 320–8
  TPC (turbo product codes), 328–34
turbo equalization for ISI channels, 644–8
  example, 646–7
turbo principle, 212

UEBR (unresolved (or unrecovered) erasure bit rate), 565
UEWR (unresolved erasure word rate), 566
uniform circulant masking, 528
unit elements (multiplicative identities) with fields, 38
universal-parity-check matrices, 495–500
universality of LDPC codes, 407–12

Vandermonde matrix, 113
vector spaces, 45–51
  basic definitions, 45–6
    scalar multiplication, 45
    scalars, 45
    vector addition, 45
  dual spaces, 50–1
  finite vector spaces over finite fields, 48–50
    binary n-tuples, 50
    n-tuple over GF, 48–50
    ordered sequences, 48
  inner products, 50–1
    example, 51
  linear dependence, 47
  linear independence and dimension, 46–8
  propositions for, 46
  spanning vector space, 48
  subspaces, 46, 49
Venn diagram, (7,4) Hamming code, 5–6
Viterbi algorithm
  differential Viterbi decoding, 177–9
  and ISI channels, 644
  and MLSD, 174
  and PCCC iterative decoder, 317
VNPs (variable-node processors), and EXIT charts, 412–17, 420, 641

WBF (weighted BF) decoding algorithms, 460, 469–76
  algorithm one, 469–72
    weighted reliability measures/profile, 471
  algorithm two and three, 472–6
    confidence coefficient, 473
    with loop detection, 476
    trapping loops, 474
WE (weight enumerators)/enumerating functions, 339, 340
  C-OWE (conditional output-WE), 341
  C-PWE (conditional parity-WE), 341
  CI-WE (cumulative information-WE), 342
  for convolutional codes, 189–93
    augmented weight enumerators, 192
    error events, 190
    examples, 190–2
  IO-WE (input-output WE), 341
  IP-WE (information-parity WE), 340, 342
weight distribution, linear block codes, 99–102
  distance distribution, 101
  ensembles of linear block codes, 100
  error-detecting capability, 101–2
  Hamming distance, 100
  Hamming weight, 99
  lower bound, 101
  minimum weight, 99, 101
  probability of an undetected error, 100
  smallest weight, 99
  triangle inequality, 100
weight-vector enumerators, 368–71
weighted BF decoding; algorithm one, weighted reliability measure/profile, 471
WER (word-error rate), 10

Z channel capacity, 13–14
zero (covering) spans, 570–2, 576, 587–8
zero elements (additive identities) with fields, 38

© in this web service Cambridge University Press www.cambridge.org