

CÁC KÍ HIỆU ĐƯỢC SỬ DỤNG

$\alpha_i(t - 1)$	là xác suất bắt đầu ở trạng thái S_i tại thời điểm t , tương ứng với tất cả các symbols nhận được từ trước cho đến thời điểm t ;
$\gamma_{i,j}(t)$	Xác suất chuyển từ trạng thái S_i sang trạng thái S_j tại thời điểm t khi nhận được từ mã $y(t)$;
β_j	Xác suất kết thúc ở trạng thái S_j tại thời điểm t , tương ứng với tất cả các symbols nhận được từ sau thời điểm t ;
ρ, τ, λ	Các hàm phân bố xác suất;
A	Ma trận thành phần A ;
\mathbf{A}^{-1}	Ma trận nghịch đảo của ma trận thành phần A ;
$F(f)$	là hàm biến đổi Fourier của hàm f ;
G	Ma trận sinh mã;
H	Ma trận kiểm tra;
\mathbf{H}^T	Ma trận kiểm tra chuyển vị;
I	Ma trận đơn vị;
K	Số bit thông tin trong một từ mã;
M	Số bit mã trong một từ mã;
M_R, M_T	Số phần tử thu và phát trong mô hình MIMO;
L_{CC}	Độ phức tạp mã chập;
L_{TC}	Độ phức tạp mã Turbo;

LR_j	(likelihood Ratio) là tỉ số xác suất bit mã thứ j có giá trị nhị phân là '0' trên xác suất bit đó có giá trị nhị phân là '1';
N	Số bit trong một từ mã;
$Q_{i,j}^a$	Xác suất truyền từ nút biến số thứ j sang nút kiểm tra thứ i ;
P_e	Xác suất lỗi bit;
$p_{ij}(t)$	xác suất bộ mã hóa thực hiện chuyển từ trạng thái S_i sang trạng thái S_j tại thời điểm t ;
$P(u(t) = 1 y)$	Xác suất có điều kiện của bit u nhận được bằng "1" khi thu được tín hiệu y ;
PR	Tỉ số xác suất;
$R_{i,j}^a$	Ước lượng xác suất tương ứng của nút kiểm tra thứ i , khi symbol thứ j ở trạng thái a ;
Rot	Hàm dịch trạng thái ô nhớ;
$Symbol$	Là một cụm bit được ánh xạ lên một sóng mang trong một cửa sổ thời gian nhất định;
T_{trmax}	Thời gian trễ của tia phản xạ;
T_{symbol}	Thời gian của một Symbol;
x	Bit được điều chế phát đi từ phía phát;
y	Bit được điều chế thu ở phía thu;
$\begin{pmatrix} n \\ L \end{pmatrix}$	ma trận một cột hai hàng, hai phần tử n, L

THUẬT NGỮ VIẾT TẮT

3G	Third Generation	Hệ thống thông tin thế hệ thứ 3
ACK	ACKnowledgement	Tín hiệu xác định đã nhận thông tin
AWGN	Additive White Gaussian Noise	Kênh truyền dẫn tạp âm trắng
BCH	Bose-Chaudhuri-Hocquenghem	Mã BCH
BEC	Binary Erasure Channel	Kênh xóa nhị phân
BER	Bit Error Ratio	Tỉ lệ lỗi bit
BP	Belief Propagation	Tích lũy độ tin cậy
BPSK	Binary Phase Shift Keying	Điều chế khóa dịch pha nhị phân
CRC	Cyclic Redundancy Check	Mã hóa kiểm tra chẵn lẻ
DVB-C	Digital TeleVision Broadcasting-Cable	Truyền hình cáp kỹ thuật số
DVB-S	Digital TeleVision Broadcasting-Satellite	Truyền hình vệ tinh kỹ thuật số
DVB-T	Digital TeleVision Broadcasting-Terrestrial	Truyền hình mặt đất kỹ thuật số
EXIT	EXtrinsic Information Transfer	Đồ thị trao đổi thông tin ngoại lai
E_b/N_0	Bit Energy per Noise power ratio	Tỉ lệ Năng lượng bit trên tạp nhiễu
FEC	Forward Error Correction	Mã sửa lỗi trước
FFT	Fast Fourier Transform	Biến đổi Fourier nhanh
GM	Generator Matrix	Ma trận sinh
G-LDPC	Generalized Low Density Parity Check	Mã có ma trận kiểm tra mật độ thấp suy rộng
H-ARQ	Hybrid Automatic Repeat reQuest	Mô hình lai ghép ARQ
IP	Internet Protocol	Giao thức mạng Internet
IPTV	Internet Protocol based TeleVision	Truyền hình sử dụng giao thức Internet
LDPC	Low Density Parity Check	Mã có ma trận kiểm tra mật độ thấp
LLR	Log Likelihood Ratio	Tỉ số Logarit hợp lệ
MAP	Maximum A Posteriori	Thuật toán cực đại xác
MIMO	Multi-Input Multi-Output	Hệ thống đa đầu vào ra

ML	Maximum Likelihood	Hợp lệ cực đại tỉ số xuất hậu nghiệm
NACK	Negative ACKnowledgement	Tín hiệu phản hồi phủ định
PCM	Parity Check Matrix	Ma trận kiểm tra
PDF	Power Density Function	Hàm mật độ công suất
QAM	Quadrature Amplitude Modulation	Điều chế biên độ cầu phương
ReqN	Request Number	Số lần yêu cầu
RN	Receive Number	Số lần nhận
RS	Reed Solomon Codes	Mã Reed-Solomon
RSC	Recursive Convolution Codes	Mã chập đệ quy
SISO	Single input Single output	Đơn kênh vào ra
SN	Sequence Number	Số thứ tự
S/N	Signal to Noise Ratio	Tỉ lệ tín hiệu trên nhiễu
SR	Shift Register	Thanh ghi dịch
URC	Unit Rate Code	Mã có tỉ lệ đơn vị
V-BLAST	Vertical Bell Labs Layered Space-Time	Hệ thống không gian thời gian phân lớp của Bell labs
XOR	Exclusive OR	Hàm logic hoặc tuyệt đối
ZF	Zero Forcing	Cưỡng bức không

Danh sách bảng

1.1	Các mốc phát triển chính trong nghiên cứu mã LDPC	30
2.1	Các thông số mô phỏng mã LDPC có hàm phân bố mật độ cho trong (2.5) và thông số $t=2$	71
2.2	Thông số mã LDPC được sử dụng trong mô phỏng	73
3.1	Các thông số mô phỏng hệ thống tích hợp V-BLAST	95
3.2	Các thông số mô phỏng	111
A.1	Các thông số của mã LDPC cho mô phỏng ảnh hưởng của số lần lặp cực đại tới khả năng sửa lỗi của mã LDPC	144
A.2	Yêu cầu tỉ số E_b/N_0 với số lần lặp giải mã cực đại khác nhau để đạt được tỉ số BER= 10^{-4} tương ứng với 3 mã LDPC có thông số cho trong bảng A.1, khi truyền dữ liệu qua kênh AWGN và kênh pha đình Rayleigh không tương quan.	149
A.3	Các thông số mã LDPC được sử dụng trong mô phỏng ảnh hưởng của độ dài từ mã đến khả năng sửa mã.	151

Danh sách hình vẽ

1	Mô hình tổng quát hệ thống truyền hình số	15
2	Các yếu tố ảnh hưởng đến khả năng của mã kênh	17
1.1	Mô hình hệ thống truyền tin số	22
1.2	Mô hình toán học kênh truyền dẫn	23
1.3	Sơ đồ bộ mã Turbo	26
1.4	Thuật toán giải mã SISO-MAP	27
1.5	Cấu trúc giải mã Turbo	28
1.6	Ma trận kiểm tra của mã LDPC	36
1.7	Ma trận kiểm tra \mathbf{H} có $N = 15$, $wc = 3$, $wr = 4, 5$, $M =$ $N - K = 10$, $r = 1/3$	37
1.8	Ma trận chuyển vị \mathbf{H}_r từ ma trận kiểm tra \mathbf{H} trong hình 4. Ma trận \mathbf{H}_r bao gồm hai ma trận thành phần \mathbf{A} và \mathbf{B} . . .	38
1.9	Tích hai ma trận thành phần trong hình 1.8 được sử dụng để tính ma trận sinh.	39
1.10	Ma trận sinh \mathbf{G} của mã LDPC được tính từ ma trận kiểm tra \mathbf{H}_r	39
1.11	Thông tin ngoại lai được bộ giải mã tạo ra từ thông tin tiền nghiệm của chuỗi bit đầu vào và thông tin kênh truyền. . .	41
1.12	Đồ thị song phương của mã LDPC	47
1.13	Lược đồ giải mã lặp của bộ mã LDPC	49

2.1	Phân bố mật độ chuẩn rời rạc hàm trọng của các cột ma trận kiểm tra thành phần.	56
2.2	Phân bố mật độ cho các bit thông tin	63
2.3	Đồ thị trao đổi thông tin EXIT của mã LDPC có hàm mật độ cho trong phương trình (2.5)	65
2.4	Đồ thị trao đổi thông tin EXIT của mã LDPC có hàm mật độ cho trong phương trình (2.5)	66
2.5	Đồ thị Histogram của mã LDPC có hàm phân bố mật độ trong 2.5 và thông số $t=2$	67
2.6	Đồ thị trao đổi thông tin EXIT của mã LDPC có hàm mật độ cho trong phương trình (2.5) và $t=2$	68
2.7	Đồ thị trao đổi thông tin EXIT của mã LDPC có hàm mật độ cho trong phương trình (2.5) và $t=3$	69
2.8	Mô phỏng khả năng sửa lỗi khác nhau của mã LDPC(1200,1800), sử dụng cùng hàm phân bố mật độ đối với các bit kiểm tra cho trong phương trình (2.5) và sử dụng các hàm phân bố mật độ khác nhau cho các bit thông tin trong từ mã LDPC.	72
2.9	Mã LDPC(1200,3600) có hàm phân bố mật độ cho trong phương trình (2.5) và $t=2$	74
2.10	Mô phỏng khả năng sửa lỗi của mã LDPC khi tăng kích thước ma trận sinh và ma trận kiểm tra lên 10 lần	75
2.11	So sánh khả năng sửa lỗi của các mã khi truyền qua kênh AWGN, sử dụng kiểu điều chế QPSK	76
2.12	So sánh khả năng sửa lỗi của các mã khi truyền qua kênh Rayleigh không tương quan, sử dụng kiểu điều chế QPSK	77
3.1	Hệ thống V-BLAST	82
3.2	Mô hình hệ thống thông tin hỏi đáp ARQ	84

3.3	Giao thức Dừng và chờ	86
3.4	Thời gian phân bố trong giao thức Dừng và chờ	86
3.5	Giao thức quay lại N bước	88
3.6	Phân bố thời gian trong giao thức quay lại N bước	88
3.7	Mô hình tích hợp mã LDPC và hệ thống V-BLAST	90
3.8	Mô hình tích hợp mã RSC-URC và hệ thống V-BLAST	91
3.9	Đồ thị EXIT của hệ thống tích hợp mã LDPC và V-BLAST với độ dài tráo $L=2.400$ bit	92
3.10	Đồ thị EXIT của hệ thống tích hợp mã LDPC và V-BLAST với độ dài tráo $L=24.000$ bit	94
3.11	Mô phỏng quan hệ BER và E_b/N_0 của các hệ thống V-BLAST tích hợp các mã kênh khác nhau	96
3.12	Sơ đồ khối bộ H-ARQ tích hợp LDPC có trợ giúp của bộ điều chế	100
3.13	Cấu trúc gói IP	101
3.14	a) Đồ thị chòm sao của kiểu ánh xạ mã Gray b) Đồ thị chòm sao của kiểu ánh xạ phân đoạn	105
3.15	Đồ thị EXIT của mô hình H-ARQ tích hợp mã LDPC sử dụng bộ ánh xạ mã Gray (Mô hình 2) trong hình 3.14	108
3.16	Đường cong đồ thị EXIT và đường hội tụ của mô hình hệ thống H-ARQ tích hợp mã LDPC sử dụng bộ ánh xạ phân đoạn (Mô hình 1) của hình 3.14	109
3.17	Khả năng hoạt động của các mô hình hệ thống 1,5 và 6, khi điều chế 16-QAM và kênh truyền là AWGN	112
3.18	Khả năng hoạt động của các mô hình hệ thống 1, 3 và 4, khi điều chế 16-QAM và kênh truyền là AWGN	114

3.19	Khả năng hoạt động của các mô hình hệ thống 1 và 2, khi điều chế 16-QAM và kênh truyền là AWGN	115
A.1	Ảnh hưởng của số lần lặp cực đại tới khả năng sửa lỗi của mã LDPC(100,200), khi truyền dữ liệu qua kênh AWGN, điều chế BPSK.	145
A.2	Ảnh hưởng của số lần lặp cực đại tới khả năng sửa lỗi của mã LDPC(250,500), khi truyền dữ liệu qua kênh AWGN, điều chế BPSK.	146
A.3	Ảnh hưởng của số lần lặp cực đại tới khả năng sửa lỗi của mã LDPC(500,1000), khi truyền dữ liệu qua kênh AWGN, điều chế BPSK.	146
A.4	Ảnh hưởng của số lần lặp cực đại tới khả năng sửa lỗi của mã LDPC(100,200), khi truyền dữ liệu qua kênh pha đình Rayleigh không tương quan, điều chế BPSK.	147
A.5	Ảnh hưởng của số lần lặp cực đại tới khả năng sửa lỗi của mã LDPC(250,500), khi truyền dữ liệu qua kênh pha đình Rayleigh không tương quan, điều chế BPSK.	147
A.6	Ảnh hưởng của số lần lặp cực đại tới khả năng sửa lỗi của mã LDPC(500,1000), khi truyền dữ liệu qua kênh pha đình Rayleigh không tương quan, điều chế BPSK.	148
A.7	Độ tăng ích của 3 mã LDPC có các thông số trong bảng A.1 tại $BER = 10^{-4}$, khi dữ liệu được điều chế BPSK và kênh truyền dẫn là AWGN và Rayleigh không tương quan.	148
A.8	Hiệu quả số lần lặp giải mã khác nhau đối với 3 mã LDPC có thông số trong bảng A.1, khi truyền dữ liệu qua các kênh AWGN và pha đình Rayleigh không tương quan.	150

A.9	Ảnh hưởng của độ dài từ mã đến khả năng sửa lỗi FER của 4 mã LDPC theo tỉ số E_b/N_0 có thông số cho trong bảng A.3, khi truyền dữ liệu qua kênh truyền AWGN và sử dụng kiểu điều chế BPSK.	152
A.10	Ảnh hưởng của độ dài từ mã đến khả năng sửa lỗi FER của 4 mã LDPC theo tỉ số E_b/N_0 có thông số cho trong bảng A.3, khi truyền dữ liệu qua kênh truyền pha đỉnh Rayleigh không tương quan và sử dụng kiểu điều chế BPSK.	152

LỜI CAM ĐOAN

Tôi xin cam đoan các kết quả trình bày trong luận án là công trình nghiên cứu của tôi dưới sự hướng dẫn của cán bộ hướng dẫn. Các số liệu, kết quả trình bày trong luận án là hoàn toàn trung thực và chưa được công bố trong bất kỳ công trình nào trước đây. Các kết quả sử dụng tham khảo đều đã được trích đầy đủ và theo đúng quy định.

Hà Nội, Ngày 8 tháng 3 năm 2014

Tác giả

Cao Văn Liết

LỜI CẢM ƠN

Trước tiên tác giả xin gửi lời cảm ơn đến Viện Điện tử, Tin học, Tự động hóa thuộc Bộ Công thương đã tạo mọi điều kiện thuận lợi cho tác giả trong thời gian nghiên cứu và hoàn thành Luận án.

Với lòng kính trọng và biết ơn sâu sắc, tác giả xin gửi lời cảm ơn tới hai Thầy giáo hướng dẫn PGS. TSKH. Nguyễn Hồng Vũ và TS. Nguyễn Thế Truyện đã tận tình giúp đỡ tác giả từ những bước đi đầu tiên xây dựng ý tưởng nghiên cứu, cũng như trong suốt quá trình nghiên cứu và hoàn thiện Luận án. Hai Thầy đã luôn ủng hộ, động viên và hỗ trợ những điều kiện tốt nhất để tác giả hoàn thiện Luận án.

Tác giả xin gửi lời cảm ơn chân thành tới các Thầy cô và các bạn đồng nghiệp và các Phòng, Ban của Đài Truyền hình Việt Nam nơi tác giả công tác đã tạo mọi điều kiện thuận lợi và giúp đỡ tác giả trong quá trình học tập và nghiên cứu.

Cuối cùng, với tình yêu từ đáy lòng, tác giả xin gửi lời cảm ơn tới bố mẹ, vợ và hai con, những người thân yêu trong gia đình đã luôn ở bên cạnh tác giả, động viên tác giả về vật chất và tinh thần để tác giả vững tâm hoàn thành Luận án của mình.

Ngày 8 tháng 3 năm 2014

Tác giả

Cao Văn Liết

Mục lục

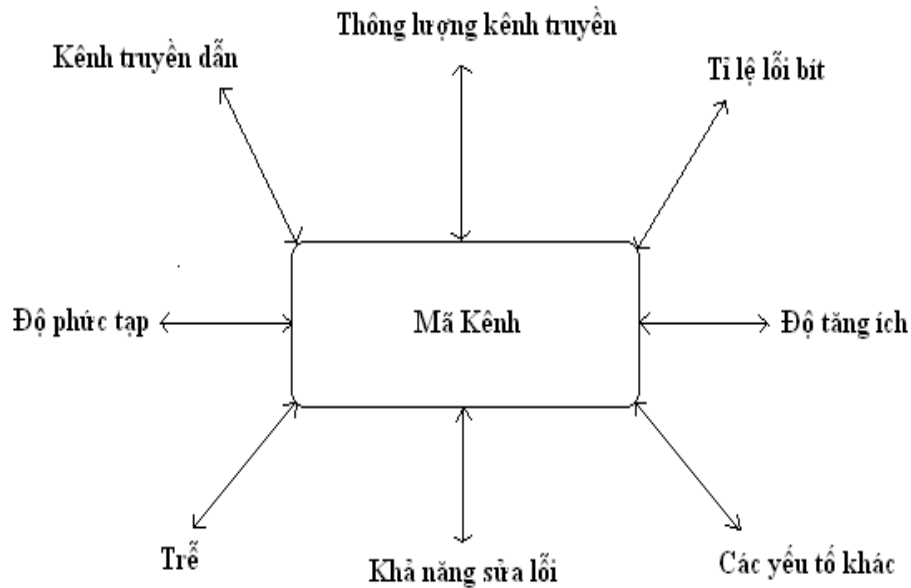
CÁC KÍ HIỆU ĐƯỢC SỬ DỤNG	1
THUẬT NGỮ VIẾT TẮT	3
DANH SÁCH CÁC BẢNG	5
DANH SÁCH CÁC HÌNH VẼ	6
MỞ ĐẦU	15
Chương 1. MÃ SỬA SAI CÓ MA TRẬN KIỂM TRA MẬT ĐỘ THẤP LDPC	22
1.1. Một số mã sửa sai thông dụng	22
1.2. Tổng quan mã LDPC	35
1.3. Các phương pháp giải mã LDPC	40
1.3.1. Phương pháp giải mã dựa theo xác suất	40
1.3.2. Phương pháp truyền giá trị thông tin <i>LLR</i>	46
1.4. Kết luận chương 1	49
Chương 2. THIẾT KẾ MA TRẬN SINH VÀ MA TRẬN KIỂM TRA CỦA MÃ LDPC	51
2.1. Xây Dựng các hàm phân bố	52
2.1.1. Xây dựng hàm phân bố cho ma trận thành phần	53
2.1.2. Xây dựng hàm phân bố cho các bit thông tin	62
2.2. Phân tích mã LDPC bằng đồ thị EXIT	64

2.3. Mô phỏng, đánh giá mã LDPC được thiết kế.....	70
2.4. Kết luận chương 2	77
Chương 3. XÂY DỰNG CÁC HỆ THỐNG TÍCH HỢP MÃ LDPC.....	79
3.1. Hệ thống thông tin.....	79
3.1.1. Hệ thống thu phát phân tập MIMO.....	79
3.1.1.1. Mô hình hệ thống phân tập.....	80
3.1.1.2. Mô hình hệ thống ghép kênh theo thời gian.....	81
3.1.1.3. Kỹ thuật tách sóng V-BLAST.....	82
3.1.1.4. Mô hình phân tập Alamouti.....	83
3.1.2. Hệ thống thông tin hỏi đáp ARQ.....	84
3.2. Hệ thống tích hợp mã hóa LDPC – V-BLAST.....	89
3.3. Hệ thống H-ARQ tích hợp mã LDPC.....	98
3.4. Kết luận chương 3	116
KẾT LUẬN VÀ ĐỀ XUẤT	117
CÁC CÔNG TRÌNH ĐÃ CÔNG BỐ.....	119
TÀI LIỆU THAM KHẢO.....	120
PHỤ LỤC A. MÔ PHỎNG ĐÁNH GIÁ MÃ LDPC...	143

nguồn tài nguyên của mạng viễn thông là hữu hạn nên việc khai thác nguồn tài nguyên của mạng viễn thông một cách hiệu quả là yêu cầu tiên quyết trong thiết kế hệ thống viễn thông số. Khi dữ liệu thông tin số được truyền qua kênh truyền dẫn có tạp nhiễu, đặc biệt là môi trường truyền dẫn có can nhiễu pha đing việc truyền dẫn số sẽ xảy ra lỗi. Để sửa lỗi, nhiều phương pháp, nhiều loại mã sửa sai đã được áp dụng trong mô hình truyền dẫn số như mã chập [1], Mã Reed-Solomon [2], mã BCH (Bose-Chaudhuri-Hocquenghem BCH codes) [3] [4], mã Turbo [5], mã có ma trận kiểm tra mật độ thấp LDPC (Low Density Parity Check) [6]. Các mô hình tích hợp giữa các mã trong và mã ngoài được áp dụng làm tăng cường khả năng sửa lỗi của hệ thống như các mô hình truyền hình số theo tiêu chuẩn DVB-T, DVB-C, DVB-S sử dụng mô hình tích hợp giữa mã chập đóng vai trò mã trong sửa lỗi bit và mã Reed-Solomon đóng vai trò mã ngoài sửa lỗi khối.

Để tăng khả năng sửa lỗi, các hệ thống truyền hình số tiêu chuẩn thế hệ thứ 2 như DVB-T2, DVB-C2, DVB-S2 sử dụng các mô hình tích hợp mã LDPC với mã BCH có độ phức tạp cao hơn thay thế cho mã chập và mã Reed-Solomon.

Trong quá trình thiết kế bộ mã kênh cần xem xét kỹ lưỡng các yếu tố để phù hợp với từng mô hình. Mỗi yếu tố sẽ quyết định một mặt của mã kênh và giữa các thông số này có mối quan hệ ràng buộc chặt chẽ với nhau như đề cập trong hình 2. Việc thay đổi bất cứ yếu tố nào trong những yếu tố này sẽ dẫn đến sự thay đổi khả năng sửa lỗi của mã. Ví dụ, giảm tỉ lệ mã sẽ làm giảm tỉ lệ lỗi bit BER (Bit Error Rate), nhưng việc này cũng giảm thông lượng của hệ thống. Mặt khác, tăng độ dài của từ mã để đạt được tỉ lệ BER tốt hơn sẽ tăng độ trễ cho các quá trình mã hóa, giải mã, đồng thời làm tăng độ phức tạp tính toán và bộ nhớ đệm



Hình 2: Các yếu tố ảnh hưởng đến khả năng của mã kênh

của thiết bị mã và giải mã.

Bên cạnh đó việc xem xét chống lỗi trong các kênh truyền có tạp âm Gauss phân bố chuẩn AWGN (Additive White Gaussian Noise) là kênh phổ biến với đường truyền cáp; kênh Rayleigh là các kênh phổ biến với đường truyền vệ tinh, mặt đất với đặc trưng tín hiệu truyền tải dữ liệu bị can nhiễu, gây méo ở phía thu; kênh xóa Binary Erasure Channel (BEC) là mô hình mạng Internet, trong đó các gói dữ liệu có thể bị thất lạc, mất trong quá trình truyền dẫn. Gần đây xu hướng sử dụng các mã khối trong các mô hình tích hợp truyền hình số là rất phổ biến. Hai loại mã LDPC và Turbo được song song phát triển, trong đó mã LDPC có lợi thế không bị ảnh hưởng của hiện tượng sàn lỗi (Error Floor), hiện tượng này làm tỉ lệ lỗi bit phía đầu ra (BER) không thể giảm xuống giá trị bằng không mặc dù tỉ số E_b/N_0 được tăng lên tới vô cùng. Khi độ dài từ mã tăng lên thì độ phức tạp tính toán của mã Turbo cao hơn so với LDPC và do vậy yêu cầu thiết bị phải có bộ vi xử lý có cấu hình cao

để đáp ứng tốc độ tính toán trong thời gian thực, dẫn đến làm cho giá thành thiết bị tăng lên.

Do đó việc xây dựng, thiết kế mã LDPC có khả năng sửa lỗi tốt hơn cùng với việc đưa ra các mô hình tích hợp mã LDPC để tăng khả năng sửa lỗi của hệ thống mà vẫn đảm bảo độ phức tạp của hệ thống là vấn đề có tính khoa học, thực tiễn và cấp thiết. *Xuất phát từ các căn cứ nêu trên, trong luận án, nghiên cứu sinh đã lựa chọn hướng nghiên cứu về mã LDPC và xây dựng các hệ thống tích hợp mã LDPC nhằm góp phần bổ sung các giải pháp cho vấn đề mang tính thời sự này.*

Cho đến nay, đang có rất nhiều hướng nghiên cứu, xây dựng, phát triển cấu trúc ma trận kiểm tra của mã nhằm tăng cường khả năng chống lỗi của mã LDPC được phát triển bởi Mackay [7], Chen [8], Zhang [9], các nghiên cứu về tối ưu thuật toán giải mã của Narayanan [10], Fossorier [11] và các hệ thống tích hợp được nghiên cứu bởi Hanzo [12, 13] và nhiều nhà khoa học khác. Quá trình phân tích các công trình nghiên cứu khoa học đã công bố cùng các mô hình thử nghiệm, tác giả nhận thấy trong quá trình xây dựng, thiết kế mã LDPC và hệ thống tích hợp có ba vấn đề quan trọng cần giải quyết là:

- Làm thế nào để tối ưu thuật toán giải mã LDPC để tăng khả năng sửa lỗi của mã, hoặc giảm độ phức tạp của quá trình giải mã?
- Làm thế nào để xây dựng một mã LDPC có khả năng sửa lỗi tốt nhất, với độ phức tạp của quá trình mã hóa, giải mã có thể chấp nhận được?
- Làm thế nào để xây dựng, tối ưu những mô hình tích hợp mã LDPC có khả năng chống lỗi tốt nhất mà độ phức tạp của hệ thống có

thể chấp nhận được?

Nội dung luận án sẽ tập trung vào việc giải quyết hai bài toán thứ hai và thứ ba ứng dụng mô phỏng trong các kênh truyền dẫn có can nhiễu như AWGN, pha đình.

Mục đích nghiên cứu

- Nghiên cứu, xây dựng các ma trận sinh và ma trận kiểm tra của mã LDPC để tăng khả năng chống lỗi của mã;
- Nghiên cứu và đề xuất các mô hình tích hợp mã LDPC, giải quyết các bài toán về độ phức tạp và khả năng chống lỗi của hệ thống.

Đối tượng nghiên cứu

- Các kênh truyền dẫn có can nhiễu tạp âm phân bố AWGN, pha đình Rayleigh;
- Các hệ thống phân tập không gian, thời gian: V-BLAST, Alamouti;
- Hệ thống Internet sử dụng giao thức Internet ARQ và lai ghép H-ARQ;
- Các mô hình hệ thống tích hợp mã URC, LDPC, ánh xạ.

Phương pháp nghiên cứu

Sử dụng phương pháp nghiên cứu so sánh, mô phỏng, so sánh các kết quả thử nghiệm hoạt động của mã LDPC thu được trên các kênh

truyền dẫn bằng các chương trình mô phỏng viết trên ngôn ngữ C++.

BỐ CỤC LUẬN ÁN

Luận án được chia thành 3 chương chính với bố cục như sau:

Chương 1: MÃ SỬA SAI CÓ MA TRẬN KIỂM TRA MẬT ĐỘ THẤP LDPC- Nội dung của chương này đề cập đến các vấn đề về sự hình thành và phát triển của mã kênh, phân tích cấu tạo các mã kênh như mã chập, mã chập đệ quy, mã Turbo, mã LDPC, các phương pháp giải mã MAP, MAX-log MAP.

Chương 2: XÂY DỰNG MA TRẬN SINH VÀ MA TRẬN KIỂM TRA CỦA MÃ LDPC- Nội dung của chương này tập trung xây dựng mối quan hệ giữa ma trận sinh và ma trận kiểm tra, các hàm phân bố ngẫu nhiên cho hàng và cột của ma trận thành phần của ma trận kiểm tra để xây dựng một loại mã LDPC mới có khả năng sửa lỗi tốt hơn loại mã LDPC phổ thông. Kiểm định bằng các mô phỏng so sánh kết quả BER và E_b/N_0 giữa mã thiết kế với mã LDPC đều và không đều, đánh giá quan hệ giữa độ dài từ mã với độ tăng ích của mã được thiết kế. Nội dung của chương 2 liên quan đến công trình nghiên cứu số 1 đã được công bố.

Chương 3: XÂY DỰNG CÁC HỆ THỐNG TÍCH HỢP MÃ LDPC- Nội dung của chương này là xây dựng, phân tích các mô hình tích hợp mã LDPC đã đề xuất với hai hệ thống V-BLAST và H-ARQ, nhằm tăng khả năng chống nhiễu, tăng thông lượng của hệ thống thông tin. Mô phỏng, tính toán, phân tích, đánh giá, các kết quả thu được. Trên cơ sở đó so sánh mô hình hệ thống được đề xuất trong luận án với các mô hình hiện hành về khả năng sửa lỗi, thông lượng và độ phức tạp của hệ thống

thông qua các chương trình mô phỏng mối quan hệ BER và E_b/N_0 . Kết quả của mô hình V-BLAST- LDPC đề xuất đạt được độ tăng ích lên tới 5 dB so với các hệ thống V-BLAST tích hợp mã URC, trong khi độ phức tạp của hệ thống chỉ tăng khoảng 3 lần. Cũng trong chương này, hệ thống H-ARQ – LDPC được thiết kế có độ lợi cao hơn tới 4 dB so với hệ thống tích hợp mã LDPC không sử dụng cơ chế ARQ với cùng một điều kiện truyền dẫn. Nội dung của chương 3 liên quan đến công trình nghiên cứu số 2 và số 3 đã được công bố.

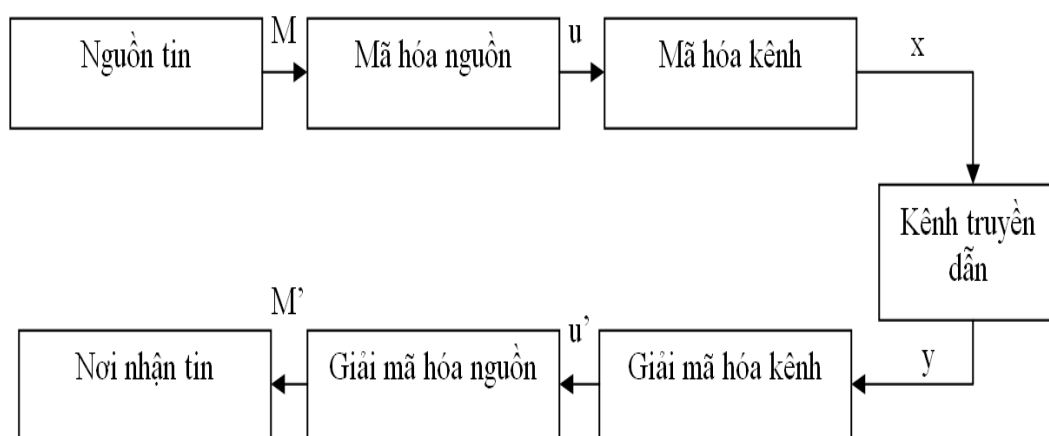
Chương 1

MÃ SỬA SAI CÓ MA TRẬN KIỂM TRA MẬT ĐỘ THẤP LDPC

1.1. Một số mã sửa sai thông dụng

Năm 1948 Claude E. Shannon đã phát hành những công trình nghiên cứu về lý thuyết toán học trong công nghệ truyền thông. Trong các công trình này Shannon phát triển các mô hình thuật toán cho phép giải quyết các vấn đề cơ bản trong truyền dẫn tín hiệu.

Nguồn tin: là nơi tạo ra tin M , với xác suất là f_M ($M = m$). Entropy



Hình 1.1: Mô hình hệ thống truyền tin số

của M được xác định như sau:

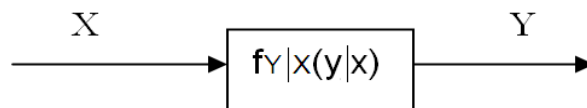
$$H(M) = - \sum_m f_M(m) \cdot \log_2 f_M(m) \quad (1.1)$$

Mã hóa Nguồn: Bộ mã hóa nguồn loại bỏ những thông tin dư thừa của chuỗi thông tin đầu vào.

Mã hóa kênh: Bộ mã hóa kênh ghép thêm thông tin dư thừa vào chuỗi dữ liệu đầu vào. Mục đích của việc ghép thêm thông tin dư thừa vào nhằm tăng khả năng tái tạo lại dữ liệu khi tín hiệu bị can nhiễu ở phía đầu thu.

Kênh: Hàm xác suất truyền dẫn của kênh được định nghĩa là $f_{Y/X}(Y/X)$ như trong mô hình thuật toán kênh 1.1. Trong đó kênh truyền dẫn là kênh không nhớ.

Có hai dòng mã sửa sai chính đó là mã chập và mã khối:



Hình 1.2: Mô hình toán học kênh truyền dẫn

- Mã chập thực hiện trên những dòng bit hoặc dòng symbol có độ dài tùy ý. Mã này thường được giải mã bằng thuật toán Viterbi, đôi khi một số thuật toán khác cũng được sử dụng. Thuật toán giải mã Viterbi cho phép hiệu quả giải mã tối ưu khi tăng độ dài từ mã, tuy nhiên độ phức tạp giải mã cũng tăng theo độ dài từ mã. Mã chập cũng có thể trở thành mã khối nếu được thiết kế bằng phương pháp "gắn bit đuôi".

- Mã khối thực hiện trên gói dữ liệu chứa một khối bit hay symbols có kích thước đã được xác định. Thời gian giải mã khối là hàm đa thức thời gian phụ thuộc vào kích thước khối của nó.

Mã chập đệ quy

Mã chập đệ quy là trường hợp mã chập thông thường có một đường hồi tiếp về đầu vào. Mã chập đệ quy (RSC) có các bit thông tin được thể hiện trong từ mã ở đầu ra, vì vậy mã RSC còn được gọi là mã hệ thống. Mặt khác có thể coi bộ mã RSC là một bộ lọc đáp ứng xung vô tận (IIR). Quá trình giải mã chập có thể thực hiện bằng phương pháp giải mã bit cứng và giải mã bit mềm dựa trên thuật toán Viterbi. Thuật toán giải mã bit mềm giải mã xác suất thông tin hậu nghiệm cực đại (MAP) mục đích là tìm xác suất $P[u(t) = 1|y]$ và xác suất $P[u(t) = 0|y]$ tại mỗi thời điểm t , đối với mỗi bit $u(t)$ nhận được ở phía thu, khi một bit y được truyền qua kênh có nhiễu.

Thuật toán MAP

Thuật toán này tuân tự thực hiện các bước sau:

- Dán nhãn các nhánh của giản đồ lưới bằng xác suất chuyển trạng thái $\gamma_{i,j}(t)$;
- Quét toàn bộ theo hướng thuận của giản đồ lưới để tính toán xác suất bắt đầu tại mỗi nút của giản đồ lưới $\alpha_i(t)$;
- Quét toàn bộ theo hướng ngược lại của giản đồ lưới để tính toán xác suất kết thúc tại mỗi nút của giản đồ lưới $\beta_i(t)$;

- Tính toán giá trị logarit tỉ số xác suất LLR của bit thông tin tại mỗi đoạn giản đồ lưới như sau:

$$\begin{aligned}\lambda(t) &= \frac{P[u(t) = 1|y]}{P[u(t) = 0|y]} \\ &= \log \frac{\sum_{S_i \rightarrow S_j: u=1} \alpha_i(t-1) \gamma_{i,j}(t) \beta_j(t)}{\sum_{S_i \rightarrow S_j: u=0} \alpha_i(t-1) \gamma_{i,j}(t) \beta_j(t)};\end{aligned}\quad (1.2)$$

- Thuật toán MAP còn được gọi là thuật toán "xuôi-ngược" (Forney).

Thuật toán MAP có thể được tính toán trong miền logarit bằng cách lấy logarit hai vế ta được:

$$\gamma_{i,j}(t) = \log P(S_i(t-1) \rightarrow S_j(t)|y(t)) \quad (1.3)$$

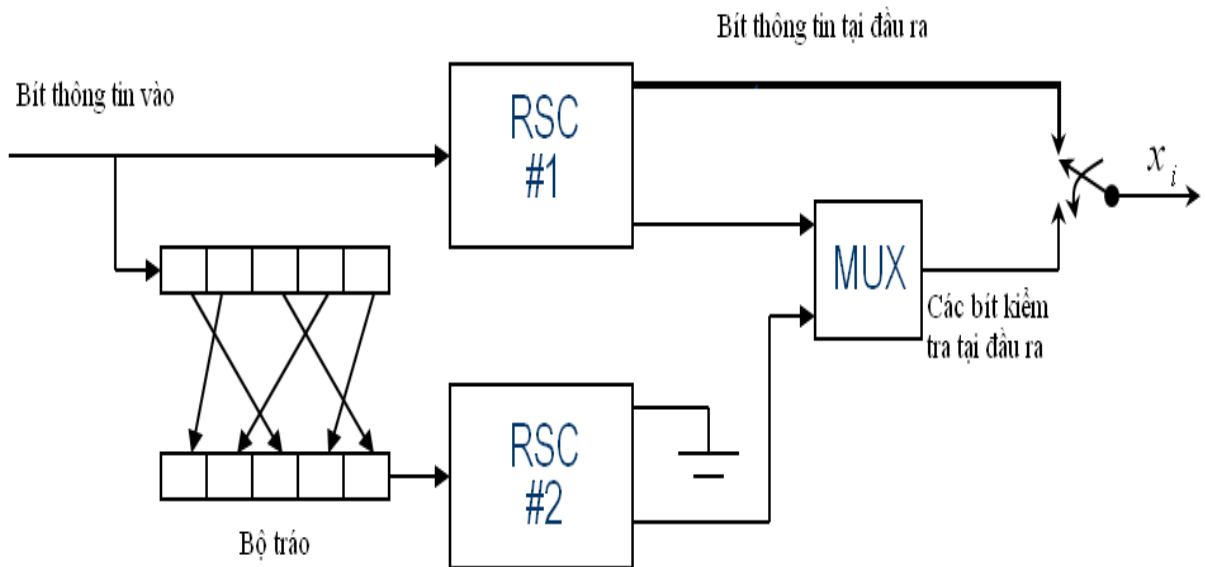
$$\alpha_i(t-1) = \log P(S_i(t-1)|y_1, y_2, \dots, y_{t-1}) \quad (1.4)$$

$$\beta_j(t) = \log P(S_j(t)|y_{t+1}, \dots, y_{L+m}) \quad (1.5)$$

Mã Turbo

Mã Turbo được cấu tạo bởi hai bộ mã hóa chấp đệ quy (RSC), tại đầu vào giữa hai bộ mã RSC sử dụng bộ tráo ngẫu nhiên các bit mã đầu vào. Quá trình giải mã Turbo là quá trình lặp giải mã giữa hai bộ mã hóa RSC, thông tin trích xuất tại đầu ra bộ mã 1 được chuyển đến đầu vào bộ mã hóa 2 sau khi đã đưa qua bộ giải tráo (ngược với quá trình tráo ở phía bộ mã hóa) và ngược lại. Bộ mã hóa Turbo có những ưu, nhược điểm sau:

- Bộ mã hóa Turbo có khả năng sửa lỗi tốt với mức tỉ số tín hiệu trên tạp nhiễu S/N rất thấp. Khả năng giải mã của bộ mã Turbo



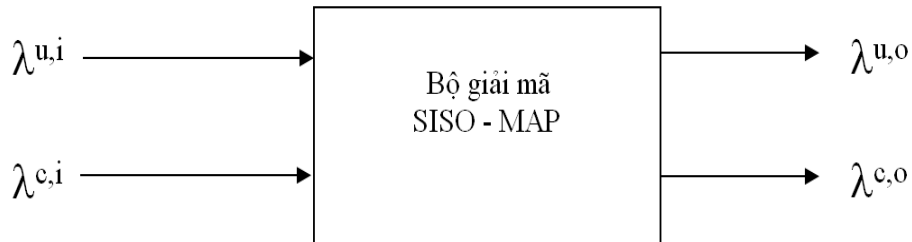
Hình 1.3: Sơ đồ bộ mã Turbo

có thể tiến sát đến giới hạn Shannon là do có rất nhiều từ mã đầu ra bộ mã Turbo có hàm trọng nhỏ.

- Tuy nhiên cũng vì có nhiều từ mã hàm trọng nhỏ, mã Turbo có hiện tượng sàn lỗi (tại dải giá trị rất nhỏ BER không tăng khi tăng S/N trong một khoảng nhất định).
- Khả năng sửa lỗi của mã Turbo tăng theo độ dài khối bit mã hóa và hiện tượng sàn lỗi sẽ suy giảm khi kích thước khung của bộ tráo tăng lên. Tuy tăng kích thước khối bit mã hóa không gây thêm độ phức tạp của bộ giải mã, nhưng sẽ gây ra hiện tượng trễ trong quá trình giải mã, điều này sẽ đòi hỏi bộ nhớ đệm của bộ giải mã phải tăng theo.
- Độ phức tạp của mã Turbo K_{TC} và bộ mã RSC K_{CC} có cùng độ dài ràng buộc L được tính như sau:

$$L_{TC} \approx 2 + L_{CC} + \log(\text{Số lần lặp giải mã})$$

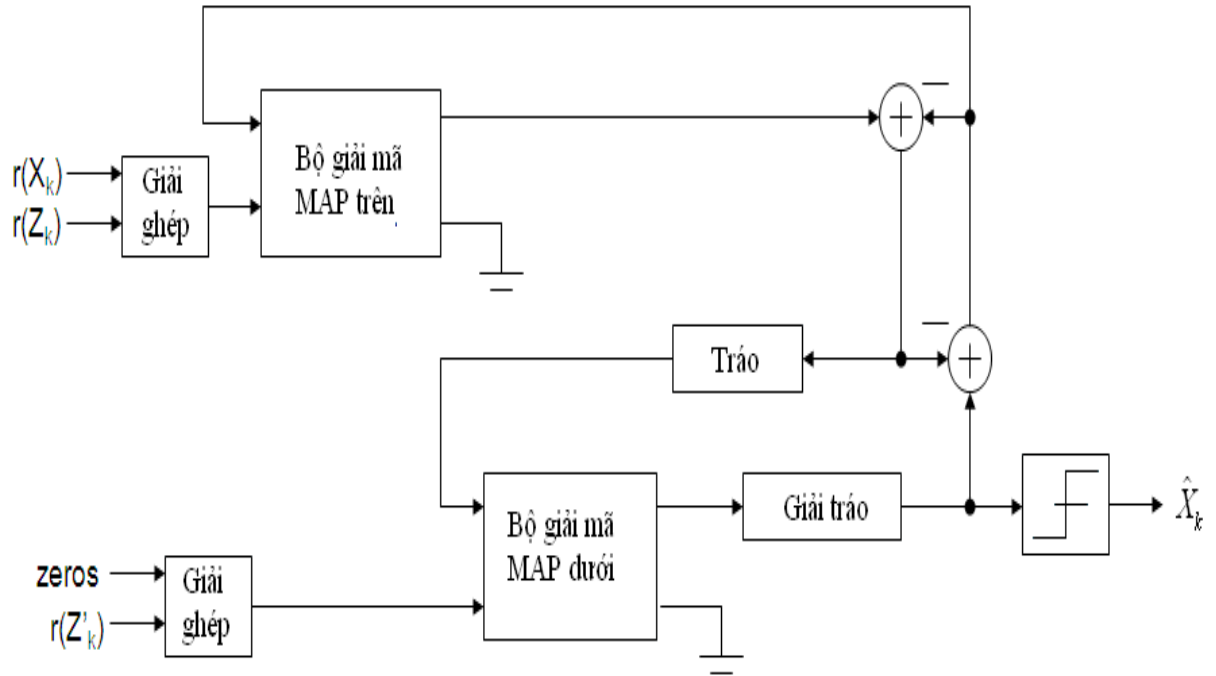
Quá trình giải mã của bộ mã Turbo được thực hiện theo thuật toán giải mã lặp các giá trị của bit mềm vào ra giữa hai bộ mã RSC (SISO-MAP) như hình dưới đây.



Hình 1.4: Thuật toán giải mã SISO-MAP

- Đầu vào bộ giải mã là:
 - Các tỉ số xác suất $LLR \lambda^{u,i}$ của các bit thông tin đến từ bộ giải mã thứ hai.
 - Các tỉ số xác suất $LLR \lambda^{c,i}$ của các bit mã đến từ kênh truyền.
- Đầu ra bộ giải mã là:
 - Các tỉ số xác suất $LLR \lambda^{u,o}$ của các bit thông tin truyền qua bộ giải mã thứ hai.
 - Các tỉ số xác suất $LLR \lambda^{c,o}$ của các bit mã không được bộ giải mã thứ 2 sử dụng .

Cấu trúc giải mã Turbo được mô tả trong hình 1.5.



Hình 1.5: Cấu trúc giải mã Turbo

Các xác suất đầu vào $\lambda^{u,i}$ bộ mã hóa trên ban đầu được gán giá trị "0". Bộ giải mã trên sẽ thực hiện giải mã trước sau đó bộ giải mã dưới sẽ thực hiện giải mã dựa vào các thông tin từ kênh truyền và từ đầu ra bộ giải mã trên.

Các bộ mã hóa Turbo gần đây được sử dụng trong các hệ thống thông tin vệ tinh [14,15], trong hệ thống Wimax [16], hay hệ thống viễn thông phân chia truy cập theo mã CDMA2000 [17].

Mã có ma trận kiểm tra mật độ thấp LDPC

Năm 1963, Gallager [6, 18] phát minh họ mã có mật độ ma trận kiểm tra thấp LDPC (Low Density Parity Check) trong quá trình làm

luận văn tiến sĩ của mình tại học viện MIT. Tại thời điểm lúc đó, mã LDPC có rất ít ảnh hưởng đến hội đồng nghiên cứu mã kênh truyền dẫn, do khả năng sửa lỗi của họ mã này kém hơn nhiều so với mã Turbo [5], trong khi đó độ phức tạp của mã cao. Vì vậy, trong hơn một thập kỉ tiếp theo họ mã này không được tiếp tục phát triển. Độ phức tạp của mã LDPC được đánh giá bởi Zyablov và Pinsker năm 1975 [19], trong khi đó Tanner [20] đưa ra phương pháp thiết kế cấu trúc lặp cho ma trận kiểm tra (Parity Check Matrix) và biểu diễn ma trận này dưới dạng đồ thị. Sipser và Spielman [21] giới thiệu ma trận kiểm tra sử dụng các đồ thị mở rộng.

Tuy nhiên, trong những năm của thập kỉ 1990, sự quan tâm của hội đồng nghiên cứu mã kênh truyền đối với mã LDPC được khôi phục lại. Như tổng kết trong bảng 1.1, các mã LDPC đã trở thành đề tài nóng bỏng đối với hội đồng nghiên cứu mã kênh truyền dẫn. Mackay và Neal [22, 23] thực nghiệm với mã LDPC có từ mã lớn và đã chứng minh rằng các mã LDPC có khả năng sửa lỗi cao hơn so với các mã turbo, khi truyền dẫn qua các kênh truyền có phân bố tạp âm trắng kiểu Gauss AWGN (Additive White Gaussian Noise). Bị thu hút bởi khả năng sửa lỗi của mã LDPC, họ đã tiếp tục nghiên cứu nhiều thuộc tính khác của họ mã LDPC. Thuật toán tiến triển hàm mật độ DE (Density Evolution) được phát minh bởi Richardson và các đồng nghiệp cho phép tính toán tiệm cận hiệu quả của các mã LDPC được sử dụng trong sửa lỗi dữ liệu qua kênh truyền AWGN.

Sau đó Chung và các đồng nghiệp của mình [30, 41] đã đơn giản hóa thuật toán DE. Thuật toán DE ngày nay được sử dụng rộng rãi để dự đoán tiệm cận hiệu quả sử dụng các mã LDPC trong kênh AWGN và nó được sử dụng bởi nhiều nhà nghiên cứu như Fossorier, Chen và các

Bảng 1.1: Các mốc phát triển chính trong nghiên cứu mã LDPC

1948	Giới hạn của Shannon [24];
1962	Các mã LDPC được phát minh bởi Gallager [18];
1975	Tính toán độ phức tạp của các mã LDPC bởi Zhang [19] ;
1983	Đồ thị Tanner cho các mã LDPC của Tanner [20];
1997	Hiệu quả của mã LDPC gần giới hạn Shannon, được thực hiện bởi Mackay [22];
1998	Mã LDPC bất nhị phân được phát minh bởi Davey [25];
	Mã LDPC không đều được thiết kế bởi Luby [26];
	Richardson phát minh thuật toán giải mã dựa trên thuật toán FFT nhằm giảm độ tính toán phức tạp của quá trình giải mã LDPC;
	Thuật toán tiến triển hàm mật độ được phát minh bởi Richardson;
1999	Lentmaier phát minh mã LDPC tổng hợp G-LDPC [27];
2001	Kou và đồng nghiệp thiết kế ma trận kiểm tra của LDPC có dạng hình học hữu hạn [28];
	Giải mã LDPC bằng phương pháp đảo bit được thực hiện bởi Kou ;
	Ten Brink phát minh đồ thị EXIT [29];
	Chung phát minh phương pháp tiến triển mật độ xấp xỉ hàm Gauss [30];

2002	Mã LDPC dựa trên thuật toán BIBD được thực hiện bởi Ammar [31];
	Phương pháp giải mã bằng thuật toán giải mã Bootstrap được phát minh bởi Nouth [32];
	Hệ thống LDPC-MIMO ba lớp nhị phân được thiết kế bởi Meshkat [33];
2004	Zhang cải tiến thuật toán giải mã dựa trên phương pháp đảo bit [34];
	Guosen thiết kế hệ thống LDPC-MIMO ba lớp Symbol [35];
	Guo cải tiến thuật toán giải mã bằng phương pháp đảo bit dựa trên tỉ số độ tin cậy [36];
2010	Huang xây dựng các ma trận thành phần của ma trận kiểm tra mã LDPC bất nhị phân nhằm tăng kích thước các vòng trao đổi thông tin giữa các nút kiểm tra và nút biến [37]; Ahn thiết kế mô hình tích hợp mã LDPC và bộ điều chế không đều [38].
2012	Wang phát triển mã LDPC bất nhị phân với quá trình xử lý hiệu quả các nút kiểm tra [39]; Arabaci xây dựng mô hình tích hợp mã LDPC bất nhị phân với bộ điều chế [40];

đồng nghiệp [42, 43], Narayanaswani [44], Anastasopoulos [45] và Kumar [46]. Biên giới hạn hiệu quả của mã LDPC được nghiên cứu bởi Burshtein và các đồng nghiệp trong [47, 48]. Họ mã LDPC được sử dụng trong rất nhiều mô hình hệ thống khác nhau như các mô hình hệ thống OFDM [12, 49], hệ thống mã hóa MIMO và phân tập theo thời gian-không gian (MIMO Space-Time coding System) [50], trong các mô hình kênh xóa nhị phân BEC (Binary Erasure Channel) [51, 52], trong các mô hình kênh đáp ứng cục bộ [53, 54] và rất nhiều mô hình khác [55, 56]. Hiệu suất sử dụng băng tần của các mô hình điều chế có mã hóa sử dụng các mã LDPC được nghiên cứu trong [57–59]. Hơn nữa, các mã LDPC cũng được sử dụng trong nhiều ứng dụng khác như ghi từ [60–62] được nghiên cứu bởi Song và các đồng nghiệp, trong truyền dẫn ảnh và trong lưu trữ bởi Zhang [63]. Rất nhiều các thuật toán được phát minh cho thiết kế mã LDPC đối với các ứng dụng phần cứng, ví dụ như các kỹ thuật được ủng hộ bởi Zhang [64, 65], Rupp [66], Hocevar [67], Thorpe [68], Lu [69] và Shanbhag [70]. Bên cạnh những nghiên cứu về đánh giá hiệu quả và các khía cạnh ứng dụng của mã LDPC, các nhà nghiên cứu cũng cố gắng tăng cường hiệu quả của mã LDPC bằng cách cải tiến các thuật toán và tối ưu cấu trúc ma trận kiểm tra. Các mã LDPC bất nhị phân được phát minh bởi Davey [25, 71, 72], trong một số trường hợp có hiệu quả cao hơn so với các mã LDPC nhị phân. Thuật toán có độ phức tạp thấp được sử dụng cho các mã LDPC bất nhị phân được phát minh bởi Barnault [73]. Họ mã LDPC bất nhị phân cũng được áp dụng bởi Song [60], Nakamura [74] và Li [75]. Bằng cách thiết kế cấu trúc ma trận mật độ phân bố không đồng đều cho các mã LDPC [7, 26, 76–79], hiệu quả sử dụng mã LDPC có thể đạt tới giới hạn Shannon [80]. Các phương pháp tạo cấu trúc ma trận khác nhau của mã LDPC có hàm phân bố mật

độ không đồng đều được thực hiện trong [78, 79, 81, 82]. Tăng độ dài các chu kỳ ngắn nhất trong ma trận kiểm tra của mã LDPC cũng là một kỹ thuật tăng hiệu quả của mã LDPC và làm suy giảm hiện tượng sàn lỗi. Moura [83–86], Lin [8] và Williamson [8] thiết kế nhiều cấu trúc ma trận kiểm tra nhằm mục đích loại các chu kỳ ngắn trong ma trận kiểm tra. Lentmaier phát minh mã LDPC tổng hợp (G-LDPC) [87], bằng cách thay các hàng trong ma trận kiểm tra bằng mã Hamming. Kỹ thuật này thu hút được sự chú ý của Zhang [88, 89], Hirst [90, 91] và Boutros [92]. Bên cạnh việc tăng khả năng sửa lỗi của các mã LDPC, các nhà nghiên cứu khác cũng cố gắng làm giảm độ phức tạp của các mã LDPC. Mackay và Neal [22] chứng minh rằng bằng cách thiết kế ngẫu nhiên cấu trúc ma trận kiểm tra của mã LDPC, hiệu quả sửa lỗi của mã LDPC cũng có thể đạt được gần so với giới hạn dung lượng kênh truyền. Khả năng sửa lỗi của các mã LDPC có cấu trúc ma trận kiểm tra theo các luật khác nhau là như nhau nếu sử dụng quá trình mã hóa có độ phức tạp giảm bằng cách sử dụng các thanh ghi dịch. Ví dụ như phép tiếp cận phân tích định dạng hình học hữu hạn của Kou [93–97] cho cấu trúc ma trận kiểm tra của các mã LDPC. Kỹ thuật này cũng được sử dụng bởi Pados [98] và Vasic [99, 100]. Honary [31, 101] sử dụng kỹ thuật thiết kế không toàn khối cân bằng BIBD (Balanced Incomplete Block Design) cho cấu trúc ma trận LDPC theo phương pháp cân bằng các chu kỳ. Một loạt các nghiên cứu về cấu trúc ma trận kiểm tra của mã LDPC được thực hiện bởi Vontobel [102], Ahn [103] và Okamura [104]. Tất cả các mô hình này đều đạt được hiệu quả tốt như các mã LDPC có cấu trúc ma trận kiểm tra ngẫu nhiên, trong khi độ phức tạp của mã thấp hơn nhiều. Thuật toán giải mã dựa trên thuật toán FFT được phát minh bởi Richardson và Urbanke [105, 106] và thuật toán giải mã tuyến tính

của Spielman [107], được thực hiện nhằm mục đích làm giảm độ phức tạp của các bộ giải mã LDPC. Hơn nữa, có rất nhiều các thuật toán khác nhằm mục đích suy giảm độ phức tạp của mã LDPC được phát minh bởi Pothier [108], Narayanan [10] và Fossorier [109].

Ngoài thuật toán cận tối ưu Tổng – Tích SPA (Sum-Product Algorithm) được phát minh cho các mã LDPC còn có thuật toán giải mã dựa trên phương pháp đảo bit có độ phức tạp thấp được thiết kế bởi Kou và các đồng nghiệp [93]. Thuật toán này được tiếp tục phát triển bởi Zhang [109]. Thuật toán giải mã đảo bit Bootstrap được thực hiện bởi Nouth [110]. Thuật toán giải mã dựa trên tỉ số độ tin cậy được phát triển bởi Guo [36]. Tất cả các thuật toán giải mã có độ phức tạp thấp được hình thành để hỗ trợ cho thuật toán SPA trong các ứng dụng có độ phức tạp của hệ thống bị giới hạn nghiêm ngặt.

So sánh với mã Turbo cũng đang được phát triển song song, mã LDPC có lợi thế không bị ảnh hưởng của hiện tượng sàn lỗi (Error Floor), hiện tượng này làm tỉ lệ lỗi bit phía đầu ra (BER) không thể giảm xuống giá trị cực nhỏ mặc dù tỉ số E_b/N_0 được tăng lên khá nhiều. Độ phức tạp tính toán của mã Turbo cao hơn so với LDPC khi độ dài từ mã tăng lên. Từ việc phân tích lịch sử phát triển của các mã LDPC và so sánh với các dòng mã tiên tiến khác như Turbo, ngoài ra xem xét độ phức tạp của mã LDPC không lớn nên có thể áp dụng cho các thiết bị truyền hình dân dụng như các đầu thu số vệ tinh tiêu chuẩn DVB-S2 (Digital Video Broadcasting- Satellite 2).

Từ nghiên cứu tổng quan và phân tích như đã trình bày ở trên, có thể nhận thấy việc nghiên cứu, phát triển mã LDPC là một đề tài có tính thời sự, cần thiết và cấp bách cho ứng dụng trong truyền hình số.

1.2. Tổng quan mã LDPC

Mã LDPC thuộc họ mã khối [6, 18], quan hệ giữa ma trận sinh \mathbf{G} và ma trận kiểm tra \mathbf{H} được biểu diễn bằng phương trình sau:

$$\mathbf{H}_{M \times N} \cdot \mathbf{G}_{N \times K}^T = \mathbf{0}_{M \times K}, \quad (1.6)$$

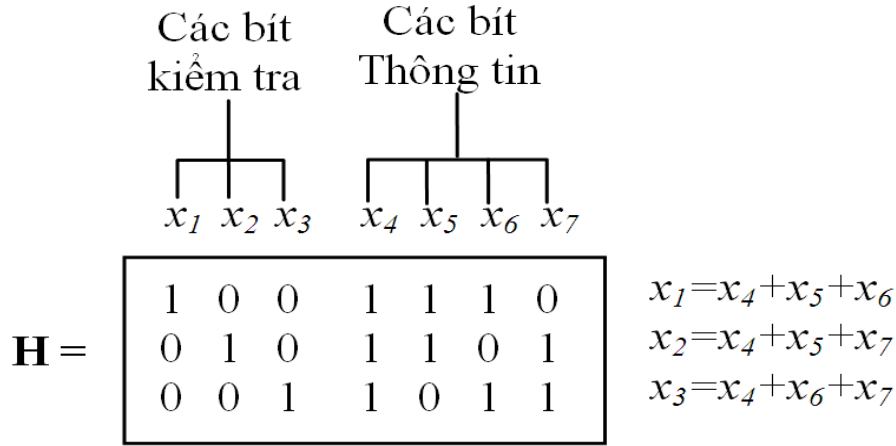
trong đó N là số bit mã, K là số bit thông tin và $M = (N - K)$ là số bit kiểm tra trong một từ mã. Giả sử một chuỗi bit thông tin $S_{1 \times K}$ ở đầu vào bộ mã hóa LDPC, có kích thước là K bit. Ở đầu ra bộ mã hóa LDPC nhận được một từ mã $\mathbf{C}_{1 \times N}$, có độ dài từ mã là N bit mã. Quá trình mã hóa chuỗi bit thông tin đầu vào $\mathbf{S}_{1 \times K}$ được thực hiện bằng cách nhân véc tơ chuỗi bit này với ma trận sinh $\mathbf{G}_{K \times N}$ của bộ mã LDPC. Quá trình này được tiến hành như sau:

$$\mathbf{C}_{1 \times N} = \mathbf{S}_{1 \times K} \cdot \mathbf{G}_{K \times N}. \quad (1.7)$$

Tính hợp lệ của một từ mã được kiểm tra bằng phương trình kiểm tra từ mã sau [18]:

$$\text{Syndrome}_{1 \times M} = \mathbf{C}_{1 \times N} \cdot \mathbf{H}_{N \times M}^T. \quad (1.8)$$

Nếu từ mã này là từ mã hợp lệ, thì véc tơ *Syndrome* là một véc tơ $\mathbf{0}$. Trường hợp từ mã không hợp lệ véc tơ *Syndrome* là một véc tơ khác $\mathbf{0}$. Ma trận kiểm tra \mathbf{H} của mã LDPC bao gồm các phần tử số nhị phân ‘0’, ‘1’ để kiểm tra phần thông tin của từ mã. Hình 1.6 minh họa ma trận kiểm tra của mã LDPC. Ma trận kiểm tra của mã LDPC có thể được biểu diễn bằng các thông số sau (N, wc, wr) , trong đó N là độ dài từ mã, wc , wr là các hàm trọng Hamming trung bình của các cột và hàng tương ứng trong ma trận kiểm tra. Nếu ma trận kiểm tra của mã



Hình 1.6: Ma trận kiểm tra của mã LDPC

LDPC có hàng đầy đủ¹, tỉ lệ mã r được xác định như sau:

$$r = \frac{K}{N}. \quad (1.9)$$

Quan hệ giữa số bit kiểm tra và độ dài từ mã được thể hiện như sau:

$$M.w_r = N.w_c. \quad (1.10)$$

Vì vậy tỉ lệ mã r cũng có thể tính như sau:

$$r = \frac{K}{N} = \frac{N - M}{N} = 1 - \frac{w_c}{w_r}. \quad (1.11)$$

Hình 1.7 là ví dụ cụ thể về một ma trận kiểm tra của mã LDPC. Quá trình tính toán ma trận sinh từ ma trận kiểm tra được tiến hành như sau. Giả sử:

$$\mathbf{C}_{1 \times N} = \mathbf{S}_{1 \times K} \cdot \mathbf{G}_{K \times N}, \quad (1.12)$$

Là từ một từ mã LDPC chứa véc tơ chuỗi bit thông tin \mathbf{S} ở phần cuối của từ mã này và phần đầu từ mã là véc tơ \mathbf{P} chứa các bit kiểm

¹Ma trận kiểm tra của mã LDPC được gọi là ma trận hàng đầy đủ, khi các hàng của ma trận này là độc lập tuyến tính với nhau

tra. Từ mã \mathbf{C} có thể viết lại dưới dạng sau:

$$\mathbf{C}_{1 \times N} = (\mathbf{P}_{1 \times M} | \mathbf{S}_{1 \times K}). \quad (1.13)$$

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	0	1	1	1	0	0	0	1	0	0	1	0	0	0	0
2	0	0	0	1	1	0	1	1	0	1	0	0	0	0	0
3	1	0	0	0	0	0	1	0	0	0	0	1	0	1	1
4	1	1	0	0	0	1	0	0	1	1	0	0	0	0	0
5	1	0	0	0	0	1	0	0	1	0	0	0	1	1	0
6	0	0	0	0	1	0	0	0	1	0	1	0	1	0	0
7	0	0	0	0	0	0	1	0	0	1	0	1	0	0	1
8	0	0	0	0	0	1	0	0	0	0	0	0	1	1	1
9	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0
10	0	0	1	0	0	0	0	1	0	0	1	1	0	0	0

Hình 1.7: Ma trận kiểm tra \mathbf{H} có $N = 15$, $w_c = 3$, $w_r = 4, 5$, $M = N - K = 10$, $r = 1/3$.

Ma trận kiểm tra $\mathbf{H}_{M \times N}$ có thể viết lại dưới dạng hai ma trận liên kề $(\mathbf{A}_{M \times M} | \mathbf{B}_{M \times K})$, trong đó ma trận thành phần \mathbf{A} là ma trận độc lập tuyến tính. Từ phương trình kiểm tra tính hợp lệ của từ mã (1.6), ta có thể viết lại như sau:

$$\mathbf{C} \cdot \mathbf{H}^T = \mathbf{C} \cdot (\mathbf{A} | \mathbf{B})^T = \mathbf{P} \cdot \mathbf{A}^T + \mathbf{S} \cdot \mathbf{B}^T = 0. \quad (1.14)$$

Ma trận \mathbf{A} là độc lập tuyến tính, cho nên ta có thể tính được ma trận đảo $(\mathbf{A}^T)^{-1}$. Từ phương trình (1.14) suy ra :

$$\mathbf{P} = \mathbf{S} \cdot (\mathbf{B}^T \cdot (\mathbf{A}^T)^{-1}). \quad (1.15)$$

Từ phương trình (1.15), ta có thể suy ra ma trận sinh của mã LDPC như sau:

$$\mathbf{G}_{K \times N} = [(\mathbf{B}^T \cdot (\mathbf{A}^T)^{-1})_{K \times M} | \mathbf{I}_{K \times K}]. \quad (1.16)$$

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	1	1	0	1	0	0	0	0	1	0	1	0	0	0	0
2	0	1	0	0	0	1	1	0	1	1	0	0	0	0	0
3	0	0	1	0	0	0	1	0	0	0	0	1	0	1	1
4	1	0	1	0	1	0	0	1	0	1	0	0	0	0	0
5	0	0	1	0	1	0	0	1	0	0	0	0	1	1	0
6	0	0	0	0	0	1	0	1	0	0	1	0	1	0	0
7	0	0	0	0	0	0	1	0	0	1	0	1	0	0	1
8	0	0	0	0	1	0	0	0	0	0	0	0	1	1	1
9	1	1	0	1	0	1	0	0	0	0	0	0	0	0	0
10	0	0	0	1	0	0	0	0	1	0	1	1	0	0	0

Hình 1.8: Ma trận chuyển vị \mathbf{H}_r từ ma trận kiểm tra \mathbf{H} trong hình 4. Ma trận \mathbf{H}_r bao gồm hai ma trận thành phần \mathbf{A} và \mathbf{B} .

Tuy nhiên để tìm được một ma trận \mathbf{A} có thể nghịch đảo được, tồn tại trong ma trận kiểm tra \mathbf{H} , ta phải tiến hành hoán vị các cột của ma trận \mathbf{H} và tiến hành kiểm tra tính độc lập tuyến tính của ma trận \mathbf{A} bằng phương pháp toán học Gauss. Sau khi thực hiện hoán vị các cột của ma trận \mathbf{H} được ma trận \mathbf{A} là độc lập tuyến tính, ma trận \mathbf{H} trở thành ma trận kiểm tra mới \mathbf{H}_r .

Như vậy ma trận sinh của mã LDPC là ma trận sinh \mathbf{G} được tính từ ma trận hoán vị \mathbf{H}_r và phương trình kiểm tra từ mã hợp lệ phải dựa trên hai ma trận mới này. Thông thường việc tính toán ma trận sinh \mathbf{G}

của mã LDPC từ ma trận kiểm tra \mathbf{H}_r là phức tạp, vì cần kiểm tra tính độc lập tuyến tính của các cột nhiều lần (sau mỗi lần hoán vị) và chiếm nhiều phép tính cho quá trình tính toán ma trận đảo. Ma trận chuyển vị \mathbf{H}_r của ma trận kiểm tra \mathbf{H} được cho trong hình 1.8. Tích của hai ma trận $\mathbf{B}^T \cdot (\mathbf{A}^T)^{-1}$ được sử dụng để tính ma trận sinh của mã LDPC, được cho trong hình 1.9.

$$\mathbf{B}^T \cdot (\mathbf{A}^T)^{-1} =$$

	1	2	3	4	5	6	7	8	9	10
1	1	1	1	0	0	0	1	1	1	1
2	0	1	0	1	0	0	1	0	0	0
3	0	0	1	0	1	0	1	1	0	1
4	0	0	0	0	1	0	1	0	0	1
5	1	1	1	0	1	0	0	0	0	1

Hình 1.9: Tích hai ma trận thành phần trong hình 1.8 được sử dụng để tính ma trận sinh.

$$\mathbf{G} =$$

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	1	1	1	0	0	0	1	1	1	1	1	0	0	0	0
2	0	1	0	1	0	0	1	0	0	0	0	1	0	0	0
3	0	0	1	0	1	0	1	1	0	1	0	0	1	0	0
4	0	0	0	0	1	0	1	0	0	1	0	0	0	1	0
5	1	1	1	0	1	0	0	0	0	1	0	0	0	0	1

Hình 1.10: Ma trận sinh \mathbf{G} của mã LDPC được tính từ ma trận kiểm tra \mathbf{H}_r .

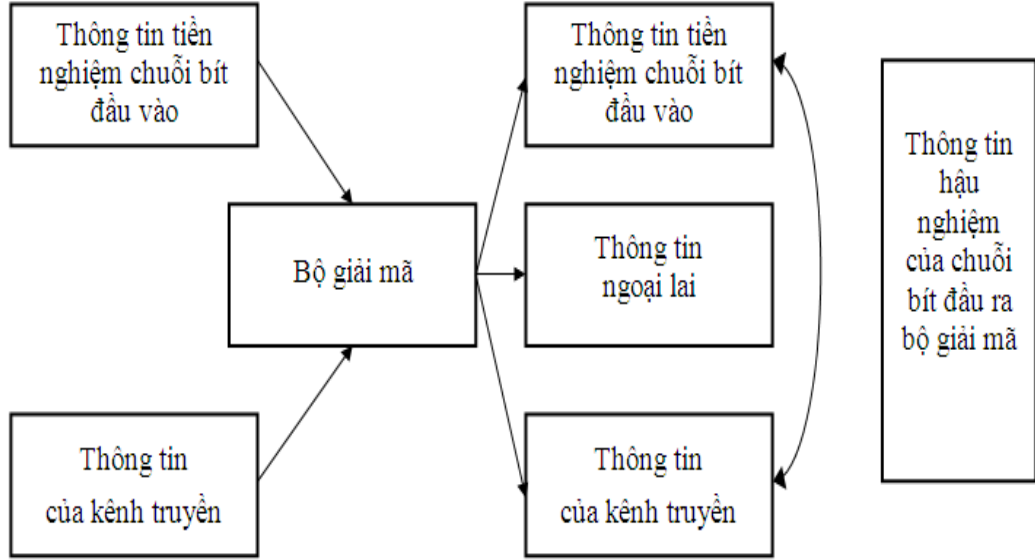
1.3. Các phương pháp giải mã LDPC

Có hai phương pháp giải mã của LDPC đó là: giải mã theo xác suất (Probability Decoding) hay còn gọi là giải mã bằng thuật toán truyền bá độ tin cậy BPA (Belief Propagation Algorithm) hoặc giải mã bằng thuật toán tích-tổng (Sum-Product Algorithm). Thuật toán này được đề xuất trong các tài liệu nghiên cứu của Gallager [6, 18]. Phương pháp giải mã thứ hai dựa trên thuật toán trao đổi thông tin MPA (Message Passing Algorithm) giữa các nút kiểm tra (Check Nodes) và nút biến số (Variable Nodes), trong ma trận kiểm tra của mã LDPC [111]. Các thuật toán giải mã này ước lượng thông tin hậu nghiệm đầu ra bộ giải mã dựa trên các thông tin tiên nghiệm của chuỗi bit mã thu được, phần thông tin của chuỗi bit được tạo ra từ bên trong bộ giải mã và thông tin của kênh truyền dẫn (Channel Information)². Mô hình ước lượng thông tin các bit đầu ra bộ giải mã được cho trong hình 1.11.

1.3.1. Phương pháp giải mã dựa theo xác suất

Phương pháp giải mã dựa theo xác suất của Gallager được thực hiện như sau. Có hai yếu tố cần cân nhắc trong quá trình giải mã chuỗi bit nhận được. Thứ nhất, bit thông tin thu được bị sai lệch do ảnh hưởng của kênh truyền như thế nào. Thứ hai là các bit dư thừa được dùng để kiểm tra bit thông tin này có thể được coi là hoàn toàn không lỗi hay không? Từ đó ta có thể tính toán xác suất có điều kiện của symbol thu được là '0' hay '1' dựa trên các giá trị bit mềm (Soft output) y bị ảnh

²Thông tin hậu nghiệm của chuỗi bit đầu ra bộ giải mã là tổng thông tin của kênh truyền, thông tin tiên nghiệm của chuỗi bit và thông tin của chuỗi bit ngoại lai được tính toán bên trong bộ giải mã.



Hình 1.11: Thông tin ngoại lai được bộ giải mã tạo ra từ thông tin tiên nghiệm của chuỗi bit đầu vào và thông tin kênh truyền.

hưởng của kênh truyền, tại đầu ra của bộ điều chế và điều kiện sự kiện S khi bit thông tin này thỏa mãn tất cả các phương trình kiểm tra chẵn lẻ w_c chứa nó. Nói một cách khác, xác suất có điều kiện này được viết lại như sau:

$$P[x_j = 1|y, S]. \quad (1.17)$$

Đặt P_j^1 là xác suất giá trị nhị phân ‘1’ được truyền tại vị trí $j = (1, \dots, N)$, dựa trên giá trị bit mềm tại đầu ra bộ giải điều chế và $P_{i,j}^1$ là xác suất giá trị nhị phân ‘1’ tại vị trí bit thứ $j = (1, \dots, N)$ và $i = (1, \dots, M)$, trong ma trận kiểm tra của mã LDPC. Gallager đã đưa ra công thức tính toán tỉ số xác suất giữa giá trị nhị phân ‘1’ và ‘0’ được truyền tại các vị trí bit $j = (1, \dots, N)$ như sau:

$$\frac{P[x_j = 0|y, S]}{P[x_j = 1|y, S]} = \frac{1 - P_j^1}{P_j^1} \prod_{i \in \{R_j\}} \left[\frac{1 + \prod_{l \in \{C_j\}, l \neq j} (1 - 2P_{i,l}^1)}{1 - \prod_{l \in \{C_j\}, l \neq j} (1 - 2P_{i,l}^1)} \right] \quad (1.18)$$

trong đó C_i là tập các chỉ số cột của các thành phần khác '0' trong hàng thứ i và R_j là tập các chỉ số hàng của các thành phần khác '0' trong cột thứ j của ma trận kiểm tra chuyển vị như trong hình 1.8. Ví dụ, trong hình 1.8 ta có tập các cột $C_2 = 2, 6, 7, 9, 10$, và tập các hàng $R_{12} = 3, 7, 10$. Hơn nữa, tập các hàng R_j cho biết các chỉ số hàng của các bit thứ j như trong hình 1.8. Tỷ số xác suất thông tin hậu nghiệm đầu ra của bit thứ j với điều kiện thông tin bit mềm đầu ra kênh truyền y và thỏa mãn các phương trình kiểm tra từ mã chứa bit thứ j . Xác suất này được tính dựa trên thông tin nội tại của bit thứ j lấy được từ thông tin mềm y_j đầu ra của kênh truyền và xác suất của các thành phần khác '0' kí hiệu là $P_{i,l}$ trong phương trình (1.18).

Tỷ số hợp lệ được xác định bằng công thức sau:

$$LR_{ij} = \frac{1 + \prod_{l \in \{C_j\}, l \neq j} (1 - 2P_{i,l}^1)}{1 - \prod_{l \in \{C_j\}, l \neq j} (1 - 2P_{i,l}^1)}; \text{ với } i = 1, \dots, M; j = 1, \dots, N \quad (1.19)$$

và tỷ số xác suất PR được tính như sau:

$$PR_{i,j} = \frac{1 - P_j^1}{P_j^1} \prod_{k \in \{R_i\}, k \neq j} LR_{k,j}; \quad (1.20)$$

và:

$$PR(x_i) = \frac{1 - P_j^1}{P_j^1} \prod_{i \in \{R_i\}} LR_{k,j}; \text{ với } j = 1, \dots, N. \quad (1.21)$$

Phương trình (1.21) là phương trình cập nhật thông tin tỷ số xác suất PR của các thành phần trong ma trận kiểm tra \mathbf{H} . Tỷ số hợp lệ $LR_{i,j}$ trong phương trình (1.19), với $i = 1, \dots, M; j = 1, \dots, N$ là tỷ số xác suất có điều kiện giữa xác suất bit mã thứ j có giá trị là '0' và xác suất nó có giá trị là '1' và phụ thuộc vào tích của các xác suất thông tin mềm $P_{i,l}$ trong phương trình (1.19) của các thành phần khác '0' trong

hàng thứ i của ma trận kiểm tra \mathbf{H} và thành phần có chỉ số j hiện đang xét như trong phương trình (1.19). Tỉ số xác suất $PR_{i,j}$ trong phương trình (1.20) cũng tương tự như tỉ số $LR_{i,j}$ trong phương trình (1.19), chỉ khác là thông tin mềm cập nhật cho tỉ số này là của tất cả các thành phần khác '0' trong cột thứ j của ma trận kiểm tra \mathbf{H} . Kí hiệu $PR(x_j)$ trong phương trình (1.21) là tỉ số xác suất hậu nghiệm tổng của bit mã thứ j . Kí hiệu $PR(x_j)$ là thông tin nội tại tỉ số xác suất của bit mã thứ j .

Quá trình giải mã LDPC được thực hiện bằng các trao đổi thông tin theo cả chiều ngang và dọc của ma trận kiểm tra \mathbf{H} trong mỗi lần lặp giải mã. Quá trình trao đổi thông tin được thực hiện bằng cách tính toán và cập nhật lặp đi lặp lại các tỉ số $PR_{i,j}$ và $LR_{i,j}$ trong phương trình (1.19) và (1.20). Ví dụ chúng ta xem xét các thành phần khác '0' trong ma trận kiểm tra \mathbf{H}_r cho trong hình 1.8 tại vị trí (3,3) cho quá trình tính toán xác suất $LR_{i,j}$ của phương trình (1.19). Có bốn thành phần khác '0' nằm trong cùng hàng thứ 3 của ma trận \mathbf{H}_r , tương ứng với vị trí (3,3), đó là các điểm giao của hàng thứ 3 với các cột thứ 7, 12, 14 và 15. Do đó, giá trị $LR_{i,j}$ của thành phần khác '0' tại vị trí (3,3) sẽ cập nhật thông tin ngoại lai từ các thành phần khác '0' trong hàng thứ 3 của ma trận kiểm tra theo phương trình (1.19). Cụ thể hơn nữa, dựa trên phương trình (1.19) thông tin ngoại lai xác định xác suất giá trị của bit thứ 3 là '0' dựa trên xác suất của tất cả các bit liên quan đến tập phương trình kiểm tra từ mã đúng của mã LDPC. Ngược lại, giá trị $PR_{i,j}$ của phương trình (1.20) được tính bằng cách tổng hợp tất cả các thông tin cập nhật từ các vị trí (4,3) và (5,3), tương ứng trong hình 1.8. Trong quá trình lặp cập nhật các phương trình (1.19), (1.20) và (1.21) các giá trị PR trong phương trình (1.20) và (1.21) được sử dụng hai lần.

Giá trị tỉ số xác suất của mỗi bit $PR(x_j)$ được tính như trong phương trình (1.21) sau mỗi lần lặp. Dựa trên tỉ số xác suất $PR(x_j)$ này, bộ giải mã LDPC đưa ra quyết định cứng cho chuỗi bit tương ứng sẽ được kiểm tra bằng các phương trình kiểm tra từ mã hợp lệ của ma trận kiểm tra. Nếu quá trình kiểm tra này không thỏa mãn thì bộ giải mã LDPC sẽ tiếp tục lặp lại quá trình giải mã để đạt được từ mã hợp lệ, các giá trị $PR_{i,j}$, với $i = 1, \dots, M; j = 1, \dots, N$ của mỗi thành phần khác '0' trong ma trận kiểm tra sẽ được tính bằng phương trình (1.20). Sự khác biệt giữa giá trị tỉ số xác suất $PR_{i,j}$ của các thành phần khác '0' trong ma trận kiểm tra trong phương trình (1.20) và $PR(x_j)$ của bit ở trên là tỉ số xác suất $PR(x_j)$ của bit đang xét trong phương trình (1.21) quan hệ chặt chẽ với thông tin được cung cấp bởi tất cả các thành phần khác '0' tại các cột tương ứng liên quan với bit thứ j của ma trận kiểm tra. Trong khi đó tỉ số $PR_{i,j}$ của một thành phần khác '0' của ma trận kiểm tra được tính dựa trên tất cả các thành phần khác '0' trong các cột, ngoại trừ giá trị của bản thân thành phần đang xét, như trong phương trình (1.20).

Các bước mô tả phương pháp giải mã dựa theo xác suất của Gallager [6, 18]:

1. Giả sử kênh truyền dẫn là kênh Gauss. Dựa trên các giá trị bit mềm (soft-bit) y_i của chuỗi bit đầu ra bộ giải điều chế, tỉ số xác suất của bit thứ j thu được giữa hai giá trị nhị phân '0' và '1' được tính như sau [112, 113]:

$$P_j^1 = P(y_j | x_j = 1) = \frac{1}{\sqrt{2\pi\rho}} e^{-\frac{(y_j-1)^2}{2\rho^2}} \quad (1.22)$$

$$P_j^0 = P(y_j | x_j = 0) = \frac{1}{\sqrt{2\pi\rho}} e^{-\frac{(y_j+1)^2}{2\rho^2}} \quad (1.23)$$

trong đó y_j và ρ tương ứng là giá trị bit mềm thứ j thu được tại đầu ra bộ giải điều chế và độ lệch chuẩn của đầu ra kênh truyền dẫn.

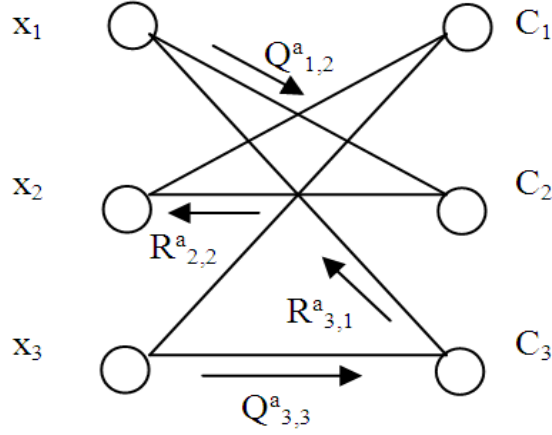
2. Đặt $P_{i,j}^1$ là xác suất giá trị nhị phân '1' tại vị trí bit mã hóa thứ j , mà tại hàng thứ i của ma trận kiểm tra \mathbf{H} , các phần tử kiểm tra từ mã là khác 0, trong đó $j = 1, \dots, N$ và $(i = 1, \dots, M)$. Các xác suất này ban đầu được gán các giá trị P_j^1 ở phương trình (1.22).
3. $LR_{i,j}$ là tỉ số xác suất bit mã thứ j có giá trị nhị phân là '0' trên xác suất bit đó có giá trị nhị phân là '1', mà xác suất này dựa trên các xác suất thông tin của các bit mã thứ l (với $l \neq j$), tương ứng với các phần tử khác '0' nằm trong hàng thứ i của ma trận kiểm tra \mathbf{H} . Các xác suất $LR_{i,j}$ được cập nhật bằng phương trình 1.19.
4. $PR_{i,j}$ là tỉ số xác suất cập nhật cho các phần tử khác '0' tương ứng với hàng thứ i trong ma trận kiểm tra \mathbf{H} , từ xác suất thông tin của các phần tử khác '0' thứ k thuộc cột thứ j trong ma trận \mathbf{H} . Tỉ số xác suất này được cập nhật bằng phương trình 1.20.
5. Đối với mỗi bit mã, tỉ số xác suất $PR(x_j)$ của bit x_j có giá trị giữa '0' và '1' được cập nhật bằng phương trình 1.21.
6. Giá trị của xác suất $P_{i,j}^1$ tương ứng với các phần tử khác không trong ma trận kiểm tra \mathbf{H} được cập nhật bằng $\frac{1}{(1+PR_{i,j})}$, trong đó giá trị $PR_{i,j}$ là giá trị được cập nhật ở bước 4.
7. Dựa trên giá trị xác suất $PR(x_i)$ được cập nhật trong bước 5, bộ giải mã tạm thời thực hiện giải mã bit cứng (hard-bit) và nhân với ma trận kiểm tra đã được chuyển vị \mathbf{H}_r^T , để kiểm tra tính hợp lệ của từ mã.

8. Nếu như véc tơ kết quả của phương trình kiểm tra từ mã hợp lệ bằng $\mathbf{0}$, thì Quá trình lặp giải mã sẽ dừng lại và đầu ra bộ giải mã là chuỗi bit đã được giải mã có giá trị nhị phân tương ứng với bước 7.
9. Trong trường hợp phương trình (1.8) không được thỏa mãn, đồng thời số lần lặp giải mã đã đạt đến mức cao nhất, thì quá trình giải mã được coi là không hoàn thành và chuỗi bit đầu ra bộ giải mã cũng tương ứng với chuỗi bit ở bước 7.
10. Trường hợp phương trình (1.8) không được thỏa mãn, mà số lần lặp vẫn chưa vượt quá mức cao nhất, thì quá trình giải mã được lặp lại từ bước 3.

1.3.2. Phương pháp truyền giá trị thông tin *LLR*

Phương pháp giải mã này dựa trên việc truyền qua lại các thông tin *LR* giữa nút kiểm tra (Check Nodes) và nút biến số (Variable Nodes) thuộc đồ thị song phương (Bipartite graph) của mã LDPC. Hình 1.12 biểu diễn đồ thị song phương của mã LDPC. Đồ thị này được biểu diễn bằng các cột và hàng của ma trận kiểm tra \mathbf{H} , tương ứng với các nút biến số (Variable nodes) và các nút kiểm tra (Check nodes) [30, 76, 106, 111, 114].

Hình 1.12 là đồ thị song phương của mã LDPC, phía bên tay trái là các nút biến số, bên tay phải là các nút kiểm tra. $Q_{i,j}^a$ là xác suất truyền từ nút biến số thứ j sang nút kiểm tra thứ i , trong đó nút biến số đang ở trạng thái $a \in (0, 1)$, $j = (1, \dots, N)$ tương ứng với các nút biến số bên trái của đồ thị song phương, $i = (1, \dots, M)$ tương ứng với các nút kiểm tra của đồ thị song phương. Ngược lại, các thông tin xác suất $R_{i,j}^a$ được truyền từ các nút kiểm tra thứ i sang các nút biến số thứ j , xác định



Hình 1.12: Đồ thị song phương của mã LDPC

lượng giá trị xác suất của nút kiểm tra thứ i dựa trên xác suất của tất cả nút thành phần có quan hệ với nút kiểm tra thứ i , ngoại trừ nút j , ở trạng thái a . Các giá trị $Q_{i,j}^a$ và $R_{i,j}^a$ được lưu trong các thành phần khác '0' của ma trận kiểm tra \mathbf{H} . Tại thời điểm đầu tiên của quá trình giải mã lặp, các giá trị xác suất $Q_{i,j}^a$ được gán các giá trị xác suất với xác suất nội tại (*intrinsic probability*) P_j^a tương ứng với symbol thứ j được truyền với giá trị nhị phân là a . $R_{i,j}^a$ ước lượng xác suất tương ứng của nút kiểm tra thứ i , khi symbol thứ j ở trạng thái a . Xác suất này có thể tính bằng công thức sau [33]:

$$R_{i,j}^a = \sum_{C:c_j=a} P(z_j = 0|C) \prod_{k \in \{C_j\}, k \neq j} Q_{i,k}^{c_k}; \quad (1.24)$$

với $i = 1, \dots, M; j = 1, \dots, N$

Các kí hiệu c_j và C_j trong phương trình (1.24) được sử dụng chỉ bit thứ j của từ mã C và tập chỉ số cột của hàng thứ j của ma trận kiểm tra tương ứng. Phương trình (1.24) là phương trình cập nhật giá trị của $R_{i,j}^a$. Xác suất có điều kiện $P(z_j = 0|C) = 1$ khi và chỉ khi phương trình kiểm tra từ mã hợp lệ thỏa mãn, trong trường hợp phương trình này không thỏa mãn, xác suất có điều kiện ở trên sẽ có giá trị bằng 0. $Q_{i,k}^{c_k}$ là

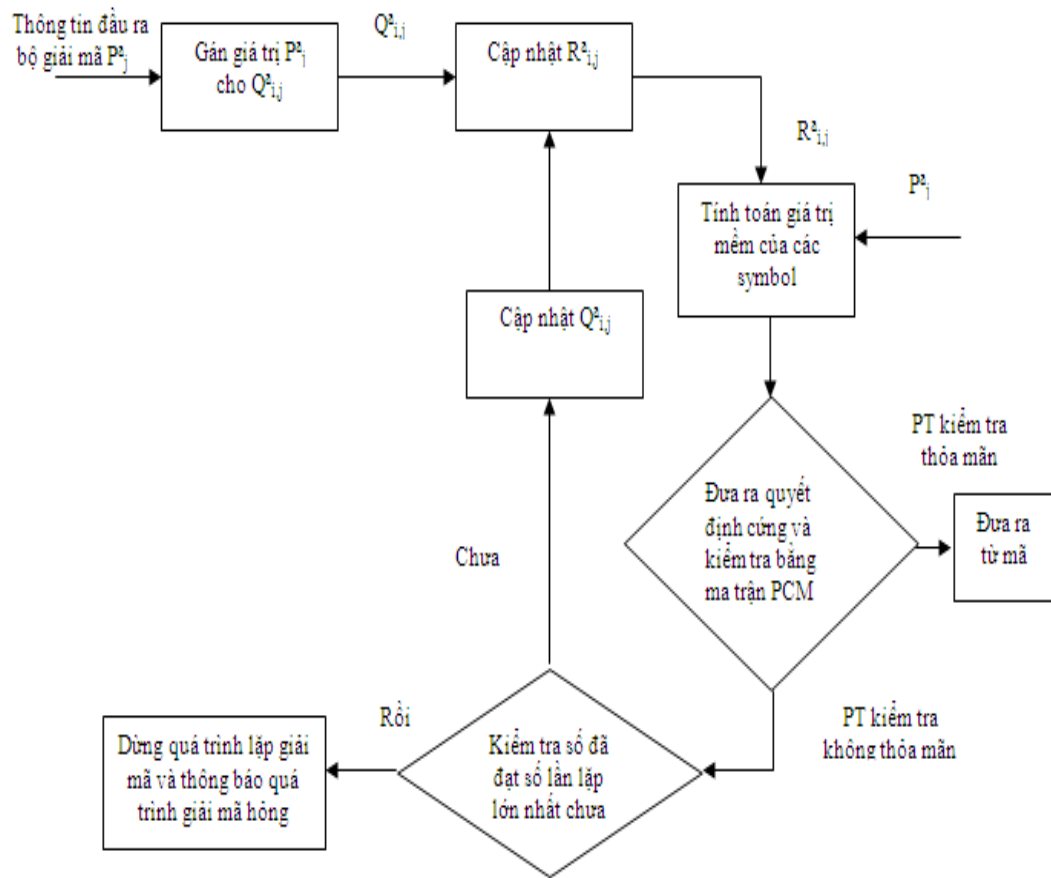
xác suất của nút biên số thứ k tại trạng thái giải mã c_k đối với phương trình kiểm tra thứ i . Cuối cùng, C_i là tập các chỉ số cột của tất cả các thành phần khác '0' trong hàng thứ i của ma trận kiểm tra. Việc cập nhật xác suất $Q_{i,j}^a$ được thực hiện bằng phương trình dưới đây [33]:

$$Q_{i,j}^a = \alpha_{i,j} P_j^a \prod_{k \in \{R_j\}, k \neq i} R_{i,j}^a, \quad (1.25)$$

trong đó R_j là tập các chỉ số hàng chứa các thành phần khác không trong cột j ; $j = 1, \dots, N$ của ma trận kiểm tra. Giá trị P_j^a là xác suất nội tại liên quan đến thông tin đầu ra của kênh truyền dẫn với giả thiết C_j ở trạng thái a , hệ số $\alpha_{i,j}$ được sử dụng để đảm bảo tổng xác suất $\sum_a Q_{i,j}^a = 1$. Quá trình giải mã LDPC được thực hiện bằng cách lặp lại các quá trình cập nhật giá trị cho các xác suất $R_{i,j}^a$ và $Q_{i,j}^a$. Sau mỗi lần cập nhật giá trị cho hai xác suất này, bộ giải mã LDPC đưa ra quyết định cứng để xác định dấu của tích hai thành phần $R_{i,j}^a$ và xác suất nội tại của mỗi symbol, ví dụ như:

$$P(c_j) = P_j^a \prod_{k \in \{R_j\}} R_{k,j}^a. \quad (1.26)$$

Symbol nhị phân có xác suất cao hơn trong tập hai symbol đang xét sẽ được giữ lại. Nếu như quyết định cứng đưa ra từ mã hợp lệ thỏa mãn các phương trình kiểm tra tính từ mã đúng của ma trận kiểm tra \mathbf{H}_r , thì quá trình giải mã sẽ kết thúc và từ mã tương ứng sẽ được xuất ra tại đầu ra bộ giải mã LDPC. Ngược lại, bộ giải mã LDPC sẽ tiếp tục quá trình cập nhật giá trị cho các xác suất $R_{i,j}^a$ và $Q_{i,j}^a$ theo các phương trình (1.24) và (1.25) cho đến khi tìm được một từ mã hợp lệ hoặc số lần lặp đạt giá trị lớn nhất đã được xác định từ trước. Quá trình giải mã lặp của bộ mã giải mã LDPC được cho trong hình 1.13.



Hình 1.13: Lược đồ giải mã lặp của bộ mã LDPC

1.4. Kết luận chương 1

Trong chương 1 đã tiến hành nghiên cứu, phân tích khả năng sửa lỗi cũng như độ phức tạp của các bộ mã hóa đang được sử dụng phổ biến trên thế giới như mã: chập, mã RSC, mã Turbo, mã LDPC.

Để có công cụ đánh giá khả năng sửa lỗi của các loại mã, trong chương này đã tiến hành xây dựng chương trình mô phỏng bằng ngôn ngữ C++. Các kết quả mô phỏng đánh giá khả năng sửa lỗi của mã LDPC và các yếu tố ảnh hưởng đến khả năng sửa lỗi của mã LDPC được trình bày trong phụ lục A của Luận án.

Qua phân tích, đánh giá các kết quả thu được ta nhận thấy mã LDPC có khả năng sửa lỗi tốt, sần lỗi hầu như không có và độ phức tạp tương đối nhỏ so với mã Turbo khi sử dụng từ mã ngắn, vì vậy có thể đưa ra khẳng định rằng mã LDPC hoàn toàn phù hợp với thiết kế chip, RAM của các thiết bị mã hóa và giải mã trong truyền hình. Tuy nhiên, mã LDPC cũng tồn tại một số nhược điểm như có độ tính toán giải mã phức tạp, phụ thuộc vào độ dài từ mã. Việc xây dựng ma trận kiểm tra \mathbf{H} và tính toán ma trận sinh \mathbf{G} từ ma trận kiểm tra \mathbf{H} là rất phức tạp.

Ma trận kiểm tra quyết định khả năng sửa lỗi của mã LDPC. Việc xây dựng quan hệ và cấu trúc ma trận sinh, ma trận kiểm tra để tăng khả năng sửa lỗi và giảm độ phức tạp tính toán tái tạo ma trận sinh từ ma trận kiểm tra là yếu tố quan trọng trong thiết kế mã LDPC.

Mã LDPC đã và đang được phổ biến rộng rãi trong hệ thống tiêu chuẩn truyền hình số mặt đất DVB - T2, DVC-2 và truyền hình số qua vệ tinh DVB-S2. Vì vậy, trong các chương tiếp theo, luận án sẽ đề xuất phát triển một mô hình mới, thiết kế và mô phỏng mô hình mã LDPC mới, phân tích, đánh giá so sánh, với các mô hình lai ghép H-ARQ với mã LDPC, V-BLAST sử dụng mã LDPC nhằm cải thiện khả năng sửa lỗi của mã LDPC và giảm độ phức tạp của hệ thống lai ghép với mã LDPC, giúp đưa hệ thống lai ghép mã LDPC có khả năng áp dụng vào thực tế trong các hệ thống thu phát truyền hình số, nhất là các bộ máy phát, đầu thu settopbox.

Chương 2

THIẾT KẾ MA TRẬN SINH VÀ MA TRẬN KIỂM TRA CỦA MÃ LDPC

Mở đầu

Hàm phân bố rời rạc cho các hàm trọng cột của ma trận kiểm tra hay các hàng tương ứng của ma trận sinh quyết định tính độc lập tuyến tính của các cột trong ma trận kiểm tra. Điều này quyết định độ dài những vòng lặp trao đổi thông tin giữa bit mã và bit thông tin, tính cân đối mức ưu tiên giữa các bit thông tin trong từ mã. Đây cũng chính là yếu tố quyết định khả năng sửa lỗi của mã LDPC và độ phức tạp của mã LDPC như phân tích trong [9, 115]. Nội dung của chương này là xây dựng hàm phân bố rời rạc cho các cột của ma trận sinh thành phần và các bậc của bit thông tin cho mã LDPC để tạo ra một loại mã LDPC với những thông số vượt trội. Các kết quả mô phỏng sẽ cho phép đánh giá tính ưu việt của phương pháp đề xuất.

2.1. Xây Dựng các hàm phân bố

Trong chương này các ma trận sinh và ma trận kiểm tra của mã LDPC sẽ được thiết kế bằng các hàm phân bố mật độ rời rạc dừng [116–118], mục đích của thiết kế này nhằm cải thiện khả năng sửa lỗi của mã LDPC tại dải giá trị C/N thấp. Ma trận kiểm tra \mathbf{H} của mã LDPC được thiết kế dựa trên các hàm phân bố mật độ cho nút thông tin và nút kiểm tra trong ma trận kiểm tra. Ma trận sinh \mathbf{G} của mã LDPC được tính toán từ ma trận kiểm tra \mathbf{H} bằng phương pháp thuật toán rút gọn Gauss. Trong chương này, luận án đi sâu vào việc phân tích và thiết kế ma trận sinh \mathbf{G} và ma trận kiểm tra \mathbf{H} của mã LDPC theo quan hệ đã được đề cập trong [6, 18]:

$$\begin{aligned}\mathbf{G}_{k \times n} &= \left[\mathbf{I}_{k \times k} | \mathbf{A}_{k \times m} \right]; \\ \mathbf{H}_{m \times n} &= \left[\mathbf{A}_{m \times k}^T | \mathbf{I}'_{m \times m} \right]; \\ \Rightarrow \mathbf{G}_{k \times n} \cdot \mathbf{H}_{m \times n}^T &= \mathbf{0}_{k \times m}.\end{aligned}\tag{2.1}$$

Trong đó k, n, m lần lượt là số bit thông tin, độ dài từ mã và số bit kiểm tra trong một từ mã LDPC; $\mathbf{I}_{k \times k}$, $\mathbf{I}'_{m \times m}$ là các ma trận đơn vị có kích thước là $(k \times k)$ và $(m \times m)$. Khả năng sửa lỗi của mã LDPC được thiết kế theo phương pháp này phụ thuộc hoàn toàn vào cấu trúc ma trận $\mathbf{A}_{k \times m}$. Để các cột trong ma trận $\mathbf{A}_{k \times m}$ có khoảng cách trung bình là lớn, thì hàm phân bố mật độ của các cột thuộc ma trận này phải là một hàm phân bố rời rạc [119–121]. Các cột của ma trận này tương ứng với các bit mã kiểm tra trong từ mã LDPC, vì vậy chúng không những cần chứa đầy đủ thông tin kiểm tra các bit thông tin của từ mã, mà còn phải thỏa mãn tính rời rạc, không lặp lại giữa chúng. Để làm được điều

này luận án đã đưa ra khái niệm hàm phân bố chuẩn rời rạc giới hạn đối với các cột của ma trận thành phần $\mathbf{A}_{k \times m}$ của ma trận sinh \mathbf{G} và hàm phân bố tương đối đều cho các hàng của ma trận này, ở dưới đây.

2.1.1. Xây dựng hàm phân bố cho ma trận thành phần

Trong phần thiết kế này, mục đích của tác giả luận án là tạo ra ma trận thành phần $\mathbf{A}_{k \times m}$ của ma trận sinh không chứa các cột có hàm trọng là 1 để tránh hiện tượng không độc lập tuyến tính giữa các cột của ma trận sinh hay tăng khả năng chứa các thông tin của các bit nguồn trong các bit mã LDPC, thiết kế các cột ma trận thành phần của ma trận sinh không chứa các cột có hàm trọng bằng 1 (Do mã LDPC là mã hệ thống, nên ma trận sinh đã chứa ma trận đơn vị thành phần có hàm trọng của các cột bằng một). Giả sử ma trận thành phần $\mathbf{A}_{k \times m}$ có số hàng là K tương ứng với K bits thông tin, được thiết kế với các cột có hàm trọng bé nhất bằng 2, ta có thể thấy xác suất để cột thứ hai được tạo ra ngẫu nhiên mà không trùng lặp với cột thứ nhất là $(1 - 2^{1-K})$, tương tự xác suất để cột thứ ba được tạo ra ngẫu nhiên mà không trùng lặp với cột thứ nhất và thứ hai là $(1 - 2^{2-K})$, cứ như vậy ta có thể nhận thấy xác suất để toàn bộ các cột của ma trận thành phần không trùng lặp nhau được tính bằng $(1 - 2^{1-K}).(1 - 2^{2-K}) \dots (1 - 1/8).(1/4)$. với K tương đối lớn xác suất này trở nên cực nhỏ, cụ thể với $K > 10$, giá trị xác suất này vào khoảng 0,125. Xác suất để các cột của ma trận thành phần không trùng lặp được tạo ra bằng $(1 - \delta)$, trong đó δ là xác suất các cột của ma trận thành phần được tạo ra là trùng lặp nhau và xác suất này được giới hạn bởi giá trị $\delta(K) \leq 2^{1-K}$, với điều kiện $K > 1$. Khi ta tăng hàm

trọng của các cột ma trận thành phần lên, thì xác suất trùng lặp cũng tăng cao, hay nói cách khác xác suất các hàng của ma trận không trùng lặp là giảm. Tuy nhiên, để tăng được hàm trọng các cột của ma trận $\mathbf{A}_{k \times m}$, mà hạn chế sự suy giảm xác suất các cột của ma trận thành phần được tạo ra độc lập tuyến tính với nhau, ta thực hiện hàm phân bố xác suất cho các cột có hàm trọng bé nhất bằng 2 bằng giá trị $\rho(2)$ sau:

$$\begin{aligned} \rho(2) &= \frac{1}{K}; \\ \rho(i) &= \frac{1}{i \cdot (i - 1)} \cdot \text{với } i = 3, 4, \dots, K \end{aligned} \quad (2.2)$$

Như vậy giá trị hàm trọng trung bình các cột của ma trận $\mathbf{A}_{k \times m}$ lúc này là bằng $\ln K$. Tuy nhiên, việc phân bố xác suất tạo ra các cột có hàm trọng nhỏ nhất bằng 2 phải đảm bảo mối liên kết giữa các cột có hàm trọng này và các cột có hàm trọng cao hơn, đồng thời cũng đảm bảo tránh trùng lặp giống nhau giữa các cột có hàm trọng tương ứng, để đảm bảo lượng thông tin của các bit nguồn có trong bit mã là lớn nhất thì số lượng các cột có hàm trọng bé nhất, cụ thể ở đây là bằng 2 phải bằng $S = c \cdot \ln(K/\delta) \sqrt{K}$. Trong đó: c là một số dương rất nhỏ và bé hơn 1, δ là xác suất đã đề cập ở trên, K là số bit nguồn thông tin, \ln là logarit cơ số tự nhiên. Để phân bố có giới hạn hàm trọng là dừng, ở đây chúng ta sử dụng một hàm phân bố chặn trên được xác định như sau:

$$\tau(i) = \begin{cases} \frac{S}{K} \frac{1}{i} & \text{với } i=2, \dots, \frac{2 \cdot K}{S} - 1, \\ \frac{S}{K} \log\left(\frac{S}{\delta}\right) & \text{với } i = \frac{2 \cdot K}{S}, \\ 0 & \text{với } i > \frac{2 \cdot K}{S}, \end{cases} \quad (2.3)$$

trong đó giá trị $(\frac{2 \cdot K}{S})$ đảm bảo mỗi bit mã được tạo ra từ ít nhất hai bit nguồn thông tin ngẫu nhiên, hay hàm trọng bé nhất của các cột trong ma trận thành phần $\mathbf{A}_{k \times m}$ bằng 2.

Tổng hợp tất cả phân bố trên ta được phân bố chuẩn rời rạc có giới hạn trên cho các hàm trọng của ma trận thành phần $\mathbf{A}_{k \times m}$, hay nói cách khác là phân bố chuẩn rời rạc giới hạn trên cho các bậc của các bit kiểm tra như sau:

$$\lambda(d) = \frac{\rho(i) + \tau(i)}{Z}, \quad (2.4)$$

trong đó, $Z = \sum_i \rho(i) + \tau(i)$ là thành phần chuẩn hóa để đảm bảo phân bố tổng $\lambda(i)$ luôn nhỏ hơn hoặc bằng 1.

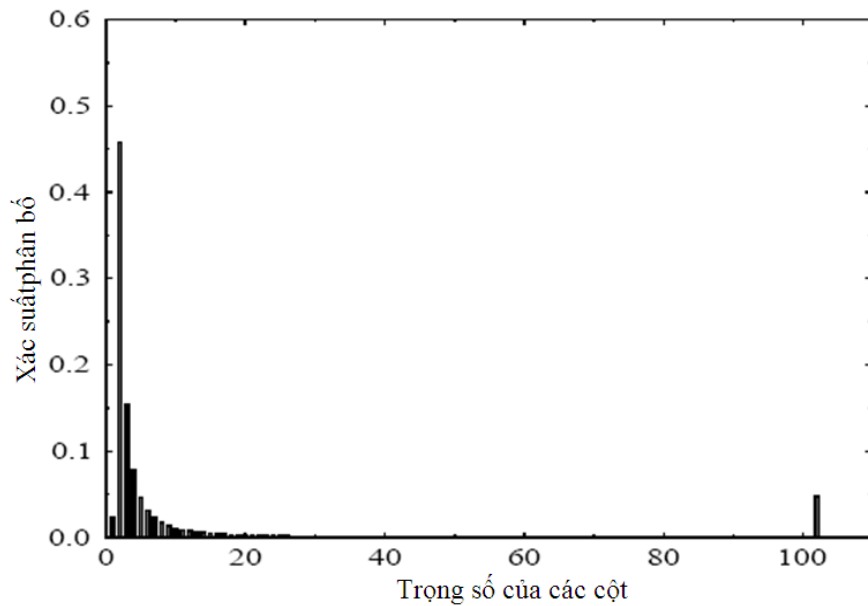
Khi tăng hàm trọng tối thiểu các cột của ma trận thành phần $\mathbf{A}_{k \times m}$ lên lớn hơn 2, ta nhận thấy xác suất để các cột của ma trận không trùng lặp nhau, hay xác suất để lượng thông tin trong các bit kiểm tra là lớn nhất bị giảm đi rõ rệt. Mặt khác, giá trị trung bình hàm trọng các cột của ma trận này cũng tăng lên, vì vậy giá trị trung bình bậc các bit kiểm tra sẽ tăng lên và điều này sẽ làm số lượng phép tính giải mã của mã LDPC tăng, hay nói cách khác độ phức tạp của quá trình giải mã LDPC tăng tỉ lệ với hàm trọng cực tiểu của ma trận $\mathbf{A}_{k \times m}$. Khả năng sửa lỗi, cũng như số lượng phép tính trong quá trình giải mã LDPC sẽ được phân tích kĩ trong phần so sánh hoạt động của các mã LDPC có hàm trọng trung bình của các cột ma trận thành phần khác nhau trong phần phân tích hoạt động của mã LDPC bằng đồ thị EXIT chart của phần này.

Trong trường hợp tổng quát, nếu chúng ta thiết kế hàm trọng bé nhất của các cột ma trận thành phần là t , có thể viết lại hàm phân bố chuẩn rời rạc có giới hạn cho các cột của ma trận $\mathbf{A}_{k \times m}$ thuộc ma trận

sinh của mã LDPC như sau:

$$\lambda(x) = \frac{1}{Z} \left[1 + \frac{S}{K} \right] \cdot x^t + \sum_{i=2t, \dots, (\frac{K \cdot t}{S} - 1)} \frac{t}{Z} \left[\frac{1}{i \cdot \binom{i}{t} - 1} + \frac{S}{K} \cdot \frac{1}{i} \right] \cdot x^i + \frac{S}{Z \cdot K} \cdot \left[\ln \frac{S}{\rho} + \frac{1}{(\frac{K}{S} - 1)} \right] \cdot x^{\frac{K \cdot t}{S}}. \quad (2.5)$$

Trong đó t là số nguyên dương, ($0 < \delta < 1$), $S = c \cdot \sqrt{K} \ln(\frac{K}{\delta})$ và cuối cùng Z là phân số chuẩn hóa đảm bảo tổng xác suất của hàm phân bố luôn bằng 1. Hàm phân bố trong phương trình (2.5) có bậc lớn nhất của cột là $\frac{K \cdot t}{S}$ để đảm bảo mỗi bit thông tin được kiểm tra ít nhất bằng t các bit kiểm tra, mà vẫn đảm bảo khoảng cách cực tiểu của các cột trong ma trận sinh thành phần là lớn nhất.



Hình 2.1: Phân bố mật độ chuẩn rời rạc hàm trọng của các cột ma trận kiểm tra thành phần.

Hình 2.1 là phân bố chuẩn rời rạc hàm trọng của các cột thuộc ma trận sinh thành phần, theo phương trình (2.5). Qua việc phân tích

phương trình (2.5), ta nhận thấy xác suất $p(t, K - t)$ khi một bit kiểm tra của mã LDPC được tạo ra mà vẫn tồn tại $(K - t)$ bit thông tin không liên quan đến bất kỳ bit kiểm tra nào đã được tạo gọi là xác suất tạo bậc hàm DRP (Degree Release Probability). Xác suất này được tính như sau:

- $p(i, L) = 1$ khi $L = K - t$ và $i = t$;
- $p(i, L) = \frac{i \cdot (i-1) \cdot L \prod_{j=1}^{i-1} (K - (L+1) - j)}{\prod_{j=1}^{i-1} (K - j)}$
trong đó $i = 2 \cdot t, 3 \cdot t, \dots, \frac{K \cdot t}{S}$ và $L = K - i + 1, \dots, t$;
- Đối với tất cả các giá trị i và L khác ta có $p(i, L) = 0$.

Trong đó L là số các bit thông tin đầu vào vẫn chưa được sử dụng để tạo các bit kiểm tra đầu ra. Do đó, với hàm phân bố mật độ là $\lambda(x)$ trong phương trình (2.5), xác suất của sự kiện khi một bit kiểm tra của mã LDPC được tạo ra mà vẫn tồn tại L bit thông tin chưa có mặt trong bất kỳ bit kiểm tra nào thuộc từ mã LDPC đang xét là bằng $P(i, L) = \lambda(x) \cdot p(i, L)$, trong đó $p(i, L)$ là DRP của bit kiểm tra có bậc bằng i thuộc từ mã LDPC đang xét.

Giá trị bậc trung bình a của các bit kiểm tra thuộc từ mã LDPC được xác định bằng công thức sau:

$$a = \sum_{i=t}^{\frac{K \cdot t}{S}} i \cdot \lambda(x) \tag{2.6}$$

Đặt ϵ là một số thực rất nhỏ lớn hơn 0 và ϵ được xác định như sau:

$$\epsilon = \frac{4}{4 + \frac{K \cdot t}{S}} \tag{2.7}$$

Chúng ta có mối quan hệ giữa ρ trong phương trình (2.5) và ϵ trong phương trình (2.6) như sau:

$$\rho = \frac{\frac{\epsilon}{4}}{1 + \epsilon}. \quad (2.8)$$

Xác suất $P_{\text{lâncận}}$ của sự kiện một bit thông tin là lân cận của L bit kiểm tra, được xác định như sau:

$$P_{\text{lâncận}} = \binom{n}{L} \left(\frac{a}{K}\right)^L \left(1 - \frac{a}{K}\right)^{n-L}, \quad (2.9)$$

trong đó: n là số bit mã trong một từ mã LDPC, a được xác định trong phương trình (2.6).

Hàm sinh của hàm phân bố mật độ các bit thông tin trong ma trận sinh của mã LDPC được xác định bằng công thức dưới đây:

$$F = \sum_L \binom{n}{L} \left(\frac{a}{K}\right)^L \cdot t^L \cdot \left(1 - \frac{a}{K}\right)^{n-L}. \quad (2.10)$$

Quá trình giải mã lặp của mã LDPC được thiết kế theo kiểu này là quá trình trao đổi các thông tin logarit hợp lệ ($LLRs$) giữa nút kiểm tra và nút thông tin của ma trận kiểm tra \mathbf{H} . Đặt Q và R là các thông tin $LLRs$ chuyển từ các nút thông tin sang nút kiểm tra và từ các nút kiểm tra sang các nút thông tin. Các thông tin $LLRs$ R truyền từ các nút thông tin sang các nút kiểm tra được xác định như sau:

$$Q = \sum_{i=0}^{d_m-1} R_i, \quad (2.11)$$

trong đó giá trị ban đầu gán cho các nút biến số được lấy từ kênh truyền và $\{R_i; \text{với } i = 1, \dots, d_m - 1\}$ là thông tin $LLRs$ truyền từ các nút kiểm tra sang nút thông tin, ngoại trừ nút kiểm tra có thông tin Q được truyền

đến; d_m là bậc của nút thông tin.

Các thông tin $LLRs$ Q truyền từ các nút kiểm tra sang các nút thông tin được xác định bằng biểu thức sau:

$$\tanh\left(\frac{R}{2}\right) = \prod_{i=1}^{d_c-1} \tanh\left(\frac{Q_i}{2}\right), \quad (2.12)$$

trong đó d_c là bậc của nút kiểm tra, \tanh là kí hiệu của tang hyperbol, $\{Q_i; i = 1, \dots, (d_c - 1)\}$ là các thông tin $LLRs$ đến từ các nút thông tin, ngoại trừ nút thông tin có thông tin $LLRs$ R được gửi đến. Đặt m_R, m_{R_0} và m_Q là các giá trị trung bình của R, R_0 và Q . Từ phương trình (2.11) ta có:

$$m_Q^{(j)} = m_{R_0} + (d_m - 1) \cdot m_R^{(j-1)}, \quad (2.13)$$

trong đó j là vòng lặp giải mã thứ j và $(m_{R_0} = 4 \cdot \frac{E}{N_0})$, với E là năng lượng bit được truyền và N_0 là mật độ phổ công suất một phía của tạp nhiễu. Giả sử R được phân bố theo hàm Gauss, m_R được cập nhật như sau [85]:

$$m_R^j = J^{-1}[I(X; R^j)], \quad (2.14)$$

trong đó $J(m_R)$ được xác định như sau:

$$J(m_R) = I(X; R^j) = \int \frac{1}{\sqrt{4\pi m_R}} \cdot e^{-\frac{j-m_R}{4m_R}} (1 - \log_2(1 + e^{-j})) dj. \quad (2.15)$$

Thông tin tương hỗ giữa đầu ra và đầu vào $I(X; R_j)$ được xác định như sau [85]:

$$I(X; R) = \frac{1}{\log_2} \sum_{i=1}^{\infty} \frac{1}{2i(2i-1)} [E(T_Q^{2i})]^{d_c-1}, \quad (2.16)$$

trong đó:

$$\phi_i(m_Q) = E(T_Q^{2i}) = \int_{-1}^{+1} \frac{2t^{2i}}{(1-t^2)\sqrt{4\pi m_Q}} \cdot e^{-\frac{(\ln(\frac{1+t}{1-t})-m_Q)^2}{4m_Q}} dt. \quad (2.17)$$

Cuối cùng ta đi đến các phương trình m_Q và m_R như sau:

$$\begin{aligned} m_Q^i &= m_{R_0} + (d_m - 1)m_R^{j-1}; \\ m_R^j &= J^{-1}\left(\frac{1}{\ln 2} \sum_{i=1}^{\infty} \frac{1}{2i(2i-1)} [E(T_{m_Q}^{2i})]\right). \end{aligned} \quad (2.18)$$

Dựa trên phương trình (2.17) ta đưa đến phương trình trao đổi thông tin EXIT cho các nút thông tin như sau:

$$\begin{aligned} I_{E_v} = I(X; Q) &= I(X; R_0, R_1, \dots, R_{d_m-1}) \\ &= f[I(X; R_0); I(X; R)] = f(I_{ch}; I_{A_v}). \end{aligned} \quad (2.19)$$

Tương tự như vậy, từ phương trình (2.16) ta có phương trình trao đổi thông tin EXIT của các nút kiểm tra như sau [85]:

$$\begin{aligned} I_{E_c} &= I(X; R) = I(X; Q_1, \dots, Q_{d_c-1}) = f(I(X; Q)) \\ &= f(I_{A_c}) = \frac{1}{\ln 2} \sum_{i=1}^{\infty} [\phi_i(J^{-1}(I_{A_c}))]^{d_c-1}, \end{aligned} \quad (2.20)$$

trong đó $I_{ch} = I(X; R_0)$ là lượng thông tin trung bình đầu ra của kênh truyền, $I_{A_v} = I(X; R)$ là lượng thông tin tiền nghiệm trung bình của các nút thông tin tại đầu vào bộ giải mã và I_{E_v} là lượng thông tin ngoại lai trung bình của các nút thông tin tại đầu ra bộ giải mã. Cuối cùng chúng ta đi đến các phương trình trao đổi thông tin EXIT của các nút thông tin và nút kiểm tra như sau:

- Hàm trao đổi thông tin EXIT của nút thông tin:

$$I_{E_v} = f(I_{ch}, I_{A_v}) = J[J^{-1}(I_{ch} + (j-1)J^{-1}(I_{A_v}))]. \quad (2.21)$$

- Hàm trao đổi thông tin EXIT của nút kiểm tra:

$$\begin{aligned}
 I_{E_c} &= f(I_{A_c}) = \sum_{j=2}^{d_c} d_{c_j} I_{E_{c_j}} \\
 &= \sum_{j=2}^{d_{cmax}} d_c(j) \cdot \frac{1}{\ln 2} \sum_{i=1}^{\infty} \left(\frac{1}{2^i(2^i - 1)} \right) [\phi_i(j^{-1}(I_{A_c}))]^{j-1}. \quad (2.22)
 \end{aligned}$$

Để tăng khả năng sửa lỗi của mã LDPC được thiết kế trong chương này, chúng ta cần phải xem xét đến hàm phân bố mật độ $\nu(x)$ cho các bit thông tin trong một từ mã LDPC. Phân bố tần suất xuất hiện các bit thông tin trong các bit kiểm tra của từ mã LDPC quyết định khả năng sửa lỗi của mã LDPC. Nếu các bit thông tin được phân bố không đồng đều trong các bit kiểm tra, điều này có nghĩa là một số bit thông tin được ưu tiên chống lỗi hơn số khác. Khi các bit thông tin ít xuất hiện trong các bit kiểm tra bị ảnh hưởng của tạp nhiễu kênh truyền, thì khả năng khôi phục lại những bit này là thấp hơn nhiều so với các bit thông tin có mặt nhiều hơn trong các bit kiểm tra của từ mã. Điều này sẽ gây nên hiện tượng sàn lỗi trong đồ thị quan hệ tỉ số lỗi bit BER và E_b/N_0 , làm suy giảm khả năng sửa lỗi của mã LDPC trong từng dải E_b/N_0 nào đó. Do đó, để tránh hiện tượng này chúng ta sẽ phân bố mật độ của các bit thông tin cấu tạo nên các bit mã, hay nói một cách khác là bậc của các bit thông tin, là đồng đều. Để phân tích thêm về vấn đề này chúng ta chuyển sang phần tạo hàm phân bố đồng đều cho các bit thông tin trong mục 2.1.2 dưới đây.

2.1.2. Xây dựng hàm phân bố cho các bit thông tin

Thông thường các hàm tạo chuỗi số nguyên ngẫu nhiên được biểu diễn bằng công thức sau [117]:

$$I_{j+1} = (a \cdot I_j + C) \pmod{M}, \quad (2.23)$$

trong đó M là số cơ số của hàm cộng modulo được sử dụng, a và c là các số nguyên được gọi là số nhân và số tăng tương ứng. Để đạt được hàm phân bố mật độ đồng đều đối với các bit thông tin, dưới đây ta áp dụng hai phương pháp tạo chuỗi ngẫu nhiên [118, 122]:

- Phương pháp thứ nhất là tạo một số nguyên ngẫu nhiên từ phép dịch tổng hai số nguyên bất kì trong chuỗi số nguyên sau khi đã được lấy modulo $2b$, trong đó $2b$ tương ứng với số phần tử nhớ của thanh ghi dịch. Hàm tạo chuỗi số nguyên bằng phương pháp thứ nhất được xác định như sau:

$$I_n = [(I_{n-j} + I_{n-k}) \pmod{2b}] \text{ rot } r, \quad (2.24)$$

trong đó, rot là kí hiệu của hàm quay (hay hàm dịch). Các bit đại diện cho các số nguyên $[(I_{n-j} + I_{n-k}) \pmod{2b}]$ được dịch quay vòng r các vị trí bit.

- Phương pháp thứ hai là phương pháp dịch trước các bit đại diện cho số nguyên I_{n-j} và I_{n-k} đi tương ứng r_1 và r_2 vị trí, trước khi thực hiện cộng modulo 2. Hàm tạo chuỗi theo phương pháp thứ hai được xác định như sau:

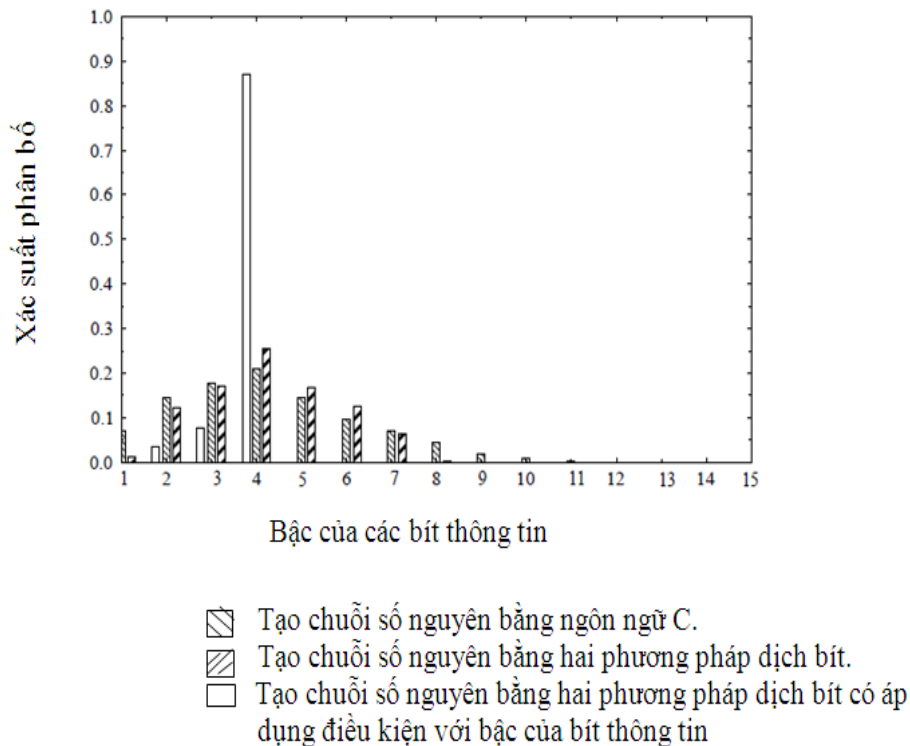
$$I_n = [(I_{n-j} \text{ rot } r_1 + I_{n-k} \text{ rot } r_2)] \pmod{2b}, \quad (2.25)$$

trong đó I_n là số nguyên được biểu diễn bằng b bit và kí hiệu $(I_{n-j} \text{ rot } r_1)$ có nghĩa là các bit của số nguyên I_{n-j} dịch sang phải r_1 vị trí, ví dụ $\{00001111_2 \text{ rot } 3 = 11100001_2\}$.

Hai phương pháp tạo chuỗi ngẫu nhiên trên được áp dụng cùng với điều kiện của bậc các bit thông tin như sau:

$$\begin{aligned} d_m &\leq \bar{D}_m \\ \bar{D}_m &= \left[\frac{1-r}{r} (\bar{d}_c - 1) \right]. \end{aligned} \quad (2.26)$$

trong đó d_m là bậc của các bit thông tin, \bar{D}_m là bậc trung bình của các bit thông tin và \bar{d}_c là bậc trung bình của các bit kiểm tra thuộc từ mã LDPC.



Hình 2.2: Phân bố mật độ cho các bit thông tin

Hình 2.2 là đồ thị Histogram phân bố mật độ bậc của các bit thông tin sử dụng các phương pháp tạo chuỗi số nguyên ngẫu nhiên khác nhau.

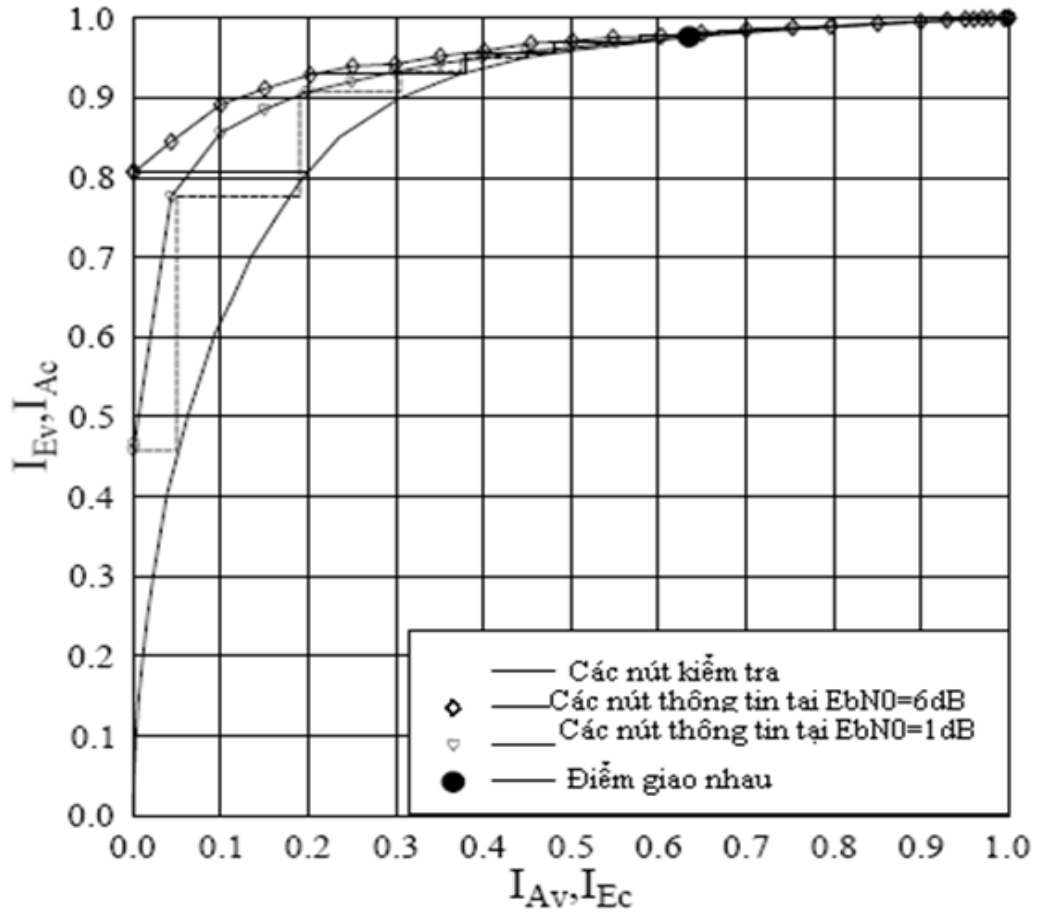
Như có thể nhận thấy trong hình 2.2, chuỗi số nguyên được tạo ra bằng hàm ngẫu nhiên trong ngôn ngữ lập trình C có phân bố rải rác từ 1 đến 12, trong khi đó chuỗi số nguyên được tạo ra bằng hai phương pháp dịch bit có phân bố trong khoảng hẹp hơn từ 2 đến 6. Khi áp dụng hai phương pháp dịch bit nói trên với điều kiện đối với bậc của các bit thông tin thỏa mãn phương trình (2.26), thì kết quả chuỗi số nguyên đầu ra có phân bố tập trung quanh giá trị 4. Điều đó có nghĩa là việc sử dụng hai phương pháp dịch bit phối hợp với điều kiện trong phương trình (2.26) cho phép tạo ra hàm phân bố mật độ đồng đều cho các bit thông tin trong từ mã LDPC. Hay nói cách khác, việc sử dụng hai phương pháp này đem lại phân bố hàm trọng đồng đều cho các hàng trong ma trận sinh thành phần.

2.2. Phân tích mã LDPC bằng đồ thị EXIT

Đồ thị EXIT của mã LDPC có hàm phân bố mật độ các cột thuộc ma trận sinh thành phần cho trong phương trình (2.5) với các tham số sau:

- Các bit mã kiểm tra của từ mã được sản sinh bằng hàm phân bố mật độ cho trong phương trình (2.5).
- Số bit kiểm tra gấp đôi số bit thông tin, có nghĩa là tỉ lệ mã $r = 1/3$.
- Chỉ số t trong phương trình (2.5) tương ứng bằng 1, $\delta = 0.5$ và $c = 0.1$.
- Số bit thông tin được sử dụng để mô phỏng là $5 \cdot 10^5$ bit.

Kết quả đồ thị trao đổi thông tin EXIT của mã LDPC được thể hiện trên hình 2.3 và 2.4.

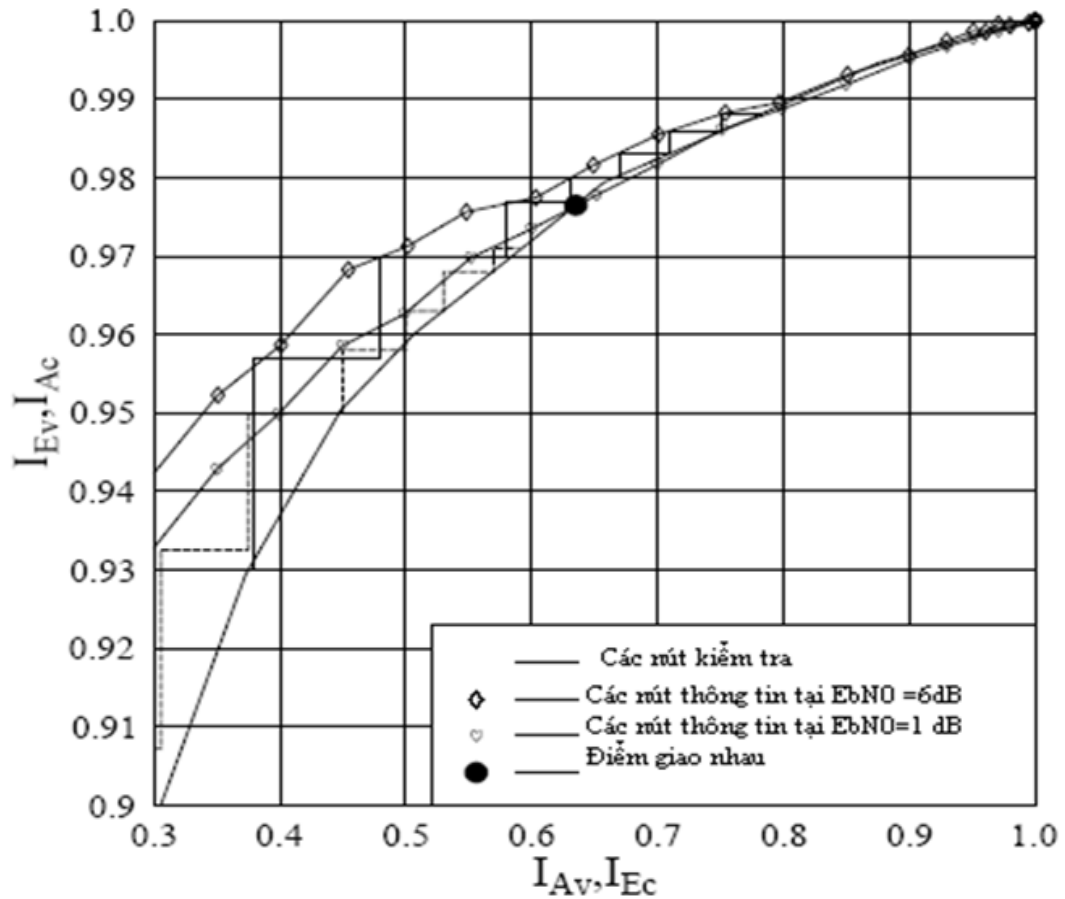


Hình 2.3: Đồ thị trao đổi thông tin EXIT của mã LDPC có hàm mật độ cho trong phương trình (2.5)

Trong hình 2.3 và 2.4 ta có thể nhận thấy tại giá trị tỉ số $E_b/N_0 = 1$ dB, đường cong EXIT của các nút thông tin và nút kiểm tra giao nhau gần điểm hội tụ hoàn toàn (1,1) của đồ thị EXIT.

Do đó, mã LDPC này không thể đạt được tỉ số lỗi bit BER cực thấp ở đầu ra bộ giải mã tại giá trị tỉ số năng lượng bit trên tạp nhiễu E_b/N_0

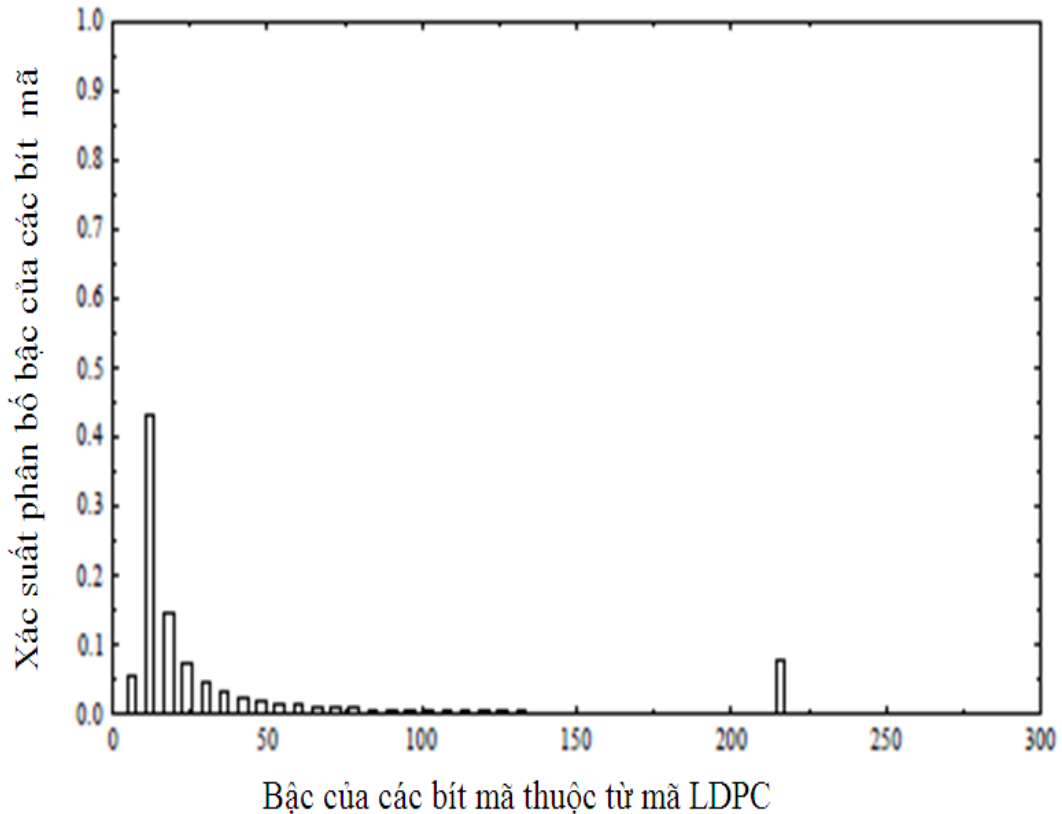
= 1 dB.



Hình 2.4: Đồ thị trao đổi thông tin EXIT của mã LDPC có hàm mật độ cho trong phương trình (2.5)

Ngược lại, tại giá trị tỉ số $E_b/N_0 = 6\text{ dB}$, đầu ra bộ giải mã LDPC này hội tụ tại điểm (1,1) của đồ thị EXIT sau khoảng 10 lần lặp giải mã. Khi thông số t tăng lên 2 phân bố có mật độ trở nên dày đặc hơn và như vậy, số phép tính cho quá trình giải mã LDPC này tăng lên. Hay nói cách khác độ phức tạp của mã LDPC tỉ lệ thuận với thông số t trong phương trình (2.5). Tuy nhiên, bù lại khả năng của mã LDPC có hàm phân bố mật độ như trong phương trình (2.5) và thông số t cao hơn cũng tăng lên. Cụ thể tại đây chúng ta phân tích khả năng sửa lỗi của mã

LDPC có hàm phân bố mật độ cho trong phương trình (2.5) và thông số $t = 2, 3$ bằng phương pháp phân tích đồ thị trao đổi thông tin EXIT.

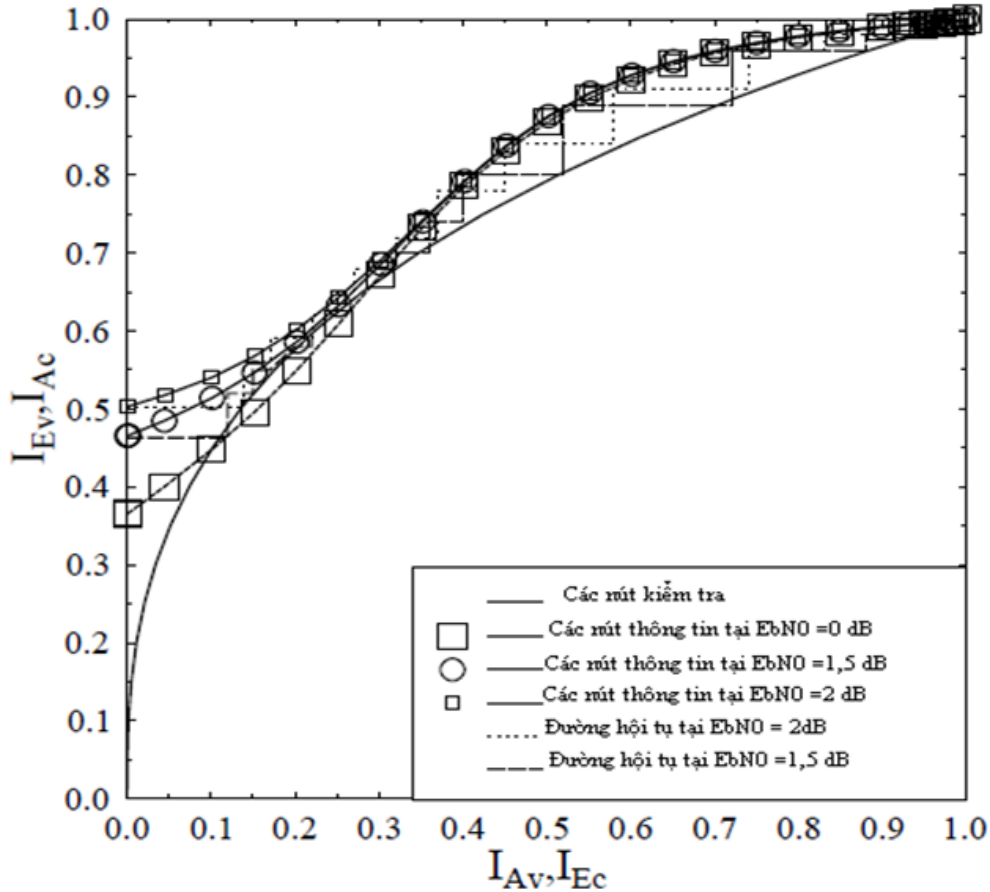


Hình 2.5: Đồ thị Histogram của mã LDPC có hàm phân bố mật độ trong 2.5 và thông số $t = 2$

Hình 2.5 là một ví dụ về đồ thị Histogram của mã LDPC có hàm phân bố mật độ cho trong phương trình (2.5) và $t = 2$. Đồ thị EXIT của mã LDPC có $t = 2, 3$ được vẽ trong hình 2.6 và 2.7.

Đồ thị EXIT của mã LDPC(1200,3600) có tỉ lệ mã $r = 1/3$, thông số $t = 2$ được vẽ trong hình 2.6. Ta có thể nhận thấy đường cong đồ thị EXIT của các nút thông tin giao đường cong đồ thị của các nút kiểm tra tại giá trị tỉ số $E_b/N_0 = 0$ dB. Điều đó có nghĩa là tại $E_b/N_0 = 0$ dB

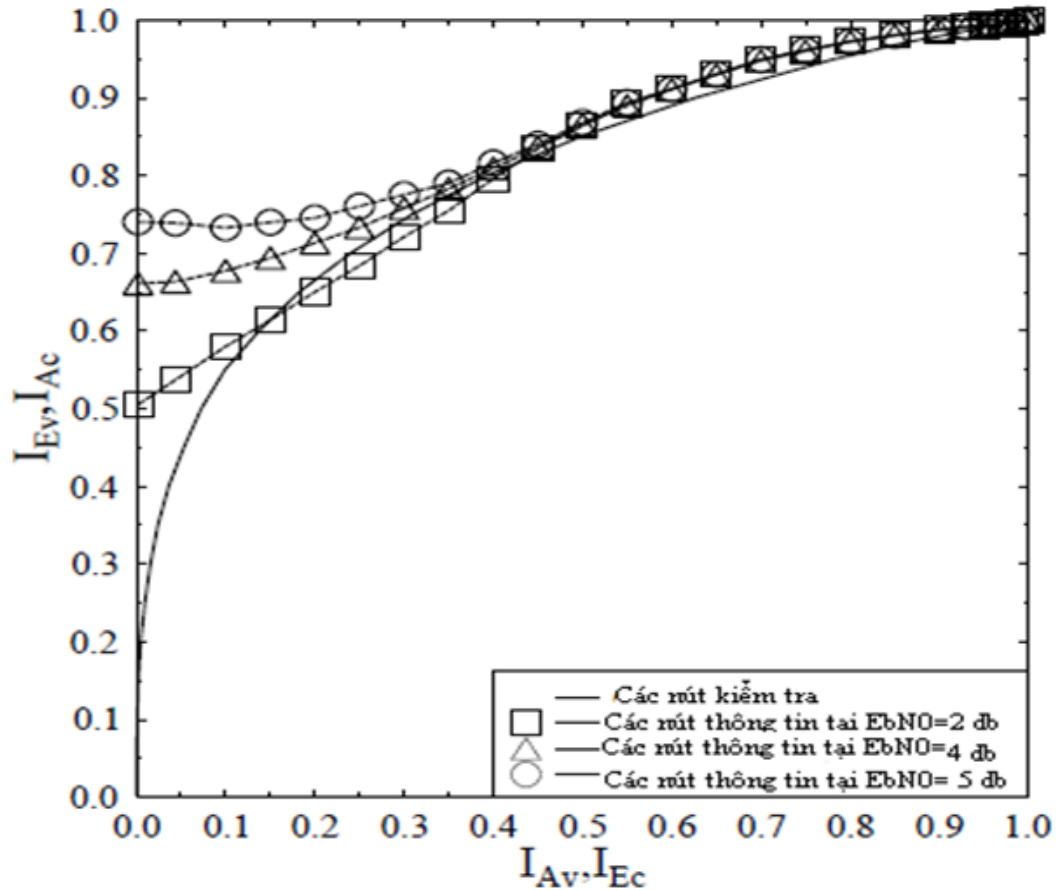
chuỗi bit đầu ra bộ giải mã LDPC không thể đạt giá trị tỉ số lỗi bit BER cực nhỏ.



Hình 2.6: Đồ thị trao đổi thông tin EXIT của mã LDPC có hàm mật độ cho trong phương trình (2.5) và $t = 2$

Ngược lại, tại giá trị tỉ số $E_b/N_0 = 1,5$ dB các đường cong của đồ thị EXIT giao nhau ở điểm (1,1) và đường hội tụ có các đỉnh nằm trọn trên các đường cong đồ thị các nút thông tin và đường cong đồ thị các nút kiểm tra. Do đó, tại giá trị $E_b/N_0 = 1,5$ dB chuỗi bit ở đầu ra bộ giải mã LDPC đạt được giá trị tỉ số BER cực nhỏ sau khoảng 20 lần lặp giải mã. Trong khi đó, tại giá trị $E_b/N_0 = 2$ dB bộ giải mã LDPC chỉ cần 12 lần lặp giải mã để đạt được giá trị tỉ số lỗi bit BER cực nhỏ ở

chuỗi bit đầu ra bộ giải mã.



Hình 2.7: Đồ thị trao đổi thông tin EXIT của mã LDPC có hàm mật độ cho trong phương trình (2.5) và $t = 3$

Khi tăng dần giá trị thông số t từ 2 lên 3 và 4, thông qua đồ thị EXIT của các mã LDPC có thông số t lớn hơn 2 trong hình 2.7 có thể nhận thấy khả năng sửa lỗi của mã LDPC có hàm phân bố mật độ cho trong phương trình (2.5) với các thông số t lớn hơn 2, bị suy giảm mạnh. Nguyên nhân của hiện tượng này là do khi tăng thông số t lên cao hơn 2 thì mật độ của ma trận kiểm tra \mathbf{H} trở nên dày đặc, khoảng cách Hamming giữa các cột trong ma trận kiểm tra \mathbf{H} giảm đáng kể và chúng gây ra lỗi tích lũy khi thực hiện trao đổi thông tin giữa các nút trong

quá trình giải mã lặp. Qua quan sát các đồ thị EXIT của mã LDPC trong hình 2.3, 2.4, 2.5, 2.6 có hàm phân bố mật độ tổng quan cho trong phương trình (2.5), ta đi đến kết luận đối với mã LDPC(1200,3600) ở trên thì giá trị $t=2$ là giá trị tối ưu cho khả năng sửa lỗi của mã. Ở phần tiếp theo sẽ phân tích các kết quả mô phỏng của mã LDPC có hàm phân bố mật độ cho trong phương trình (2.5) với các thông số t khác nhau.

2.3. Mô phỏng, đánh giá mã LDPC được thiết kế

Bảng 2.1 là các thông số mô phỏng mã LDPC có hàm phân bố mật độ cho trong phương trình (2.5) với $t=2$.

Hình 2.8 là kết quả mô phỏng khả năng sửa lỗi khác nhau của mã LDPC(1200,1800) sử dụng cùng hàm phân bố mật độ cho các bit kiểm tra cho trong phương trình (2.5) và các hàm phân bố mật độ cho các bit thông tin khác nhau ở trên. Các thông số của mã LDPC cho trong phương trình (2.5) là: $\delta = 0,5$; $c = 0.1$; $t = 2$.

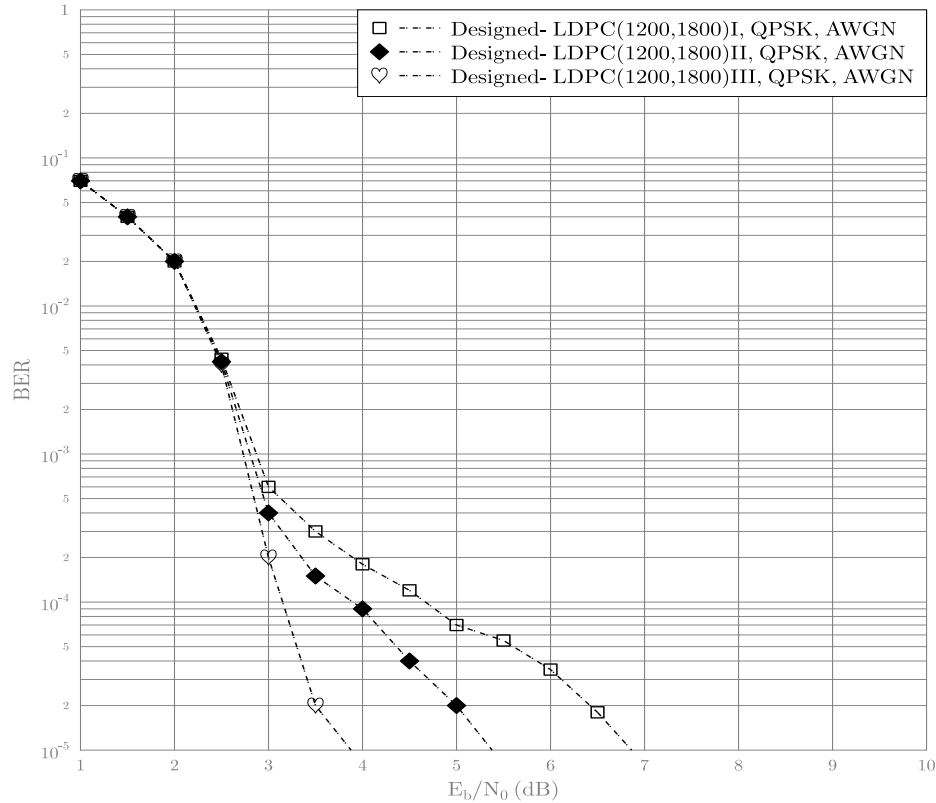
Có thể thấy khả năng sửa lỗi của mã LDPC(1200,1800), khi sử dụng bộ tạo chuỗi số nguyên ngẫu nhiên bằng phương pháp dịch bit kết hợp với điều kiện đối với bậc của các bit thông tin trong phương trình (2.26), tốt hơn hẳn so với các phương pháp khác. Cụ thể, mã LDPC(1200,1800) đạt $BER \leq 10^{-5}$ tại $E_b/N_0 = 3,8$ dB, tương ứng với đường cong đồ thị có hình \heartsuit của mã LDPC(1200,1800) loại III khi sử dụng bộ tạo chuỗi ngẫu nhiên bằng phương pháp dịch bit kết hợp với điều kiện đối với bậc của các bit thông tin cho trong phương trình (2.26).

Bảng 2.1: Các thông số mô phỏng mã LDPC có hàm phân bố mật độ cho trong (2.5) và thông số $t=2$.

Kênh tạp nhiễu	AWGN
Các thông số trong Phương trình (2.5)	$\delta = 0.5, c = 0.1, t = 2$
Tổng số bit thông tin	10^6 bit
Tỉ lệ mã	1/3
Kiểu điều chế	QPSK
Số lần lặp giải mã	0, 2, 4, 6
Mã LDPC sử dụng hàm phân bố bậc các bit thông tin trong phương trình (2.23)	Kí hiệu là LDPC loại I
Mã LDPC sử dụng hàm phân bố bậc các bit hồng tin trong các phương trình (2.23) và (2.24)	Kí hiệu là LDPC loại II
Mã LDPC sử dụng hàm phân bố bậc các bit thông tin trong các phương trình (2.23), (2.24) cùng với điều kiện bậc trung bình của các bit thông tin trong phương trình (2.26)	Kí hiệu là LDPC loại III

Trong khi đó đường cong đồ thị có kí hiệu hình \blacklozenge tương ứng với mã LDPC(1200,1800) loại II sử dụng bộ tạo chuỗi số nguyên ngẫu nhiên cho các bit thông tin bằng phương pháp dịch bit, đạt $BER < 10^{-5}$ tại $E_b/N_0 = 5,3$ dB. Cũng với tỉ số lỗi bit BER như trên, đường cong đồ thị có kí hiệu \square tương ứng với mã LDPC(1200,1800) loại I đòi hỏi tỉ số E_b/N_0 cao hơn 3 dB so với LDPC(1200,1800) loại III, vào cỡ khoảng 7 dB như

trong hình kết quả mô phỏng 2.8.



Hình 2.8: Mô phỏng khả năng sửa lỗi khác nhau của mã LDPC(1200,1800), sử dụng cùng hàm phân bố mật độ đối với các bit kiểm tra cho trong phương trình (2.5) và sử dụng các hàm phân bố mật độ khác nhau cho các bit thông tin trong từ mã LDPC.

Như vậy mã LDPC có hàm phân bố mật độ cho các bit thông tin sử dụng phương pháp tạo chuỗi số nguyên được thiết kế như đã trình bày trong mục này, có khả năng sửa lỗi tốt hơn nhiều so với các phương pháp thông thường khác.

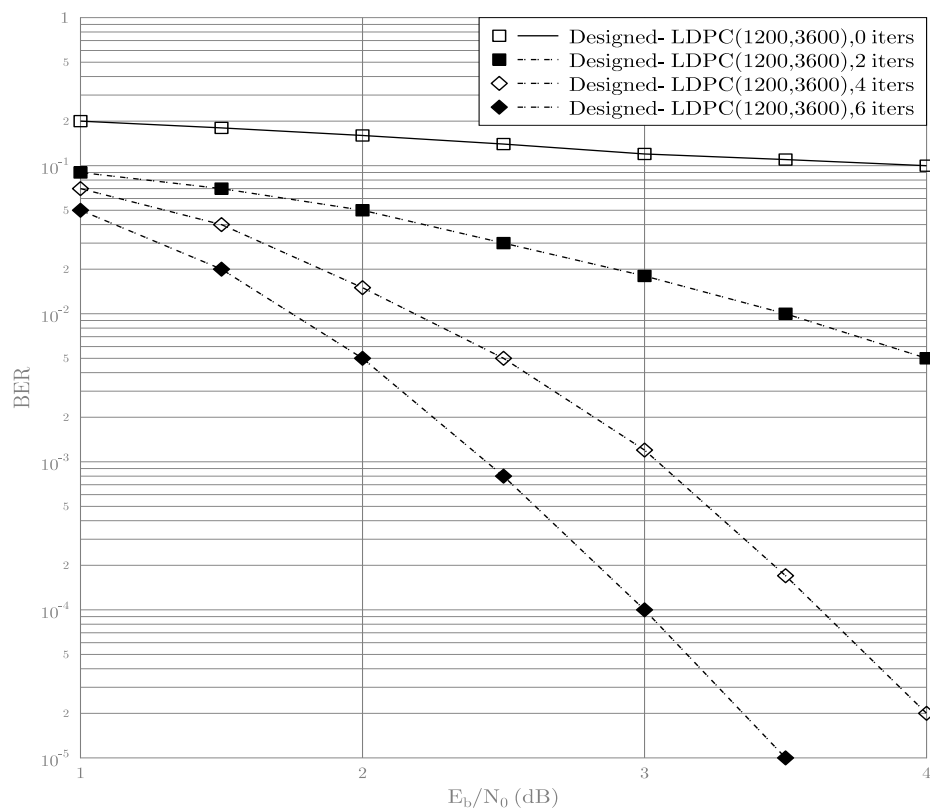
Khả năng sửa lỗi của mã LDPC(1200,3600) có hàm phân bố mật độ cho trong phương trình (2.5) và $t > 1$, phụ thuộc vào số lần lặp giải mã.

Bảng 2.2: Thông số mã LDPC được sử dụng trong mô phỏng .

Các thông số trong phương trình (2.5)	$\delta = 0.5, c = 0.1$
Tổng số bit thông tin	1200 · 1000 bit
Mã LDPC(1200,2400)	có $t=2$
Mã LDPC(1200,3600)	có $t=2$
Mã LDPC(1200,1800)	có $t=3$
Tỉ lệ mã LDPC tương ứng	1/3, 1/2, 2/3
Số lần lặp cực đại	20
Kiểu điều chế	QPSK
Kênh truyền	AWGN, pha đỉnh Rayleigh không tương quan

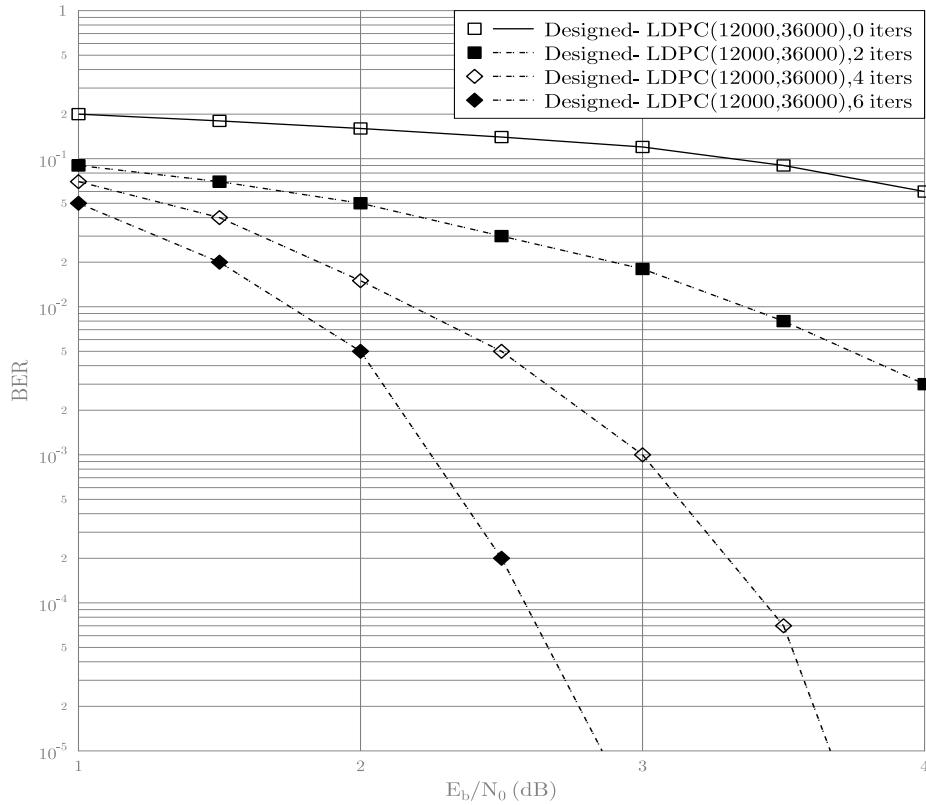
Ví dụ giữa số lần lặp giải mã là 4 và 6, khoảng cách giữa hai giá trị tỉ số E_b/N_0 yêu cầu để đạt được giá trị BER ở đầu ra $< 10^{-5}$ là 1 dB. Hay nói cách khác, với 6 lần lặp giải mã, mã LDPC(1200,3600) lợi hơn 1 dB giá trị E_b/N_0 so với sử dụng 4 lần lặp giải mã. Ta có thể thấy với thông số $t = 2$, mã LDPC(1200,3600) đạt giá trị BER= 10^{-5} tại $E_b/N_0 = 3,5$ dB.

Khi tăng kích thước ma trận sinh và ma trận kiểm tra của mã LDPC lên đến LDPC(12.000, 36.000) có thể thấy khả năng sửa lỗi của mã LDPC cũng tăng lên. Như có thể nhận thấy trong hình 2.10, mã LDPC(12.000, 36.000) đạt tỉ số lỗi bit BER $< 10^{-5}$ tại giá trị $E_b/N_0=2,8$ dB lợi 0,7 dB so với mã LDPC(1200,3600) có kích thước ma trận sinh và ma trận kiểm tra nhỏ hơn.



Hình 2.9: Mã LDPC(1200,3600) có hàm phân bố mật độ cho trong phương trình (2.5) và $t = 2$

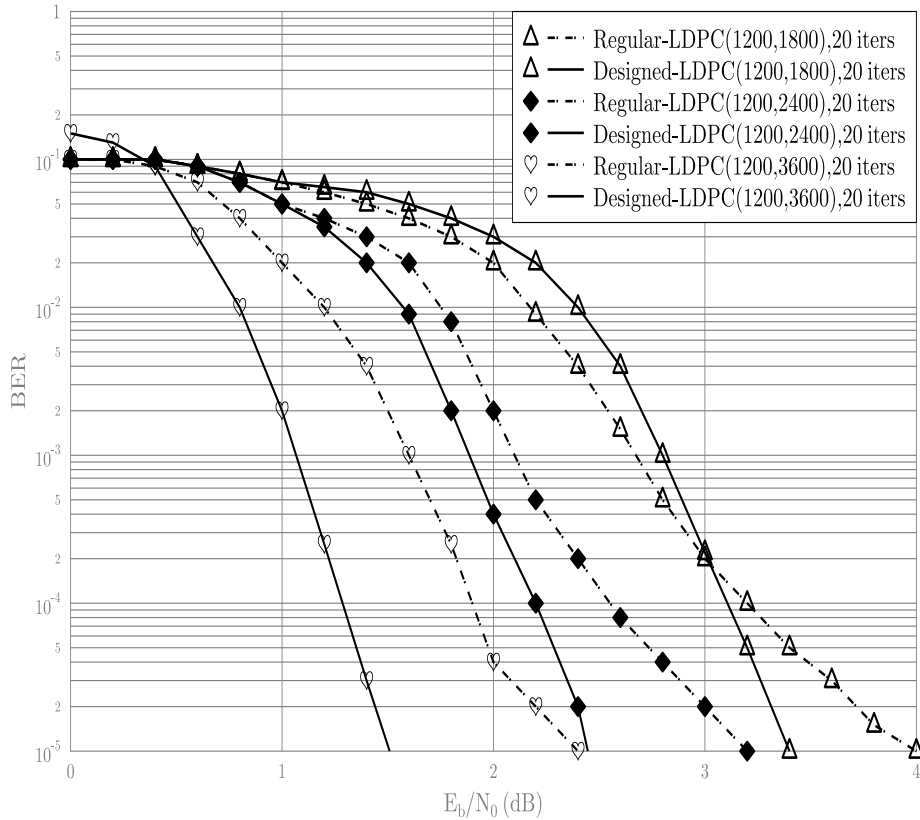
Phân tích khả năng sửa lỗi của mã LDPC có hàm phân bố mật độ cho trong phương trình (2.5) với các thông số $t > 1$ và các tỉ lệ mã khác nhau cho trong bảng 2.2.



Hình 2.10: Mô phỏng khả năng sửa lỗi của mã LDPC khi tăng kích thước ma trận sinh và ma trận kiểm tra lên 10 lần

Hình 2.11 là mô phỏng so sánh khả năng sửa lỗi của các mã LDPC có hàm phân bố mật độ cho trong phương trình (2.5) và mã LDPC có hàm phân bố mật độ của ma trận sinh là đồng đều. Qua quan sát kết quả mô phỏng trong hình 2.11, với kênh truyền chịu tác động tạp nhiễu AWGN, sử dụng kiểu điều chế QPSK ta có thể nhận xét: Các mã LDPC có hàm phân bố mật độ cho trong phương trình (2.5) có đường cong đồ thị quan hệ BER và E_b/N_0 là đường cong liền nét như trong hình vẽ 2.11, đạt giá trị tỉ lệ lỗi bit BER $< 10^{-5}$ tại E_b/N_0 thấp hơn khoảng

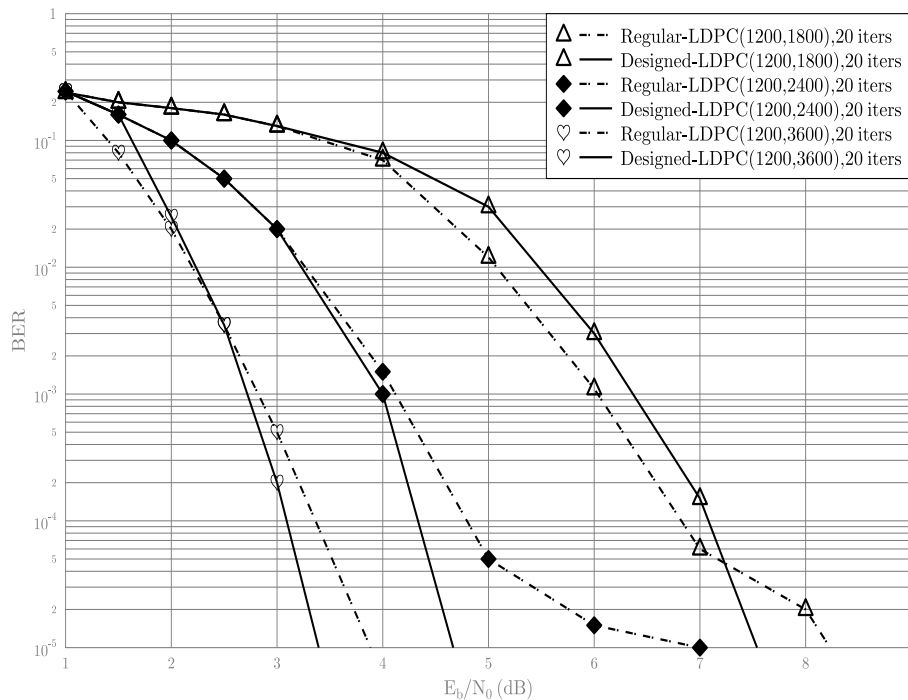
1 dB so với mã LDPC có hàm phân bố mật độ đồng đều có đường cong đồ thị là đường đứt nét.



Hình 2.11: So sánh khả năng sửa lỗi của các mã khi truyền qua kênh AWGN, sử dụng kiểu điều chế QPSK

Nói cách khác mã LDPC có hàm phân bố mật độ cho trong phương trình (2.5) có khả năng sửa lỗi tốt hơn so với mã LDPC có hàm phân bố mật độ đồng đều, khi truyền trong kênh có tạp nhiễu AWGN. Tương tự như vậy đối với kênh tạp nhiễu pha đỉnh Rayleigh không tương quan, các mã LDPC có hàm phân bố mật độ cho trong phương trình (2.5) đạt cùng giá trị BER $< 10^{-5}$ ở các giá trị BER thấp hơn so với mã LDPC

có hàm phân bố mật độ đồng đều, như được vẽ trong hình 2.12.



Hình 2.12: So sánh khả năng sửa lỗi của các mã khi truyền qua kênh Rayleigh không tương quan, sử dụng kiểu điều chế QPSK

2.4. Kết luận chương 2

Mã LDPC có hàm phân bố $\lambda(x)$ và $\nu(x)$ tương ứng với cột và hàng của ma trận sinh thành phần \mathbf{G} được thiết kế trong chương này, có khả năng sửa lỗi tốt hơn 1 dB so với mã LDPC có hàm phân bố mật độ đồng đều được thiết kế trong [115, 123].

Hơn nữa mã LDPC được thiết kế có ma trận sinh được suy trực tiếp từ ma trận kiểm tra của mã, mà không cần thiết phải thực hiện các phép tính toán phức tạp để tính ma trận sinh từ ma trận kiểm tra như đối

với các mã LDPC thông thường.

Khả năng sửa lỗi của mã LDPC được thiết kế trong chương 2 sẽ được thể hiện rõ hơn khi nó được sử dụng trong các hệ thống thông tin tích hợp mã. Trong chương 3, sẽ xây dựng các hệ thống tích hợp mã LDPC có ma trận sinh và ma trận kiểm tra được đề xuất và thiết kế trong chương 2 và thực hiện phân tích và so sánh với các mô hình tích hợp mã khác để làm rõ những ưu điểm của phương án đề xuất.

Chương 3

XÂY DỰNG CÁC HỆ THỐNG TÍCH HỢP MÃ LDPC

Mở đầu

Ở chương 2 luận án đã tiến hành xây dựng mã LDPC có khả năng sửa lỗi tốt hơn so với các mã LDPC thông dụng khác khi hoạt động độc lập. Để đánh giá được khả năng sửa lỗi của mã LDPC trong các hệ thống tích hợp mã, đồng thời để xây dựng, tối ưu các mô hình hệ thống thông tin tích hợp mã nhằm tăng cao hiệu ích của hệ thống khi sử dụng mã LDPC đã được thiết kế trong chương 2, tác giả đề xuất hai hệ thống tích hợp mã LDPC đó là hệ thống V-BLAST và hệ thống lai ghép hồi đáp H-ARQ.

3.1. Hệ thống thông tin

3.1.1. Hệ thống thu phát phân tập MIMO

Hệ thống MIMO có thể được phân chia làm hai loại tăng ích:

- Tăng ích ghép kênh theo không gian: cung cấp thông lượng¹ truyền dẫn cao nhất.
- Tăng ích theo phân tập: mục đích tối thiểu xác suất lỗi P_e , cho phép chống lại pha đình và tăng khả năng độ tin cậy, chất lượng dịch vụ QoS của hệ thống.

Tùy thuộc vào mục đích thiết kế, ta có thể xây dựng các hệ thống thỏa mãn một trong hai tiêu chí hay phối hợp hài hòa giữa hai tiêu chí.

3.1.1.1. Mô hình hệ thống phân tập

- Mỗi cặp ăng ten thu và phát tạo ra một đường truyền dẫn thẳng từ máy phát đến máy thu. Bằng cách gửi cùng một thông tin trên nhiều đường truyền dẫn thẳng giữa máy phát và thu khác nhau, nhờ đó máy thu có thể thu được nhiều bản sao của symbol dữ liệu bị can nhiễu pha đình bởi kênh truyền khác nhau. Vì vậy độ tin cậy thu tăng lên.
- Nếu độ tăng ích phân tập là d có nghĩa là trong khoảng tỉ số S/N cao, xác suất lỗi suy giảm với tốc độ $\frac{1}{(S/N)^d}$ so với tốc độ suy giảm $\frac{1}{(S/N)}$ của hệ thống đơn máy thu-phát SISO.
- Độ tăng ích phân tập cực đại d_{max} là số các đường tín hiệu độc lập tuyến tính tồn tại giữa máy phát và máy thu.
- Một hệ thống (M_R, M_T) sẽ có tổng số đường truyền dẫn từ máy thu đến máy phát là $M_R.M_T$, với điều kiện:

¹Thông lượng có ích được định nghĩa là tỉ số giữa các bit thông tin đúng trên tổng số các bit được truyền.

$$1 \leq d \leq d_{max} = M_R \cdot M_T \quad (3.1)$$

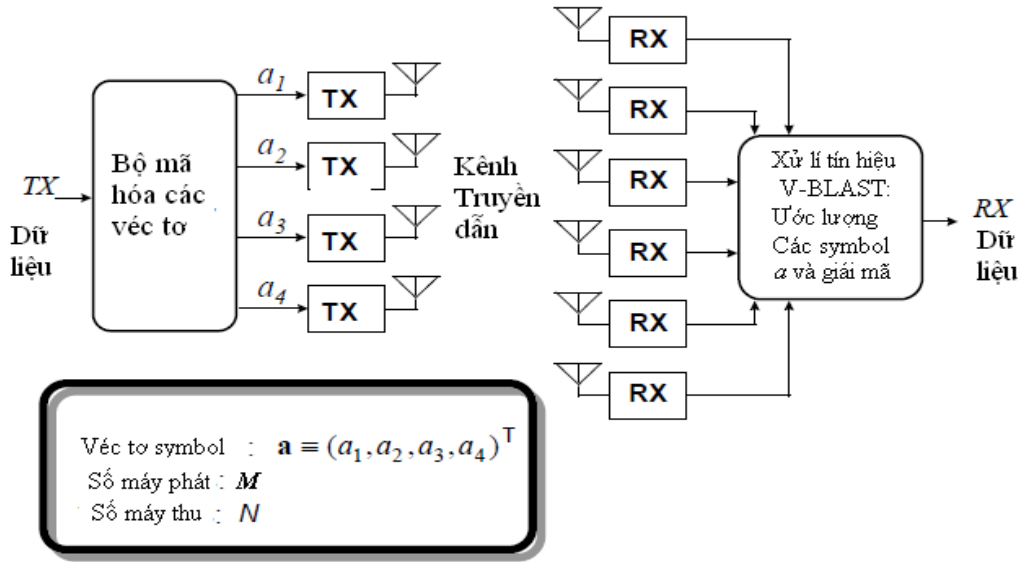
- Một hệ thống có độ tăng ích phân tập càng lớn thì xác suất lỗi P_e càng nhỏ.

3.1.1.2. Mô hình hệ thống ghép kênh theo thời gian

Giả sử các kênh truyền đơn của hệ thống MIMO M_T, M_R tạo ra $m = \min(M_T, M_R)$ kênh truyền dẫn độc lập tuyến tính SISO giữa máy phát và máy thu. Chúng ta có thể truyền một lượng m symbols dữ liệu khác nhau tại bất cứ thời điểm nào.

Hệ thống V-BLAST [124] là một mô hình cụ thể của hệ thống đa đầu vào ra (MIMO) bao gồm nhiều bộ thu phát có ăng ten riêng biệt được thiết kế trên các lớp khác nhau trong cùng một hệ thống truyền dẫn. Mô hình này ban đầu được thiết kế trong phòng thí nghiệm của tập đoàn Alcatel [124]. Mục đích của hệ thống V-BLAST nhằm tăng khả năng thông lượng kênh truyền bằng cách phân chia dòng dữ liệu đầu vào hệ thống thành M dòng dữ liệu thứ cấp và mỗi dòng dữ liệu này được mã hóa thành các symbol, sau đó các symbol này được truyền đến các máy phát tương ứng như biểu diễn trong hình 3.1. Các máy phát từ 1 đến M phát trên các kênh kề nhau, với tốc độ symbol là $1/T$ symbols/sec, trong đó T là thời gian hữu ích của một symbol. Các máy phát này được đồng bộ thời gian với nhau. Mỗi máy phát được coi là một máy phát điều chế pha biên độ thông thường (QAM). Giả sử chế độ điều chế cho các dòng dữ liệu thứ cấp tới các máy phát là như nhau và có kích thước là L symbols. Công suất của mỗi máy phát trong hệ thống là $1/M$, điều này có nghĩa là tổng công suất của hệ thống không phụ thuộc vào số

lượng máy phát trong hệ thống.



Hình 3.1: Hệ thống V-BLAST

Tổng số $1 - N$ máy thu là các máy thu QAM tiêu chuẩn. Các máy thu này cũng hoạt động trên các kênh liền kề. Mỗi máy thu thu M tín hiệu từ các máy phát. Ma trận hàm kênh truyền $\mathbf{H}^{N \times M}$, trong đó h_{pq} là độ lợi kênh truyền từ máy phát thứ q sang máy phát thứ p và $M \leq N$. Phía máy thu sử dụng kỹ thuật tách sóng tín hiệu V-BLAST nhằm giảm thiểu can nhiễu giữa các kênh kề phía phát, thu và tạp nhiễu pha định do kênh truyền gây nên.

3.1.1.3. Kỹ thuật tách sóng V-BLAST

Trong kỹ thuật này, quá trình tách sóng một véc tơ symbol cơ bản được coi như là quá trình rời rạc hóa theo thời gian, với giả thiết các véc tơ symbol đồng bộ với máy thu tuyệt đối về thời gian và tốc độ lấy mẫu.

Gọi véc tơ $a = (a_1, a_2, \dots, a_M)^T$ là véc tơ symbol được truyền, ở phía thu véc tơ symbol thu được tương ứng là:

$$r_{(N \times 1)} = \mathbf{H}_{(N \times M)} \mathbf{a}_{(M \times 1)} + \nu_{(N \times 1)} \quad (3.2)$$

trong đó ν là tạp âm. Một phương pháp để thực hiện quá trình tách sóng là kĩ thuật sử dụng mảng ăng ten thu thích nghi (AAA). Các giá trị ước lượng đầu ra bộ tách sóng y_k được quyết định như sau:

$$\hat{a}_{k1} = Q(y_{k1}) \quad (3.3)$$

trong đó $Q(\cdot)$ là lượng tử hóa.

3.1.1.4. Mô hình phân tập Alamouti

Là hệ thống có cấu trúc hệ ăng ten phân tập $(2, M_R)$ [125], trong đó tín hiệu phát được mã hóa theo qui luật sau:

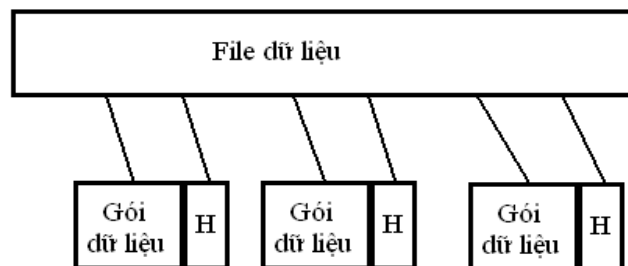
$$\begin{bmatrix} x_1 & -x_2^* \\ x_2 & x_1^* \end{bmatrix}$$

trong đó x^* là liên hợp phức của x .

- Mô hình thu phát này dễ thực hiện.
- Mô hình phân tập không gian thu phát vì số phần tử ăng ten phát bằng 2 và mô hình phân tập về thời gian vì khoảng thời gian truyền dẫn lớn hơn 2 lần chu kì symbols.
- Máy thu sử dụng cả hai phương pháp tách sóng kết hợp và ước lượng tỉ số cực đại logarit xác suất LLR .
- Mô hình này được sử dụng rộng rãi trong hệ thống CDMA 2000, WCDMA và IEEE 802.16-2004 OFDM-256 [126–128]

3.1.2. Hệ thống thông tin hỏi đáp ARQ

Song song với việc sử dụng mã kênh trong các hệ thống thông tin số, mô hình sử dụng hệ thống ARQ đã được thực hiện trong các hệ thống mạng thông tin Internet từ những năm 1984 [129]. Trong giao thức mạng ARQ [130, 131], nguồn tín hiệu được chia thành nhiều gói dữ liệu, mỗi gói dữ liệu chứa thông tin phần đầu H , như vẽ trong hình 3.2. Các gói dữ liệu thu được ở phía thu sẽ được kiểm tra bằng mô hình kiểm tra mã dư thừa có chu kỳ CRC (Cyclic Redundancy Checking), để phát hiện các lỗi trong gói thu được. Khi máy thu phát hiện lỗi, nó sẽ gửi tín hiệu qua kênh hồi tiếp đến máy phát để yêu cầu máy phát phát lại gói bị lỗi.



Hình 3.2: Mô hình hệ thống thông tin hỏi đáp ARQ

Có ba mô hình ARQ cơ bản sử dụng các giao thức khác nhau, đó là: Dừng và chờ [132], Quay lại N bước [133] và giao thức lặp có lựa chọn [130].

Giao thức Dừng và chờ [132] được xây dựng như sau:

- Giao thức Dừng và chờ là giao thức chính của giao thức Internet nguyên bản [131], trong đó bên phát phát một gói dữ liệu tại một thời điểm và bên thu gửi tín hiệu xác nhận ACK (ACKnowledgment), bất kì khi nào nó nhận được một gói dữ liệu đúng. Bên

phát sau đó sẽ phát lại gói dữ liệu vừa phát sau một khoảng thời gian mặc định, nếu như nó không nhận được tín hiệu ACK từ máy thu ngoại vi, như có thể thấy trong hình 3.3.

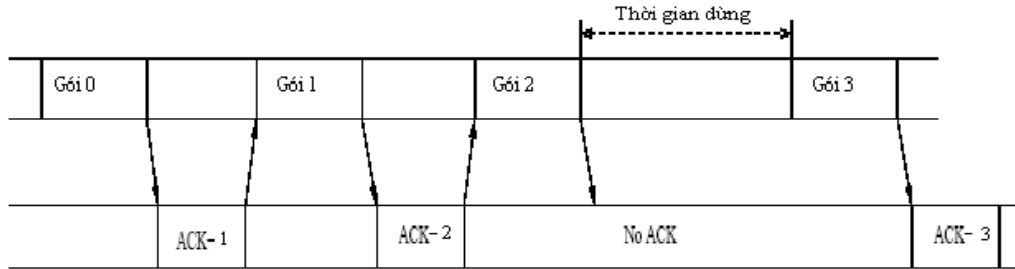
- Máy phát sử dụng các số thứ tự SN (Sequence Numbers) để đánh số các gói dữ liệu phát và máy thu sử dụng các số yêu cầu $ReqN$ (Request Numbers) để giám sát các gói thu được. Ví dụ, máy thu yêu cầu phát lại gói dữ liệu thứ $(k - 1)$ bằng cách gửi thông tin $ReqN = k$ tới máy phát.
- Máy phát sẽ tạm ngừng để chờ tín hiệu ACK, như có thể thấy trong hình 3.4.
- Giao thức này không yêu cầu bộ nhớ cho việc lưu trữ các gói dữ liệu.
- Tuy nhiên, giao thức này trở nên không hiệu quả, khi thời gian trễ tích lũy dài hơn thời gian một gói dữ liệu.
- Hiệu quả E của giao thức này bị giới hạn bởi đường trề hình tròn và có thể biểu diễn bằng công thức từ hình 3.4 như sau:

$$E = \frac{D_{TP}}{S} \quad (3.4)$$

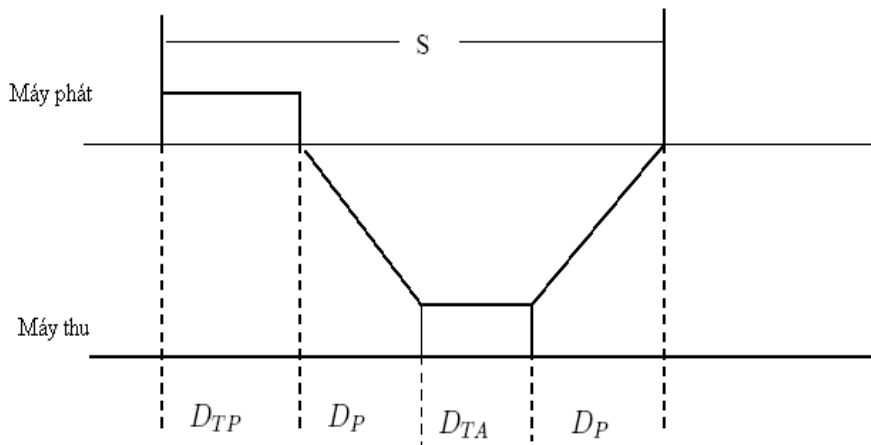
trong đó $[S = D_{TP} + 2.D_p + D_{TA}]$ là thời gian trề tương đối, D_{TA} là khoảng thời gian truyền thông tin ACK và D_{TP} là khoảng thời gian truyền một gói dữ liệu.

Giao thức quay lại N bước được mô tả như sau [133]:

- Giao thức này sử dụng cơ chế dựa trên cửa sổ thời gian, trong đó máy phát gửi các gói dữ liệu trong một cửa sổ thời gian, cho phép



Hình 3.3: Giao thức Dừng và chờ



Hình 3.4: Thời gian phân bố trong giao thức Dừng và chờ

truyền những gói dữ liệu mới trước một số các gói được xác nhận. Hình 3.5 vẽ quá trình truyền dẫn của giao thức quay lại N bước.

- Nếu kích thước của cửa sổ thời gian là N gói, thì máy phát không thể phát gói thứ $l + N$, cho đến khi nó nhận được ACK của gói thứ l .
- Máy thu hoạt động như trong giao thức Dừng và chờ và nó không chấp nhận các gói nằm ngoài chuỗi trật tự ban đầu. Khi máy thu gửi thông tin yêu cầu $RN = (l + 1)$, thông tin nhận biết này được gửi đến tất cả các gói nằm trong cửa sổ thời gian này.
- Giao thức quay lại N bước yêu cầu kích thước cửa sổ thời gian

là N tương đối cao để cho phép quá trình truyền dẫn là liên tục, trong khi chờ đợi thông tin ACK cho gói thu đúng trong khoảng thời của cửa sổ, trong đó N phải thỏa mãn:

$$N > S/D_{TP}. \quad (3.5)$$

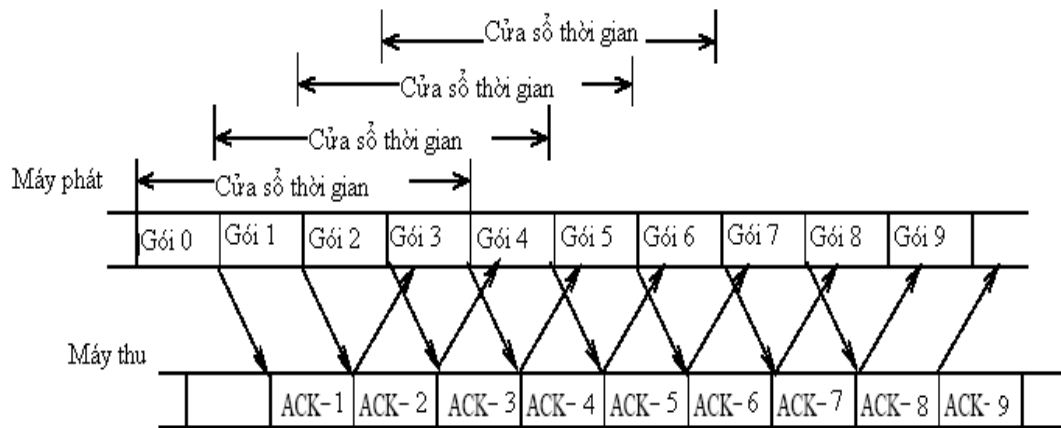
- Giao thức quay lại N bước được vẽ trong hình 3.5.
- Trong trường hợp không tính đến lỗi truyền dẫn, hiệu quả của giao thức Quay lại N bước được tính như sau:

$$E = \min\{1, N \times D_{TP}/S\}. \quad (3.6)$$

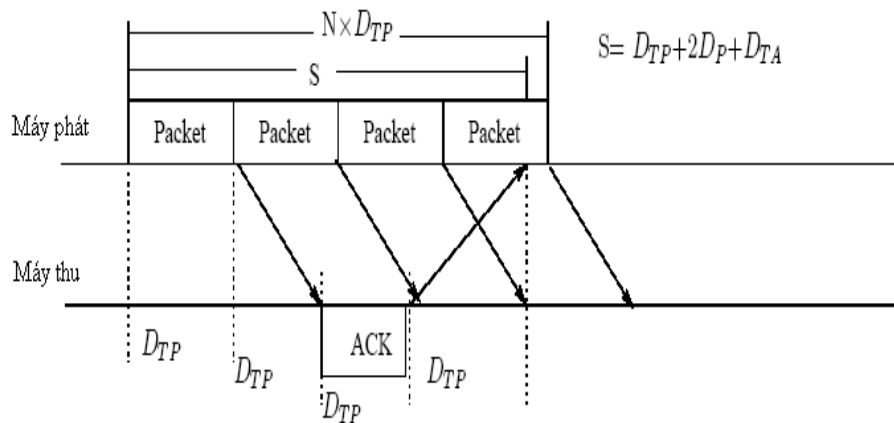
- Giao thức này không yêu cầu bộ nhớ đệm các gói dữ liệu tại máy thu.
- Máy phát phải lưu trữ đệm đến N gói dữ liệu trong khi chờ thông tin ACK của các gói này.
- Khi xảy ra lỗi, máy phát phải phát lại tất cả các gói trong khoảng thời gian cố định của cửa sổ thời gian.
- Máy thu chỉ nhận các gói dữ liệu theo đúng trật tự ban đầu của chúng. Nói cách khác, máy thu không thể nhận gói dữ liệu thứ $(l + 1)$ trước gói dữ liệu thứ l .

Cuối cùng giao thức phát lại có lựa chọn được mô tả như sau [130]:

- Máy phát chỉ cố gắng phát lại các gói bị mất do lỗi.
- Máy thu phải chấp nhận thu các gói không theo thứ tự ban đầu. Do đó, đòi hỏi máy thu phải có bộ nhớ đệm.



Hình 3.5: Giao thức quay lại N bước



Hình 3.6: Phân bố thời gian trong giao thức quay lại N bước

- Máy thu sẽ gửi thông tin xác nhận cho từng gói thu được không bị lỗi, các gói dữ liệu không được xác nhận trước trong khoảng thời gian xác định được coi là bị mất hoặc bị lỗi.
- Máy phát có thể gửi thông tin không nhận được NAK để yêu cầu máy phát phát lại từng gói dữ liệu.
- Giao thức dựa trên cửa sổ thời gian giống như giao thức quay lại N bước và độ rộng của sổ là W .
- Trong trường hợp lý tưởng, chỉ có các gói thu được mà bị lỗi sẽ

được phát lại, do đó hiệu quả của giao thức này có thể tính như sau:

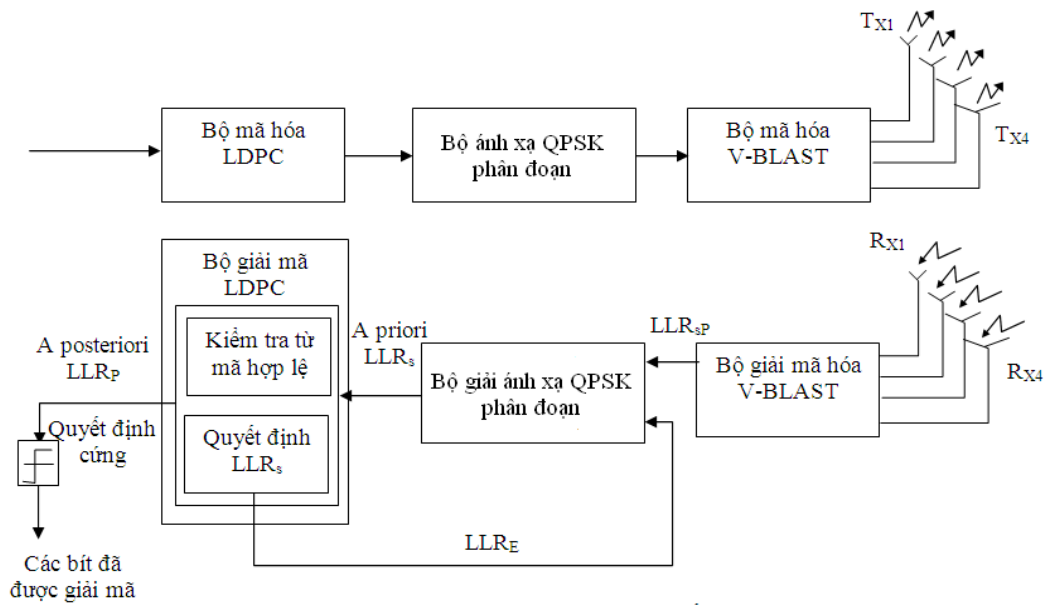
$$E = 1 - P; \quad (3.7)$$

trong đó P là xác suất lỗi gói.

3.2. Hệ thống tích hợp mã hóa LDPC – V-BLAST

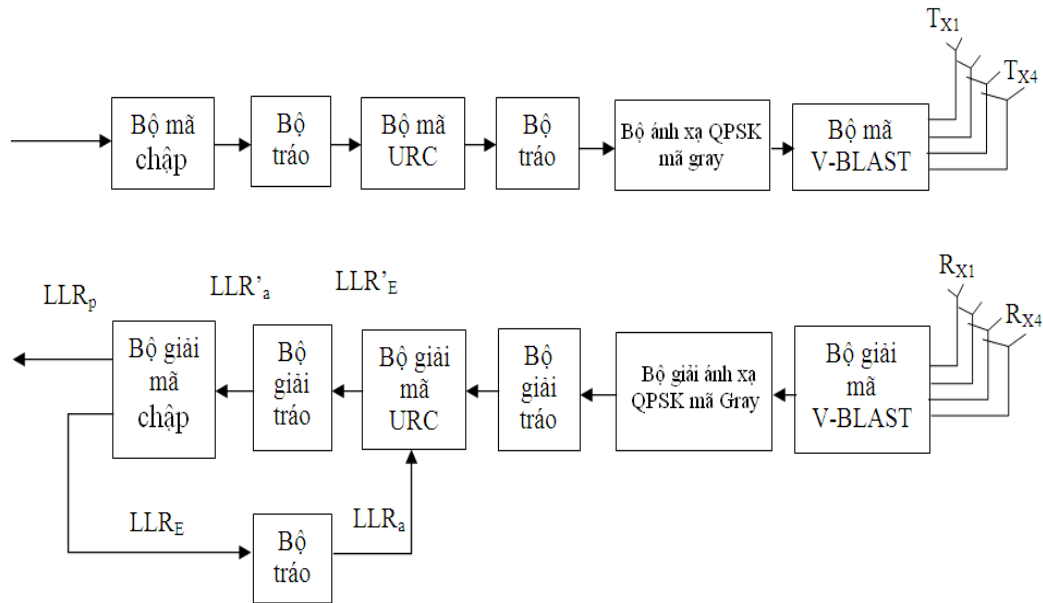
Mô hình hệ thống tích hợp V-BLAST và mã LDPC được vẽ trong hình 3.7 dưới đây. Hệ thống này gồm có bốn ăng ten phát và bốn ăng ten thu. Tại phía phát, các bit thông tin được mã hóa bằng mã LDPC thiết kế trong chương 2, sau đó được ánh xạ thành các symbol QPSK tại bộ điều chế sử dụng chế ánh xạ phân đoạn [134]. Các symbol này sau đó được mã hóa bằng bộ mã hóa V-BLAST và truyền dẫn bằng 4 ăng ten qua kênh tạp nhiễu Rayleigh băng hẹp có tương tác và tần số Doppler chuẩn hóa bằng 0,01. Tại phía máy thu, các tín hiệu thu được được giải mã bằng bộ giải mã V-BLAST. Đầu ra bộ giải mã V-BLAST được chuyển sang bộ giải điều chế QPSK, bộ này đồng thời nhận thông tin tiên nghiệm từ bộ giải mã LDPC. Các giá trị $LLRs$ tại đầu ra bộ giải điều chế QPSK sau đó lại được chuyển tới bộ giải mã LDPC và được coi là thông tin tiên nghiệm. Quá trình giải mã LDPC được hỗ trợ bởi khối kiểm tra từ mã hợp lệ như vẽ trong hình 3.7. Các giá trị $LLRs$ đầu ra của bộ giải mã LDPC tương ứng với các giá trị thông tin ngoại lai $LLRs$ đầu ra, khi và chỉ khi bộ kiểm tra từ mã đúng của bộ giải mã LDPC xác định từ mã \mathbf{C} là hợp lệ bằng phương trình kiểm tra mối quan hệ $[\mathbf{S} = \mathbf{C} \cdot \mathbf{H}^T = \mathbf{0}]$. Tuy nhiên, nếu giá trị $\mathbf{S} \neq \mathbf{0}$, thì các $LLRs$ đầu

ra của bộ giải mã LDPC không còn tiếp tục được coi là thông tin ngoại lai $LLRs$ nữa. Thay vào đó, chúng đóng vai trò là thông tin ngoại lai, được tính toán bằng hiệu giữa thông tin đầu ra ngoại lai và thông tin tiên nghiệm đầu vào.



Hình 3.7: Mô hình tích hợp mã LDPC và hệ thống V-BLAST

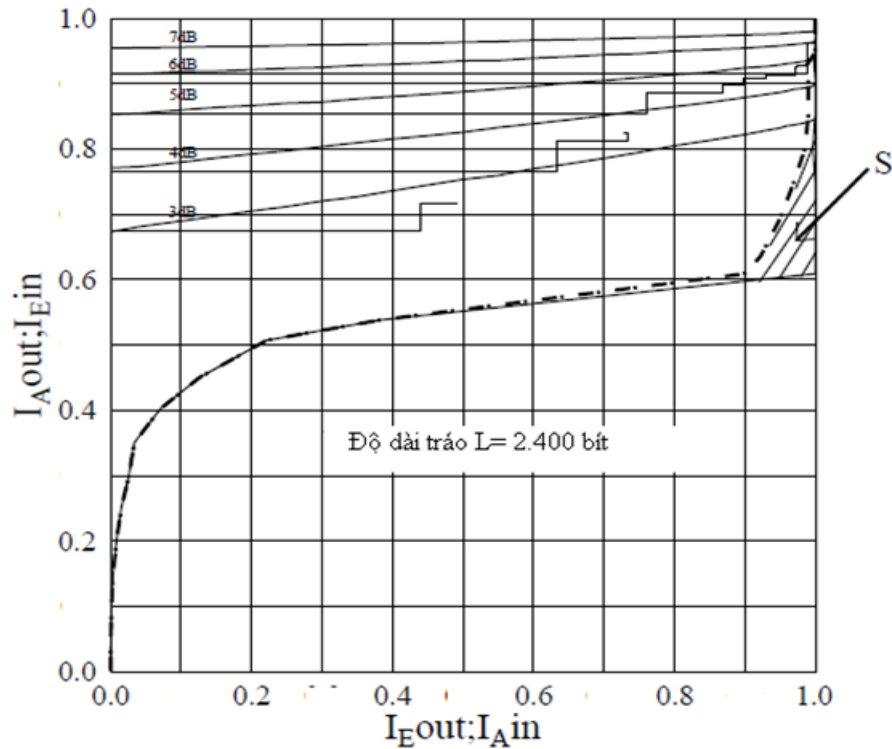
Như trong hình 3.7, các giá trị $LLRs$ đầu ra bộ giải mã LDPC được hồi tiếp về đầu vào bộ giải điều chế QPSK và được coi là thông tin tiên nghiệm $LLRs$. Quá trình giải mã được thực hiện bằng cách lặp hồi tiếp liên tục các thông tin ngoại lai giữa hai bộ giải mã LDPC và bộ giải điều chế QPSK, cho đến khi tất cả từ mã LDPC thu được là hợp lệ hoặc số lần lặp đạt giá trị cực đại đã được xác định từ trước. Trong lần lặp cuối cùng, các giá trị $LLRs$ ở đầu ra bộ giải mã LDPC được truyền đến khối quyết định cứng để khôi phục lại các bit thông tin tiên nghiệm.



Hình 3.8: Mô hình tích hợp mã RSC-URC và hệ thống V-BLAST

Mô hình tích hợp mã kênh khác và hệ thống V-BLAST được đề cập trong chương này là mô hình ưu việt [124, 135, 136], đó là mô hình tích hợp mã chập đệ quy RSC được sử dụng làm mã ngoài và mã có tỉ lệ mã bằng một URC đóng vai trò là mã trong, như vẽ trong hình 3.8. Lợi ích của việc sử dụng mã RSC tích hợp với mã URC là đáp ứng xung của hệ thống có một phần tử nhớ vô hạn, nó sẽ hỗ trợ hệ thống khai thác một cách hiệu quả các thông tin ngoại lai, thậm chí khi sử dụng hệ thống tráo có thời gian trễ cực ngắn. Trong hệ thống ưu việt này, các bit thông tin tiền nghiệm được mã bằng bộ mã hóa chập và bộ mã hóa URC, trước khi được ánh xạ bằng bộ điều chế QPSK bằng phương pháp ánh xạ Gray và cuối cùng được phát đi bằng hệ thống V-BLAST như trong sơ đồ tích hợp với mã LDPC trên hình 3.7. Máy thu của hệ thống này thực hiện quá trình giải mã lặp giữa bộ giải mã RSC và bộ giải mã URC.

Phân tích hệ thống bằng đồ thị EXIT

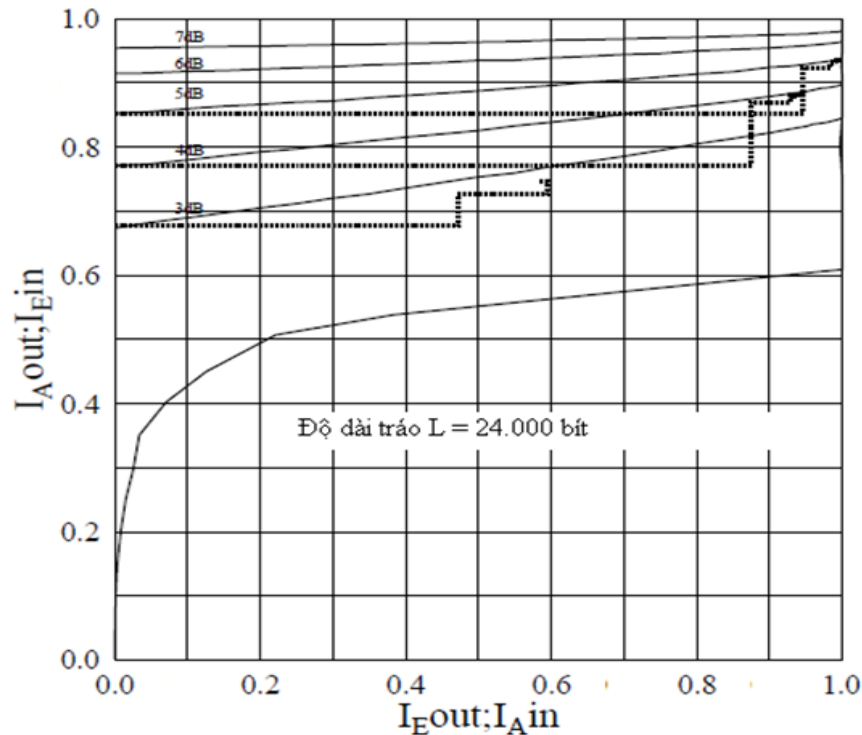


Hình 3.9: Đồ thị EXIT của hệ thống tích hợp mã LDPC và V-BLAST với độ dài tráo $L=2.400$ bit

Trong sơ đồ hệ thống tích hợp mã LDPC và V-BLAST vẽ trong hình 3.7, bộ mã hóa LDPC đóng vai trò bộ mã ngoài và bộ điều chế QPSK với phân bố theo kiểu phân đoạn đóng vai trò bộ mã trong. Quá trình giải mã lặp được thực hiện bằng cách trao đổi thông tin ngoại lai LLRs giữa bộ giải mã LDPC và bộ giải điều chế tại máy thu. Đồ thị EXIT của hệ thống này được thể hiện trong hình 3.9. Như trên hình 3.9, đồ thị EXIT của hệ thống bao gồm các đường cong đồ thị EXIT ngược của mã LDPC và các đường cong đồ thị EXIT của bộ giải điều chế tại các giá trị E_b/N_0 khác nhau. Độ dài tráo của hệ thống là 2400 bit. Độ dài tráo của hệ thống là 2400 bit. Kênh dẫn EXIT nằm giữa đường cong

đồ thị của bộ giải điều chế và mã LDPC là tương đối rộng tại giá trị E_b/N_0 , khi chúng ta sử dụng bộ ánh xạ với kiểu phân bố phân đoạn, nhưng các bậc của đường cong hội tụ không trùng khít với các đường cong đồ thị EXIT của bộ giải điều chế và bộ giải mã LDPC. Sự không trùng khít này xảy ra là do độ dài tráo quá ngắn, điều đó có nghĩa là phân bố của các $LLRs$ không phải là phân bố Gauss [82, 137]. Vì vậy, giá trị lỗi bit BER của hệ thống không trùng hợp với dự đoán của đồ thị EXIT. Ngược lại, đường cong hội tụ hoàn toàn trùng khít với các đường cong EXIT khi giá trị $E_b/N_0 > 6 \text{ dB}$. Qua quan sát hai đồ thị EXIT của mã LDPC được vẽ trong hình 3.9, một đường đồ thị được vẽ bằng nét liền và một được vẽ bằng nét đứt. Đường đồ thị vẽ bằng nét đứt tương ứng với đồ thị EXIT của mã LDPC không sử dụng khối kiểm tra từ mã hợp lệ như đã vẽ trong hình 3.7. Đường thứ hai tương ứng với đồ thị EXIT của mã LDPC có sử dụng bộ kiểm tra từ mã hợp lệ. Nói cách khác, việc sử dụng khối kiểm tra từ mã hợp lệ trong bộ giải mã LDPC làm khả năng sửa lỗi của hệ thống tăng lên và kích thước vùng mở S của đồ thị EXIT như vẽ trong hình 3.9 đặc trưng cho độ tăng ích BER đạt được.

Để tăng khả năng sửa lỗi của hệ thống tích hợp LDPC và V-BLAST, chúng ta tăng độ dài tráo lên 24.000 bit. Hệ thống lúc này có thể đạt hội tụ tại giá trị $E_b/N_0 = 5 \text{ dB}$ sau 2 lần lặp, như vẽ trong hình 3.10. Việc tăng độ dài tráo sẽ làm tăng khả năng sửa lỗi của hệ thống, nhưng bên cạnh đó độ phức tạp của hệ thống cũng tăng lên, điều này sẽ được phân tích trong mục tiếp theo.



Hình 3.10: Đồ thị EXIT của hệ thống tích hợp mã LDPC và V-BLAST với độ dài tráo $L=24.000$ bit

Phân tích khả năng sửa lỗi của hệ thống

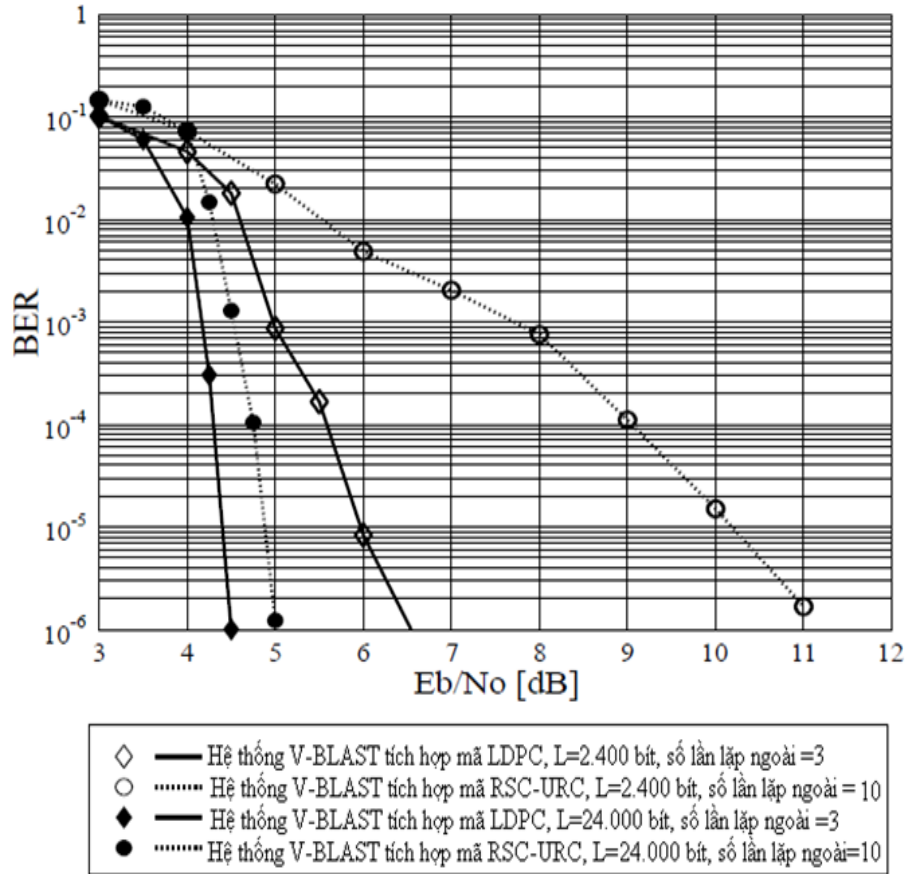
Trong mục này, tác giả so sánh khả năng sửa lỗi của hệ thống tích hợp mã LDPC với các hệ thống tích hợp mã RSC-URC thể hiện trong hình 3.8. Hệ thống tích hợp mã RSC-URC sử dụng mã RSC có hai phần tử nhớ tỉ lệ mã $1/2$, mã RSC có các đa thức sinh biểu diễn bằng hệ bát phân $G_r = 7$ và $G = 5$, trong đó G_r là kí hiệu cho đa thức sinh hàm hồi tiếp, còn G là kí hiệu đa thức sinh hàm truyền đạt. Mã URC có bộ nhớ là 1. Tất cả các thông số mô phỏng được liệt kê trong bảng 3.1. Hình 3.11 là các đồ thị quan hệ BER và E_b/N_0 của hệ thống V-BLAST tích hợp mã LDPC và hệ thống tích hợp mã RSC-URC, khi độ dài tráo bằng $L= 2.400$ bit và $L= 24.000$ bit. Khi sử dụng độ dài

Bảng 3.1: Các thông số mô phỏng hệ thống tích hợp V-BLAST

Các thông số hàm phân bố mật độ của mã LDPC cho trong phương trình (2.5) của chương 2	$\delta = 0,5; c = 0,1; t = 2$
Tỉ lệ mã LDPC	1/2
Số lần lặp cực đại của mã LDPC	30
Thông số mã RSC	$G_r = 7; G = 5$
Kiểu điều chế cho hệ thống V-BLAST tích hợp mã LDPC	QPSK- ánh xạ phân đoạn
Kiểu điều chế cho hệ thống V-BLAST tích hợp mã RSC-URC	QPSK- Gray
Thông số hệ thống V-BLAST	
Số ăng ten phát	4
Số ăng ten thu	4
Độ dài tráo	2.400 bit; 24.000 bit

tráo $L = 2.400$ bit, hệ thống V-BLAST tích hợp mã LDPC đạt tỉ số BER $\leq 10^{-6}$ tại $E_b/N_0 = 6,5$ dB, trong khi đó hệ thống V-BLAST tích hợp mã RSC-URC yêu cầu tỉ số E_b/N_0 hơn 11 dB để đạt được cùng tỉ số lỗi bit BER. Khi tăng độ dài tráo lên 24.000 bit, như quan sát trong hình 3.11, đường cong quan hệ tỉ lệ lỗi bit BER và E_b/N_0 của hệ thống V-BLAST tích hợp mã RSC-URC được cải thiện nhiều hơn so với hệ thống V-BLAST tích hợp mã LDPC, tuy nhiên hệ thống V-BLAST tích hợp mã RSC-URC vẫn đòi hỏi tỉ số E_b/N_0 cao hơn 5 dB để đạt được tỉ số lỗi bit BER đầu ra $\leq 10^{-6}$. Trong khi đó, hệ thống V-BLAST tích hợp mã LDPC chỉ yêu cầu tỉ số $E_b/N_0 = 4,5$ dB để đạt được cùng tỉ lệ

lỗi bit trên.



Hình 3.11: Mô phỏng quan hệ BER và E_b/N_0 của các hệ thống V-BLAST tích hợp các mã kênh khác nhau

Sự khác nhau về độ phức tạp của hai hệ thống này chủ yếu là do hai bộ giải mã LDPC và bộ giải mã RSC-URC, khi tính toán bằng phương pháp đã được đề cập trong [138]. Ở đây, đã sử dụng thuật toán giải mã Maxlog-MAP [139] cho các bộ giải mã RSC-URC và thuật toán truyền thông tin giữa các nút kiểm tra và nút thông tin cho mã LDPC. Chúng ta hãy xem xét với trường hợp độ dài tráo $L = 2.400$ bit, tỉ lệ mã của cả mã LDPC và mã RSC là bằng $1/2$. Số lần lặp giải mã ngoài cực đại hệ thống V-BLAST tích hợp mã RSC-URC bằng $I_{RSC-URC_{max}} = 10$ lần,

trong khi đó $m_1 = 2, m_2 = 1$ là các bộ nhớ tương ứng với các bộ mã RSC và URC. Các thông số $i = 8$ và $j = 9$ là số các phần tử khác 0 trong hàng và cột của ma trận kiểm tra mã LDPC, sử dụng hàm phân bố mật độ cho trong phương trình (2.5) được nêu trong chương 2. Số lần lặp giải mã cực đại của mã LDPC là $I_{max-inner} = 30$ và số lần lặp ngoài là $I_{LDPC-outer} = 3$. Độ phức tạp Λ_{LDPC} của mã được tính bằng số phép tính toán học Cộng-so sánh- lựa chọn ASC (Add-Compare-Select) như sau [138]:

$$\begin{aligned} \Lambda_{LDPC} &= I_{LDPC-outer} \times (I_{max-inner} \times 4 \times i + I_{max-inner} \times j \times L) \\ &= 3 \times (30 \times 4 \times 8 \times 2400 + 30 \times 9 \times 2400) \\ &= 8.856.000 \text{ ACS}. \end{aligned} \quad (3.8)$$

Độ phức tạp của hệ thống V-BLAST tích hợp mã RSC-URC, kí hiệu là $\Lambda_{RSC-URC}$ được tính như sau [93]:

$$\begin{aligned} \Lambda_{RSC-URC} &= I_{RSC-URC_{outer}} \times L \times (10 \times 2^{m_1} + 10^{m_2} + 2 \times 4 \times 2^{m_1} + \\ &\quad + 2 \times 2^{m_2} + 2 \times 4 \times 2^{m_2} + 4^{m_1} - 2 + 4^{m_2} - 2) \\ &= 10 \times 2400 \times (10 \times 2^2 + 10 \times 2^1 + 2 \times 4 \times 2^2 + \\ &\quad + 2 \times 4 \times 2^1 + 4 \times 2^2 - 2 + 4 \times 2^1 - 2) \\ &= 3.072.000 \text{ ACS}. \end{aligned} \quad (3.9)$$

Từ các phương trình (3.8) và (3.9) chúng ta có thể tính tỉ số phức tạp Λ_{ratio} giữa các hệ thống V-BLAST tích hợp mã LDPC và hệ thống V-BLAST tích hợp mã RSC-URC như sau:

$$\Lambda_{ratio} = \frac{\Lambda_{LDPC}}{\Lambda_{RSC-URC}} = \frac{8.856.000}{3.072.000} = 2,88 \quad (3.10)$$

Số lần lặp giải mã sử dụng cho mã LDPC để đạt được tỉ số BER $\leq 10^{-5}$ phụ thuộc vào tỉ số E_b/N_0 và quá trình kiểm tra từ mã hợp lệ.

Nói cách khác, khi tỉ số E_b/N_0 tăng số lần lặp giải mã yêu cầu cho mã LDPC để đạt được từ mã hợp lệ giảm. Do đó, thực tế độ phức tạp của hệ thống tích hợp mã LDPC $\Lambda_{LDPC} \leq 2,8 \times \Lambda_{RSC-URC}$.

Thông qua việc mô phỏng so sánh hai hệ thống tích hợp mã ở trên, chúng ta có thể thấy hệ thống tích hợp mã LDPC được thiết kế trong chương 2 tốt hơn hẳn hệ thống tích hợp mã RSC-URC đã được biết đến trong [116, 140–143]. Cụ thể, với độ dài tráo ngẫu nhiên $L = 2.400$ bit hệ thống tích hợp LDPC lợi hơn 5 dB so với các hệ thống tích hợp mã RSC-URC với $L = 24.000$ bit độ tăng ích này vào khoảng 0,5 dB. Khi truyền dẫn tín hiệu qua kênh can nhiễu tương quan MIMO sử dụng các thông số trong bảng 3.1, độ phức tạp của hệ thống V-BLAST tích hợp mã LDPC cao hơn 2,88 lần độ phức tạp của hệ thống V-BLAST tích hợp mã RSC-URC. Dưới đây tác giả thiết kế một hệ thống thông tin Lai ghép- Tự động yêu cầu phát lại, gọi tắt là H-ARQ (Hybrid- Automatic Repeat reQuest) tích hợp với mã LDPC được thiết kế trong chương 2.

3.3. Hệ thống H-ARQ tích hợp mã LDPC

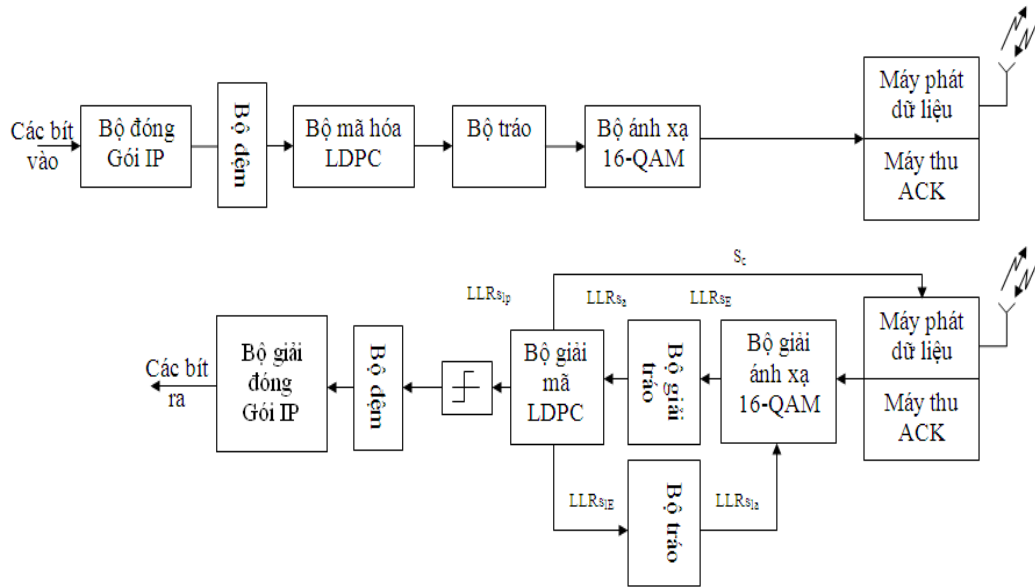
Trong mục này, đề xuất thiết kế mô hình H-ARQ dựa trên giao thức phát lại có điều kiện kết hợp với mã sửa lỗi trước FEC. Có hai loại mô hình H-ARQ cơ bản, đó là mô hình H-ARQ loại I và mô hình H-ARQ loại II [129, 144]. Máy phát của mô hình H-ARQ loại I phát lại cả phần thông tin và phần thông tin dư thừa để sửa lỗi của các gói dữ liệu bị lỗi, khi máy phát nhận được thông tin yêu cầu phát lại từ máy thu ngoại vi. Quá trình này được thực hiện ở lớp điều khiển truy cập thông tin MAC (Media Access Control). Thông thường, việc phát lại cả phần bit thông tin và bit kiểm tra sẽ gây ra sự suy giảm hiệu quả thông lượng truyền

dẫn. Do đó, hệ thống H-ARQ loại I [129] thường được thay thế bằng hệ thống H-ARQ loại II [144]. Trong hệ thống H-ARQ loại II, phần thông tin và phần kiểm tra được phát đi lần thứ nhất. Tuy nhiên, trong lần thứ hai truyền dẫn chỉ có phần bit kiểm tra được truyền thêm. Máy thu sẽ sử dụng phần kiểm tra trong tất cả các quá trình truyền dẫn để sửa lỗi thông tin thu được.

Hệ thống H-ARQ có thể kết hợp được cả kỹ thuật phát lại và các thuật toán sửa lỗi hay phát hiện lỗi nhằm mục đích tăng hiệu quả hoạt động của các hệ thống thông tin vô tuyến trong các kênh truyền bị ảnh hưởng tạp nhiễu [145, 146]. Để đạt được giá trị BER thấp, các hệ thống H-ARQ có sự trợ giúp của điều chế được thực hiện trong [147, 148], những hệ thống này liên quan đến nhiều kỹ thuật điều chế [149–151]. Quá trình giải mã lậ được thực hiện bằng cách trao đổi thông tin giữa bộ giải mã FEC và bộ giải ánh xạ bit sang symbol của bộ giải điều chế.

Các bộ mã LDPC của chương 2 đã tăng đáng kể khả năng sửa lỗi cho quá trình truyền dẫn qua cả kênh nhiễu và pha đình. Mô hình được đề xuất và thiết kế sau đây sẽ có khả năng sửa lỗi tốt hơn các mô hình H-ARQ tích hợp mã sửa sai khác đã được thực hiện trong [152, 153]. Ở đây đã tiến hành cải tiến phiên bản hệ thống H-ARQ loại II đã từng biết [154]. Trong hệ thống H-ARQ loại II nguyên bản, máy phát thực hiện quá trình phát lại bằng cách loại bỏ một số bit trong phần kiểm tra và sau đó từ từ phát lại các mẫu thông tin kiểm tra nhỏ, bất cứ khi nào máy phát nhận được yêu cầu gửi thêm thông tin dư thừa kiểm tra từ máy thu. Máy thu sau đó sẽ sử dụng cả phần thông tin kiểm tra đã thu được trước đó và phần thông tin kiểm tra hiện tại để tiếp tục lại quá trình giải mã, để có cơ hội giải khôi phục thành công dữ liệu ban đầu. Nếu như máy thu vẫn không thể khôi phục hoàn toàn phần thông

tin được truyền, quá trình phát lại sẽ được lặp lại đến khi nào số lần lặp vượt quá giá trị mặc định.



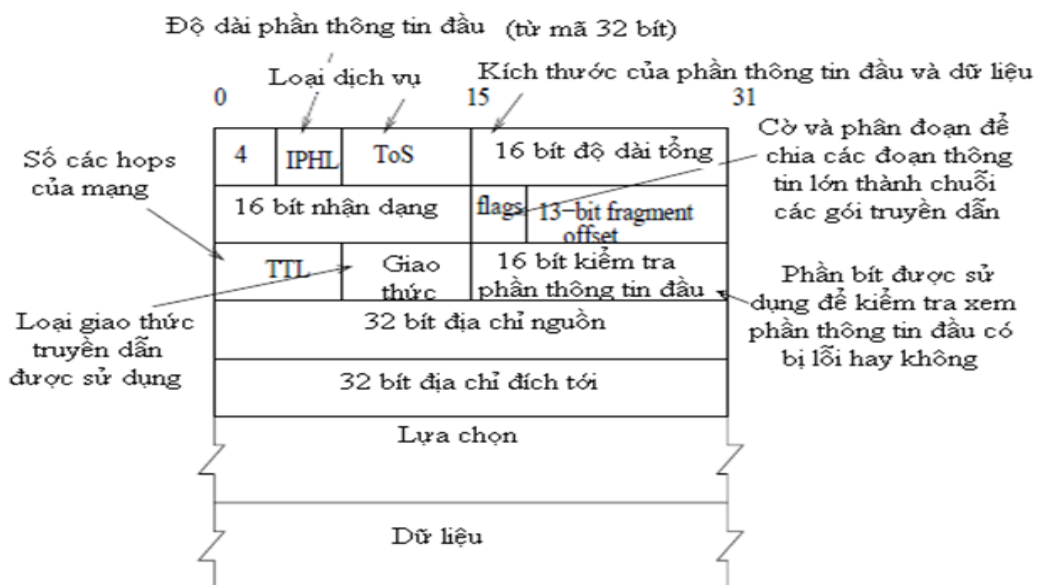
Hình 3.12: Sơ đồ khối bộ H-ARQ tích hợp LDPC có trợ giúp của bộ điều chế

Mô hình H-ARQ tích hợp mã LDPC đề xuất trong luận án được thực hiện như sau: Máy thu được tích hợp kỹ thuật kiểm tra từ mã hợp lệ. Khi đầu ra bộ giải mã LDPC là một từ mã không hợp lệ, máy thu sẽ yêu cầu máy phát phát lại một phần thông tin kiểm tra của từ mã không hợp lệ. Bộ giải mã LDPC vẫn giữ các giá trị $LLRs$ của các bit thông tin được tạo ra từ quá trình trao đổi thông tin giữa phần thông tin và phần kiểm tra trước đó của ma trận kiểm tra \mathbf{H} . Phần thông tin kiểm tra mới của ma trận kiểm tra \mathbf{H} tiếp tục thực hiện thuật toán trao đổi thông tin với sự trợ giúp của các giá trị $LLRs$ đã được cập nhật trước đó, cho đến khi đạt được từ mã hợp lệ hoặc số lần phát lại đạt giá trị cực đại mặc định.

Cấu trúc hệ thống H-ARQ tích hợp mã LDPC

Như trong hình 3.12 và 3.13, dữ liệu nguồn được đóng gói tại bộ đóng gói giao thức mạng Internet tại máy phát [53], trước khi đưa tới khối mã hóa LDPC. Một khung của K gói IP được lưu tại bộ đệm và sau đó đưa qua bộ mã hóa LDPC. Thực tế, do số các bit trong phần đầu gói IP là ít hơn rất nhiều so với phần dữ liệu, vì vậy tác giả sử dụng mã LDPC có tỉ lệ mã thấp để bảo vệ phần thông tin đầu này, mà không làm thay đổi nhiều kích thước phần đầu. Một từ mã LDPC bao gồm K bit của gói IP và M bit kiểm tra. Các từ mã LDPC tại đầu ra bộ mã hóa LDPC được thực hiện tráo bằng bộ tráo có độ dài tráo bằng n bit và sau đó được đưa tới bộ ánh xạ 4 bit cho một symbol của bộ điều chế 16-QAM, trong đó n là độ dài tổng của từ mã LDPC. Tác giả gọi mô hình nối tiếp như trên là mô hình H-ARQ tích hợp mã LDPC có sử dụng giải mã lặp giữa bộ ánh xạ và bộ giải mã LDPC.

Bộ ánh xạ của bộ điều chế 16-QAM có các kiểu ánh xạ khác nhau,



Hình 3.13: Cấu trúc gói IP

đó là ánh xạ theo mã Gray và ánh xạ theo từng phân đoạn [142, 155],

như có thể thấy trong hình 3.14a và 3.14b. Các bộ ánh xạ được thiết kế với mục đích để đạt được khoảng cách Euclid cực tiểu lớn nhất giữa các điểm pha có khoảng cách Hamming bằng 1 và giá trị trung bình nhỏ nhất số các điểm tín hiệu lân cận [143]. Quan sát đồ thị chòm sao, giá trị trung bình số các bit N_{min} khác nhau giữa hai điểm có pha gần nhau trong đồ thị chòm sao được tính như sau [143]:

$$N_{min} = \frac{1}{L \cdot N_0} \sum_{s \in S} \sum_{s' \in S_s} d_H(s, s') \quad (3.11)$$

trong đó s kí hiệu pha đang được quan sát trong tập tín hiệu S trong tọa độ hai chiều, s' kí hiệu pha của các tín hiệu lân cận nhau trong tập tín hiệu con S_s , $L = 2m$ kí hiệu số điểm pha trong đồ thị chòm sao, với m là số các bit tương ứng với một symbol trong đồ thị chòm sao và cuối cùng $d_H(s, s')$ kí hiệu khoảng cách Hamming giữa s và s' . Số các điểm pha lân cận gần nhất với điểm pha đang xét được tính như sau:

$$N_0 = \frac{1}{L} \sum_{s \in S} N_s, \quad (3.12)$$

trong đó N_s là số các điểm pha gần điểm pha đang xét s nhất, ví dụ như đối với kiểu ánh xạ mã Gray trong hình 3.14a, số điểm pha trong tập con là $N_{min} = 1$. Mỗi điểm trong đồ thị chòm sao tương ứng với 4 bit. Máy phát cũng được tích hợp máy thu tín hiệu ACK từ máy thu dữ liệu phát trở lại. Máy phát ACK tại phía máy thu dữ liệu được điều khiển bởi bộ giải mã LDPC và bộ kiểm tra từ mã hợp lệ trong hình 3.12. Khi một từ mã không hợp lệ bị phát hiện tại đầu ra bộ giải mã LDPC, bộ giải mã LDPC sẽ gửi tín hiệu điều khiển s_c đến máy phát tín hiệu ACK được tích hợp trong máy thu dữ liệu, máy phát này sẽ gửi tín hiệu NACK đến máy thu tín hiệu ACK được tích hợp trên máy phát dữ liệu để yêu cầu máy phát phát lại một phần thông tin kiểm tra của từ mã

không hợp lệ.

Đối với mỗi symbol QAM $x = f(b)$, trong đó b là chuỗi m bit $b = (b_0, b_1, \dots, b_{m-1})$, tương ứng với các giá trị $LLRs$ từ $LLR(b_0)$ cho đến $LLR(b_{m-1})$, chúng ta đi đến [139]:

$$LLR(b_j) = \log \left[\frac{P(b_j = 1)}{P(b_j = 0)} \right], j = 0, \dots, m - 1 \quad (3.13)$$

Tín hiệu thu được y_k được biểu diễn như sau [139]:

$$y_k = a_k \cdot x_k + n_k, \quad (3.14)$$

trong đó n_k là biến ngẫu nhiên phức có giá trị trung bình 0 của tập nhiễu phân bố theo hàm Gauss có phương sai bình phương theo mỗi chiều. Hệ số a_k kí hiệu độ tăng ích pha định biến thiên theo thời gian có trị số phức và x_k kí hiệu symbol QAM được phát. Hàm phân bố mật độ (PDF) của tín hiệu y thu được có thể tính như sau [139]:

$$P(y|x, a) = \left(\frac{1}{\pi N_0} \right) e^{[-(\frac{E_s}{N_0})|y-ax|^2]}. \quad (3.15)$$

Lấy logarit hai phía phương trình (3.15), chúng ta đi đến xác suất dưới dạng logarit [139]:

$$\ln P(y|x, a) = \ln \frac{1}{\pi N_0} - \frac{E_s}{N_0} |x|^2 - |a|^2 \frac{E_s}{N_0} (y_1 x_1 + y_Q x_Q), \quad (3.16)$$

trong đó các thành phần cùng pha và vuông góc của tín hiệu truyền x và tín hiệu thu được y có thể được biểu diễn tương ứng như sau $x = x_1 + jx_Q, y = y_1 + jy_Q$ và E_s/N_0 là tỉ số năng lượng trên nhiễu của symbol được điều chế. Nếu tất cả các bit b_j trong chuỗi bit b là độc lập, chúng ta có:

$$\log P(x) = \log P(b) = \log P(b_0)P(b_1) \cdots P(b_{m-1}), \quad (3.17)$$

Từ các phương trình (3.15) và (3.17), chúng ta có thông tin bits mềm đầu ra của bộ giải ánh xạ có thể biểu diễn theo các giá trị $LLRs$ [139]:

$$\begin{aligned} LLR(\hat{b}_j) &= \log \left[\frac{P(b_j = 1)}{P(b_j = 0)} \right] \\ &= \log \left[\frac{\sum_{x:b_j=1} e^{[\log P(y|x,a) + \log P(x)]}}{\sum_{x:b_j=0} e^{[\log P(y|x,a) + \log P(x)]}} \right] \\ &= \log \left[\frac{\sum_{x:b_j=1} e^{[\gamma(y,x)]}}{\sum_{x:b_j=0} e^{[\gamma(y,x)]}} \right], \end{aligned} \quad (3.18)$$

trong đó ước lượng x của máy thu được kí hiệu là \hat{x} và chúng ta có $\hat{b} = (\hat{b}_0, \dots, \hat{b}_{m-1})$, trong đó kí hiệu $\gamma(y, x)$ được biểu diễn như sau [139]:

$$\gamma(y, x) = -|a|^2 \frac{E_s}{N_0} |x|^2 + 2|a| \frac{E_s}{N_0} (y_1 x_1 + y_Q x_Q) + \sum_{j=0}^{m-1} b_j \cdot LLR(b_j). \quad (3.19)$$

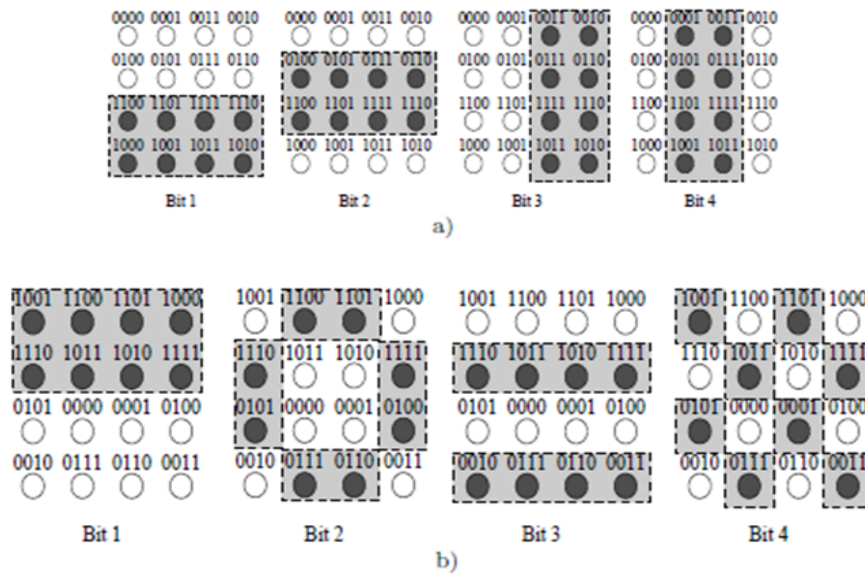
Thông tin ngoại lai đầu ra được tính như sau [53]:

$$LLR_E(\hat{b}_j) = LLR(\hat{b}_j) - LLR(b_j). \quad (3.20)$$

Thông lượng của mô hình H-ARQ tích hợp mã LDPC được kí hiệu là C , khi truyền dữ liệu qua kênh AWGN và sử dụng kiểu điều chế 16-QAM, chúng ta có [149]:

$$C = \frac{1}{16} \sum_{x \in 16} I(x, y) = \frac{1}{16} \sum_{x \in 16} \int_{-\infty}^{\infty} P(y|x) \log_2 \frac{P(y|x)}{P(y)} dy, \quad (3.21)$$

trong đó $I(x; y)$ kí hiệu cho thông tin tương tác giữa tín hiệu thu được y và tín hiệu được truyền dẫn x , trong đó $P(y|x)$ được cho trong phương trình (3.15) và a là hằng số.



Hình 3.14: a) Đồ thị chòm sao của kiểu ánh xạ mã Gray b) Đồ thị chòm sao của kiểu ánh xạ phân đoạn

Trong sơ đồ khối của hệ thống H-ARQ tích hợp mã LDPC vẽ trong hình 3.13, bộ giải ánh xạ đóng vai trò bộ giải mã trong, trong khi đó bộ giải mã LDPC đóng vai trò bộ giải mã ngoài. Tiếp sau bộ giải tráo các thông tin ngoại lai *LLRs* đầu ra bộ giải ánh xạ được đưa vào đầu vào bộ giải mã LDPC. Các *LLRs* này được xử lý trong bộ giải mã LDPC. Thông tin ngoại lai *LLRs* tại đầu ra bộ giải mã LDPC được hồi tiếp về đầu vào bộ giải ánh xạ sau khi được thực hiện tráo như trong hình 3.13. Nếu khối kiểm tra từ mã hợp lệ của bộ giải mã LDPC phát hiện đầu ra bộ giải mã là một từ mã không hợp lệ, nó sẽ gửi tín hiệu điều khiển để mở máy phát ACK. Tín hiệu NACK được gửi đến máy phát, yêu cầu máy phát phát thêm phần thông tin kiểm tra. Sau khi nhận được thêm phần thông tin kiểm tra mới máy thu tiếp tục quá trình giải mã lặp. Quá trình phát lặp phần thông tin kiểm tra được lặp lại cho đến khi đầu ra bộ giải mã là một từ mã hợp lệ, hoặc số lần phát lặp đạt giá trị cực

đại đã được đặt từ ban đầu.

Phân tích hoạt động của hệ thống bằng đồ thị EXIT

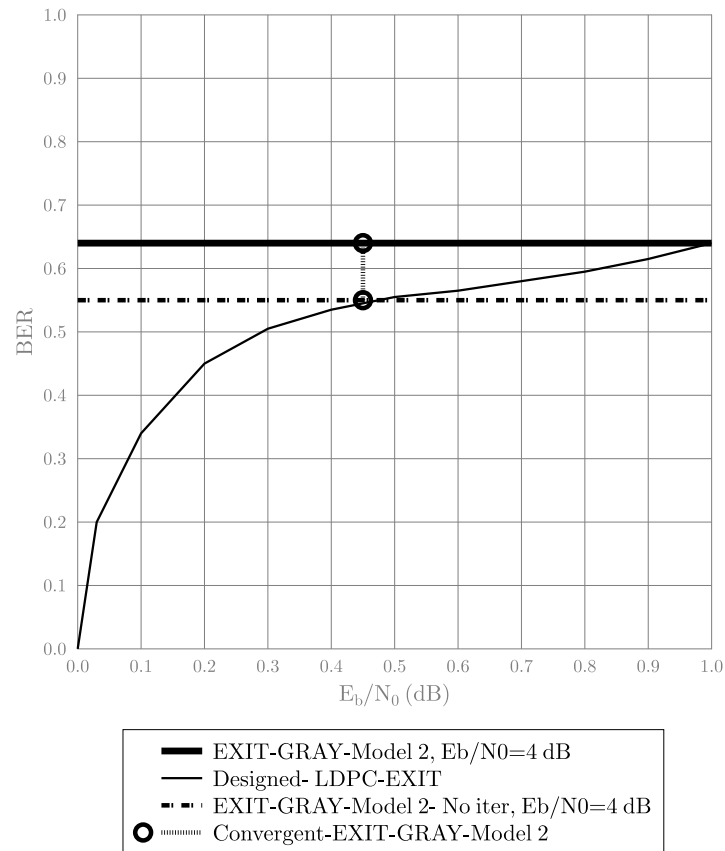
Trong mục này tác giả sử dụng đồ thị EXIT trong [156,157] để phân tích hoạt động của hệ thống H-ARQ tích hợp mã LDPC. Đồ thị EXIT của mô hình này bao gồm 3 đường cong trong hình 3.15 và 3.16, đó là đường cong đồ thị bộ giải mã trong, đường cong đồ thị của bộ giải mã ngoài và đường cong hội tụ dựa trên mô phỏng của Monte-Carlo. Đường cong đồ thị của bộ giải mã trong được sinh ra do bộ giải ánh xạ, đường cong đồ thị của bộ giải mã ngoài được sinh ra do bộ giải mã LDPC và đường cong hội tụ được sinh ra do quá trình trao đổi thông tin ngoài lai giữa bộ giải mã LDPC và bộ giải ánh xạ.

Đường cong của bộ giải mã trong trong hình 3.15 và 3.16 là quan hệ giữa thông tin tiên nghiệm $I_{A_{map}}$ và thông tin ngoài lai của bộ giải ánh xạ. Ngược lại, đường cong đồ thị EXIT của bộ giải mã LDPC trong hình 3.15 và 3.16 thể hiện mối quan hệ giữa thông tin đầu vào bộ giải mã $I_{A_{ldpc}}$ và thông tin đầu ra bộ giải mã LDPC $I_{E_{ldpc}}$. Nếu không có quá trình phát lại xảy ra giữa máy thu và phát trong mô hình H-ARQ tích hợp mã LDPC, thì thông tin tiên nghiệm $LLRs$ của bộ giải điều chế LDPC không thay đổi giữa hai lần lặp, bởi vì thông tin $LLRs$ của bộ giải điều chế không thay đổi. Do đó, đường cong hội tụ của mô hình này thể hiện mối quan hệ giữa thông tin tiên nghiệm $LLRs$ tại đầu vào bộ giải điều chế và thông tin ngoài lai $LLRs$ tại đầu ra bộ giải mã LDPC sau 1 lần lặp giải mã. Thông tin tương tác hai chiều đầu ra tuân theo hàm dưới đây [157]:

$$I_{E_{ldpc}} = f(I_{E_{map}}, I_{A_m}), \quad (3.22)$$

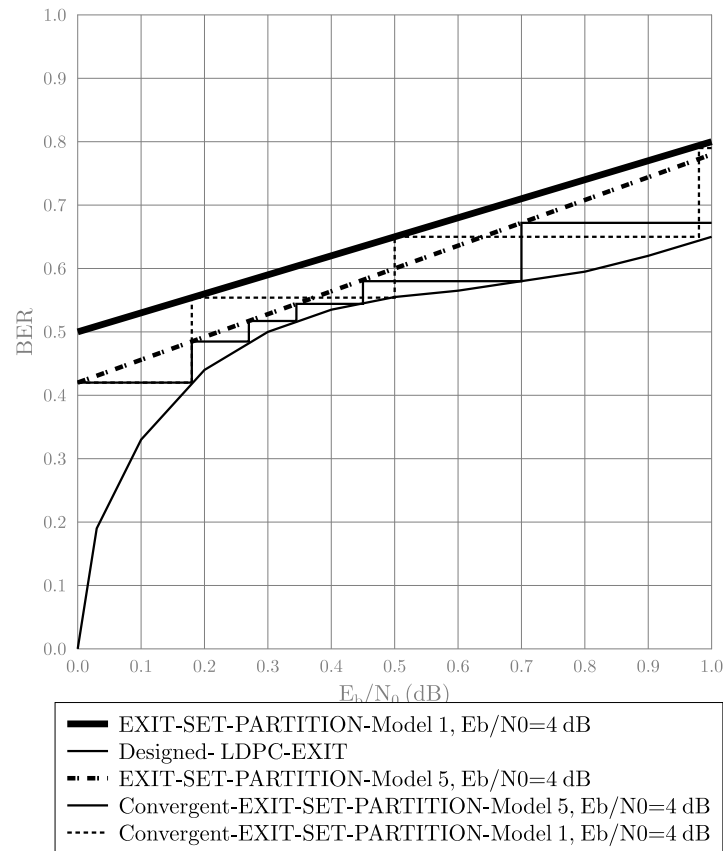
trong đó $I_{A_m} = I(X; R)$ là giá trị trung bình thông tin tiên nghiệm tại đầu vào các nút thông tin và R là thông tin LLR tới từ các nút kiểm

tra của bộ giải mã LDPC. Khi có quá trình phát lại giữa máy phát và thu của mô hình, thì phương trình đường cong hội tụ là hàm của cả các thông tin tương tác đầu vào của bộ giải mã LDPC và các thông tin tương tác đầu ra của bộ giải mã LDPC và cả thông tin tương tác $I_{A_{retrans}}$ tại đầu vào bộ giải mã LDPC. Lượng thông tin $I_{A_{retrans}}$ tại đầu vào bộ giải mã LDPC phụ thuộc vào số lần phát lặp được thực hiện. Hình 3.15 biểu diễn các đường cong đồ thị EXIT và đường cong hội tụ của mô hình H-ARQ tích hợp mã LDPC tại $E_b/N_0 = 4$ dB, khi sử dụng bộ ánh xạ mã Gray và điều chế 16-QAM. Đường đồ thị có nét chấm gạch là đường đồ thị EXIT nguyên bản của bộ ánh xạ tại $E_b/N_0 = 4$ dB, trong khi đó đường đậm nét và liên tục là đường đồ thị EXIT của bộ ánh xạ mã Gray trong mô hình hệ thống H-ARQ tích hợp mã LDPC. Ban đầu, máy thu của mô hình H-ARQ tích hợp mã LDPC cố gắng giải mã từ mã thu được bằng cách lặp lại quá trình trao đổi thông tin giữa bộ giải mã LDPC và bộ giải mã ánh xạ. Tuy nhiên, đường hội tụ bị chặn bởi giao điểm của các đường đồ thị EXIT, sau chỉ một lần lặp, điểm này được đánh dấu bằng hình tròn như trong hình 3.15. Trong trường hợp có lỗi tại đầu ra bộ giải mã LDPC, máy phát sẽ được yêu cầu phát lại phần thông tin kiểm tra của từ mã lỗi. Tại thời điểm này, đường cong đồ thị EXIT của bộ ánh xạ sẽ chuyển lên trên và đường EXIT này được biểu diễn bằng đường cong đậm, như có thể thấy trong hình 3.15. Thông tin tiên nghiệm $LLRs$ tại đầu vào bộ giải mã LDPC được gán bằng giá trị $LLRs$ tại đầu ra bộ giải mã LDPC của lần lặp trước đó. Máy thu tiếp tục giải mã từ mã thu được và do đó đường hội tụ bây giờ đạt được đến điểm $(0,64, 1)$ sau một lần lặp giải mã, như thể hiện trong hình 3.15.



Hình 3.15: Đồ thị EXIT của mô hình H-ARQ tích hợp mã LDPC sử dụng bộ ánh xạ mã Gray (Mô hình 2) trong hình 3.14

Đường cong đồ thị EXIT của bộ giải mã LDPC có tỉ lệ mã bằng 0,5 có hình dạng đặc biệt, như trong hình 3.15 và sau chỉ 4 lần lặp giải mã, hệ thống H-ARQ tích hợp mã LDPC đạt được giá trị $BER \leq 10^{-5}$ tại giá trị $E_b/N_0 = 4$ dB, như có thể thấy trong hình 3.17 và 3.18.



Hình 3.16: Đường cong đồ thị EXIT và đường hội tụ của mô hình hệ thống H-ARQ tích hợp mã LDPC sử dụng bộ ánh xạ phân đoạn (Mô hình 1) của hình 3.14

Bây giờ ta cải thiện khả năng sửa lỗi của hệ thống này bằng cách tạo ra các dạng đường cong EXIT có lợi cho bộ giải ánh xạ với sự trợ giúp của kỹ thuật ánh xạ phân đoạn. Đồ thị EXIT của hệ thống H-ARQ tích hợp mã LDPC sử dụng kỹ thuật ánh xạ phân đoạn được thể hiện trong hình 3.16. Đường đồ thị EXIT nguyên bản của bộ ánh xạ phân đoạn được thể hiện bằng đường chấm gạch, trong khi đó đường đồ thị EXIT của bộ ánh xạ phân đoạn sau khi thực hiện lặp một lần được thể hiện bằng đường nét liền đậm, như có thể thấy trong hình 3.16. Nếu không

có sự trợ giúp của quá trình phát lặp, thì hệ thống này không đạt được giá trị BER cực nhỏ như mong muốn sau 4 lần lặp giải mã, vì sau bốn lần lặp đường hội tụ dừng tại điểm được đánh dấu bằng hình hoa thị trong đồ thị 3.16. Để đạt được giá trị $BER \leq 10^{-5}$, máy thu phải thực hiện giải mã lặp đến 8 lần.

Qua việc quan sát đồ thị EXIT trong hình 3.15 và 3.16 ta có nhận xét sau. Mô hình hệ thống H-ARQ tích hợp mã LDPC sử dụng bộ ánh xạ kiểu mã Gray đòi hỏi số lần lặp giải mã thấp hơn so với khi sử dụng kiểu ánh xạ phân đoạn, để đạt được tỉ số $BER \leq 10^{-5}$. Ngược lại nếu hệ thống sử dụng bộ ánh xạ kiểu phân đoạn có điểm giao giữa các đường cong đồ thị EXIT và đường hội tụ gần với điểm (1, 1) hơn khi sử dụng số lần lặp giải mã cao, điều đó chứng tỏ tại cùng giá trị E_b/N_0 nếu sử dụng số lần lặp lớn, hệ thống sử dụng bộ ánh xạ phân đoạn có BER thấp hơn so với hệ thống sử dụng bộ ánh xạ kiểu mã Gray.

Kết quả mô phỏng

Các thông số mô phỏng được sử dụng trong Mô hình 1, Mô hình 2, Mô hình 3, Mô hình 4, Mô hình 5 và Mô hình 6 được đưa trong bảng 3.2. Hình 3.17 và 3.18 biểu diễn quan hệ BER và E_b/N_0 của các mô hình hệ thống H-ARQ sử dụng các thông số khác nhau trong bảng 3.2. Trục đồ thị thẳng đứng bên trái là các giá trị BER, trong khi đó trục tọa độ thẳng đứng bên phải là thông lượng chuẩn hóa của hệ thống và trục đồ thị ngang là trục giá trị E_b/N_0 .

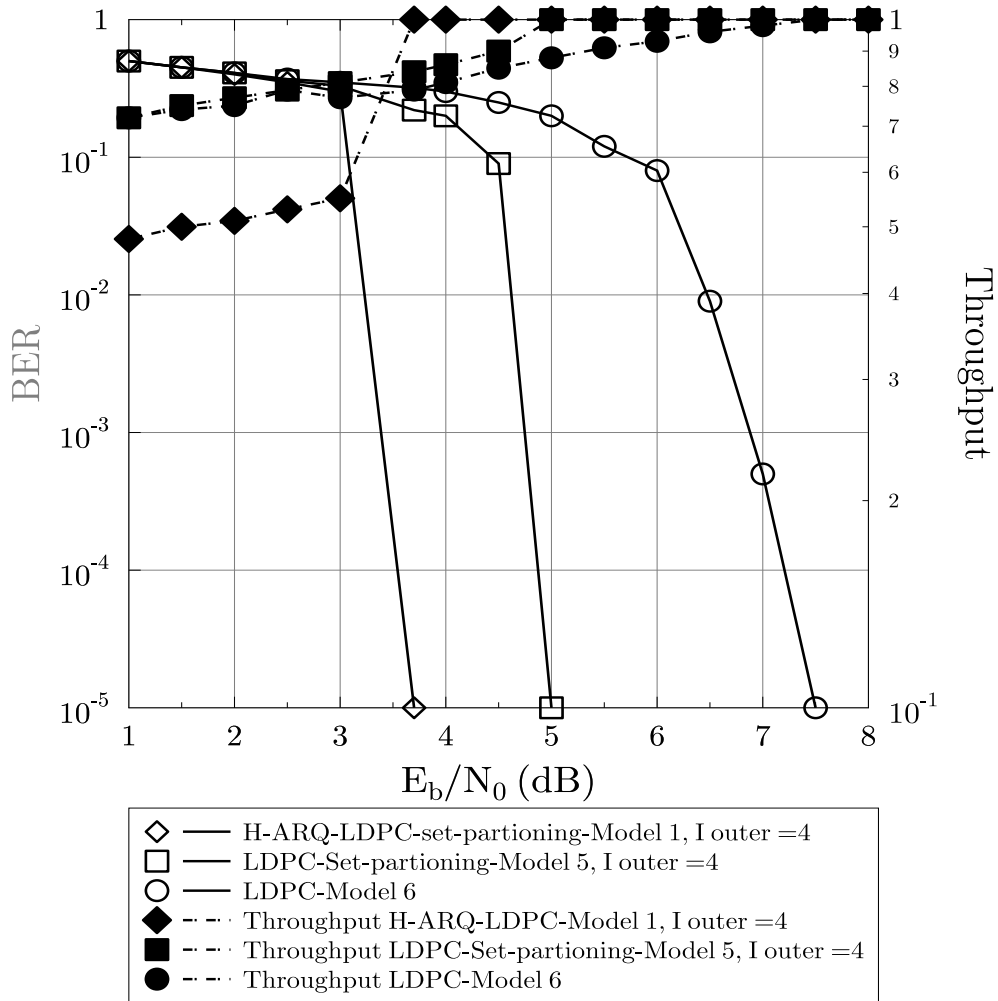
Các giá trị chuẩn hóa thông lượng được tính toán bằng cách chuẩn hóa thông lượng (Trong hình thông lượng là Throughput) có ích với thông lượng² của hệ thống H-ARQ tích hợp mã LDPC, ghi tại giá trị E_b/N_0 cao, ví dụ như khi truyền không có lỗi xảy ra. Bộ mã LDPC của

²Thông lượng có ích được định nghĩa là tỉ số giữa các bit thông tin đúng trên tổng số các bit được truyền.

Bảng 3.2: Các thông số mô phỏng

Tỉ lệ mã r của mã LDPC	1/2
Các thông số của hàm phân bố mật độ cho trong 2.5	$\delta = 0, 5; c = 0, 1; t = 2$
Số lần lặp cực đại mặc định cho bộ giải mã LDPC I_{max}	12 , 20
Số lần lặp cực đại giữa bộ giải ánh xạ và bộ giải mã LDPC	2 , 4
Số bit thông tin	10^6 bit
Kiểu điều chế	16-QAM
Loại kênh	AWGN
Kiểu ánh xạ	Kiểu phân đoạn và kiểu mã Gray
Kiểu H-ARQ	H-ARQ loại II
Mô hình hệ thống H-ARQ tích hợp mã LDPC sử dụng kiểu ánh xạ phân đoạn và có giải mã lặp giữa bộ giải mã LDPC và bộ ánh xạ	Kí hiệu là Mô hình 1
Mô hình hệ thống H-ARQ tích hợp mã LDPC sử dụng kiểu ánh xạ mã Gray và có giải mã lặp giữa bộ giải mã LDPC và bộ ánh xạ	Kí hiệu là Mô hình 2
Mô hình hệ thống H-ARQ tích hợp mã LDPC không thực hiện giải mã lặp giữa bộ giải mã LDPC và bộ ánh xạ	Kí hiệu là Mô hình 3
Mô hình hệ thống ARQ không có mã sửa sai	Kí hiệu là Mô hình 4
Mô hình hệ thống LDPC có giải mã lặp giữa bộ ánh xạ sử dụng kiểu phân đoạn và bộ giải mã LDPC	Kí hiệu là Mô hình 5
Mô hình hệ thống chỉ sử dụng mã LDPC	Kí hiệu là Mô hình 6

hệ thống có tỉ lệ mã $r=0,5$, do đó thông lượng cực đại của mô hình hệ thống H-ARQ tích hợp mã LDPC- Mô hình 2 là 2 bit/symbol, giá trị này được sử dụng làm hệ số chuẩn hóa.



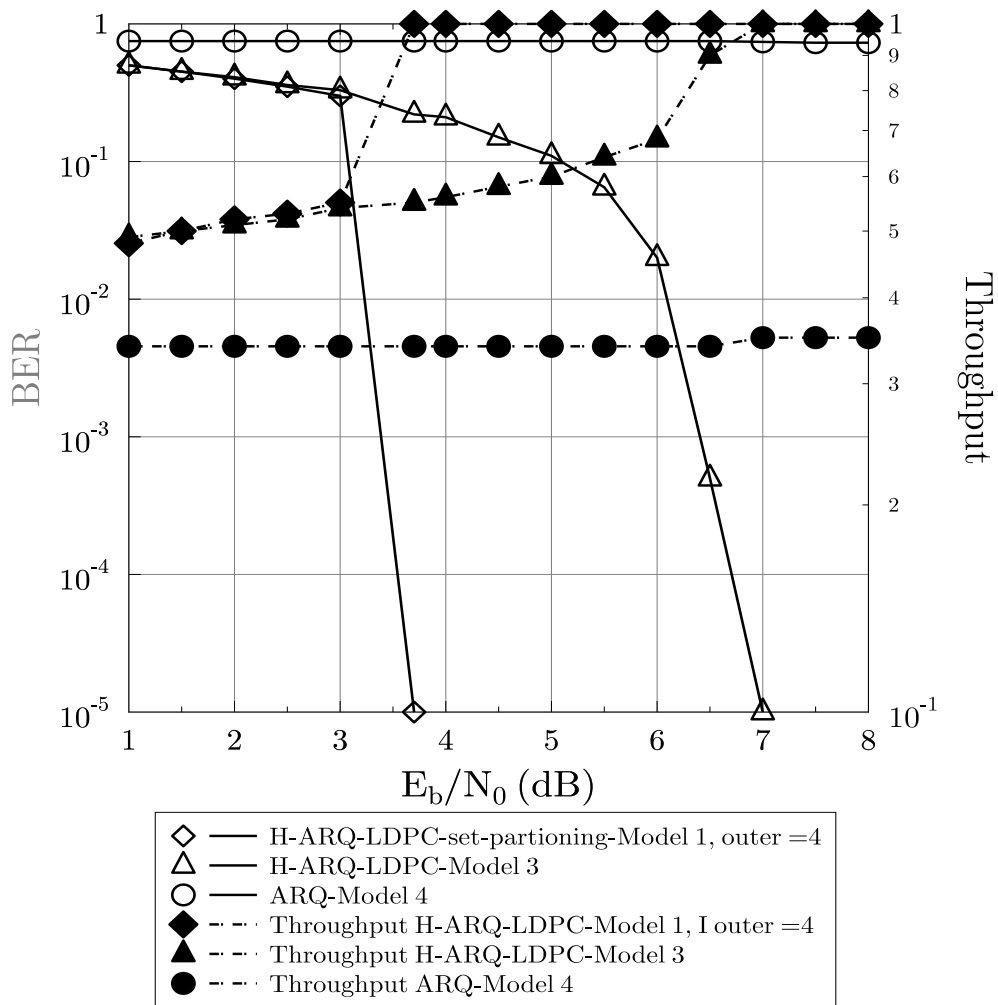
Hình 3.17: Khả năng hoạt động của các mô hình hệ thống 1,5 và 6, khi điều chế 16-QAM và kênh truyền là AWGN

Như có thể thấy trong hình 3.17 và 3.18 các đường liền nét được đánh dấu bằng hình thoi rỗng biểu diễn quan hệ BER và E_b/N_0 của mô hình hệ thống H-ARQ tích hợp mã LDPC có sử dụng lặp giải mã giữa LDPC và bộ ánh xạ, Mô hình này đạt giá trị BER $\leq 10^{-5}$ tại giá trị

E_b/N_0 cỡ 3,8 dB. Nếu hệ thống này không sử dụng chế độ phát lặp và hồi tiếp, thì để đạt được tỉ số BER $\leq 10^{-5}$, nó yêu cầu tỉ số $E_b/N_0 = 5$ dB, như được biểu diễn bằng đường liền nét được đánh dấu bằng hình vuông rỗng trong hình 3.17. Một mô hình hệ thống khác được mô phỏng ở đây là mô hình hệ thống không thực hiện giải mã lặp giữa bộ giải mã LDPC và bộ ánh xạ, đường cong đồ thị được biểu diễn bằng đường liền nét đánh dấu bằng hình tròn rỗng, như thể hiện trong hình 3.17. Khi so sánh khả năng sửa lỗi và thông lượng có ích của các hệ thống trên, ta có thể thấy mô hình hệ thống H-ARQ tích hợp mã LDPC được tác giả thiết kế có độ tăng ích trên 4 dB và thông lượng lớn hơn so với hai hệ thống còn lại, như trong hình 3.17.

Tiếp tục phân tích kết quả mô phỏng trong hình 3.18. Trong hình này chúng ta so sánh khả năng hoạt động của hệ thống H-ARQ tích hợp mã LDPC có sử dụng giải mã lặp giữa bộ giải ánh xạ kiểu phân đoạn và bộ giải mã LDPC với các mô hình hệ thống ARQ thông thường và mô hình hệ thống chỉ có mã sửa sai LDPC. Như có thể thấy trong hình 3.18, các đường cong kí hiệu bằng hình tam giác biểu diễn mô hình hệ thống có mã LDPC độc lập với bộ ánh xạ của bộ điều chế 16-QAM. Cuối cùng, đường cong kí hiệu bằng hình tròn có dấu cộng ở giữa biểu diễn hệ thống ARQ không sử dụng mã sửa sai, với mô hình này thông lượng có ích của hệ thống là 4 bit/symbol. Ở dải tỉ số tín trên tạp rất cao, hệ thống ARQ có thông lượng lớn hơn hệ thống mã hóa LDPC. Ngược lại, trong vùng tỉ số tín trên tạp thấp, thông lượng chuẩn hóa của hệ thống ARQ vào khoảng 0,33, bởi vì tại vùng tỉ số tín trên tạp thấp máy thu không thể đạt giá trị BER thấp như mong muốn, do đó nó yêu cầu phát lặp nhiều nhất. Hệ thống H-ARQ tích hợp mã LDPC không sử dụng lặp giải mã giữa bộ giải mã LDPC và bộ giải ánh xạ yêu cầu tỉ số E_b/N_0 cao hơn 3

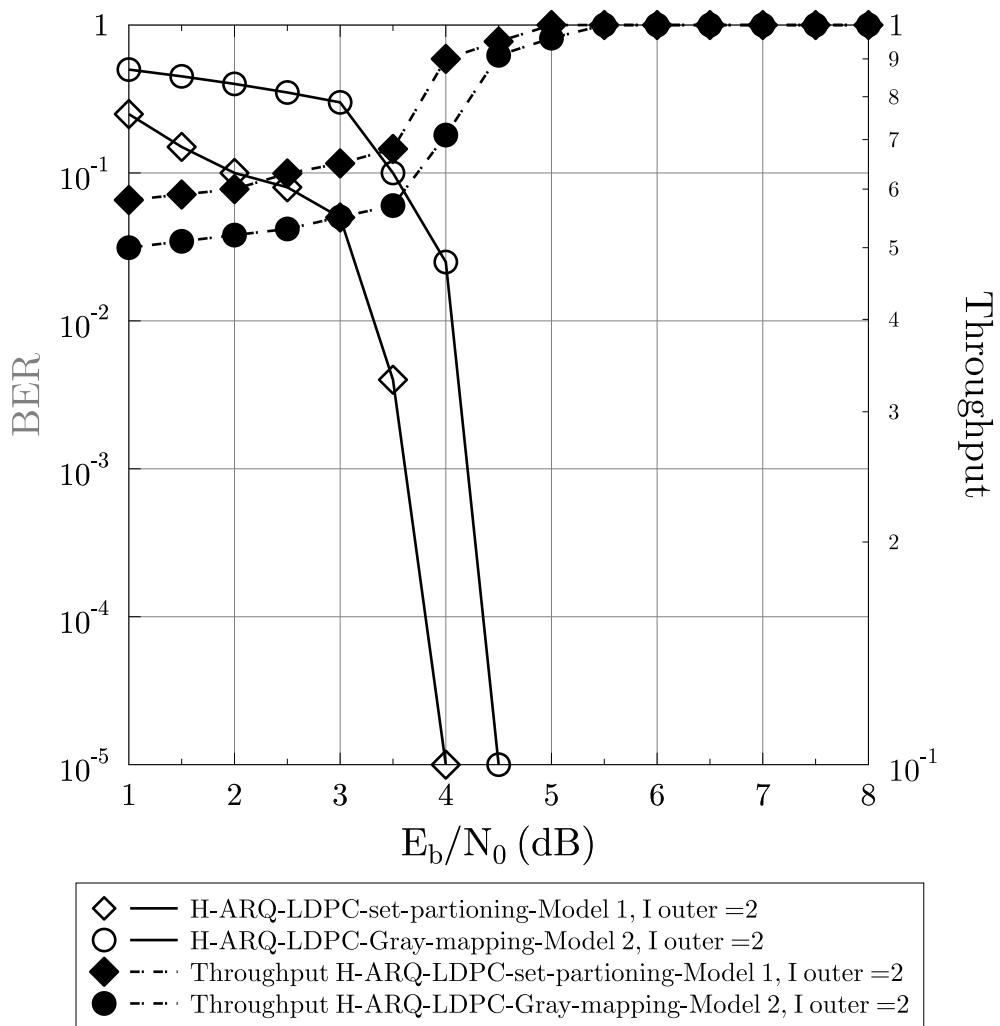
dB so với hệ thống H-ARQ tích hợp mã LDPC có sử dụng lập giải mã giữa bộ giải mã LDPC và bộ giải ánh xạ kiểu phân đoạn, đường cong đồ thị tương ứng được thể hiện bằng đường liền nét kí hiệu bằng hình tam giác trong hình 3.18. Hệ thống ARQ không sử dụng mã kênh yêu cầu tỉ số E_b/N_0 cao hơn 18 dB để đạt được tỉ số BER $\leq 10^{-5}$.



Hình 3.18: Khả năng hoạt động của các mô hình hệ thống 1, 3 và 4, khi điều chế 16-QAM và kênh truyền là AWGN

Cuối cùng, trong hình 3.19 chúng ta so sánh khả năng hoạt động của

hai mô hình hệ thống H-ARQ tích hợp mã LDPC sử dụng bộ ánh xạ kiểu mã Gray và kiểu ánh xạ phân đoạn. Qua quan sát, có thể thấy mô hình hệ thống sử dụng kiểu ánh xạ mã phân đoạn có khả năng sửa lỗi và thông lượng ra của hệ thống tốt hơn so với kiểu ánh xạ Gray khi chỉ dùng 2 lần lặp giải mã giữa bộ giải mã LDPC và bộ giải ánh xạ của bộ điều chế 16-QAM.



Hình 3.19: Khả năng hoạt động của các mô hình hệ thống 1 và 2, khi điều chế 16-QAM và kênh truyền là AWGN

3.4. Kết luận chương 3

Các mô hình tích hợp mã LDPC trong chương 3 có khả năng hoạt động tốt hơn so với các mô hình hệ thống cơ bản hiện có. Cụ thể hệ thống V-BLAST tích hợp mã LDPC cho ta thấy khả năng vượt trội của mã LDPC đã được thiết kế trong chương 2 so với các mã thuộc họ mã Turbo tiên tiến được thiết kế từ bộ mã tích hợp RSC-URC. Đặc biệt với độ dài tráo ngẫu nhiên mô hình hệ thống V-BLAST tích hợp mã LDPC có độ tăng ích trên 4 dB so với mô hình hệ thống V-BLAST tích hợp mã RSC-URC. Trong chương 3 cũng đề xuất thiết kế một hệ thống H-ARQ tích hợp mã LDPC đã được thiết kế trong chương 2. Khả năng hoạt động của hệ thống H-ARQ trở nên vượt trội hơn so với các hệ thống thông tin khác kể cả về khả năng sửa lỗi BER và thông lượng đầu ra hệ thống.

KẾT LUẬN VÀ ĐỀ XUẤT

Thực hiện mục tiêu đề ra luận văn đã tiến hành nghiên cứu, xây dựng ma trận sinh và ma trận kiểm tra của mã LDPC, nhằm tối ưu khả năng sửa lỗi của mã LDPC, đã xây dựng, tối ưu hóa các mô hình tích hợp giữa mã LDPC với các hệ thống thông tin nhằm tăng khả năng chống lỗi của hệ thống với độ phức tạp phù hợp, đạt được mục tiêu đề ra. Đề tài đã đạt được các kết quả nghiên cứu và đóng góp chính sau đây:

1. Xây dựng mối quan hệ trực tiếp giữa ma trận sinh và ma trận kiểm tra của mã LDPC, nhằm giảm độ phức tạp trong quá trình tính toán xây dựng ma trận sinh của mã LDPC truyền thống. Ma trận sinh của mã LDPC được suy trực tiếp từ ma trận kiểm tra.
2. Xây dựng cấu trúc ma trận thành phần của ma trận sinh bằng các hàm phân bố ngẫu nhiên cho hàng và cột, hiệu quả của quá trình xây dựng này là làm tăng khả năng sửa lỗi của mã LDPC so với các mã LDPC phổ thông từ 0,5 dB đến 1 dB, tùy theo từng điều kiện cụ thể.
3. Xây dựng mô hình tích hợp giữa mã LDPC và hệ thống V-BLAST, nhằm tăng cường khả năng sửa lỗi của hệ thống tốt hơn đến 5 dB so với mô hình tích hợp mã URC và V-BLAST, tối ưu các thông số trong hệ thống để giảm thiểu độ phức tạp của hệ thống so với hệ thống URC-VBLAST.
4. Xây dựng mô hình lai ghép giữa mã LDPC và H-ARQ, nhằm tăng cường khả năng chống lỗi do can nhiễu của đường truyền và thông lượng của hệ thống. Như đã phân tích trong chương 3, hệ thống

lai ghép LDPC và H-ARQ có độ tăng ích lớn hơn 4dB so với các hệ thống H-ARQ khác và hơn 10dB so với hệ thống ARQ không sử dụng mã sửa sai, với cùng tỉ lệ bit lỗi $BER \leq 10^{-5}$.

Phương hướng nghiên cứu tiếp theo

- Về mặt lý luận nghiên cứu: Tối ưu các hàm phân bố chuẩn để tạo ra ma trận kiểm tra có khả năng làm tăng khả năng sửa lỗi của mã LDPC trong vùng E_b/N_0 thấp.
- Về mặt thực hành nghiên cứu: Nghiên cứu phát triển, tối ưu các mô hình tích hợp mã LDPC với mã BCH ứng dụng trong các hệ thống truyền hình theo tiêu chuẩn DVB-T2, DVB-C2, DVB-S2.

CÁC CÔNG TRÌNH ĐÃ CÔNG BỐ

1. Liet. C. V¹, Thanh. N. D, Vu. N. H, "A FAST DESIGN FOR LDPC MATRICES", IEEE International Symposium on Signal Processing and Information Technology December 12-15, 2012 - Ho Chi Minh City-Vietnam.
2. Cao Văn Liết¹, Nguyễn Đăng Thành, Nguyễn Hồng Vũ, "TÍCH HỢP HỆ THỐNG LDPC V-BLAST", Chuyên san Công nghệ Thông tin và Truyền thông, Tạp chí Khoa học và Kỹ thuật, Học viện Kỹ thuật Quân sự, tháng 10 năm 2012, số 150, trang 23-31.
3. Cao Văn Liết¹, Nguyễn Đăng Thành, Nguyễn Đức Minh, "HỆ THỐNG H-ARQ TÍCH HỢP MÃ LDPC", Tạp chí Nghiên cứu Khoa học và Công nghệ Quân sự, tháng 02 năm 2014, số 29, trang 70-76.

Tài liệu tham khảo

- [1] P. Elias, “Coding for Noisy Channels,” *IRE International Convention Record*, vol. Part IV, pp. 37–46, 1955.
- [2] I. S. Reed and G. Solomon, “Polynomial Codes over Certain Finite Fields,” *SIAM Journal of Applied Math*, vol. 8, pp. 300–304, 1960.
- [3] A. Hocquenghem, “Codes Correcteurs D’erreurs,” *Chiffres*, vol. 2, pp. 147–156, September 1959.
- [4] R. C. Bose and C. R. Ray-Chaudhuri, “On a Class of Error correcting Binary Group Codes,” *Information and Control*, vol. 3, pp. 68–79, 1960.
- [5] C. Berrou, A. Glavieux and P. Thitimajshima, “Near Shannon Limit Error-Correcting Coding and Decoding: Turbo Codes,” *Proceedings of the International Conference on Communications*, pp. 1064–1070s, May 1993.
- [6] R. Gallager, “Low Density Parity Check Codes,” *IRE Transactions on Information Theory*, pp. 21–28, Jan 1962.

-
- [7] D. J. C MacKay, S. T Wilson, and M. C Davey, “Comparison of constructions of irregular gallager codes,” *IEEE Transactions on Communications*, vol. 47, pp. 1449–1454, October 1999.
- [8] L. Lan, Y. Y. Tai, L. Chen, S. Lin and K. Abdel-Ghaffar, “A trellis-based method for removing cycles from bipartite graphs and construction of low-density parity-check codes,” *IEEE Communications Letters*, vol. 81, pp. 443–445, July 2004.
- [9] A. Viswanathan, K. J. Zhang, “Stopping Set Distribution of LDPC Code Ensembles,” *IEEE Transactions on Information Theory*, vol. 51, pp. 929–953, March 2005.
- [10] H. Sankar and K. R. Narayanan, “Memory-efficient sum-product decoding of LDPC codes,” *IEEE Transactions on Communications*, vol. 52, pp. 1225–1230, August 2004.
- [11] J. Chen, A. Deholakia, E. Eleftheriou, M. Fossorier, X. Y. Hu, “Near optimal reduced-complexity decoding algorithms for LDPC codes,” in *IEEE International Symposium on Information Theory*, p. 455, 2002.
- [12] M. Y. Alias, F. Guo, S. X. Ng, T. H. Liew and L. Hanzo, “LDPC and turbo coding assisted space-time block coded OFDM,” in *IEEE Vehicular Technology Conference, Jeju, Korea*, vol. 4, pp. 2309–2313, 22-25 April 2003.
- [13] J. Y. Chung, M. Y. Alias, F. Guo and L. Hanzo, “LDPC and turbo coding assisted space-time block coded OFDM for H.26L,” in *Proceedings of PIMRC, Beijing, China*, pp. 2702–2706, September 2003.

- [14] S. Y. L. Goff, “Digital Video Broadcasting (DVB); Interaction channel for Satellite Distribution Systems,” *ETSI EN 301 790, V1.5.1*, May 2009.
- [15] M.C. Valenti, S. Cheng, and R. Iyer Seshadri, “Turbo and LDPC codes for digital video broadcasting,” *Chapter 12 of Turbo Code Applications: A Journey from a Paper to Realization*, Springer, 2005.
- [16] S. Y. L. Goff, “Comparison of power consumption of mobile WiMAX, HSPA and LTE access networks,” <http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=5557715>, 30 October 2012.
- [17] , “What really is a Third Generation (3G) Mobile Technology,” *International Telecommunications Union*, June 2011.
- [18] R. Gallager, “Low Density Parity Check Codes,” *Ph.D thesis, M.I.T, USA*, 1963.
- [19] A. Viswanathan, K. J. Zhang, “Estimation of the error correction complexity for gallager’s low-density codes,” *Problemy Peredachi Informatsii*, vol. 11, no. 1, pp. 23–36, 1975.
- [20] M. R. Tanner, “A recursive approach to low complexity codes,” *IEEE Transactions on Information Theory*, vol. 27, September 1981.
- [21] M. Sipser and D. A. Spielman, “Expander codes,” *IEEE Transactions on Information Theory*, vol. 42, pp. 1710–1722, November 1996.

- [22] . J. C MacKay, and R. M. Neal, “shannon limit performance of low density parity check codes,” *Electronics Letters*, vol. 33, pp. 457–458, 13 March 1997.
- [23] D. J. C. MacKay and R. M. Neal, “Good error-correction codes based on very sparse matrices,” *IEEE Transactions on Information Theory*, vol. 45, pp. 399–431, March 1999.
- [24] C. E. Shannon, “A Mathematical Theory of Communication,” *Bell System Technical Journal*, vol. 27, pp. 379–423, Oct 1948.
- [25] M. C. Davey and D. J. C MacKay, “Low density parity check codes over $GF(q)$,” *IEEE Communications Letters*, vol. 2, pp. 165–167, June 1998.
- [26] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, and D. A. Spielman, “Improved low density parity check codes using irregular graphs and belief propagations,” in *Proceedings of the IEEE International Symposium on Information Theory*, p. 117, 1998.
- [27] M. Lentmaier and K. S. Zigangirov, “On Generalized Low-Density Parity-Check Codes Based on Hamming Component Codes,” *IEEE Communications Letters*, pp. 248–250, Aug 1999.
- [28] Y. Kou, S. Lin and M. Fossorier, “Low Density Parity Check Codes Based on Finite Geometries: A Rediscovery,” *IEEE International Symposium on Information Theory*, pp. 25–30, June 2000.
- [29] S. ten Brink, “Designing Iterative Decoding Schemes with the Ex-

- trinsic Information Transfer Chart,” *AEÜ International Journal of Electronics Communications*, vol. 55, pp. 389–398, 2000.
- [30] S. Y. Chung, T. J. Richardson and R. L. Urbanke, “Analysis of Sum-Product Decoding of Low Density Parity Check Codes using A Gaussian Approximation,” *IEEE Transactions on Information Theory*, vol. 47, pp. 657–670, Feb 2001.
- [31] B. Ammar, B. Honary, Y. Kou and S. Lin, “Construction of low density parity check codes: a combinatoric design approach,” in *IEEE International Symposium on Information Theory*, p. 311, 2003.
- [32] B. Ammar, B. Honary, Y. Kou and S. Lin, “Bootstrap Decoding of Low-Density Parity-Check Codes,” *IEEE Communications letters*, vol. 6, pp. 391–393, 9 September 2002.
- [33] P. Meshkat and H. Jafarkhani, “Space-Time Low-Density Parity-Check Codes,” vol. 2, p. 1117, 2002.
- [34] J. Zhang and M. P. C. Fossorier, “A Modified Weighted Bit-Flipping Decoding of Low-Density Parity-Check Codes,” *IEEE Communications Letters*, vol. 8, pp. 165–167, March 2004.
- [35] B. Lu, Y. Guosen and Y. Xiaodong, “Performance Analysis and Design Optimization of LDPC-Coded MIMO OFDM systems,” *IEEE Transactions on Signal Processing*, vol. 52, no. 2, p. 348, 2004. 1053-587X.

- [36] F. Guo, L. Hanzo, “Reliability ratio based weighted bit-flipping decoding for low-density parity check codes,” *IEE Communications Letters*, vol. 40, pp. 1356 – 1357, 14 October 2004.
- [37] Jie Huang ; Lei Liu ; Wuyang Zhou ; Shengli Zhou , “Large-Girth Nonbinary QC-LDPC Codes of Various Lengths,” *Communications, IEEE Transactions*, vol. 58, pp. 3436 – 3447, December 2010.
- [38] Seok-Ki Ahn ; Kyeongcheol Yang , “Adaptive Modulation and Coding Schemes Based on LDPC Codes with Irregular Modulation ,” *Communications, IEEE Transactions*, vol. 58, pp. 2465 – 2470, December 2010.
- [39] Kai He ; Jin Sha ; Zhongfeng Wang , “Nonbinary LDPC Code Decoder Architecture With Efficient Check Node Processing ,” *Communications, IEEE Transactions*, vol. 59, pp. 381 – 385, June 2012.
- [40] Arabaci, M. ; Djordjevic, I.B. ; Lei Xu ; Ting Wang, “Nonbinary LDPC-Coded Modulation for Rate-Adaptive Optical Fiber Communication Without Bandwidth Expansion,” *Photonics Technology Letters, IEEE*, vol. 24, pp. 1402 – 1404, 15 August 2012.
- [41] S. Y. Chung, *On the Construction of Some Capacity-Approaching Coding Schemes*. 2000. at <http://portal.acm.org/citation.cfm?id=934066>.
- [42] J. Chen and M. Fossorier, “Density evolution for two improved BP-based decoding algorithms of LDPC codes,” *IEEE Communication Letters*, vol. 6, pp. 208–210, May 2002.
- [43] W. Lin, X. Juan and G Chen, “Density evolution method and

- threshold decision for irregular LDPC codes,” in *IEEE International Conference on Communications, Circuits and Systems*, vol. 1, pp. 25 – 28, 27-29 June 2004.
- [44] K. R. Narayanan, I. Altunbas and R. S. Narayanaswami, “Design of serial concatenated MSK schemes based on density evolution,” *IEEE Transactions on Communications*, vol. 51, pp. 1283– 1295, August 2003.
- [45] A. Anastasopoulos, “A comparison between the sum-product and the min-sum iterative detection algorithms based on density evolution,” in *IEEE Global Telecommunications Conference*, vol. 2, pp. 1021 – 1025, 25 - 29 November 2001.
- [46] H. Song, J. Liu and B. V. Kumar, “Convergence analysis of iterative soft decoding in partial response channels,” *IEEE Transactions on Magnetics*, vol. 48, pp. 2437 – 2449, September 2002.
- [47] D. Burshtein, “Bounds on the performance of belief propagation decoding,” *IEEE Transactions on Information Theory*, vol. 48, pp. 112 – 122, January 2002.
- [48] G. Miller and D. Burshtein, “Bounds on the maximum-likelihood decoding error probability of low-density parity-check codes,” *IEEE Transactions on Information Theory*, vol. 47, pp. 2696 – 2710, November 2001.
- [49] H. Futaki and T. Ohtsuki, “Low-density parity-check (LDPC) coded OFDM systems with MPSK,” in *IEEE 55th Vehicular Technology Conference, Birmingham, AL, USA*, vol. 2, pp. 1035– 1039, May 2002.

- [50] H. Futaki and T. Ohtsuki, “LDPC-based space-time transmit diversity schemes with multiple transmit antennas,” in *IEEE 57th Vehicular Technology Conference*, vol. 4, pp. 2589 – 2593, 22 - 25 April 2003.
- [51] H. Pishro-Nik and F. Fekri, “On decoding of low-density parity-check codes over the binary erasure channel,” *IEEE Transactions on Information Theory*, vol. 50, pp. 439 – 454, March 2004.
- [52] D. Burshtein and G. Miller, “An efficient maximum-likelihood decoding of LDPC codes over the binary erasure channel,” *IEEE Transactions on Information Theory*, vol. 50, pp. 2837–2844, November 2004.
- [53] H. Song, J. Liu and B. V. Kumar, “Low complexity LDPC codes for partial response channel,” in *IEEE Global Telecommunications Conference*, vol. 2, pp. 1294 – 1299, November 2002.
- [54] T. Mittelholzer, A. Dholakia and E. Eleftheriou, “Reduced-complexity decoding of low density parity check codes for generalized partial response channels,” *IEEE Transactions on Magnetics*, vol. 37, pp. 721–728, November 2001.
- [55] M. Yang and W. E. Ryan, “Performance of (quasi-)cyclic LDPC codes in noise bursts on the EPR4 channel,” in *IEEE Global Telecommunication Conference*, vol. 5, pp. 2961 – 2965, 25-29, November 2001.
- [56] M. Yang and W. E. Ryan, “Performance of efficiently encodable low-density parity-check codes in noise bursts on the EPR4 chan-

- nel,” *IEEE Transactions on Magnetism*, vol. 40, pp. 507–512, March 2004.
- [57] D. Yang, R. Molstad and Y. Yip, “Performance evaluation of LDPC code on VR2 channel,” in *IEEE International Magnetism Conference*, p. AP2, 28 April - 2 May 2002.
- [58] E. Eleftheriou, S. Olcer, “Further results on the performance of LDPC coded modulation for AWGN channels,” *ITU-Telecommunication Standardization Sector*, May 2001.
- [59] J. Hou, P. H. Siegel, L. B. Milstein and H. D. Pfister, “Capacity-approaching bandwidth-efficient coded modulation schemes based on low-density parity-check codes,” *IEEE Transactions on Information Theory*, vol. 49, pp. 2141–2155, September 2003.
- [60] F. Guo, L. Hanzo, “LDPC assisted block coded modulation for transmission over Rayleigh fading channels,” in *Vehicular Technology Conference, Spring, Jeju, Korea*, vol. 3, pp. 1867 – 1871, 22-25 April 2003.
- [61] H. Song, J. Liu and B. V. Kumar, “Concatenated low density parity check (LDPC) codes for magnetic recording channels,” in *IEEE International Magnetism Conference*, vol. 42, pp. DT–12, 28 March - 3 April 2003.
- [62] H. Song, R. M. Todd and J. R. Cruz, “Applications of low-density parity-check codes to magnetic recording channels,” *IEEE Journal on Selected Areas in Communications*, vol. 19, pp. 918–923, May 2001.

- [63] H. Song, R. M. Todd and J. R. Cruz, “Low density parity check codes for magnetic recording channels,” *IEEE Transactions on Magnetics*, vol. 36, pp. 2183–2186, September 2000.
- [64] H. Zhong and T. Zhang, “Design of VLSI implementation-oriented LDPC codes,” in *IEEE Vehicular Technology Conference*, vol. 1, pp. 670–673, 6-9 Oct 2003.
- [65] H. Zhong and T. Zhang, “Joint code-encoder-decoder design for LDPC coding system VLSI implementation,” in *International Symposium on Circuits and Systems*, vol. 2, pp. 389–392, 23-26 May 2004.
- [66] G. Lechner, J. Sayir and M. Rupp, “Efficient DSP implementation of an LDPC decoder,” in *International Conference on Acoustics, Speech and Signal Processing*, vol. 4, pp. 665–668, 17-21 May 2004.
- [67] D. E. Hocevar, “LDPC code construction with exible hardware implementation,” in *IEEE International Conference on Communications*, vol. 4, pp. 2708–2712, 11-15 May 2003.
- [68] J. Thorpe, “Design of LDPC graphs for hardware implementation,” in *IEEE International Symposium on Information Theory*, p. 483, 2002.
- [69] Y. Pei, L. Yin, J. Lu, “Design of irregular LDPC codec on a single chip FPGA,” in *IEEE 6th Circuits and Systems Symposium on Emerging Technologies: Frontiers of Mobile and Wireless Communications*, vol. 1, pp. 221 – 224, 31 May - 2 June 2004.
- [70] M. M. Mansour, M. Shanbhag, “Architecture-aware low-density

- parity-check codes,” in *International Symposium on Circuits and Systems*, vol. 2, pp. 57–60, 25-28 May 2003.
- [71] M. C. Davey and D. J. C. MacKay, “Low density parity check codes over $GF(q)$,” in *Proceedings of the 1998 IEEE Information Theory Workshop*, vol. 2, pp. 70–71, June 1998.
- [72] M.C. Davey, “Error-correction using low density parity check codes,” *Ph.D thesis, University of Cambridge, UK*, 1999.
- [73] L. Barnault and D. Declercq, “Fast decoding algorithm for LDPC over $GF(2q)$,” in *IEEE Information Theory Workshop*, pp. 70–73, 31 March - 4 April 2003.
- [74] K. Nakamura, Y. Kabashima and D. Saad, “Statistical mechanics of low-density parity check error-correcting codes over Galois fields,” *Europhysics Letters*, vol. 56, pp. 610–616, November 2001.
- [75] X. Li, M. R. Soleymani, J. Lodge and P. S. Guinand, “Good LDPC codes over $GF(q)$ for bandwidth efficient transmission,” in *IEEE workshop on Signal Processing Advances in Wireless Communications*, vol. 56, pp. 95–99, 15-18 June 2003.
- [76] T. Richardson, M. A Shokrollahi, R. Urbanke, “Design of Capacity Approaching Irregular Low Density Parity Check Codes,” *IEEE Transaction on Information Theory*, vol. 47, February 2001.
- [77] X. Yang, D. Yuan, P. Ma and M. Jiang, “New research on unequal error protection (UEP) property of irregular LDPC codes,” in *IEEE Consumer Communications and Networking Conference*, pp. 361 – 363, 5-8 Jan 2004.

- [78] S. J. Johnson and S. R. Weller, “A family of irregular LDPC codes with low encoding complexity,” *IEEE Communications Letters*, vol. 7, pp. 79–81, February 2003.
- [79] T. Tao, C. Jones, J. D. Villasenor and R. D. Wesel, “Construction of irregular LDPC codes with low error floors,” in *IEEE International Conference on Communications*, vol. 5, pp. 3125–3129, 11-15 May 2003.
- [80] C. E. Shannon, “A mathematical theory of communication,” *The Bell system Technical Journal*, vol. 27, pp. 379–656, July 1948.
- [81] S. Sankaranarayanan, B. Vasic and E. M. Kurtas, “Irregular low-density parity-check codes: construction and performance on perpendicular magnetic recording channels,” *IEEE Transactions on Magnetics*, vol. 39, pp. 2567–2569, September 2003.
- [82] M. Ardakani, F. R. Kschischang, “Designing irregular LDPC codes using EXIT charts based on message error rate,” in *IEEE International Symposium on Information Theory*, p. 454, 2002.
- [83] J. Lu, J. M. F. Moura, “Turbo design for LDPC with large girth,” in *IEEE workshop on Signal Processing Advanced in Wireless Communications*, pp. 90–94, 15-18 June 2003.
- [84] H. Zhang and J. M. F. Moura, “The design of structured regular LDPC codes with large girth,” in *IEEE Global Telecommunications Conference*, vol. 7, pp. 4022–4027, 1-5 December 2003.
- [85] H. Zhang and J. M. F. Moura, “Large-girth LDPC codes based

- on graphical models,” in *IEEE workshop on Signal Processing Advances in Wireless Communications*, pp. 100–104, 15-18 June 2003.
- [86] J. M. F. Moura, J. Lu and H. Zhang, “Structured low-density parity-check codes,” *IEEE Signal Processing Magazine*, vol. 21, pp. 42–55, January 2004.
- [87] M. Lentmaier, “Soft iterative decoding of generalized low-density parity-check codes based on MAP decoding of component hamming codes,” *Diploma Thesis, University of Ulm, Germany*, 1997.
- [88] T. Zhang and K. K. Parhi, “A class of efficient-encoding generalized low-density parity-check codes,” in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 4, pp. 2477 – 2480, 7-11 May 2001.
- [89] T. Zhang and K. K. Parhi, “High-performance, low-complexity decoding of generalized low density parity-check codes,” in *IEEE Global Telecommunications Conference*, vol. 1, pp. 181–185, 25-29 November 2001.
- [90] S. Hirst and B. Honary, “Decoding of generalized low-density parity-check codes using weighted bit-ip voting,” in *IEE Proceedings on Communications*, vol. 149, pp. 1 – 5, 2002.
- [91] S. Hirst and B. Honary, “Application of efficient Chase algorithm in decoding of generalized low-density parity-check codes,” in *IEEE Communications Letters*, vol. 6, pp. 385 – 387, 2002.
- [92] J. Boutros, O. Pothier and G. Zemor, “Generalized low density

- (Tanner) codes,,” in *IEEE International Conference on Communications*, vol. 1, pp. 441 – 445, 6-10 June 1999.
- [93] Y. Kou, S. Lin and M. Fossorier, “Low density parity check codes: construction based on finite geometries,” in *IEEE Global Telecommunications Conference*, vol. 2, pp. 825–829, 27 Nov. – 1 Dec 2000.
- [94] Y. Kou, S. Lin and M. Fossorier, “Low-density parity-check codes based on finite geometries: a rediscovery and new results,” *IEEE Transactions on Information Theory*, vol. 47, pp. 2711–27369, November 2001.
- [95] J. Xu, H. Tang, Y. Kou, S. Lin and K. Abdel-Ghafar, “A general class of LDPC finite geometry codes and their performance,” in *IEEE International Symposium on Information Theory*, p. 309, November 2002.
- [96] H. Tang, J. Xu, Y. Kou, S. Lin and K. Abdel-Ghafar, “On algebraic construction of Gallager and circulant low-density parity-check codes,” *IEEE Transactions on Information Theory*, vol. 50, pp. 1269–1279, June 2004.
- [97] H. Tang, J. Xu, Y. Kou, S. Lin and K. Abdel-Ghafar, “On circulant low density parity check codes,” in *IEEE International Symposium on Information Theory*, p. 200, 2002.
- [98] Z. Liu and D. A. Pados, “Low complexity decoding of finite geometry LDPC codes,” in *IEEE International Conference on Communications*, vol. 4, pp. 2713–2717, 11-15 May 2003.
- [99] I. B. Djordjevic and B. Vasic, “Projective geometry LDPC codes

- for ultralong-haul WDM highspeed transmission,” *IEEE Photonics Technology Letters*, vol. 15, pp. 784–786, May 2003.
- [100] O. Milenkovic, I. B. Djordjevic, B. Vasic, “Block-circulant low-density parity-check codes for optical communication systems,” *IEEE Journal of Selected Topics in Quantum Electronics*, vol. 10, pp. 294 – 299, March-April 2004.
- [101] B. Ammar, B. Honary, Y. Kou, J. Xu and S. Lin, “Construction of low-density parity-check codes based on balanced incomplete block designs,” *IEEE Transactions on Information Theory*, vol. 50, pp. 1257–1269, June 2004.
- [102] J. Rosenthal and P. O. Vontobel, “Constructions of regular and irregular LDPC codes using Ramanujan graphs and ideas from Margulis,” in *International Symposium on Information Theory*, p. 4, 24-29 June 2004.
- [103] K. S. Kim, S. H. Lee, Y. H. Kim and J. Y. Ahn, “Design of binary LDPC code using cyclic shift matrices,” *Electronics Letters*, vol. 40, pp. 325–326, March 2004.
- [104] T. Okamura, “Designing LDPC codes using cyclic shifts,” in *IEEE International Symposium on Information Theory*, p. 151, 29 June - 4 July 2003.
- [105] T. J. Richardson and R. L. Urbanke, “The Capacity of Low Density Parity Check Codes under Message-Passing Decoding,” *IEEE Transactions on Information Theory*, vol. 47, pp. 599–618, Feb 2001.

- [106] T. J. Richardson and R. L. Urbanke, “Efficient Encoding of Low-Density Parity-Check Codes,” *IEEE Transactions on Information Theory*, vol. 47, pp. 638 – 656, February 2001.
- [107] D. A. Spielman, “Linear-time encodable and decodable error-correcting codes,” *IEEE Transactions on Information Theory*, vol. 42, pp. 1723–1731, November 1996.
- [108] O. Pothier, L. Brunel and J. Boutros, “A low complexity FEC scheme based on the intersection of interleaved block codes,” in *IEEE 49th Vehicular Technology Conference*, vol. 1, pp. 274 – 278, 16-20 May 1999.
- [109] J. Zhang, M. P. C. Fossorier, “A Modified Weighted Bit-Flipping Decoding of Low-Density Parity-Check Codes,” *IEEE Communications Letters*, vol. 8, pp. 165–167, March 2004.
- [110] A. Nough, A. H. Banihashemi, “Bootstrap decoding of low-density parity-check codes,” *IEEE Communications Letters*, vol. 6, pp. 391–393, September 2002.
- [111] T. Richardson, R. Urbanke, “The Capacity of Low-Density Parity Check Codes Under Message-Passing Decoding,” *IEEE Transaction on Information Theory*, vol. 47, pp. 599–618, Feb 2001.
- [112] L. Hanzo, P. Cherriman, J. Streit, “Wireless Video Communications: Second to Third Generation Systems and Beyond,” *Wiley & IEEE*, 2001.
- [113] A. V. Oppenheim and A. S. Willsky, “Signal and Systems,” *Prentice Hall*, pp. 177–284, 1999.

- [114] S. ten Brink, “Convergence Behavior of Iteratively Decoded Parallel Concatenated Codes,” *IEEE Transactions on Communications*, vol. 49, pp. 1727–1737, Oct 2001.
- [115] S. Chae and Y. Park, “Low complexity encoding of regular low density parity check codes,” in *IEEE Vehicular Technology Conference*, vol. 3, pp. 1822 – 1826, 6-9 Oct 2003.
- [116] T. Clevorn, S. Godtmann and P. Vary, “BER Prediction Using EXIT Charts For BICM With Iterative Decoding,” *IEEE Communications Letters*, vol. 10, p. 49–51, January 2006.
- [117] C. Kuo, T. W. Cadman and J. R. Arsenault, “Sequential Random Integer Generator,” *Computer Physics Communications*, vol. 12, p. 163–171, May 1996.
- [118] A. Fog, “Chaotic Random Number Generators with Random Cycle Lengths,” <http://www.agner.org/random/theory>, December 2004.
- [119] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, D. A. Spielman and V. Stemann, “Practical loss-resilient codes,” in *Proceedings of 29th Annual ACM Symp. Theory of Computing*, pp. 150– 159, 1997.
- [120] Joachim Hagenauer, Elke Offer and Lutz Papke, “Iterative decoding of binary block and convolutional codes,” *IEEE Transactions on Information Theory*, vol. 42, pp. 429–445, March 1996.
- [121] S. Y. Chung, G. D. Jr. Forney, T. J. Richardson and R. Urbanke, “On the design of low-density parity-check codes within 0.0045

- dB of the Shannon limit,” *IEEE Communications Letters*, vol. 5, pp. 58–60, February 2001.
- [122] M. Matsumoto and T. Nishimura, “Mersenne Twister: A623-Dimensionally Equidistributed Uniform Pseudo-Random Number Generator,” *ACM Trans, Modeling and Computer Simulation*, vol. 8, no. 1, pp. 31–42, 1998.
- [123] T. Zhang and K. K. Parhi, “VLSI implementation-oriented (3,k)-regular low-density parity-check codes,” in *IEEE Workshop on Signal Processing Systems*, vol. 2, pp. 25–36, 26-28 Sept 2001.
- [124] P. W. Wolniansky, G. J. Foschini, G. D. Golden and R. A. Valenzuela, “V-BLAST: An Architecture for Realizing Very High Data Rates over The Rich-Scattering Wireless Channel,” *International Symposium on Signals, Systems, and Electronics*, pp. 295–300, 1998.
- [125] S. M. Alamouti, “A Simple Transmit Diversity Technique for Wireless Communications,” *IEEE Journal on Selected Areas in Communications*, vol. 16, pp. 1451–1458, October 1998.
- [126] TIA 45.5 Subcommittee, “The cdma2000 Candidate Submission,” tech. rep., Draft, 1998.
- [127] R. Wichman and A. Hottinen, “Transmit diversity WCDMA system,” tech. rep., Nokia Research Center, 1998.
- [128] T. H. Liew, B. J. Choi and L. Hanzo, “Comparative Study of Concatenated Turbo Coded and Space-Time Block Coded as well as

- Space-Time Trellis Coded OFDM,” *IEEE Vehicular Technology Conference*, p. 107 (CDROM), May 2001.
- [129] S. Lin, D. J. Costello and M. J. Miller, “Automatic-Repeat-Request Error-Control Schemes,” *Communications Magazine, IEEE*, vol. 22, pp. 5–17, December 1984.
- [130] M. Miller and S. Lin, “The Analysis of Some Selective-Repeat ARQ Schemes with Finite Receiver Buffer,” *IEEE Transactions on Communications*, vol. 29, pp. 1307–1315, September 1981.
- [131] D. Towsley, “A Statistical Analysis of ARQ Protocols Operating in a Nonindependent Error Environment,” *IEEE Transactions on Communications*, vol. 29, pp. 971–981, July 1981.
- [132] G. Benelli, “Throughput Optimisation of a Stop-And-Wait ARQ Protocol,” *Electronics Letters*, vol. 24, pp. 735–736, June 1988.
- [133] C. Pimentel and R. L. Siqueira, “Analysis of the Go-Back-N Protocol on Finite State Markov Fading Channels,” *Vehicular Technology Conference, 2006. VTC 2006-Spring. IEEE 63rd*, vol. 4, pp. 2042–2046, May 2006.
- [134] G. Caire, G. Taricco and E. Biglieri, “Bit-Interleaved Coded Modulation,” *IEEE Transactions on Information theory*, vol. 44, pp. 927–946, May 1998.
- [135] D. Pham, K. R. Pattipati, P. K Willet and J. Luo, “An Improved Complex Sphere Decoder for V-Blast Systems,” *IEEE Signal Processing Letters*, vol. 11, pp. 748–751, 2004.

- [136] G. D. Golden, G. J. Foschini, R. A. Valenzuela and P. W. Wolniansky, “Detection Algorithms and Initial Laboratory Results using V-BLAST Space-Time Communication Architecture,” *IEE Electronics Letters*, vol. 35, pp. 14–16, January 1999.
- [137] T. Clevorn, S. Godtmann and P. Vary, “BER Prediction Using EXIT Charts For BICM With Iterative Decoding,” *IEEE Communications Letters*, vol. 10, pp. 49–51, January 2006.
- [138] M. Sabbaghian, D. Falconer, “Comparison between Convolutional and LDPC Code-based Turbo Frequency Domain Equalization,” *IEEE International Conference on Communications*, vol. 12, pp. 5432–5437, June 2006.
- [139] L. Hanzo, T. H. Liew and B. L. Yeap, *Turbo Coding, Turbo Equalisation and Space-Time Coding for Transmission over Fading Channels*. Wiley & IEEE Press, 2002.
- [140] F. Shreckenbach, P. Henkel, N. Görtz and G. Bauch, “Analysis and Design of Mappings for Iterative Decoding of BICM,” *XI National Symposium of Radio Sciences, Poznan*, pp. 82–87, April 2005.
- [141] G. Caire, G. Taricco and E. Biglieri, “Bit-Interleaved Coded Modulation,” *IEEE Transactions on Information Theory*, vol. IT-44, pp. 927–946, May 1998.
- [142] A. Chindapol, J. A. Ricey, “Design, Analysis, and Performance Evaluation for BICM-ID with Square QAM Constellations in Rayleigh Fading Channels,” *IEEE Journal on Selected Areas in Communications*, vol. 19, pp. 944–957, May 2001.

- [143] S. Y. L. Goff, “Signal Constellations for Bit-Interleaved Coded Modulation,” *IEEE Transactions on Information Theory*, vol. 49, pp. 307–313, January 2003.
- [144] S. Kallel, “Analysis of a Type II Hybrid ARQ Scheme with Code Combining,” *IEEE Transactions on Communications*, vol. 38, pp. 1133–1137, August 1990.
- [145] A. Shiozaki, K. Okuno, K. Suzuki and T. Segawa, “A Hybrid ARQ Scheme with Adaptive forward Error Correction for Satellite Communications,” *IEEE Transactions on Communications*, vol. 39, pp. 482–484, April 1991.
- [146] F. Babich, “Performance of Hybrid ARQ Schemes for the Fading Channel,” *IEEE Transactions on Communications*, vol. 50, pp. 1882–1885, December 2002.
- [147] R. H. Deng, “Hybrid ARQ Schemes Employing Coded Modulation and Sequence Combining,” *IEEE Transactions on Communications*, vol. 42, pp. 2239–2245, June 1994.
- [148] Q. Luo and P. Sweeney, “Hybrid-ARQ Protocols Based on Multilevel Coded Modulation,” *IEE Electronics Letters*, vol. 39, pp. 1063–1065, July 2003.
- [149] N. H. Tran and H. H. Nguyen, “Signal Mappings of 8-ary Constellations for Bit Interleaved Coded Modulation with Iterative Decoding,” *IEEE Transactions on Broadcasting*, vol. 52, pp. 92–99, March 2006.
- [150] L. F. Wei, “Coded Modulation with Unequal Error Protection,”

- IEEE Transactions on Communications*, vol. 41, pp. 1439–1450, October 1993.
- [151] F. Guo, S. X. Ng and L. Hanzo, “LDPC assisted Block Coded Modulation for Transmission over Rayleigh Fading Channels,” in *IEEE Vehicular Technology Conference*, vol. 3, (Florida, USA), pp. 1867–1871, April Spring 2003.
- [152] A. Avudainayagam, J. M. Shea and A. Roongta, “Improving the Efficiency of Reliability-based Hybrid-ARQ with Convolutional Codes,” *Military Communications Conference - MILCOM. IEEE, Atlantic City*, vol. 1, pp. 448–454, October 2005.
- [153] J. Hamorsky and L. Hanzo, “Performance of the Turbo Hybrid Automatic Repeat Request System Type II,” *ITW 1999, Metsovo, Greece, 27 June - 1 July*, p. 51, 1999.
- [154] R. H. Deng, “A Hybrid ARQ Scheme with Adaptive forward Error Correction for Satellite Communications,” *IEEE Transactions on Communications*, vol. 42, pp. 2239–2245, June 1994.
- [155] G. Lechner, J. Sayir and I. Land, “Optimization of LDPC Codes for Receiver Frontends,” *International Symposium on Information Theory*, pp. 2388–2392, July 2006.
- [156] A. Ashikhmin, G. Kramer and S. ten Brink, “Extrinsic Information Transfer Functions: Model and Erasure Channel Properties,” *IEEE Transactions on Information Theory*, vol. 50, pp. 2657–2673, Nov 2004.
- [157] E. Sharon, A. Ashikhmin and S. Litsyn, “Analysis of Low-Density

Parity-Check Codes Based on EXIT Functions,” *IEEE Transactions on Communications*, vol. 54, pp. 1407–1414, Aug 2006.

Phụ lục A

MÔ PHỎNG ĐÁNH GIÁ MÃ LDPC

Ảnh hưởng của số lần lặp giải mã LDPC

Bảng A.1 là các thông số của mã LDPC được sử dụng trong phần mô phỏng này. Dữ liệu được thực hiện điều chế và giải điều chế BPSK. Tương tự như các họ mã Turbo, khả năng sửa lỗi của mã LDPC sẽ tăng lên khi ta tăng số lần lặp trao đổi thông tin giữa các nút biến số và nút kiểm tra trong đồ thị Bipartite của mã LDPC, nhưng đồng thời điều này có nghĩa là độ phức tạp tính toán của bộ giải mã LDPC cũng tăng lên. Trong phần mô phỏng này chúng ta sử dụng các mã LDPC có cùng tốc độ bit R , nhưng có độ dài từ mã khác nhau. Hàm trọng của các cột trong ma trận kiểm tra tương ứng bằng 3. Để tiện theo dõi chúng ta sử dụng kí hiệu G và UR viết tắt cho các kênh AWGN và pha đỉnh Rayleigh không tương quan tương ứng.

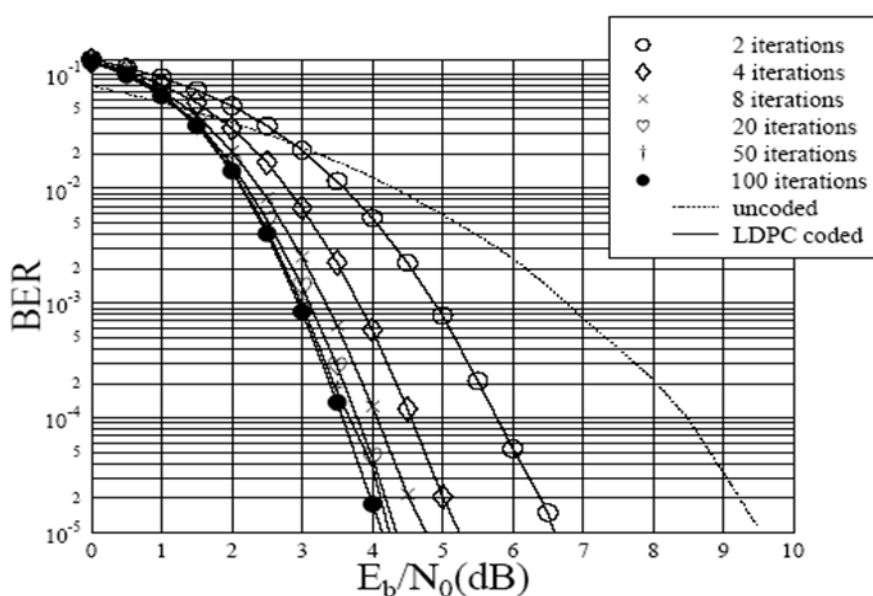
Bảng A.1: Các thông số của mã LDPC cho mô phỏng ảnh hưởng của số lần lặp cực đại tới khả năng sửa lỗi của mã LDPC

Mã LDPC	Kênh	Số lần lặp giải mã cực đại	Hàm trọng của các cột trong ma trận kiểm tra
(500,1000)	AWGN	2, 4, 8, 20, 50, 100	3
(250,500)	AWGN	2, 4, 8, 20, 50, 100	3
(100,200)	AWGN	2, 4, 8, 20, 50, 100	3
(500,1000)	Pha đỉnh Rayleigh không tương quan	2, 4, 8, 20, 50, 100	3
(250,500)	Pha đỉnh Rayleigh không tương quan	2, 4, 8, 20, 50, 100	3
(100,200)	Pha đỉnh Rayleigh không tương quan	2,4,8,20,50,100	3

Hình (A.1 - A.5) minh họa tỉ số lỗi bit BER (Bit Error Ratio) theo năng lượng bit trên nhiễu E_b/N_0 khi sử dụng các mã LDPC có thông số như trong bảng 5 và kênh truyền là AWGN và Rayleigh không tương quan. Như trong các kết quả mô phỏng, ta có thể thấy khả năng sửa lỗi của các mã LDPC(100,200), LDPC(250,500) và LDPC(500,1000) phụ thuộc vào số lần lặp giải mã cực đại. Ví dụ mã LDPC(100,200) chỉ yêu cầu tỉ số E_b/N_0 vào khoảng 4,3 dB để đạt được tỉ số lỗi bit $BER \geq 10^{-5}$ với số lần lặp giải mã cực đại là 20, khi truyền dữ liệu qua kênh AWGN. Trong khi đó, cũng với mã này cần tỉ số E_b/N_0 đến 9,8 dB để đạt được cùng tỉ số BER như trên khi số lần lặp giải mã cực đại là 2, khi truyền dữ liệu qua kênh AWGN. Các kết quả phân tích tương tự cho các mã

LDPC(250,500) và LDPC(500,1000). Từ các hình minh họa mô phỏng trên ta cũng nhận thấy kênh truyền dẫn Rayleigh không tương quan can nhiễu nhiều hơn đến dữ liệu truyền so với kênh AWGN.

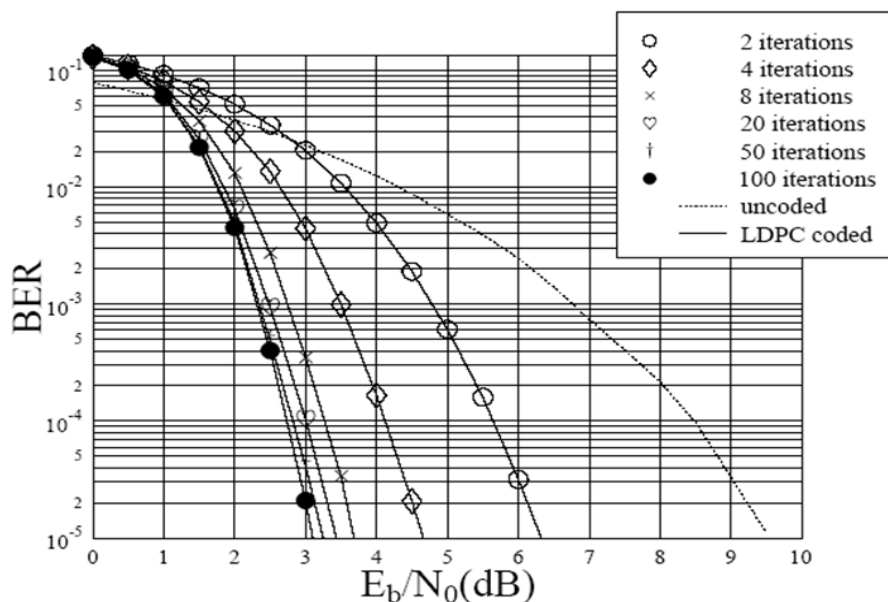
Độ tăng ích của 3 mã LDPC có thông số cho trong bảng A.1



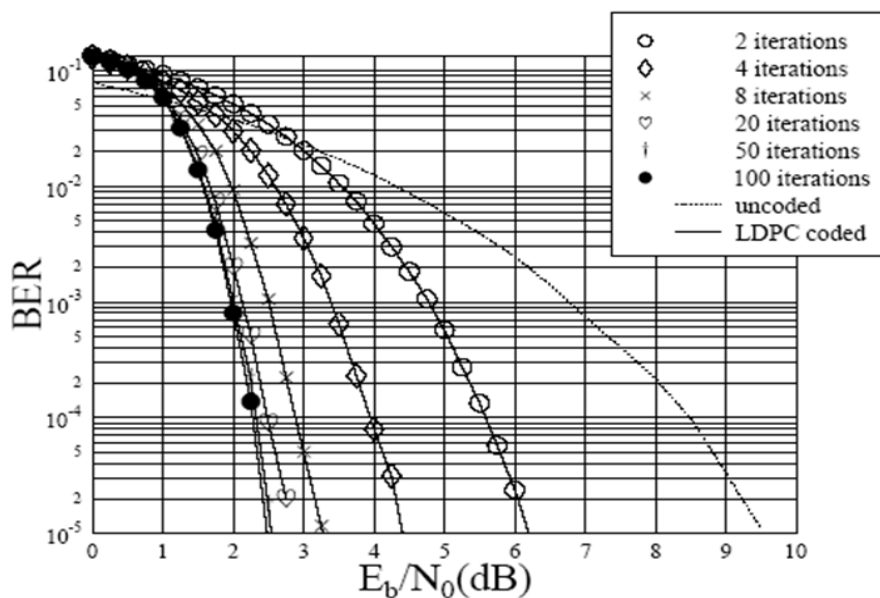
Hình A.1: Ảnh hưởng của số lần lặp cực đại tới khả năng sửa lỗi của mã LDPC(100,200), khi truyền dữ liệu qua kênh AWGN, điều chế BPSK.

tại $BER = 10^{-4}$, phụ thuộc vào số lần lặp giải mã cực đại như vẽ trong hình A.7. Từ hình A.7 ta có thể thấy 3 mã LDPC(100,200), LDPC(250,500) và LDPC(500,1000) có độ tăng ích đạt bão hòa sau khoảng 20 lần lặp. Vì vậy khả năng sửa lỗi của ba mã LDPC này đạt được tỉ số BER nhỏ nhất tương ứng với số lần lặp giải mã cực đại bé nhất là 20 lần. Sau 20 lần lặp giải mã khả năng sửa lỗi của 3 mã LDPC không tăng, trong khi đó số lượng phép tính tăng lên. Vì vậy số lần lặp giải mã cực đại tối ưu cho ba mã LDPC ở trên là 20 lần.

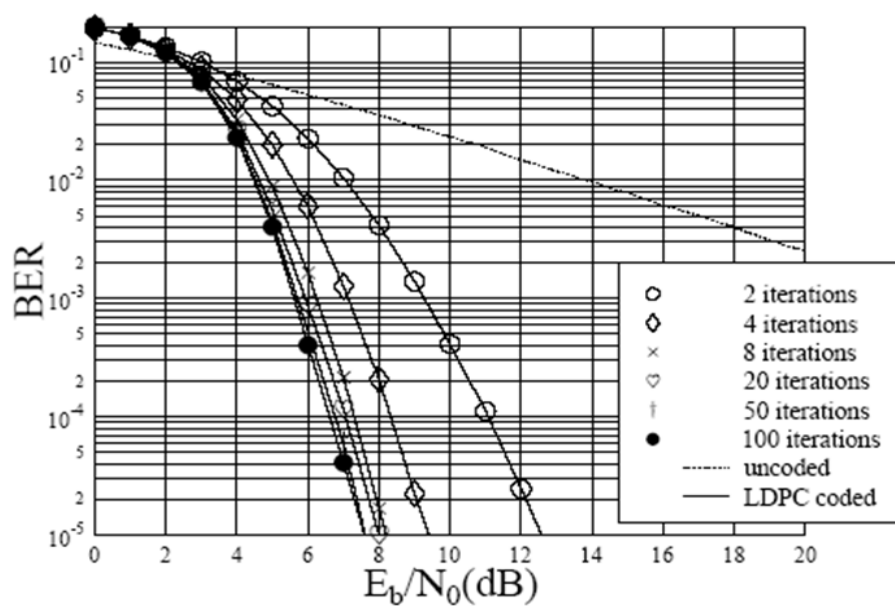
So sánh ba đường cong thể hiện hiệu quả sử dụng số lần lặp khác nhau của ba mã LDPC có thông số cho trong bảng A.1, được vẽ trong



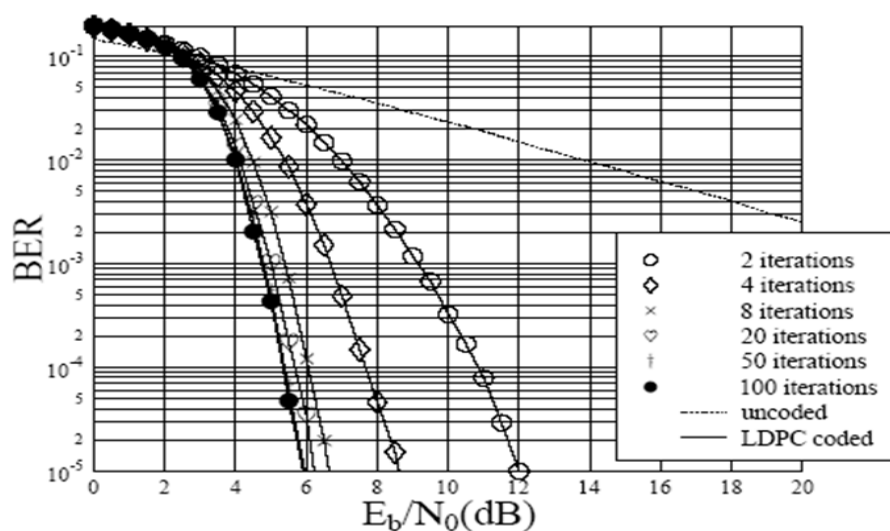
Hình A.2: Ảnh hưởng của số lần lặp cực đại tới khả năng sửa lỗi của mã LDPC(250,500), khi truyền dữ liệu qua kênh AWGN, điều chế BPSK.



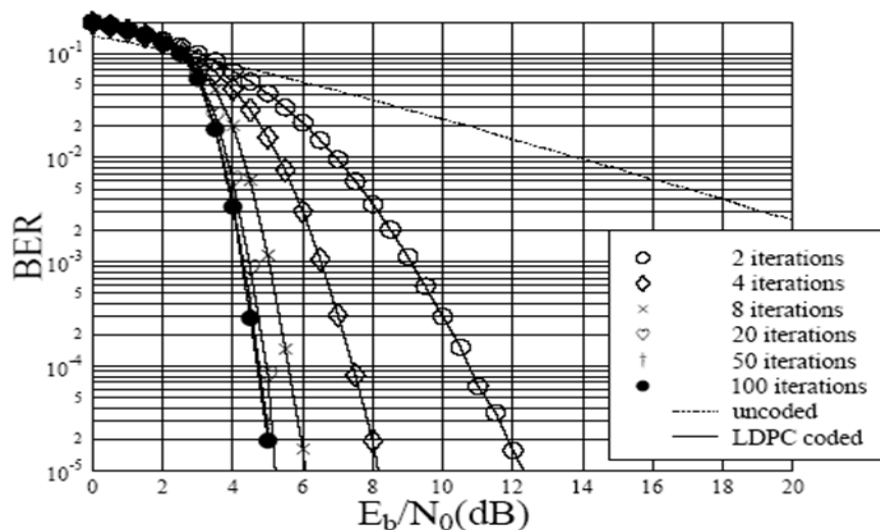
Hình A.3: Ảnh hưởng của số lần lặp cực đại tới khả năng sửa lỗi của mã LDPC(500,1000), khi truyền dữ liệu qua kênh AWGN, điều chế BPSK.



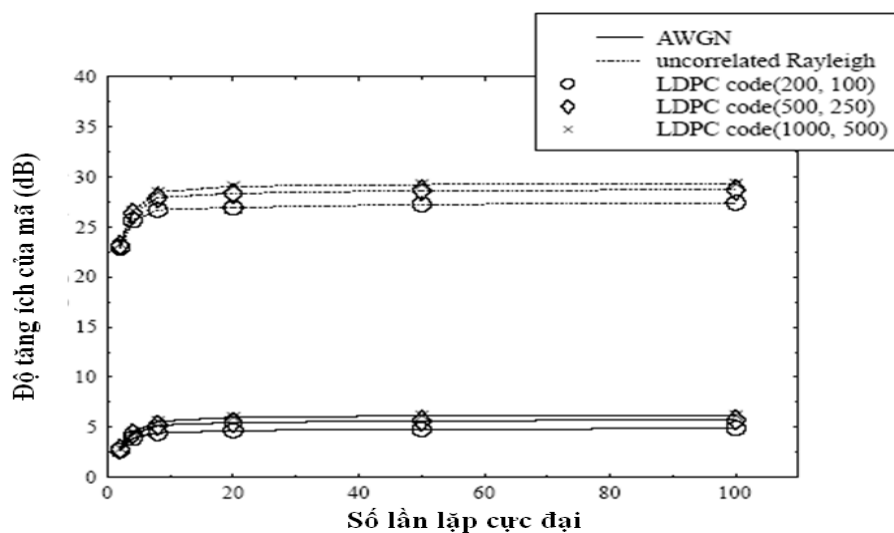
Hình A.4: Ảnh hưởng của số lần lặp cực đại tới khả năng sửa lỗi của mã LDPC(100,200), khi truyền dữ liệu qua kênh pha đình Rayleigh không tương quan, điều chế BPSK.



Hình A.5: Ảnh hưởng của số lần lặp cực đại tới khả năng sửa lỗi của mã LDPC(250,500), khi truyền dữ liệu qua kênh pha đình Rayleigh không tương quan, điều chế BPSK.



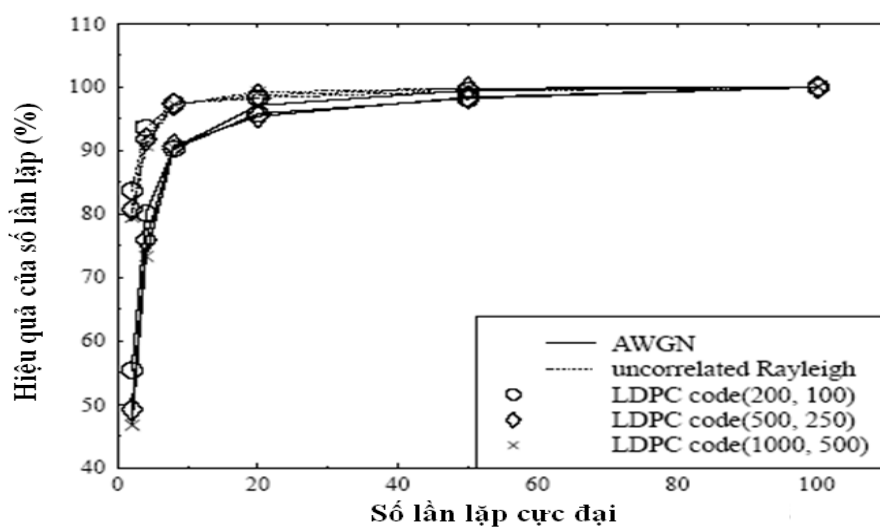
Hình A.6: Ảnh hưởng của số lần lặp cực đại tới khả năng sửa lỗi của mã LDPC(500,1000), khi truyền dữ liệu qua kênh pha định Rayleigh không tương quan, điều chế BPSK.



Hình A.7: Độ tăng ích của 3 mã LDPC có các thông số trong bảng A.1 tại $BER = 10^{-4}$, khi dữ liệu được điều chế BPSK và kênh truyền dẫn là AWGN và Rayleigh không tương quan.

Bảng A.2: Yêu cầu tỉ số E_b/N_0 với số lần lặp giải mã cực đại khác nhau để đạt được tỉ số BER = 10^{-4} tương ứng với 3 mã LDPC có thông số cho trong bảng A.1, khi truyền dữ liệu qua kênh AWGN và kênh pha đỉnh Rayleigh không tương quan.

Mã LDPC	Số lần lặp cực đại	Tỉ số E_b/N_0 yêu cầu (AWGN) (dB)	Tỉ số E_b/N_0 yêu cầu (Pha đỉnh Rayleigh không tương quan) (dB)
(100,200)	2	5,765	11,059
	4	4,559	8,353
	8	4,059	7,294
	20	3,794	7,059
	50	3,676	6,745
	100	3,588	6,589
(250,500)	2	5,647	10,824
	4	4,118	7,647
	8	3,265	6,509
	20	3,000	5,647
	50	2,824	5,353
	100	2,735	5,294
(500,1000)	2	5,588	10,706
	4	3,941	7,412
	8	2,882	5,529
	20	2,470	4,941
	50	2,324	4,764
	100	2,294	4,706
Không sử dụng mã	8	8,47	34



Hình A.8: Hiệu quả số lần lặp giải mã khác nhau đối với 3 mã LDPC có thông số trong bảng A.1, khi truyền dữ liệu qua các kênh AWGN và pha đỉnh Rayleigh không tương quan.

hình A.8, ta có thể nhận thấy khả năng sửa lỗi của cả ba mã có độ dài từ mã khác nhau này đều tăng khi số lần lặp giải mã cực đại tăng. Theo như hình A.8 với số lần lặp giải mã cực đại là 8 lần, cả ba mã LDPC đều đạt độ tăng ích mã trên 90%, đối với kênh AWGN và 95% đối với kênh pha đỉnh Rayleigh không tương quan.

Ảnh hưởng của độ dài từ mã LDPC

Dưới đây chúng ta nghiên cứu ảnh hưởng của độ dài từ mã đến khả năng sửa lỗi của mã LDPC, khi truyền dữ liệu qua các kênh AWGN và pha đỉnh Rayleigh không tương quan, sử dụng kiểu điều chế BPSK. Thông số của 4 mã LDPC có tỉ lệ mã $\frac{1}{2}$, có độ dài từ mã khác nhau trong phần mô phỏng này được cho trong bảng A.3. Mô phỏng này được thực hiện bằng cách so sánh tỉ lệ lỗi khung (hay từ mã) FER (Frame Error Ratio) theo tỉ số E_b/N_0 của các mã LDPC có thông số cho trong

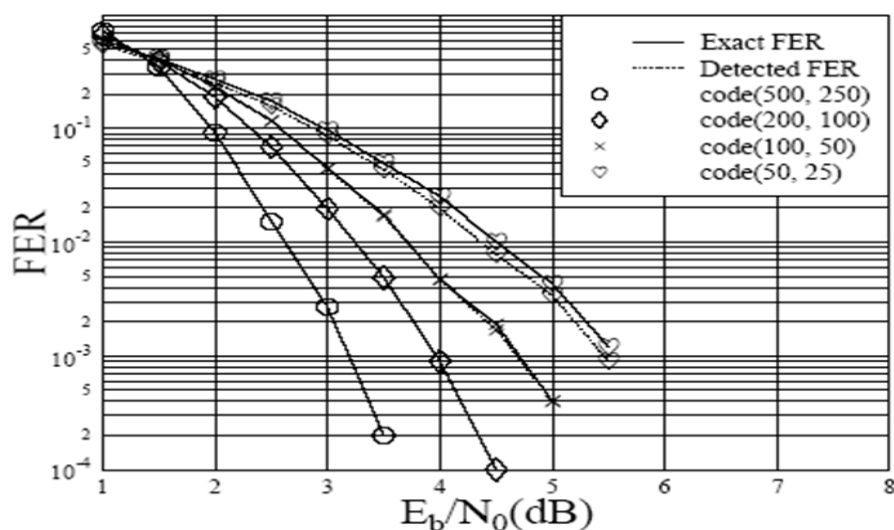
bảng A.3.

Bảng A.3: Các thông số mã LDPC được sử dụng trong mô phỏng ảnh hưởng của độ dài từ mã đến khả năng sửa mã.

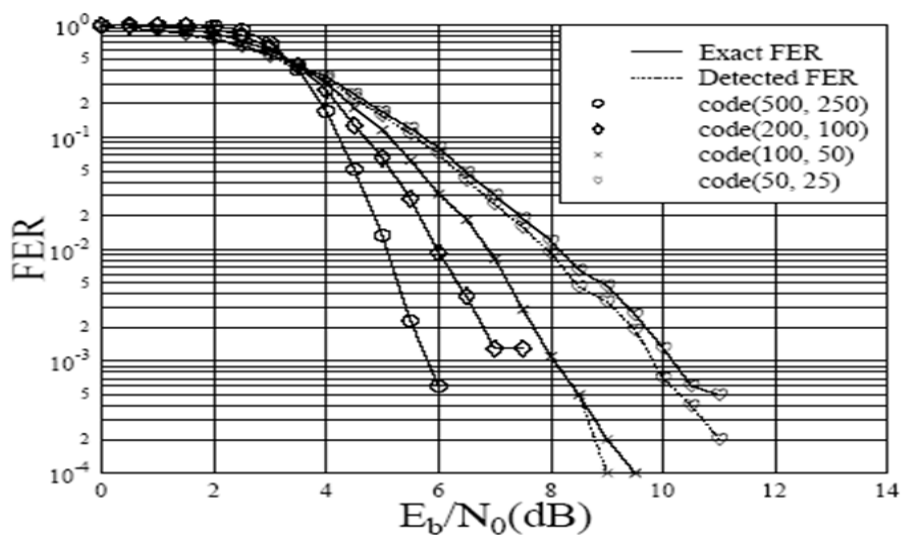
Mã LDPC	Kênh	Số lần lặp giải mã cực đại	Số từ mã được sử dụng cho mô phỏng
LDPC(25,50)	AWGN	25	10000
LDPC(50,100)	AWGN	25	10000
LDPC(100,200)	AWGN	25	10000
LDPC(250,500)	AWGN	25	10000
LDPC(25,50)	Rayleigh	25	10000
LDPC(50,100)	Rayleigh	25	10000
LDPC(100,200)	Rayleigh	25	10000
LDPC(250,500)	Rayleigh	25	10000

Hình A.9 và A.10 là kết quả mô phỏng ảnh hưởng của độ dài từ mã đến khả năng sửa lỗi khung FER của các mã LDPC có thông số cho trong bảng 6, khi điều chế BPSK và kênh truyền dẫn tương ứng là AWGN và pha đỉnh Rayleigh không tương quan. Như trong hình A.9 và A.10 độ tăng ích của mã LDPC tăng đến 3 dB trong kênh AWGN và 5 dB trong kênh Rayleigh khi sử dụng mã LDPC(25,50) và LDPC(250,500) tương ứng.

Nói tóm lại khả năng sửa lỗi của mã LDPC tăng khi độ dài từ mã tăng là do khoảng cách cực tiểu giữa các cột trong ma trận kiểm tra của mã LDPC tăng khi độ dài từ mã LDPC tăng lên, khi và chỉ khi hàm trọng của các cột trong ma trận kiểm tra không vượt quá 3.



Hình A.9: Ảnh hưởng của độ dài từ mã đến khả năng sửa lỗi FER của 4 mã LDPC theo tỉ số E_b/N_0 có thông số cho trong bảng A.3, khi truyền dữ liệu qua kênh truyền AWGN và sử dụng kiểu điều chế BPSK.



Hình A.10: Ảnh hưởng của độ dài từ mã đến khả năng sửa lỗi FER của 4 mã LDPC theo tỉ số E_b/N_0 có thông số cho trong bảng A.3, khi truyền dữ liệu qua kênh truyền pha đình Rayleigh không tương quan và sử dụng kiểu điều chế BPSK.