Doctoral Dissertation
(Shinshu University)

Differentially Encoded LDPC Coded Systems with Iterative Multiple-Symbol Differential Detection

March 2013

Yang Yu

SHINSHU UNIVERSITY
Nagano, Japan

Differentially Encoded LDPC Coded Systems with Iterative Multiple-Symbol Differential Detection

A dissertation submitted in partial satisfaction
of the requirements for the degree
Doctor of Engineering

by

Yang Yu

March 2013

Dedicated to my parents and wife

Table of Contents

List of Figures

List of Tables

Acknowledgments

I would like to express my heartfelt gratitude and appreciation to my advisor Prof. Shiro Handa for his active guidance and support throughout my stay at Shinshu University.

I would also like to thank Dr. Fumihito Sasamori and Dr. Osamu Takyu for their support and helpful advice.

I am grateful to Mr. Kimiaki Takizawa for his assistance.

I am grateful to all the current and former lab-mates in Handa-Sasamori-Takyu laboratory for their friendly and generous help.

Finally, I would like to thank my parents, wife, sister and friends. Without their support and love I would have not been the person I am, and the completion of this thesis would not be possible.

Abstract

Low-Density Parity-Check (LDPC) codes have attracted significant attention since they have several advantages compared to other error correcting codes (ECCs), such as simple structure, low decoding complexity and outstanding performance. Many reports have confirmed the amazing performance of LDPC codes with coherent detection. However, since the performance of coherent detection relies on accurate phase tracking and reliable estimation of channel state information (CSI), coherent detection becomes expensive or infeasible in some cases. In this case, differential detection, which circumvents the need for phase tracking and channel estimation, is an attractive scheme. In addition, to compensate the performance loss caused by differential detection, multiple-symbol differential detection (MSDD) has been proposed and proven to be an effective approach. On the other hand, wireless communication systems with ECCs can achieve a performance close to the channel capacity with accepted decoding complexity by employing the coded modulation scheme and the iterative decoding technique.

Against this background, in this thesis, we focus on improving the performance of LDPC coded systems without using synchronization information and CSI. We propose an iterative MSDD scheme for differentially encoded LDPC (DE-LDPC) coded systems, and study in detail how to optimize the system performance and how to reduce the decoding complexity of the system. The main contributions of this paper are the following four aspects:

1) We propose an iterative MSDD scheme for DE-LDPC coded systems and analyze the characteristics and performance of the studied systems. In the studied systems, the outer iterative decoding is performed between the MSDD soft-input soft-output demodulator (SISOD) and the LDPC decoder. The transfer characteristics of the MSDD SISOD and the LDPC decoder are analyzed by the extrinsic information transfer (EXIT) chart. It is shown that the performance can be improved by increasing the observation window size (OWS) of the MSDD SISOD and the number of outer iterations. Whereas, the performance gain cannot be achieved by iterative decoding when the inner decoder employs the conventional differential detection. The bit error rate (BER) performance of the systems under consideration is evaluated in slow and fast Rayleigh fading channels, respectively. It is shown that the analysis results obtained from the EXIT charts are supported by the computer simulation results.

2) In order to improve the performance of the studied systems with large OWS, the irregular LDPC codes are optimized for the MSDD SISOD with different OWS. We optimize irregular LDPC codes by using the EXIT chart to optimize the degree distributions of them. The simulation results demonstrate that the performance of the DE-LDPC coded systems with a large OWS can be significantly improved by using the optimized irregular LDPC codes. Moreover, when the optimized irregular LDPC codes with very long length, the performance of the systems under consideration can close to the capacity of the noncoherent AWGN channel.

3) To solve the problem of high complexity of the MSDD SISOD, which becomes prohibitively high as the OWS and the order of the modulation become large, we propose an improved soft-output M algorithm (ISOMA) by combining the features of the existing SOMA approaches. The proposed ISOMA can guarantee that the LLR of each coded bit can be computed with high reliability by using only small number of search paths. By computer simulations, it is shown that the computational complexity of the MSDD SISOD as well as the iterative decoding complexity of DE-LDPC coded systems with iterative MSDD can be significantly reduced by the proposed ISOMA. Compared with the existing SOMA approaches, the proposed ISOMA has better performance in terms of the BER performance and the ability of reducing the decoding complexity of DE-LDPC systems with iterative MSDD.

4) In order to further reduce the iterative decoding complexity and delay of the systems under consideration with finite-length LDPC codes, an adaptive iterative decoding approach (AIDA) which can adaptively adjust the OWS of the MSDD SISOD and the outer iteration number of the iterative decoder is proposed. In AIDA, the OWS and the outer iteration number are adaptively adjusted by using a SC to judge whether the iterative decoding converges or not. To circumvent the disadvantages of the existing SCs, a new SC, which we call DMI criterion, is proposed for tracking the convergence status of the iterative decoding by tracking the difference of the output mutual information of the LDPC decoder between two consecutive outer iterations of the considered systems. Simulation results show that AIDA with the proposed DMI criterion can significantly reduce the iterative decoding complexity and delay of the considered systems at all SNRs. Moreover, compared with the existing SCs, it is demonstrated that the DMI criterion is more effective for the considered systems in terms of reducing the average number of outer iterations, performance loss and robustness.

CHAPTER 1

Introduction

1.1 Back Ground and Motivation

Multipath fading characteristics of wireless channels are the important factor affecting the data transmission rate and quality of wireless communication systems. Especially in the case of users moving at high speed, the fast fading of wireless channels will seriously affect the performance of wireless communication systems. In the future, the wireless systems will require high speed and high quality data transmission even in fast fading channels. In order to meet this requirement, powerful error correcting codes (ECCs) need to be applied in wireless communication systems.

Low-density parity-check (LDPC) codes are a class of capacity-approaching linear block codes [1]-[4]. LDPC codes have several advantages compared to other channel coding codes. First, LDPC codes can achieve very good performance in data transmission with low decoding complexity. Second, the error performance of LDPC codes does not always exhibit an error floor. Furthermore, LDPC codes are more flexible in their construction in terms of the code rate and other parameters. Due to the advantages of LDPC codes, they have been adopted in current and next generation wireless standards, such as worldwide interoperability for microwave access (WiMAX), wireless fidelity (WiFi), second generation satellites for digital video broadcasting (DVB-S2), and also as a potential candidate for fourth-generation (4G) mobile wireless systems.

It should be noted that the amazing performance of LDPC codes confirmed by most of the reports in the literature is achieved by coherent detection. However, since the performance of coherent detection relies on accurate phase tracking and reliable estimation of channel state information (CSI), coherent detection becomes expensive or infeasible in some cases, such as rapid relative motion between transmitter and receiver or phase noise in local oscillators. Therefore, differentially encoded LDPC (DE-LDPC) coded systems with differential detection, which circumvent the need for phase tracking and channel estimation, have attracted a lot of attention [5]-[9].

It is well known that conventional differential detection has considerably worse performance than coherent detection. Multiple-symbol differential detection (MSDD) [10]-[11] has been proven to be an effective approach to compensate for this performance loss. By extending the length of observation window size to more than two symbols, MSDD makes a joint decision on those several symbols simultaneously rather than symbol-by-symbol detection as in conventional differential detection. Moreover, in continuous fading channels, MSDD considering fading autocorrelations can enhance the system performance possible. Up to now, MSDD for different communication systems

have been widely studied [12]-[18]. However, the high complexity of MSDD limits its application in practical systems.

On the other hand, wireless communication systems with ECCs need to employ the coded modulation scheme and the iterative decoding technique, in order to achieve a performance close to the channel capacity. That is, at the transmitter, ECCs and modulation schemes should be designed together for optimizing the system performance. On the other hand, at the receiver, the inner decoder and the outer decoder are soft-input soft-output (SISO) decoders. Moreover, the iterative decoding is performed between the two decoders to make the system performance close to the channel capacity with accepted decoding complexity.

On the basis of the above, in this thesis, the application of the iterative MSDD for DE-LDPC coded systems is considered. In the systems under consideration, the demodulator with MSDD is viewed as an inner decoder, and the iterative decoding is performed at the receiver. The objective of this thesis mainly includes three aspects:

(1) confirm the effectiveness of the iterative MSDD scheme for DE-LDPC coded systems, and analyze the impact of the parameters of iterative MSDD and LDPC codes on the system performance;

(2) optimize the parameters of the system to improve the system performance;

(3) develop the complexity reduction algorithms for iterative MSDD to greatly reduce the iterative decoding complexity and delay of the studied system.

1.2 Thesis Outline

The remainder of this thesis is organized as follows.

- In Chapters 2 and 3, the basics for the study of LDPC codes and MSDD are presented, respectively.
- In Chapter 4, the system model DE-LDPC coded systems with iterative MSDD is first illustrated. Then, the performance of the MSDD soft-input soft-output demodulator (SISOD) and the LDPC decoder is analyzed by the extrinsic information transfer (EXIT) chart. The effectiveness of the iterative MSDD scheme is then evaluated using computer simulations. Moreover, the importance of designing the irregular LDPCs for the MSDD SISOD with different observation window size (OWS) is discussed and confirmed.
- In Chapter 5, to solve the problem of high complexity of the MSDD SISOD, we propose an improved soft-output M algorithm (ISOMA) by combining the features of the existing SOMA approaches. The complexity of the MSDD SISOD is first discussed. Then, the idea and the principle of ISOMA are described. Finally, the effectiveness and advantages of ISOMA are confirmed and analyzed using computer simulations.

- In Chapter 6, an adaptive iterative decoding approach (AIDA) is proposed to further reduce the iterative decoding complexity and delay for finite-length DE-LDPC coded systems with iterative MSDD. The characteristics of the systems under consideration with finite-length LDPC codes are first analyzed by EXIT band chart. Then, the motivation and idea of the proposed AIDA are described. And then, a new stopping criterion (SC) called differential mutual information (DMI) criterion, which can track the convergence status of the iterative decoding, is proposed. Finally, the performance of AIDA using the DMI criterion for the systems under consideration is evaluated. Moreover, the performance comparison of the proposed DMI criterion and the existing SCs for AIDA is also presented.

- Finally, this thesis is concluded in Chapter 7.

CHAPTER 2

Low-Density Parity-Check Codes

LDPC codes are a class of linear block ECCs, which provide near-capacity performance on a large set of data-transmission and data-storage channels. They were invented by Robert Gallager in 1960 [1]. However, at that time, since the technology of the hardware could not satisfy the requirements needed by their encoding, they were ignored for more than 30 years. With the development of the Tanner graph [2] and Turbo codes, LDPC codes were rediscovered by Mackay and Neal in 1996 [3]. In the recent years, the study of LDPC codes has become the research hotspot in the research filed of ECCs. With the great progress in the study of LDPC codes, LDPC codes have been or will be used in many kinds of communications standards.

In this chapter, we provide an overview on LDPC codes as the basis for the following chapters. In section 2.1, we first introduce the representation of LDPC codes. In section 2.2, regular and irregular LDPC codes are briefly introduced. Then, encoding of LDPC codes is presented in 2.3. Finally, the iterative decoding of LDPC codes is described in 2.4.

2.1 Representations of LDPC Codes

LDPC codes are a special class of linear block codes, which can be represented by the party-check matrixes $\mathbf{H}$. The particularity of LDPC codes is that parity check matrixes $\mathbf{H}$ of LDPC codes are sparse matrixes, most elements of which are 0 and only a small part of elements are 1. LDPC codes can be denoted as (N, d_v, d_c), where N is the length of LDPC codes, d_v is the column weight, which represents the number of nonzero elements in a column of the parity-check matrix), and d_c is the row weight, which represents the number of nonzero elements in a row of the parity-check matrix. We illustrate the matrix representation of LDPC codes with a simple example as shown in Fig. 2.1, which shows the party-check matrix of a (10, 3, 6) LDPC code.

Except using the matrix to represent LDPC codes, we can also use the Tanner graph to represent LDPC codes. Fig. 2.2 shows the Tanner graph corresponding to the above (10, 3, 6) LDPC code. As shown in Fig. 2.2, the Tanner graph consists of two types of nodes and edges connecting only nodes of different types. The two types of nodes in the Tanner graph are called the variable nodes (VNs) and the check nodes (CNs) respectively. VNs correspond to the code bits, while CNs correspond to the parity-check equations. An edge connects a VN and a CN if that bit is 1 included in the corresponding parity-check equation. Therefore, the number of edges in the Tanner graph is equal to the number of ones in the parity-check matrix. The number of the

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

Figure 2.1: Matrix representation of the (10, 3, 6) LDPC code.

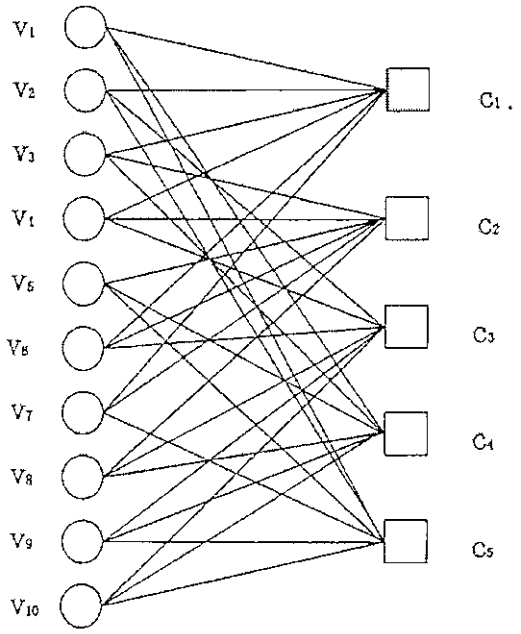

Figure 2.2: Tanner graph of the (10, 3, 6) LDPC code.

edges connecting a node is called the degree of the node. The degree of each VN is equal to the weight of its corresponding column, while the degree of each CN is equal to the weight of its corresponding row. In this thesis, we use d_v and d_c to denote the degree of VN and the degree of CN, respectively.

2.2 Regular and Irregular LDPC Codes

According to the number of the non-zero elements of the sparse matrix in the row and column, LDPC codes can be divided into regular LDPC codes and irregular LDPC codes. For a parity-check matrix $\mathbf{H}$, if the numbers of the non-zero elements of each row are the same, and the numbers of the non-zero elements of each column are also the same, the LDPC code corresponding to $\mathbf{H}$ is the regular LDPC code. On the contrary, if the above conditions do not satisfy, the LDPC code is the irregular LDPC code.

From [4], the basic structure of an LDPC code is defined by its degree distribution, which are two polynomials that give the fraction of edges in the graph that are connected to the VNs and the CNs, respectively. We call them degree distribution polynomials, denoted by $\lambda(x)$ and $\rho(x)$, respectively. We assume that d_{vmax} and d_{cmax} denote the maximum VN degree and CN degree, respectively. The degree distribution polynomials can be expressed as

$$\lambda(x) = \sum_{i=2}^{d_{vmax}} \lambda_i x^{i-1} \tag{2.1}$$

$$\rho(x) = \sum_{j=2}^{d_{cmax}} \rho_j x^{j-1} \tag{2.2}$$

where λ_i corresponds to the fraction of edges connected to VNs, and ρ_j corresponds to the fraction of edges connected to CNs. The coefficients λ_i and ρ_j must satisfy the following constraints

$$0 \leqslant \lambda_i \leqslant 1, i \geqslant 2; 0 \leqslant \rho_j \leqslant 1, j \geqslant 2; \tag{2.3}$$

$$\sum_{i=2}^{d_{vmax}} \lambda_i = 1; \sum_{j=2}^{d_{cmax}} \rho_j = 1. \tag{2.4}$$

Furthermore, in order to be compatible with a given code rate R, the following linear constraint must be satisfied for a degree distribution [5]:

$$\sum_{j=2}^{d_{cmax}} \frac{\rho_j}{j} = (1-R) \sum_{i=2}^{d_{vmax}} \frac{\lambda_i}{i}. \tag{2.5}$$

2.3 Encoding of LDPC Codes

The process of the encoding of LDPC codes basically includes two steps. The first step is to construct a sparse parity-check matrix $\mathbf{H}$. The second step is to generate LDPC codewords $\mathbf{c}$ using this matrix, and $\mathbf{c}$ must satisfy $\mathbf{cH}^T = 0$, where the superscript 'T' denotes the transpose operation.

2.3.1 Construction of Sparse Parity-Check Matrix

LDPC code is constructed based on the definition of the parity check matrix. Therefore, how to obtain the high sparse parity-check matrix with excellent performance is a key technology in the encoding of LDPC codes.

In general, constructing an LDPC code parity check matrix can be divided into two steps as follows:

1. Select a degree distribution expression;

2. Construct the structure of the parity check matrix; that is, arrange the specific placement of the edge between the variable node and the check node.

We can design the optimal degree distribution with the aid of various types of performance analysis tools, such as differential evolution, density evolution and EXIT chart [19]. For the construction of the parity check matrix, there are many ways which are summarized into two main classes: random constructions [22] and structural constructions [23]. For the sake of conciseness, we do not present these approaches in detail at here.

2.3.2 Encoding Based on Gauss-Jordan Elimination

The conventional encoding algorithm is based on the approach of Gauss-Jordan elimination. Let the $M \times N$ matrix $\mathbf{H}$ is a randomly generated party-check matrix. According to the definition of linear block codes, for the input binary source bits u, the encoded codeword $\mathbf{c}$ satisfies

$$\mathbf{c}\mathbf{H}^T = 0. \tag{2.6}$$

Since $\mathbf{H}$ is randomly generated, it is a non-systematic form in general. $\mathbf{H}$ is converted to systematic form and then divided into an $M \times (N - M)$ matrix $\mathbf{A}$ and an $M \times M$ matrix $\mathbf{B}$ as $\mathbf{H} = [\mathbf{A}|\mathbf{B}]$. We can also partition the codeword $\mathbf{c}$ into message bits u, and check bits $\mathbf{p}$ using the similar form. Based on (2.6), we have

$$[\mathbf{A}\ \mathbf{B}] \begin{bmatrix} u \\ p \end{bmatrix} = 0. \tag{2.7}$$

From this, we have

$$\mathbf{A}u + \mathbf{B}p = 0 \tag{2.8}$$

Hence, we can obtain the check bits $\mathbf{p}$ by

$$p = -\mathbf{B}^{-1}\mathbf{A}u \tag{2.9}$$

This approach is performed with complexity $O(M^3)$, which is mainly decided by the computation of $\mathbf{B}^{-1}\mathbf{A}$. Equation (2.9) can be used to compute the check bits as long as $\mathbf{B}$ is non-singular and not just when $\mathbf{A}$ is an identity matrix. In general, the parity-check matrix $\mathbf{H}$ will not be sparse after the pre-processing. Thus the complexity of conventional methods for the encoding of LDPC codes is high when the length of LDPC codes is large.

2.3.3 Encoding Based on Approximate Lower Triangular Matrix

To solve the problem of the high complexity of encoding based on the Gauss-Jordan elimination, an efficient LDPC encoding algorithm, which encodes based on approximate lower triangular matrix, was proposed by Richardson [23]. The detailed description of this encoding approach is introduced in the following.

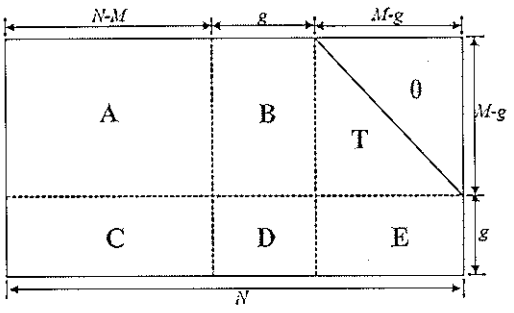

Figure 2.3: Approximate lower triangular matrix.

The principle of this approach is that in order to keep the sparsity of the parity-check matrix H, the transformation of the parity-check matrix is performed by using only row and column permutations. Any arbitrary sparse matrix can be converted into the parity check matrix H with an approximate lower triangular form as shown in Fig. 2.3. The flow of the encoding algorithm based on approximate lower triangular matrix is illustrated as follows:

Step 1: Covert H into an approximate lower triangular form by performing row and column permutation. H can be represented as

$$\mathrm{H} = \begin{bmatrix} \mathrm{A} & \mathrm{B} & \mathrm{T} \\ \mathrm{C} & \mathrm{D} & \mathrm{E} \end{bmatrix} \tag{2.10}$$

where A is $(M-g)\times(N-M)$, B is $(M-g)\times g$, T is an $(M-g)\times(M-g)$ lower triangular matrix, C is $g\times(N-M)$, D is $g\times g$ and finally E is $g\times(M-g)$. The g rows of H are called the gap of the approximate representation, and the smaller g is, the lower is the encoding complexity for LDPC codes.

Step 2: Use Gauss elimination to clear E, which is performed as follows:

$$\begin{bmatrix} \mathrm{I} & 0 \\ -\mathrm{ET}^{-1} & \mathrm{I} \end{bmatrix} \begin{bmatrix} \mathrm{A} & \mathrm{B} & \mathrm{T} \\ \mathrm{C} & \mathrm{D} & \mathrm{E} \end{bmatrix} = \begin{bmatrix} \mathrm{A} & \mathrm{B} & \mathrm{T} \\ -\mathrm{ET}^{-1}\mathrm{A}+\mathrm{C} & -\mathrm{ET}^{-1}\mathrm{A}+\mathrm{D} & 0 \end{bmatrix} \tag{2.11}$$

where I is an identity matrix.

Step 3: Consider the codeword c consisting of a systematic part s and two parity parts p_1 and p_2, with lengths g and $(M-g)$, respectively. That is, the codeword $\mathrm{c} = [s\ p_1\ p_2]$ must satisfy the parity-check equation (2.6), we have

$$\begin{cases} \mathrm{A}s^T + \mathrm{B}p_1^T + \mathrm{T}p_2^T = 0 \\ (-\mathrm{ET}^{-1}\mathrm{A}+\mathrm{C})s^T + (-\mathrm{ET}^{-1}\mathrm{B}+\mathrm{D})p_1^T = 0 \end{cases} \tag{2.12}$$

From (2.12) we can obtain the two parity check bits p_1 and p_2. Then combining them with the systematic part s, we can obtain the LDPC codewords.

The computation of p_1 and p_2 can be accomplished with complexity $O(N+g^2)$ and $O(N)$, respectively, which are much lower than that of the method of encoding based on Gauss-Jordan elimination. Therefore, this method is the most popular one for encoding LDPC codes.

2.4 Decoding of LDPC Codes

Decoding is a crucial factor for the performance of channel coding techniques. LDPC codes have a variety of decoding methods, which are essentially based on the iterative message-passing algorithm since their operation can be explained by the passing of messages iteratively along the edges of a Tanner graph. LDPC decoding algorithm can be decomposed into bit-flipping (BF) algorithm [1] and belief-propagation (BP) algorithm [3] based on the different forms of messages sent in the iterative process. The messages are binary bits in the BF algorithm; while the messages are probabilities which represent the belief about each bit in the BP algorithm. The BF algorithm has a lower complexity, but with worse decoding performance. In contrast, the BP algorithm can achieve near-capacity performance with a higher implementation complexity. Since the BP algorithm has been used in practical systems with the development of hardware technology, we only present the BP algorithm in detail at here.

For the illustration of the following description, we will use the following assumption conditions. We assume that an LDPC codeword $\mathbf{c} = \{c_1, c_2, ..., c_N\}, c_i \in \{0, 1\}$ is first mapped to binary phase shift keying (BPSK) symbol sequence $\mathbf{x} = \{x_1, x_2, ..., x_N\}, x_i \in \{-1, +1\}$, and then $\mathbf{x}$ is transmitted over an additive white Gaussian noise (AWGN) channel. At the receiver, the received sequence is $\mathbf{y} = \{y_1, y_2, ..., y_N\}$. According $\mathbf{y}$, the decoded sequence is $\hat{\mathbf{c}}$.

The following notations will be used to describe the decoding algorithm of LDPC codes. R_j is the set of checks in which bit j participates. $R_{j\backslash i}$ is the set R_j with check node i excluded. C_i is the set of bits that participate in check i. $C_{i\backslash j}$ is the set C_i with bit j excluded. $r_{ji}(b)$ is the probability of check i when bit i of $\mathbf{c}$ ($c_i = b$) and other bits have a separable distribution given by the probabilities $\{q_{ij'}\}_{j' \neq j}$. $q_{ij}(b)$ is the probability that bit i of $\mathbf{c}$ $c_i = b$, given the information obtained via checks other than check j. P_i is the *a posterior* probability of the transmitted codeword bit $c_i = 1$, which is obtained by the receive signals and the channel feature.

The flow of the BP decoding algorithm is as follows:

Step 1: Initialization

In the case of an AWGN channel with BPSK, the variables $q_{ij}(0)$ and $q_{ij}(1)$ are initialized using (2.13) and (2.14). Set the maximum number of iterations as IN_{max}.

$$q_{ij}(0) = 1 - P_i = P(x_i = +1|y_i) = \frac{1}{1 + e^{-2y_i/\sigma^2}} \tag{2.13}$$

$$q_{ij}(1) = P_i = P(x_i = -1|y_i) = \frac{1}{1 + e^{2y_i/\sigma^2}} \tag{2.14}$$

Step 2: Row operation

$$r_{ji}(0) = \frac{1}{2} + \frac{1}{2} \prod_{i' \in R_{j\backslash i}} (1 - 2q_{i'j}(1)) \tag{2.15}$$

$$r_{ji}(1) = 1 - r_{ji}(0) \tag{2.16}$$

Step 3: Column operation

$$q_{ij}(0) = K_{ij}(1 - P_i) \prod_{j' \in C_{i \setminus j}} r_{j'i}(0) \tag{2.17}$$

$$q_{ij}(1) = K_{ij} P_i \prod_{j' \in C_{i \setminus j}} r_{j'i}(1) \tag{2.18}$$

where K_{ij} is chosen such that $q_{ij}(0) + q_{ij}(1) = 1$.

Step 4: Decision

$$Q_i(0) = K_i(1 - P_i) \prod_{j \in C_i} r_{ji}(0) \tag{2.19}$$

$$Q_i(1) = K_i P_i \prod_{j \in C_i} r_{ji}(1) \tag{2.20}$$

where K_i is chosen such that $q_i(0) + q_i(1) = 1$.

$$\hat{c}_i = \begin{cases} 0, & Q(0) > Q(1) \\ 1, & elsewhere \end{cases} \tag{2.21}$$

Step 5:Iteration stopping judgement

If $\hat{\mathrm{c}}\mathrm{H}^T = 0$, output $\hat{\mathrm{c}}$ and stop the decoding. Otherwise, go to step 2 until the maximum number of iterations IN_{max} is reached.

If we use the log-likelihood ratio (LLR) to represent the probabilistic message of the BP algorithm, we can obtain an enhanced version of the probabilistic BP algorithm, which is called Log BP algorithm. Using this approach can reduce most multiplications to additions. Therefore, the decoding complexity is greatly reduced. For more details, we refer to [1] and [3].

CHAPTER 3

Multiple-Symbol Differential Detection

In the communication systems with differential detection, the receiver can demodulate the transmission of information bits without requiring carrier phase recovery and CSI estimation. Therefore, differential detection can greatly reduce the implementation complexity of the system, and can also be implemented in the severe channel, such as fast fading channel, where the coherent detection systems are difficult to implement since the CSI cannot be accurately estimated or carrier recovery is difficult. However, differential detection also brings a performance loss of the system, thereby reducing its practical value.

Conventional differential detection makes a decision using only the two adjacent reception signal, which results in a considerable performance loss compared with coherent detection. In order to compensate this performance loss, MSDD [10] has been proposed. By extending the length of the observation window size to more than two symbols, MSDD makes a joint decision on those several symbols simultaneously rather than symbol-by-symbol detection as in conventional differential detection. The performance of MSDD for uncoded multiple-phase shift keying (MPSK) and amplitude/phase-shift keying (APSK) systems has been analyzed and evaluated over AWGN, slow flat Rayleigh fading and fast Rayleigh fading channels, respectively in [11] and [12]. It was shown that MSDD is an effective approach to compensate for the performance loss between the differential detection and the coherent detection.

In this chapter, we provide the fundamentals of MSDD as the basis for the following chapters. In Section 3.1, we first describe the uncoded communication system model with differential coding and MSDD. Then, the decision metric of MSDD is presented in Section 3.2. Finally, the complexity of MSDD is analyzed in detail in Section 3.3.

3.1 System Model

A differential encoded transmission system with MSDD is shown in Fig. 3.1. At the transmitter, the transmitted message bits b_k are first mapped to MPSK symbols d_k. Then, the MPSK symbols are differentially encoded to symbols s_k. The process of the differential coding is

$$s_k = s_{k-1} d_k. \tag{3.1}$$

s_0 is a reference symbol and is known by the receiver. In this thesis, s_0 is set to 1.

We consider the transmission of MPSK signal over a channel with unknown random phase and multiplicative fading. Differentially encoded MPSK symbols transmitted in

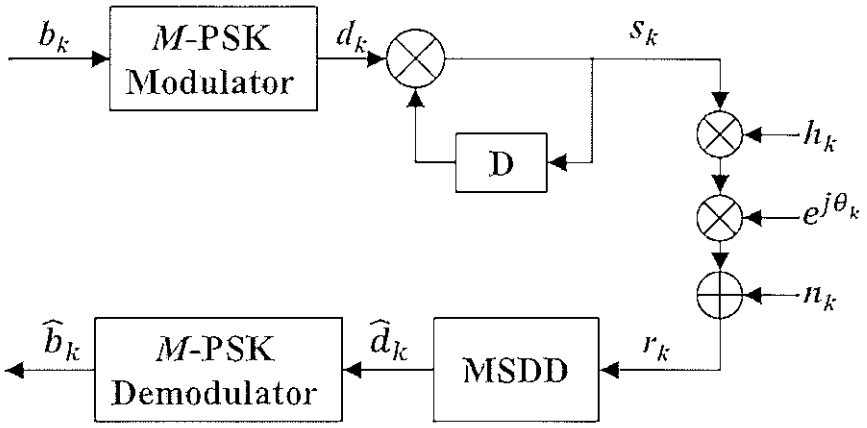

Figure 3.1: System model of MPSK systems with MSDD.

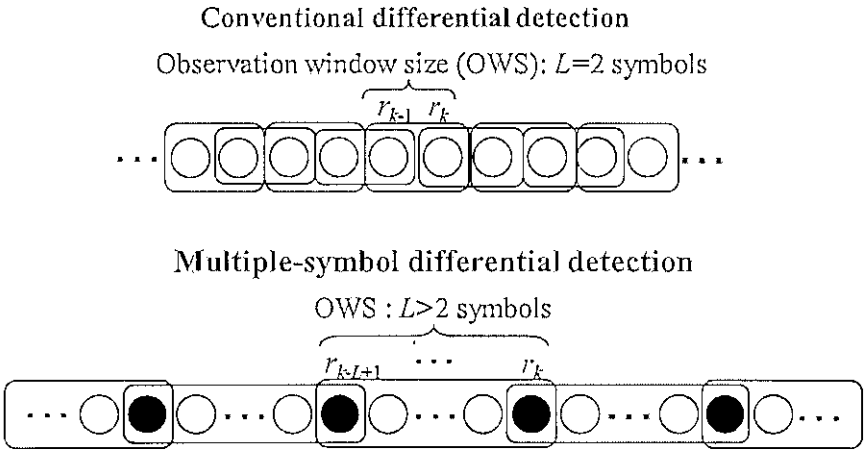

Figure 3.2: Partitioning of the received signals into groups (observation windows) for MSDD.

the interval k can be expressed as

$$s_k = a_k e^{j\phi_k} \tag{3.2}$$

where a_k is the transmitted signal amplitude, and ϕ_k is the transmitted signal phase. Based on the above channel, the corresponding received signal in the same interval is

$$r_k = s_k h_k e^{j\theta_k} + n_k \tag{3.3}$$

where h_k is a sample of a normalized complex Gaussian process with mean $\overline{h}_k$ and variance σ_h^2, n_k is a sample of a zero mean complex Gaussian noise with variance σ_n^2, and θ_k, which is assumed to be uniformly distributed in the interval $(0, 2\pi)$, is an arbitrary phase introduced by the channel.

At the receiver, MSDD is employed before performing the MPSK demodulator. The received sequence of samples r_k is partitioned into groups (observation windows) of L samples each in such a way that the groups overlap in one sample as shown in Fig. 3.2. Based on the principle of the differential coding, the overlap is necessary since the last signal in the group r_{k-1} need to be used as the reference symbol for the next group.

In the following, we use L to represent the observation window size of MSDD. The L sample received sequence from interval $k-L+1$ to interval k can be rewritten into the following vector form:

$$\mathrm{r} = \mathrm{H}^T\mathrm{S} + \mathrm{n} \tag{3.4}$$

where $\mathrm{r}_k = [r_k, r_{k-1}, ..., r_{k-L+1}]^T$, $\mathrm{H} = diag\{h_k,\ h_{k-1}, ...,\ h_{k-L+1}\}$,
$\mathrm{S}_k = [s_k e^{j\theta_k}, s_{k-1}e^{j\theta_{k-1}}, ..., s_{k-L+1}e^{j\theta_{k-L+1}}]^T$, and $\mathrm{n}_k = [n_k, n_{k-1}, ..., n_{k-L+1}]^T$, and the superscript '$T$' denotes the transpose operation.

3.2 Decision Metric of Multiple-Symbol Differential Detection

In many practical channels, it is reasonable to assume the channel phase θ_k is independent with k within an observation window. Therefore, (3.4) can be rewritten into

$$\mathrm{r} = \mathrm{H}^T\mathrm{s}e^{j\theta} + \mathrm{n}. \tag{3.5}$$

For MSDD over the assumed channel model, the CSI is unknown at the receiver. From [11], the conditional probability density function (PDF) of r given s and θ is given as:

$$p(\mathrm{r}|\mathrm{s},\theta) = \frac{1}{(2\pi)^L \det \mathrm{R}} \exp\left\{-\frac{1}{2}(\mathrm{r}-\bar{\mathrm{r}})^T\mathrm{R}^{-1}(\mathrm{r}-\bar{\mathrm{r}})^*\right\}, \tag{3.6}$$

where the superscript '$*$' denotes complex conjugation, and

$$\bar{r} = \overline{\mathrm{H}}^T\mathrm{s}e^{j\theta} = \mathrm{X}e^{j\theta}, \tag{3.7}$$

and R is the covariance matrix of r, which is expressed as

$$\mathrm{R} = \frac{1}{2}E\{(\mathrm{r}-\bar{\mathrm{r}})^*(\mathrm{r}-\bar{\mathrm{r}})^T\}. \tag{3.8}$$

R has $\mathrm{R}^T = \mathrm{R}^*$ and $\det \mathrm{R} > 0$, because it is a Hermitian positive definite. The i,jth element of R can be calculated as

$$\begin{aligned} R_{i,j} &= \frac{1}{2}E\left\{(r_i-\bar{r}_i)^*(r_j-\bar{r}_j)^T\right\} \\ &= s_i^* s_j \sigma_h^2 \rho_{i,j} + \sigma_n^2 \delta_{i,j}, \end{aligned} \tag{3.9}$$

where $\rho_{i,j}$ denotes the normalized covariance function of the fading process as

$$\rho_{i,j} = \frac{\frac{1}{2}E\{(h_i-\bar{h}_i)^*(h_j-\bar{h}_j)^T\}}{\sigma_h^2}, \tag{3.10}$$

and

$$\delta_{i,j} = \begin{cases} 1; & i = j \\ 0; & i \neq j. \end{cases} \tag{3.11}$$

Based on the Hermitian of R^{-1}, submitting (3.7) into $(\mathrm{r} - \bar{\mathrm{r}})^T R^{-1} (\mathrm{r} - \bar{\mathrm{r}})^*$, we can get

$$(\mathrm{r} - \bar{r})^T \mathrm{R}^{-1} (\mathrm{r} - \bar{r})^* = \mathrm{r}^T \mathrm{R}^{-1} \mathrm{r}^* + \mathrm{X}^T \mathrm{R}^{-1} \mathrm{X}^* - 2 \left| \mathrm{r}^T \mathrm{R}^{-1} \mathrm{X}^* \right| \cos(\theta - \alpha), \tag{3.12}$$

where

$$\alpha = \tan^{-1} \frac{\mathrm{Im}\left\{ r^T R^{-1} X^* \right\}}{\mathrm{Re}\left\{ r^T R^{-1} X^* \right\}}. \tag{3.13}$$

Averaging over the uniform distribution of θ, and substituting (3.12) in (3.6), the PDF of r given s can be expressed as

$$\begin{aligned} p(\mathrm{r}|\mathrm{s}) &= \int_{-\pi}^{\pi} p(\mathrm{r}|\mathrm{s}, \theta) p(\theta) d\theta \\ &= \frac{1}{(2\pi)^L \det \mathrm{R}} \exp\left\{ -\frac{1}{2} (\mathrm{r}^T \mathrm{R}^{-1} \mathrm{r}^* + \mathrm{X}^T \mathrm{R}^{-1} \mathrm{X}^*) \right\} I_0(|\mathrm{r}^T \mathrm{R}^{-1} \mathrm{X}^*|), \end{aligned} \tag{3.14}$$

where $I_0(\cdot)$ is the zero-order modified Bessel function of the first kind.

Using (3.14), MSDD makes a joint decision on the symbols in the observation window based on the maximum likelihood (ML) principle. Therefore, the transmitted MPSK symbols vector d is decided by

$$\hat{\mathrm{d}} = \underset{d_{k-i}(i=0,\ldots,L-1)}{\mathrm{argmax}} \; p(\mathrm{r}|\mathrm{s}) = \underset{d_{k-i}(i=0,\ldots,L-1)}{\mathrm{argmax}} \; \ln p(\mathrm{r}|\mathrm{s}). \tag{3.15}$$

Since the natural logarithm is a monotonically increasing function, maximizing $p(\mathrm{r}|\mathrm{s})$ over s is equivalent to maximizing $\ln p(\mathrm{r}|\mathrm{s})$ over s. Therefore, equation (3.14) is equivalent to choose sequence s that maximizes the decision metric:

$$\eta = -\ln(\det \mathrm{R}) - \frac{1}{2} (\mathrm{r}^T \mathrm{R}^{-1} \mathrm{r}^* + \mathrm{s}^T \ddot{\mathrm{H}} \mathrm{R}^{-1} \ddot{\mathrm{H}}^{*T} \mathrm{s}^*) + \ln I_0(|\mathrm{r}^T \mathrm{R}^{-1} \ddot{\mathrm{H}}^{*T} \mathrm{s}^*|). \tag{3.16}$$

Equations (3.14) and (3.16) are general decision metrics for several kind of channels. In the following, we will discuss the decision metric of MSDD for different channels which will be considered in the following chapters.

3.2.1 MSDD of MPSK over AWGN Channels

In the case of AWGN channels, $h_k = 1$ for all k and $\sigma_h^2 = 0$. Submitting them into (3.9) and (3.14), the decision metric (3.14) is simplified to

$$p(\mathrm{r}|\mathrm{s}) = \frac{1}{(2\pi\sigma_n^2)^L} \exp[-\frac{1}{2\sigma_n^2} \sum_{i=0}^{L-1} (|r_i|^2 + |s_i|^2)] \times I_0(\frac{1}{\sigma_n^2} \left| \sum_{i=0}^{L-1} r_i s_i^* \right|), \tag{3.17}$$

and the decision metric (3.16) is simplified to

$$\eta = |\sigma_n^{-2} \mathrm{r}^T \mathrm{s}^*|^2. \tag{3.18}$$

3.2.2 MSDD of MPSK over Rayleigh Fading Channels

In the case of Rayleigh Fading channels, $E(h_k) = 0$ for any number of k. In this condition, the decision metric (3.14) can be simplified to

$$p(\mathrm{r}|\mathrm{s}) = \frac{1}{(2\pi)^L \det \mathrm{R}} \exp[-\frac{1}{2}\{\mathrm{r}^T \mathrm{R}^{-1} \mathrm{r}^*\}], \tag{3.19}$$

and the decision metric (3.16) can be simplified to

$$\eta = -\ln(\det \mathrm{R}) - \frac{1}{2}(\mathrm{r}^T \mathrm{R}^{-1} \mathrm{r}^*), \tag{3.20}$$

and the element of R_{ij} is

$$R_{i,j} = s_i^* s_j \sigma_h^2 \rho_{i,j} + \sigma_n^2 \delta_{i,j}, \tag{3.21}$$

where $\rho_{i,j}$ is the correlation coefficient of the fading process, which is given by [20] as

$$\rho_{i,j} \approx J_0(2\pi f_D T_s |i-j|), \tag{3.22}$$

where $f_D T_s$ is the normalized Doppler frequency, and $J_0(\cdot)$ is the zeroth order Bessel function of the first kind.

3.3 Complexity of Multiple-Symbol Differential Detection

From the decision metric of MSDD, we can find that MSDD needs to consider all candidate bit sequences. From this point, MSDD can be viewed as a tree decoding problem. The hard output of MSDD is actually the path with the ML value of the metric. Assuming the order of MPSK is M and the observation widow size is L, the tree diagram of MSDD searching the maximum likelihood sequence is illustrated in Fig. 3.3, where the first symbol of the received sequence is the reference symbol, which is the last symbol of the previous observation window in the received sequence.

From this figure, we can observe that the number of search paths equals to M^{L-1}. Thus, the number of search paths increases exponentially with the observation window size and the order of the modulation. MSDD calculates the metrics of all paths and compares them at the receiver, and then the path that has the maximum metric is decided as the L transmitted symbols. In calculating the metric of (3.18), it is necessary to compute the inverse and the determinant of the matrix R. In case of a fairly large L and M, the computation of the metric will consume much time. Generally, the computational complexity of the inverse of the matrix is $O(N^3)$ [21]. Therefore, this high complexity will result in an unacceptable decoding time delay, which limits the practical application of MSDD.

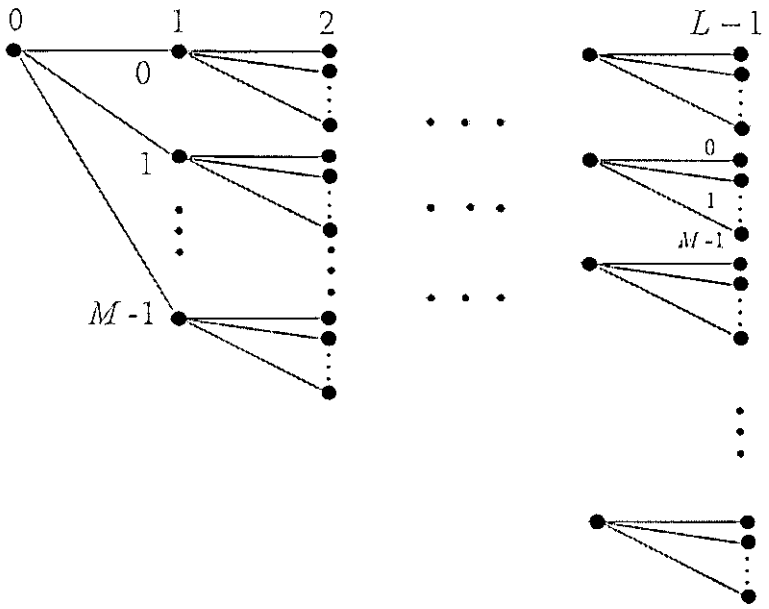

Figure 3.3: Tree diagram for MSDD searching the maximum likelihood sequence (MP-SK).

CHAPTER 4

Iterative Multiple-Symbol Differential Detection for Differentially Encoded LDPC Coded Systems

The performance of LDPC codes with differential detection over AWGN channels were studied in [5] and [24]. The two papers mainly focused on how to design LDPC codes for differential encoding with iterative differential detection. In [6]-[9], the performance of LDPC codes with differential detection over flat Rayleigh fading channels was studied. In [6] and [7], two kinds of metrics of a serial concatenation system of LPDC codes and differential binary phase shift keying (DBPSK) were proposed. In [8] and [9], a simple iterative differential detection and decoding receiver was proposed, and also mainly focused on how to optimize the LDPC codes to fit for differential detection.

It is well known that conventional differential detection has considerably worse performance than coherent detection. MSDD [10]-[12] has been proven to be an effective approach to compensate for this performance loss. For serially concatenated coded systems consisting of a channel encoder and a differential encoder, an iterative decoding scheme based on MSDD was proposed in [13]. In this scheme, the soft-input soft-output demodulator (SISOD) with MSDD (called MSDD SISOD in this thesis) is used as the inner decoder, and the iterative decoding is performed between the MSDD SISOD and the outer decoder. It was shown that this iterative MSDD scheme can achieve a large performance improvement compared with conventional differential detection. Up to now, the iterative MSDD scheme has been widely studied, such as iterative MSDD for turbo coded systems [14], for cooperative communication systems [15] and for spatial division multiple access systems [16].

In this chapter, an iterative MSDD scheme for differentially encoded LDPC coded system is studied. To make the MSDD suit for iterative decoding, a metric computation algorithm which output soft information for MSDD are derived, and the performance of the system is analyzed using the EXIT chart and computer simulations. Moreover, the importance of the optimization of LDPC codes for MSDD SISOD with different observation window size is studied.

4.1 System Model

A. Transmitter

The system model is shown in Fig. 4.1. A random message bit sequence $\mathrm{b} = \{b_1, b_2, ..., b_K\}$, $b_i \in \{0, 1\}$ is first encoded by a rate K/N LDPC encoder to a code sequence $\mathrm{c} = \{c_1, c_2, ..., c_N\}$, $c_i \in \{0, 1\}$. The code sequence is then mapped to an M-ary

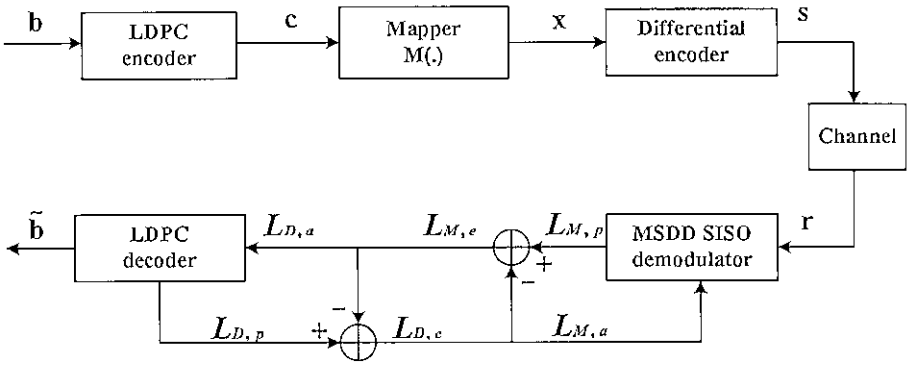

Figure 4.1: System model.

PSK symbol sequence $\mathbf{x} = \{x_1, x_2, ..., x_{N/m}\}, x_i \in \{e^{j2\pi i/M} \mid i = 0, 1, ..., M\}$, where m is the number of bits of each M-ary PSK symbol. The sequence $\mathbf{x}$ is then differentially encoded to a sequence $\mathbf{s} = \{s_0, s_1, ..., s_{N/m}\}$, where s_k is given by $s_k = x_k s_{k-1}$. s_0 is a reference symbol and is known by the demodulator. In this thesis, s_0 is set to 1. Different from general serially concatenated codes [25], the considered systems omit the interleaver between the encoder and the modulator to reduce the complexity of the system, because LDPC codes have an inherent interleaving nature since their parity check matrixes are randomly constructed with a high degree of sparsity [26].

B. Channel

The differential encoded sequence $\mathbf{s}$ is transmitted to the receiver through the Rayleigh fading channel with AWGN. The received discrete-time baseband signal, at time k, can be written as

$$r_k = h_k s_k e^{j\theta_k} + n_k \tag{4.1}$$

where fading coefficient h_k is a sample of a normalized complex Gaussian process with mean zero and variance σ_h^2, θ_k, which is uniformly distributed over $[0, 2\pi)$, is the unknown phase introduced by the channel, and n_k is a sample of a zero mean complex Gaussian noise with variance σ_n^2. It is assumed that h_k and n_k are mutually independent.

C. Receiver

At the receiver, the received signal is iteratively decoded by mutually exchanging soft information between the inner decoder and outer decoder (LDPC decoder). The inner decoder named MSDD SISOD is a soft-input soft-output demodulator based on MSDD. The process of this demodulator-decoder iteration is denoted by the outer iteration in this thesis. At each outer iteration, the MSDD SISOD produces the *a posteriori* information $L_{M,p}$ based on the received signals and the *a priori* information $L_{M,a}$ provided by the LDPC decoder, and produces the extrinsic information $L_{M,e}$ by subtracting $L_{M,a}$ from $L_{M,p}$. Then $L_{M,e}$ is passed to the LDPC decoder as the *a priori* information $L_{D,a}$. Based on $L_{D,a}$, the LDPC decoder performs a number of inner

iterations, and makes a tentative hard decision. If the hard decision is determined to be a legitimate codeword by checking the parity check constraints, the iterative decoding will be terminated. Otherwise, the extrinsic information $L_{D,e}$ of the LDPC decoder, which is obtained in a similar way to $L_{M,e}$, will be fed back to the MSDD SISOD as the *a priori* information $L_{M,a}$ for the next outer iteration. This process is repeated until the predefined maximum outer iteration number is reached or a legitimate codeword is found.

4.2 Metric Derivation for MSDD SISOD

MSDD SISO outputs the soft information as the input to the LDPC decoder. This is different from the conventional MSDD [11], which makes a hard decision through Maximum Likelihood Detection (MLD). So we should derive the metric computation algorithm to output the soft information for MSDD.

Assume that the OWS of MSDD SISOD is L, and θ_k remains constant over the entire received sequence. From Chapter 3, we know that the received sequence is divided into subblocks of L symbols each in such a way that the subblocks overlap in one symbol. That is, the number of subblocks is $(N/m)/(L-1)$. For the kth subblock, we can rewrite (4.1) in the following vector form

$$\mathrm{r}_k = \mathrm{H}^T \mathrm{s}_k + \mathrm{n}_k, \tag{4.2}$$

where $\mathrm{r}_k = [r_{k,0}, r_{k,1}, ..., r_{k,L-1}]^T$, $\mathrm{s}_k = [s_{k,0}e^{j\theta}, s_{k,1}e^{j\theta}, ..., s_{k,L-1}e^{j\theta}]^T$,
$\mathrm{n}_k = [n_{k,0}, n_{k,1}, ..., n_{k,L-1}]^T$.

Let $\mathrm{c}_k = [c_{k,1}, c_{k,2}, ..., c_{k,m(L-1)}]^T$ denote the code bits corresponding to the kth subblock of the received symbols. For the sake of clarity, we drop the index k in the following. At the MSDD SISOD, the *a posteriori* probability (APP) of each coded bits is computed. The APP of the code bit c_i is written in terms of LLR

$$L_{M,p}(c_i) = \log \frac{p(c_i = 0|\mathrm{r})}{p(c_i = 1|\mathrm{r})}. \tag{4.3}$$

Based on the Bayesian formula, and assume that the coded bits are independent with each other due to the inherent interleaving nature of LDPC codes, (4.3) is equivalent to

$$L_{M,p}(c_i) = \log \frac{\sum_{\mathrm{s}:c_i=0} p(\mathrm{r}|\mathrm{c})p(\mathrm{c})}{\sum_{\mathrm{s}:c_i=1} p(\mathrm{r}|\mathrm{c})p(\mathrm{c})} = \log \frac{\sum_{\mathrm{s}:c_i=0} p(\mathrm{r}|\mathrm{c}) \prod_{j=1}^{m(L-1)} p(c_j)}{\sum_{\mathrm{s}:c_i=1} p(\mathrm{r}|\mathrm{c}) \prod_{j=1}^{m(L-1)} p(c_j)}, \tag{4.4}$$

where the sums in the numerator and denominator are taken over all sequences c whose bit in position i is the value 0 or 1, respectively. $p(c_j)$ is the *a priori* probability provided by the LDPC decoder.

For the coded sequence c, M-ary PSK symbols sequence x and differential encoded

sequence s are one to one mapping, we can rewrite (4.4) as

$$L_{M,p}(c_i) = \underbrace{\log \frac{p(c_i=0)}{p(c_i=1)}}_{a\ priori\ probability} + \underbrace{\log \frac{\sum_{s:c_i=0} p(\mathrm{r}|\mathrm{s}) \prod_{j=1,j\neq i}^{m(L-1)} p(c_j)}{\sum_{s:c_i=1} p(\mathrm{r}|\mathrm{s}) \prod_{j=1,j\neq i}^{m(L-1)} p(c_j)}}_{extrinsic\ probability}. \tag{4.5}$$

From (4.5), we can find that the LLR is the summation of the *a priori* information and extrinsic information. The first part of (4.5) is related to the *a priori* probability of the coded bit c_i. The second part of (4.5) is related to the extrinsic information of the coded bit c_i, which is outputted into the LDPC decoder. In the first iteration, because no *a priori* probabilities of the coded bits are fed back, the transmitted bits are assumed to have equal *a priori* probabilities, i.e. $P(c_i = 0) = P(c_i = 1) = 1/2$. In the following iteration, the extrinsic information $L_{D,e}$ of LLR of the coded bits is fed back to MSDD SISO as *a priori* probability, so $P(c_i)$ is given as

$$p(c_i = 0) = \frac{e^{L_{D,e}(c_i)}}{1 + e^{L_{D,e}(c_i)}}, \quad p(c_i = 1) = \frac{1}{1 + e^{L_{D,e}(c_i)}}, \tag{4.6}$$

where $L_{D,e}(c_i)$ is the extrinsic information outputted from the LDPC decoder corresponding to the coded bit c_i.

From Chapter 3, the likelihood function $P(\mathrm{r}|\mathrm{s})$ is given as

$$p(\mathrm{r}|\mathrm{s}) = \frac{1}{(2\pi)^L \det \mathrm{R}} \exp[-\frac{1}{2}\{\mathrm{r}^T \mathrm{R}^{-1} \mathrm{r}^*\}], \tag{4.7}$$

where R is the covariance matrix of the received sequence r, and R^{-1} denotes its inverse. R_{ij} of R is evaluated by

$$R_{i,j} = s_i^* s_j \sigma_h^2 \rho_{ij} + \sigma_n^2 \delta_{ij}, \tag{4.8}$$

where $\delta_{ij} = 1$ when $i = j$; otherwise, $\delta_{ij} = 0$.

When we consider a Rayleigh fading channel, the correlation coefficient ρ_{ij} is given by [20]

$$\rho_{ij} \approx J_0(2\pi f_D T_s |i - j|), \tag{4.9}$$

where $f_D T_s$ is the normalized Doppler frequency, and $J_0(\cdot)$ is the zeroth order Bessel function of the first kind.

4.3 Extrinsic Information Transfer (EXIT) Chart Analysis of the System

To better understand the behavior of the iterative decoding process of the considered systems, we apply EXIT chart analysis [19] and [27] which can visualize the transfer characteristics of the inner decoder and the outer decoder, and also the convergence behavior of iterative decoding based on tracking the exchange of mutual information between the component decoders.

In EXIT chart, the component decoders are characterized by the EXIT functions, which describe the output mutual information as a function of the input mutual information. The mutual information between the transmitted coded bits c and the respective LLR values $L(C)$ is defined as [19]

$$I_L = I(L(C); C) = \frac{1}{2} \sum_{c=0,1} \int_{-\infty}^{\infty} p_{L(c)}(l|C = c) \cdot \log_2 \frac{2p_{L(c)}(l|C = c)}{p_{L(c)}(l|C = 0) + p_{L(c)}(l|C = 1)} dl, \tag{4.10}$$

where $p_{L(c)}(l|C = c)$ is the conditional PDF of the LLR values $L(c)$ given $c \in \{0, 1\}$, and $0 \leqslant I_L \leqslant 1$.

Using (4.10), the input mutual information $I_{L_{M,a}}$ and the output mutual information $I_{L_{M,e}}$ of the MSDD SISO demodulator can be obtained. Viewing $I_{L_{M,e}}$ as a function of $I_{L_{M,a}}$ and parameters of channel, the transfer characteristic of the MSDD SISOD is defined as

$$I_{L_{M,e}} = T_1(I_{L_{M,a}}, \frac{E_b}{N_0}, f_D T_s), \tag{4.11}$$

where $\frac{E_b}{N_0}$ is the signal-to-noise ratio (SNR) of the channel.

Similarly, $I_{L_{D,a}}$ and $I_{L_{D,e}}$ of the LDPC decoder can be computed, and the transfer characteristic of the LDPC decoder is defined as

$$I_{L_{D,e}} = T_2(I_{L_{D,a}}). \tag{4.12}$$

To generate the EXIT chart, the PDFs of the LLRs corresponding to $I_{L_{M,a}}$ and $I_{L_{D,a}}$ are assumed to be Gaussian distributed. When $I_{L_{M,e}}$ and $I_{L_{D,e}}$ are calculated, the PDFs of $L_{M,e}$ and $L_{D,e}$ are obtained by the histogram method [27]. As shown in Fig. 4.1, the extrinsic information of the MSDD SISO demodulator is the *a priori* information of the LDPC decoder and vice versa, which implies that $I_{L_{D,a}} = I_{L_{M,e}}$ and $I_{L_{M,a}} = I_{L_{D,e}}$. Therefore, the transfer characteristics of the MSDD SISO demodulator and the LDPC decoder can be plotted into a signal diagram by switching the x-axis and the y-axis. This diagrame is referred to as EXIT chart.

We plot the EXIT chart of MSDD SISOD of the systems under consideration in Fig. 4.2-4.5. The system considers a regular rate-1/2 (3, 6) LDPC code with code length 100800 with BPSK modulation over an AWGN channel and a Rayleigh Fading channel, respectively. From [19], we know that if the EXIT curve of inner decoder has a steep slope, the strong potential performance improvement can be got by iterative decoding. And the bit error rate (BER) performance is determined by the location of the intersection of the EXIT curves of the MSDD SISOD and the LDPC decoder. If the location is at the left side of the EXIT chart, this means that the iterative decoding stops quickly and high BER will be got. On the contrary, if the location is at the very right side, it means that iterative decoding can converge at low BER.

Fig. 4.2 shows the transfer characteristics of MSDD SISOD with different OWS over AWGN channels at $E_b/N_0 = 3.5$ dB. In the figure, L denotes the value of the OWS of the MSDD SISOD. From Fig. 4.2, we can observe that the slopes of the MSDD SISOD curves increase with the increase of the OWS, which implies that the performance of

MSDD SISOD can been improved by increasing the OWS and the number of iterations. Whereas, because the MSDD SISOD curves are horizontal lines, the iterative decoding is not valid with $L = 2$ (conventional differential detection). In addition, we can also observe that the gap of the slopes of these curves becomes smaller and smaller with an increase of OWS. This means that the performance improvement is not significant when L increases to a relatively large value. Fig. 4.3 shows the transfer characteristics of MSDD SISOD with different OWS over Rayleigh fading channels with $f_D T_s = 0.01$ at $E_b/N_0 = 6$ dB. It is shown that the similar analysis results can be obtained for Rayleigh fading channels. The difference is that the slops of the curves of Fig. 4.3 are smaller than than that of Fig. 4.2 at the same value of L, which implies that for Rayleigh fading channels, the performance improvement by increasing the OWS of MSDD SISOD is not as significant as that for AWGN channels.

Fig. 4.4 shows the transfer characteristic curve of the LDPC decoder with different iteration. It is shown that the steepness of the curves converges after the $10th$ iteration number. From the 10th iteration to the $100th$ iteration, the curves of them are almost overlapping with each other. That means the performance LDPC decoder can hardly be improved after the $10th$ iteration, which will be supported by the BER performance as shown in Fig. 4.7.

Plotting the transfer characteristics of MSDD SISOD and the LDPC decoder into a signal figure by switching the x-axis and the y-axis of the EXIT chart of the LDPC decoder, as shown in Fig. 4.5, we can obtain a EXIT chart which can help us to analyze the convergence characteristics of the iterative decoding and predict the decoding threshold of the systems under consideration. The decoding threshold is the SNR where a tunnel opens between the curves of the MSDD SISOD and the LDPC decoder. It can be observed that a tunnel opens between the curves of MSDD SISOD and the LDPC decoder with an increase of OWS. For example, the tunnel opens at SNR = 6 dB for $L = 6$ in Fig. 4.5, which means that the iterative decoding can improve the performance effectively and achieve low BER performance. So we can predict that the decoding threshold appears near SNR = 6 dB for $L = 6$. All the above analysis results by the EXIT chart will be further supported by the simulation results in the next section.

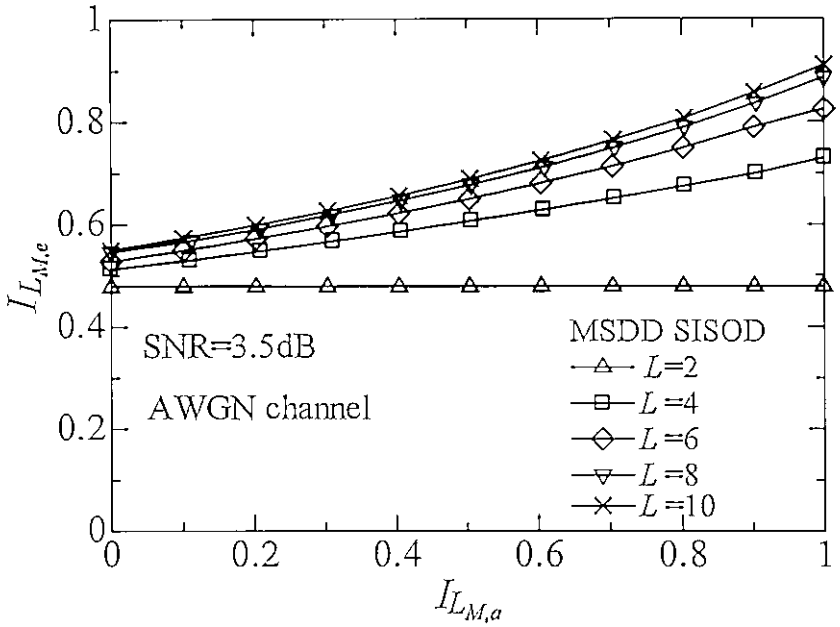

Figure 4.2: EXIT chart of MSDD SISOD with different OWS over AWGN channels.

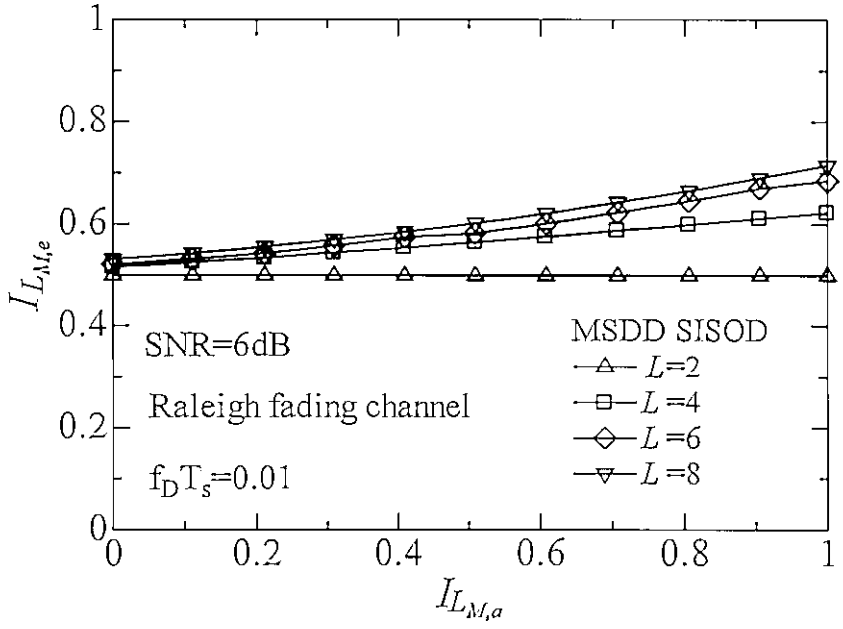

Figure 4.3: EXIT chart of the MSDD SISOD with different OWS over Rayleigh fading channels.

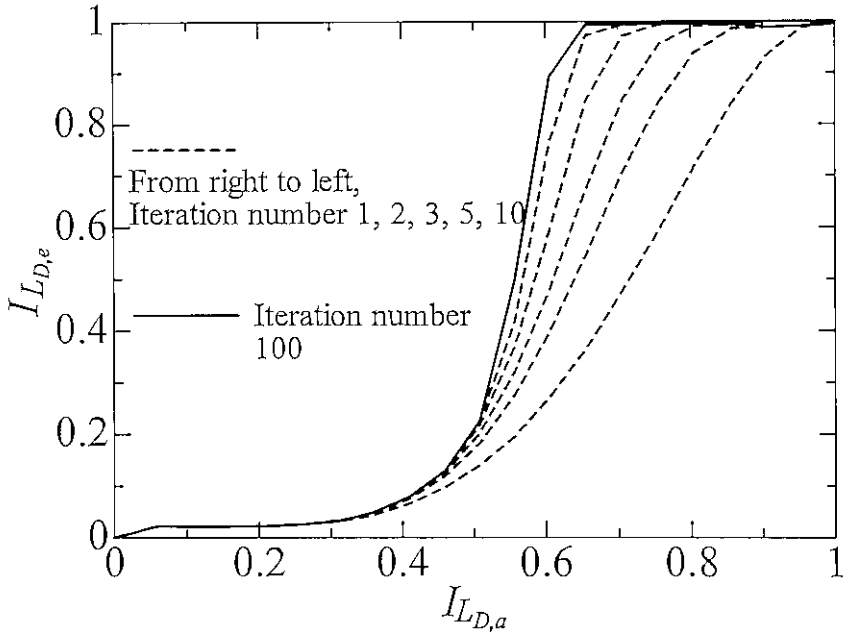

Figure 4.4: EXIT chart of the LDPC decoder with different number of iterations.

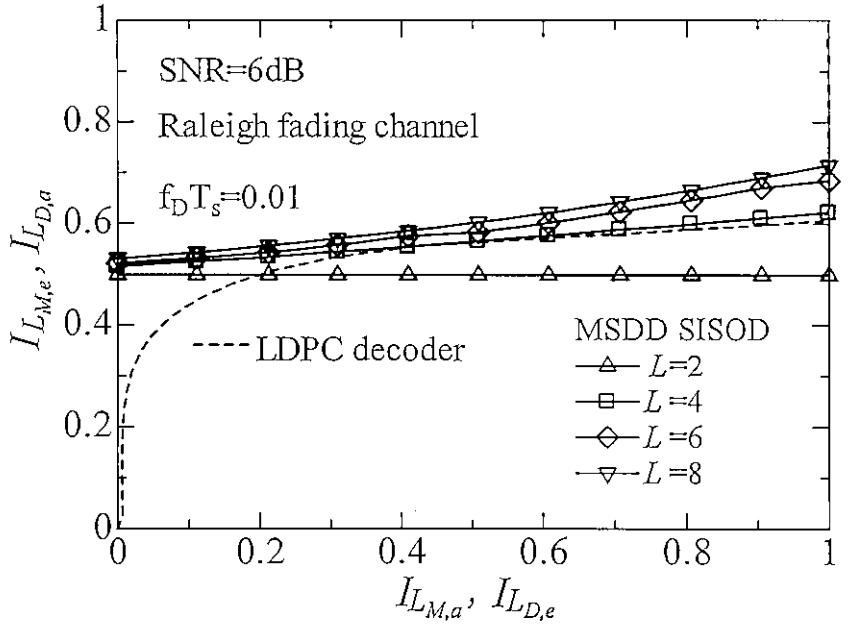

Figure 4.5: EXIT chart of the systems under consideration with different OWS; rate-1/2 (3, 6) LDPC code with length 100800 over Rayleigh fading channels with BPSK; inner iteration number of LDPC decoder is 10.

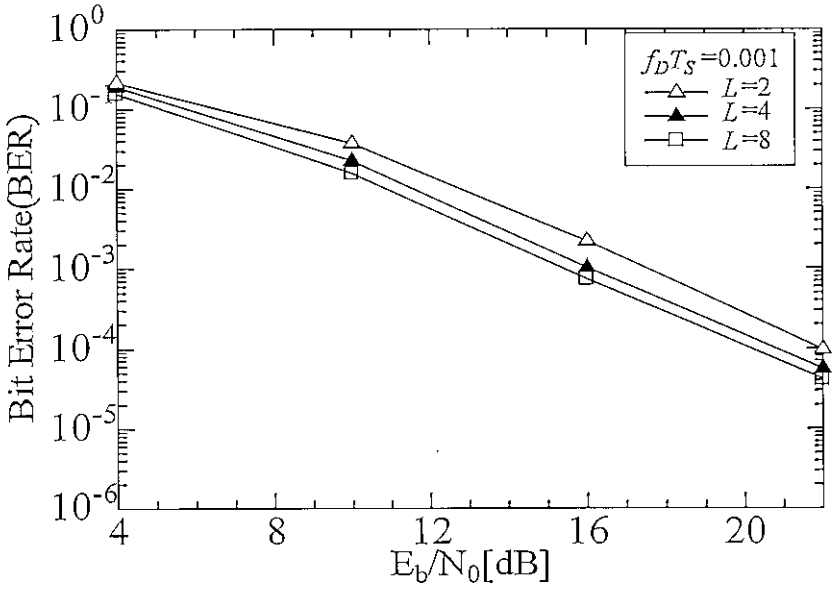

Figure 4.6: Effect of observation interval size on BER performance.

4.4 Simulation Results and Analysis

In our simulations, unless otherwise indicated, regular rate-1/2 (3, 6) LDPC code with length $N = 1008$ is used. The coded bits are modulated using BPSK for simplicity. The Rayleigh fading channel considered at here is Jake's model, and the normalized maximum Doppler frequency f_DT_S is set to 0.001 for slow fading channels and 0.01 for fast fading channels.

Fig. 4.6 shows the BER performance of MSDD SISOD with different OWS. In this simulation, we focus on the effectiveness of the proposed metric computation algorithm based on MSDD, the iteration number of sum-product algorithm is set to a small number of 3. The iteration number between the MSDD SISOD and LDPC decoder is set to 6. From Fig. 4.6 we can observe that the BER performance is improved with the increase of L. And we note that the improvement of BER performance is not significant when L is bigger than 4. That means a significant gain can be got without a large observation window size.

Fig. 4.7 shows the BER performance of the LDPC decoder with different iteration numbers. L is set to 4, other simulation parameters is same as the above simulation. It can be seen that the performance is improved with the increase of iteration number of the LDPC decoder but not significantly when the iteration number is bigger than 10. Considering from the balance between the performance and algorithm complexity, the iteration number of sum-product algorithm is selected as 10 in the following simulations. From the above analysis results, we can also find that the above simulation results support the EXIT chart analysis results of Section 4.3.

After testing the impact of the MSDD SISOD and the LDPC decoder on BER performance, the impact of the iterative decoding between MSDD SISOD and LDPC decoder on BER performance is shown in Fig. 4.8. It can be observed that the performance is effectively improved with the increase of the iteration number. And we can also observe that the performance improvement is very small after the third iteration. That means the convergence rate of the iterative decoding of the proposed system is fast. This conclusion can also be explained by the EXIT chart. From Fig 4.3, we can observe that the slope of the curve of MSDD SISOD with $L = 4$ is not steep and close to the slope of the curve of the LDPC decoder, which means that the convergence of the iterative decoding only requires a small number of iterations [19],[27].

In Fig. 4.9, the BER performances of the proposed system over slow and fast fading channels with different OWS are compared. And the performance of coherent detection of BPSK of the proposed system with CSI known is also included for comparison. We can observe that the performance of the proposed system can get close to that of coherent detection. We can also find that the performance is much better in fast fading channels. And extending L in fast fading channels, the improvement of performance is more significant than that in slow fading channels. The same phenomena can be also observed from Fig. 4.7 and Fig. 4.8. This is because the system under consideration can exploit the time diversity benefit which is available in fast fading channels.

In addition, it is known that increasing the length of LDPC codes can greatly improve the performance of the LDPC coded systems. The performance of the systems under consideration with long LDPC codes is not shown on here, since the characteristics of the performance is similar to that of the above simulation results. Furthermore, in the case of DE-LDPC coded systems with regular LDPC codes, regardless of using short or long codes, the performance almost cannot be improved by increasing the OWS of MSDD SISOD when the OWS is bigger than 8. Moreover, there is a relatively large gap between the performance of the systems under consideration with regular codes and the capacity of the noncoherent MPSK channel with MSDD, which will be shown in the next section. This is because regular LDPC codes, which have good performance for coherent detection, are not optimized for the DE-LDPC coded systems with MSDD. Therefore, in order to further improve the performance of the DE-LDPC coded systems with MSDD, it is necessary to design the irregular LDPC codes for MSDD SISOD with different OWS.

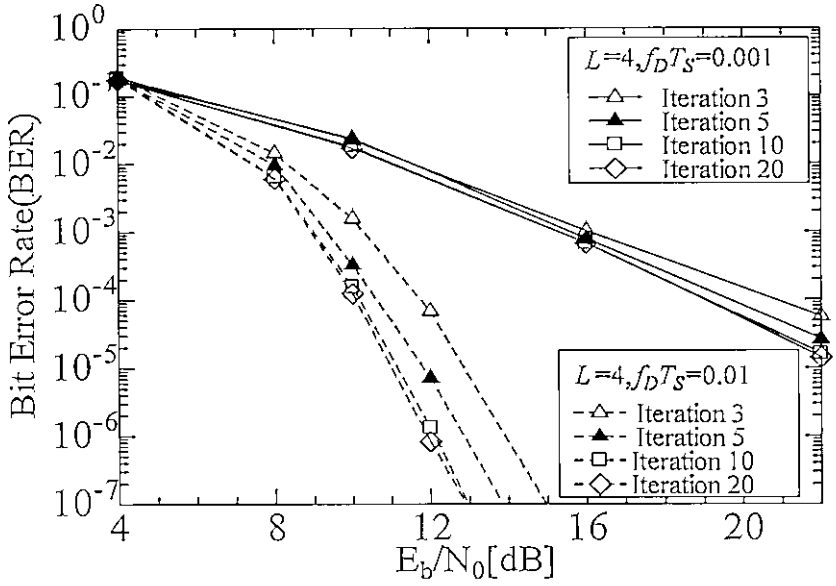

Figure 4.7: Effect of iteration number of LDPC decoder on BER performance.

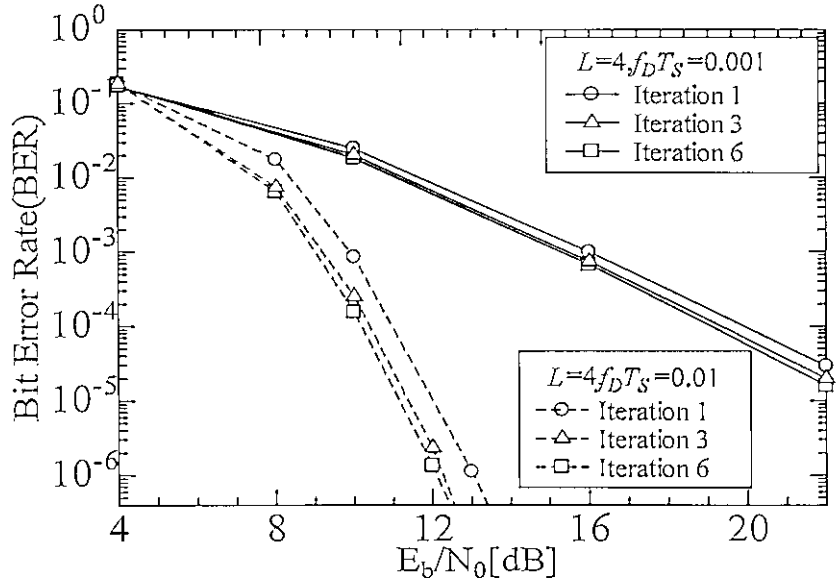

Figure 4.8: Effect of iteration number of iterative decoding on BER performance.

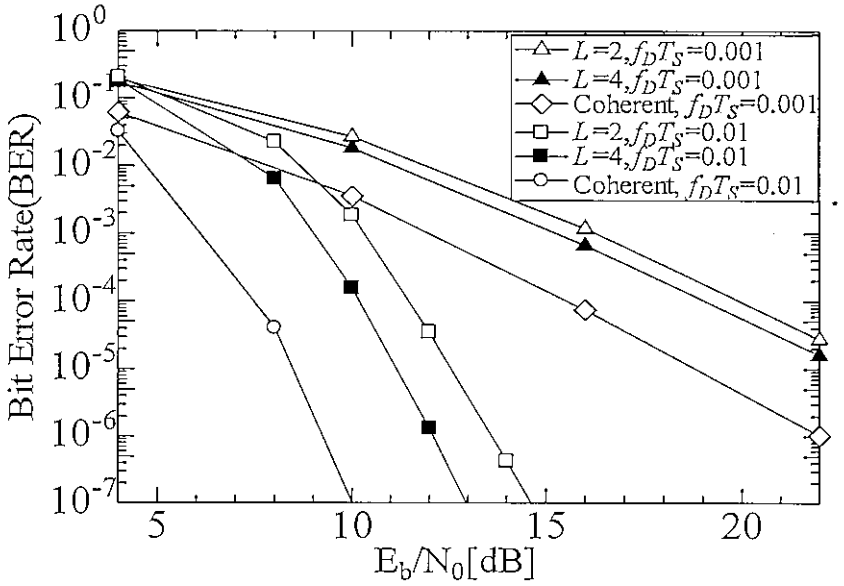

Figure 4.9: BER performance comparison of proposed system over slow and fast fading channels.

4.5 LDPC Codes Optimization for DE-LDPC Coded Systems with Iterative MSDD

Many works have been reported on how to design the irregular LDPC codes to close the channel capacity of different systems, such as coherent detection systems over AWGN channels, MIMO systems, etc. [22], [28]. In [28] and [5], an EXIT curve-fitting approach was proposed to design the optimized irregular LDPC codes for MIMO systems. With this approach, irregular LDPC codes optimized for serial concatenation systems of LDPC codes and differential modulation (DM) with coherent detection was studied in [5]. It was shown that the optimized LDPC codes for DM perform significantly better than non-optimized LDPC codes. However, until now, optimizing the LDPC codes for the iterative MSDD systems has not been studied.

In this section, we first introduce the irregular LDPC codes and its EXIT charts. Then, we describe how to optimize irregular LDPC codes by EXIT chart for the systems under consideration. Finally, the performance of the systems under consideration with the optimized LDPC codes is evaluated using computer simulations.

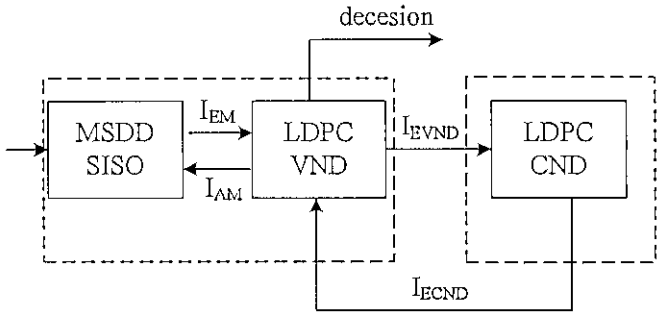

Figure 4.10: Block Diagram of the receiver of the systems under consideration.

4.5.1 Irregular LDPC Codes and Its EXIT Chart

From Section 2.2 of Chapter 2, an irregular LDPC code ensemble can be specified by two polynomials:

$$\lambda(x) = \sum_{i=2}^{d_{vmax}} \lambda_i x^{i-1}, \quad \rho(x) = \sum_{j=2}^{d_{cmax}} \rho_j x^{j-1}, \tag{4.13}$$

where λ_i (ρ_j) is the fraction of edges with variable (check) nodes of degree i (j), and d_{vmax} (d_{cmax}) is the maximum degree of variable (check) nodes, respectively. The coefficients λ_i and ρ_j must satisfy the following constraints:

$$\sum_{i=2}^{d_{vmax}} \lambda_i = 1, 0 \leqslant \lambda_i \leqslant 1; \qquad \sum_{j=2}^{d_{cmax}} \rho_j = 1, 0 \leqslant \rho_j \leqslant 1; \tag{4.14}$$

and

$$\sum_{j=2}^{d_{cmax}} \frac{\rho_j}{j} = (1-R) \sum_{i=2}^{d_{vmax}} \frac{\lambda_i}{i}. \tag{4.15}$$

In the LDPC decoder, the sets of variable and check nodes are referred to as the variable-node decoder (VND) and check-node decoder (CND), respectively. In order to design irregular LDPC codes, in the EXIT chart analysis, similar to [5], we combine the MSDD SISOD and VND into one component for the EXIT curve analysis and treats the CND as the other as shown in Fig. 4.10, in contrast to treat the MSDD SISOD as one component and the entire LDPC decoder as the other which is used in our previous studies. In Fig. 4.10, the mutual information at the output of LDPC VND and CND are denoted as I_{EVND} and I_{ECND}, respectively; the mutual information at the input and output of the MSDD SISOD are labeled I_{AM} and I_{EM}, respectively. The approximate formulas of the EXIT curves I_{EVND} and I_{ECND} are given as:

$$I_{EVND} = \sum_i \lambda_i J(\sqrt{(i-1)(J^{-1}(I_{ECND}))^2 + (J^{-1}(I_{EM}))^2}), \tag{4.16}$$

$$I_{ECND} = 1 - \sum_j \rho_j J(\sqrt{j-1} J^{-1}(1 - I_{EVND})), \tag{4.17}$$

where the calculation of the functions J and J^{-1} can refer to [28]. The examples of the EXIT chart for the DE-LDPC coded systems under consideration with OWS $L = 4$ using a rate-1/2 (3, 6) regular LDPC code and an irregular LDPC code are shown in Fig. 4.11 and 4.12, respectively. The irregular LDPC code is the optimized LDPC code obtained by [22], of which the threshold value is only 0.06 dB away from the capacity of AWGN channels. The degree distributions of this code are shown in Table 4.1.

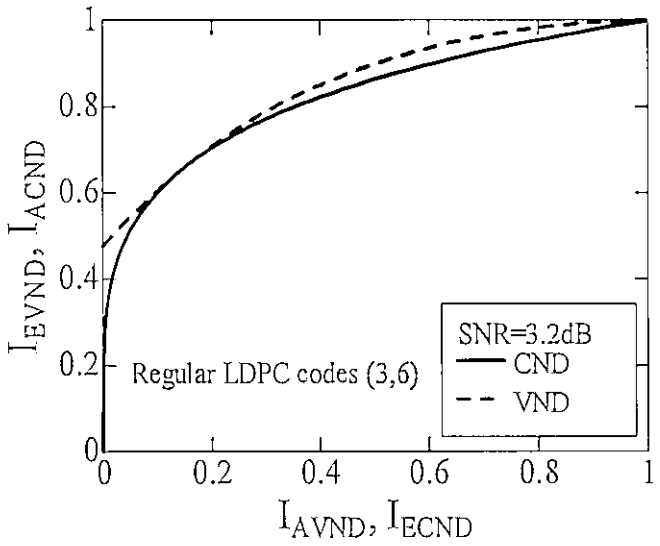

Figure 4.11: EXIT charts of the systems under consideration with OWS $L = 4$ and rate-1/2 (3, 6) regular LDPC codes over AWGN channels with BPSK.

Table 4.1: Degree distribution of irregular LDPC codes [22]

Degree distribution
$\lambda_2 = 0.17120$, $\lambda_3 = 0.21053$, $\lambda_4 = 0.0.00273$, $\lambda_7 = 0.00009$, $\lambda_8 = 0.15269$, $\lambda_9 = 0.09227$, $\lambda_{10} = 0.02802$, $\lambda_{15} = 0.01206$, $\lambda_{30} = 0.0.07212$, $\lambda_{50} = 0.25830$.
$\rho_9 = 0.33260$, $\rho_{10} = 0.08883$, $\rho_{11} = 0.57497$.

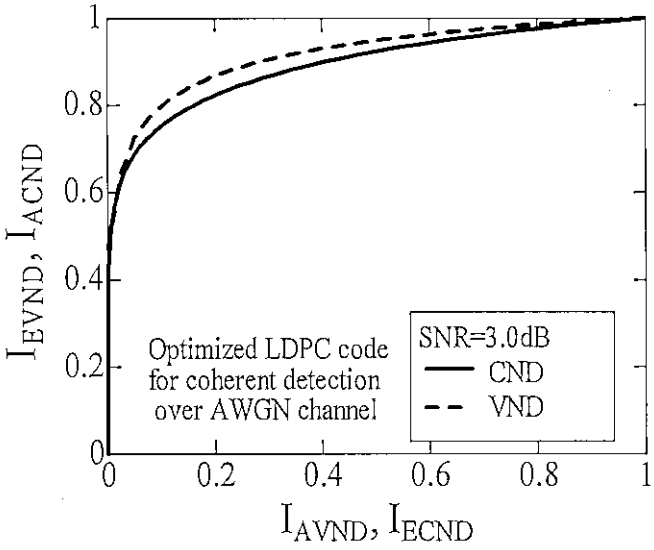

Figure 4.12: EXIT charts of the systems under consideration with OWS $L = 4$ and optimized irregular LDPC codes of [22] over AWGN channels with BPSK.

4.5.2 Design and Optimization of Irregular LDPC Codes Based on EXIT Charts

It is known that the iterative decoding can successfully converge when a tunnel is opened between the I_{EVND} curve and the I_{ECND} curve. We can use the EXIT curve-fitting approach to optimize irregular LDPC codes. The basic principle of this approach is that searching the set of degree distributions of irregular LDPC codes makes the I_{ECND} curve stay strictly below the I_{EVND} curve at an SNR as low as possible.

We construct a cost function to represent the relation of the two curves as follows

$$f = min\left\{I_{EVND}(I_i) - I_{ECND}^{-1}(I_i)\right\}, \quad 0 < I_i < 1, \tag{4.18}$$

where I_i is average selected between 0 and 1. Here should be noted that $I_{ECND}^{-1}(I_i)$ denotes the inverse of $I_{ECND}(I_i)$, since the EXIT curve of the LDPC CND in the EXIT chart corresponds to the inverse of I_{ECND}^{-1}. When $f > 0$, it means that the tunnel is opened between the two curves. In contrast, when $f < 0$, it means that the two curves intersect, that is, the tunnel is closed. Whereas, when $f = 0$, it means that the two curves touch with each other, which means that the tunnel can be opened by a small SNR increment.

Using the cost function of (4.18), the procedure of optimizing irregular LDPC codes is described as follows:

(1) Given a certain SNR, we first obtain the EXIT curve I_{EM} of the MSDD SISOD. Set the maximum number of searches as S_{max}.

(2) By differential evolution method [29], we search $\{\lambda_i\}$ and $\{\rho_j\}$ to maximize the minimum of (4.18).

Table 4.2: Degree distribution of optimized LDPC codes for $L = 6$ and 10

	Degree distribution
$L = 6$	$\lambda_2 = 0.527,\ \lambda_3 = 0.1217,\ \lambda_{12} = 0.3513,$ $\rho_6 = 1.0.$
$L = 10$	$\lambda_2 = 0.781,\ \lambda_3 = 0.0037,\ \lambda_{26} = 0.2153,$ $\rho_5 = 1.0.$

(3) If $f > 0$, decrease SNR and go back to step 2. Otherwise, the set of $\{\lambda_i\}$ and $\{\rho_j\}$ corresponding to the maximum value of (4.18) is outputted.

Based on the principle of the EXIT curve-fitting approach, it is can be viewed as a global optimization problem. When the numbers of $\{\lambda_i\}$ and $\{\rho_j\}$ which are required to be optimized are large, the complexity of the process of optimizing LDPC codes is prohibitively high. To overcome this problem, in this thesis, following the method of [28], we limit the number of different VN degrees and different CN degrees to 3 and 1, respectively. In this case, based on the objective code rate and the degree distribution constraints (4.14) and (4.15), the only one degree of VNs needs to be optimized. By this way, the complexity of optimization of LDPC codes is greatly reduced. In the following, we will show that we can obtain the well-designed LDPC codes using this method.

In the following, we give the two examples of the optimized LDPC codes using the above method. We consider the optimization of a rate 1/2 irregular LDPC code for the DE-LDPC coded systems with BPSK over AWGN channels. The optimized examples of the systems under consideration with OWS $L = 6$ and 10 are detailed in Table 4.2. The EXIT charts corresponding to the systems under consideration using the two optimized codes and rate-1/2 (3, 6) regular LDPC codes are shown in Fig. 4.13 and 4.14, respectively. It is easy to see that using the optimized LDPC codes, tunnels are opened for $L = 6$ and $L = 10$ at SNR > 2.2 dB and SNR > 1.7 dB, respectively. Thus, we can predict that the decoding thresholds of the system with the two optimized LDPC codes are SNR $= 2.2$ dB for $L = 6$ and SNR $= 1.7$ dB for $L = 10$, respectively. Whereas, It is obvious that the decoding threshold corresponding to the system with ate-1/2 (3, 6) regular LDPC codes is much larger than that of the two optimized LDPC codes. In order to more clearly understand the relation of the characteristics of the MSDD SISOD and the LDPC decoder, we also present the EXIT charts for the system model of Fig. 2.1 as shown in Fig. 4.15 and 4.16. From the two figures, we can also obtain the same analysis results.

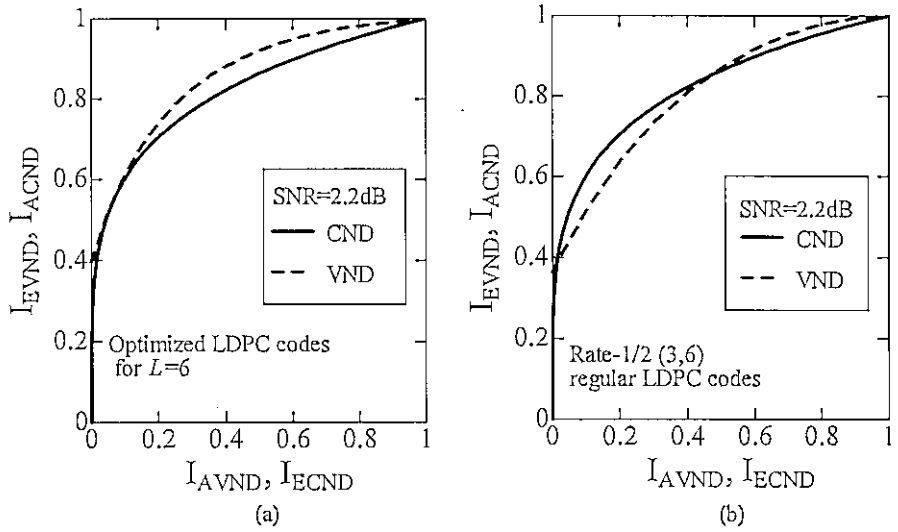

Figure 4.13: EXIT chart of the systems under consideration with OWS $L = 6$ and optimized LDPC codes for $L = 6$ with length 100800 over AWGN channels.

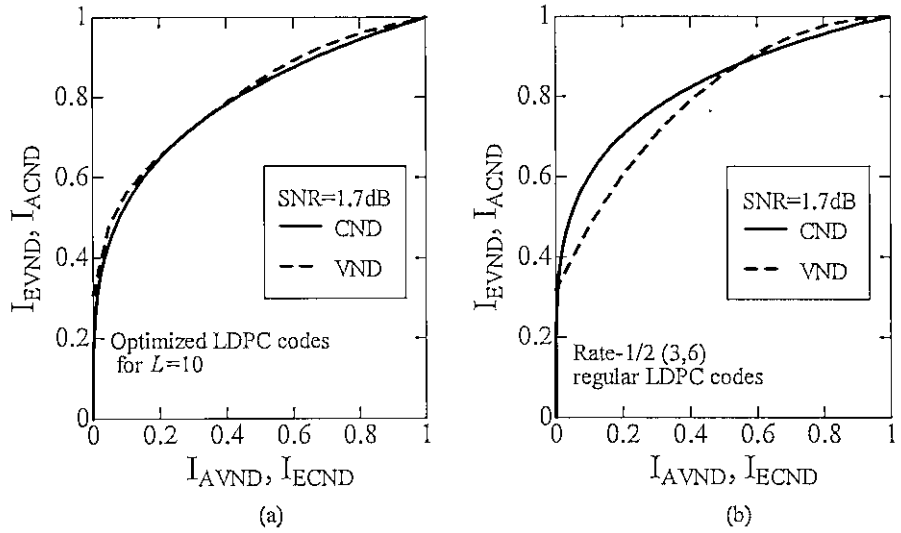

Figure 4.14: EXIT chart of the systems under consideration with OWS $L = 10$ and optimized LDPC codes for $L = 10$ with length 100800 over AWGN channels.

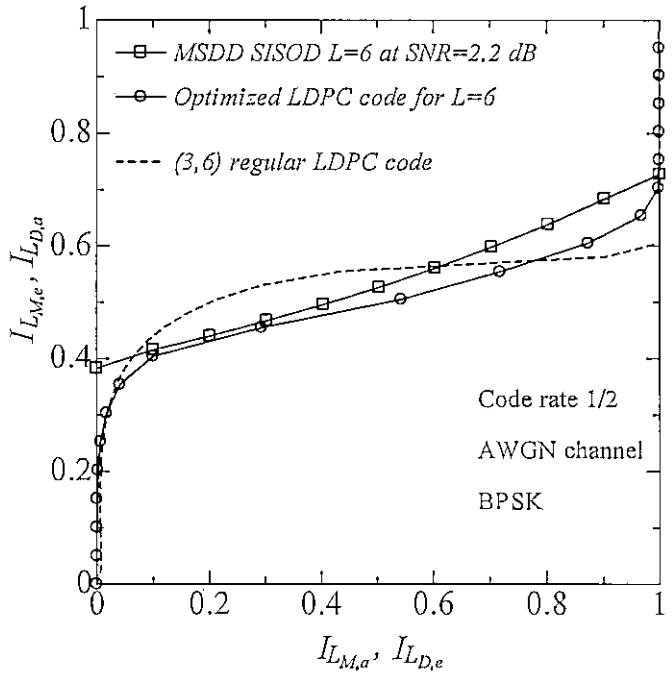

Figure 4.15: EXIT chart of the systems under consideration with OWS $L = 6$ and optimized LDPC codes for $L = 6$ with length 100800 over AWGN channels.

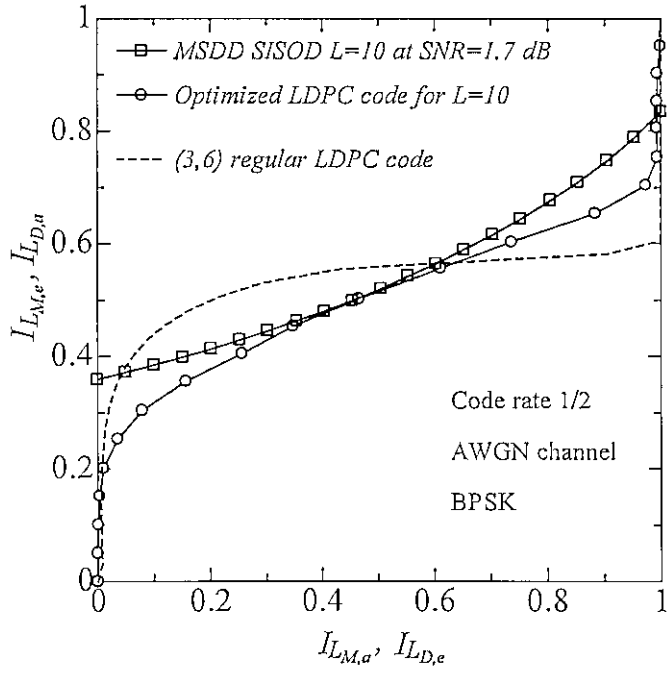

Figure 4.16: EXIT chart of the systems under consideration with OWS $L = 10$ and optimized LDPC codes for $L = 10$ with length 100800 over AWGN channels.

4.5.3 Evaluation of the Design Examples

Decoding thresholds corresponding to the design results of irregular LDPC codes for MSDD SISOD with $L = 6$ and 10, which are predicted by Fig. 4.13 and 4.14, are illustrated in Table 4.3. The results of the BER performance for the considered systems with the optimized and non-optimized codes are shown in Fig. 4.15 and 4.16. In the simulation, the code length of LDPC codes is 100800, and the maximum iteration number of the LDPC decoder is set 20. Unless otherwise indicated, the maximum iteration number between the MSDD SISOD and LDPC decoder is set to 8.

From the simulation results, we observe that compared to using the regular (3, 6) LDPC codes, optimized irregular LDPC codes provide much better performance for the considered systems. It is also shown that the gap to the decoding threshold can be narrowed by using more iterations as shown in Fig. 4.16 and longer codes. Moreover, the decoding thresholds observed from the BER performance is very close to that of design results as shown in Table 4.3, which proves the accuracy of our optimization method and results.

In addition, we also evaluate the performance of the systems under consideration using the optimized LDPC codes for different OWS. As shown in Fig. 4.17, compared with using non-optimized LDPC codes for the considered systems, the performance of the system with $L = 2, 4, 10$ can also be significantly improved by the optimized LDPC codes for $L = 6$. This result shows the importance of optimizing LDPC codes for the DE-LDPC coded systems with MSDD once again.

The above simulations are all for the very long LDPC codes. Whereas, in practical systems, LDPC codes are generally limited to blocks of a few hundred to a few thousand code bits. In the following, we evaluate the performance of the systems under consideration using the optimized LDPC codes with finite length. It is known that we can obtain a well-designed LDPC code with a very long length using the random construction approach to construct the LDPC code based on the degree distributions. However, for finite length LDPC codes, we need to construct the LDPC codes to avoid the error floor, which is easily appeared for the irregular LDPC codes with finite length. To improve the error floor performance, we use the progressive-edge-growth (PEG) algorithm [30] to construct the optimized LDPC codes. PEG is well-known for the construction of good LDPC codes with short to medium block length. Using this algorithm, we can remove the short cycles in the code graph. We use the PEG algorithm to construct an LDPC code with length 1008 based on the degree distributions that is optimized for $L = 6$. Using this LDPC code, the performance of the system under consideration over Rayleigh fading channel with $f_D T_s = 0.01$ is shown in Fig. 4.18. We can observe that in the case of short length code, the performance can also be improved significantly using the optimized LDPC codes.

Table 4.3: Comparison of decoding thresholds of the systems under consideration with different LDPC codes over AWGN channels.

	Decoding threshold with optimized codes	Decoding threshold with rate-1/2 (3,6) regular codes	Capacity of noncoherent MPSK AWGN channel with MSDD at rate 1/2 [31]
$L = 6$	2.2 dB	3.0 dB	2.0 dB
$L = 10$	1.7 dB	2.7 dB	1.3 dB

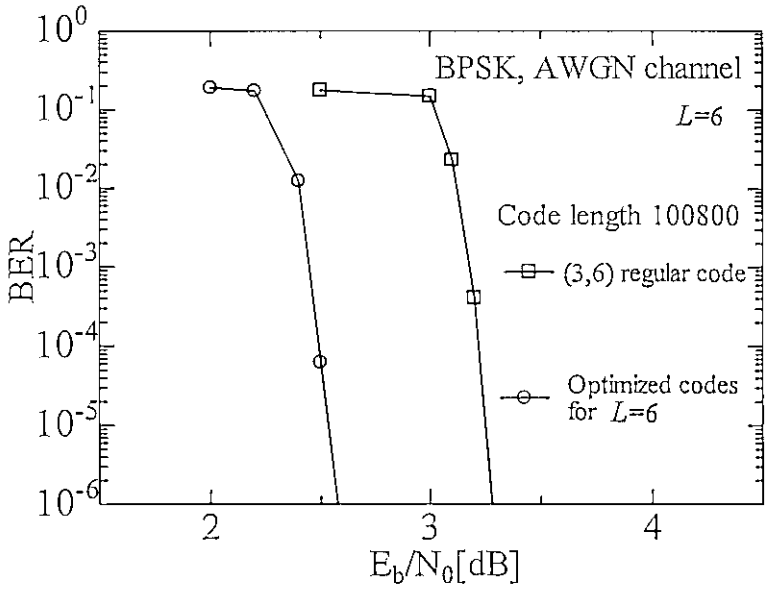

Figure 4.17: BER performance comparison of the systems under consideration with non-optimized LDPC codes and optimized LDPC codes for OWS $L = 6$.

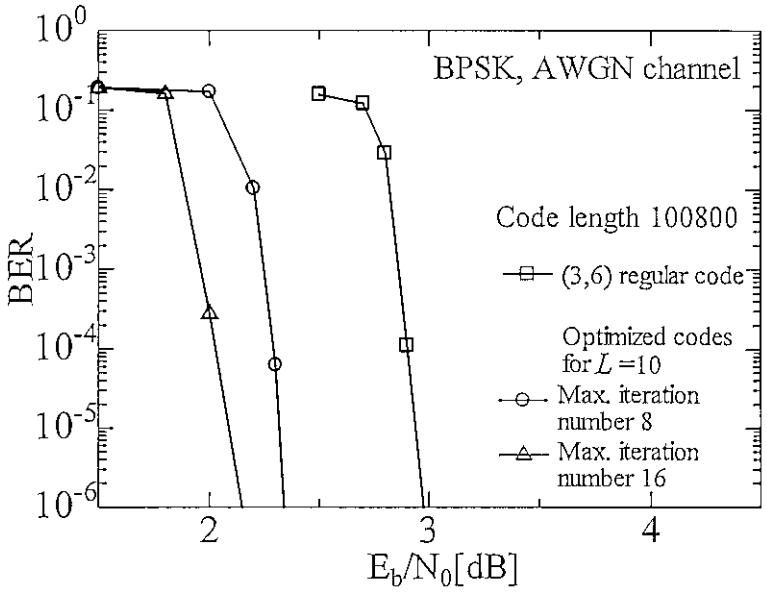

Figure 4.18: BER performance comparison of the systems under consideration with non-optimized LDPC codes and optimized LDPC codes for OWS $L = 10$.

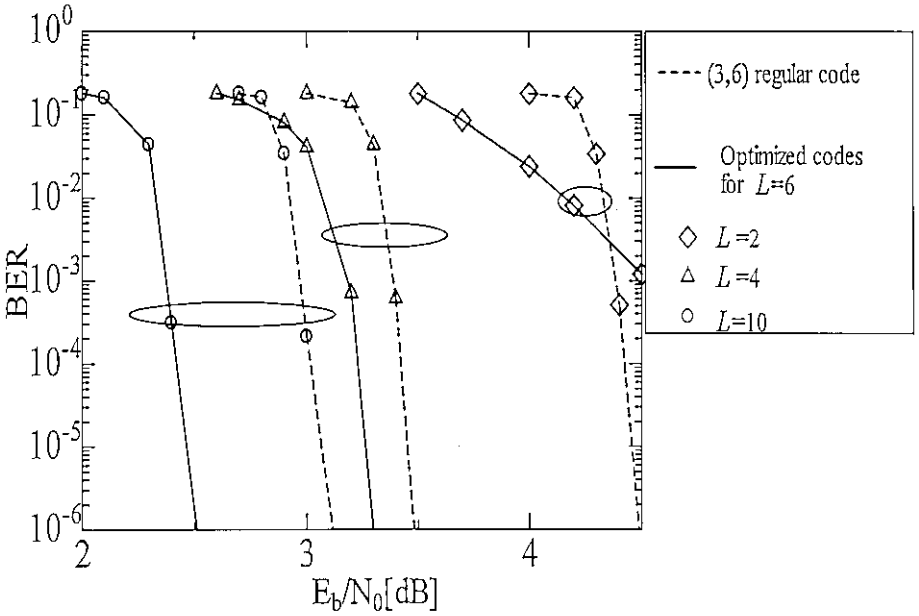

Figure 4.19: BER performance of the systems under consideration with optimized LDPC codes for $L = 6$ using different OWS.

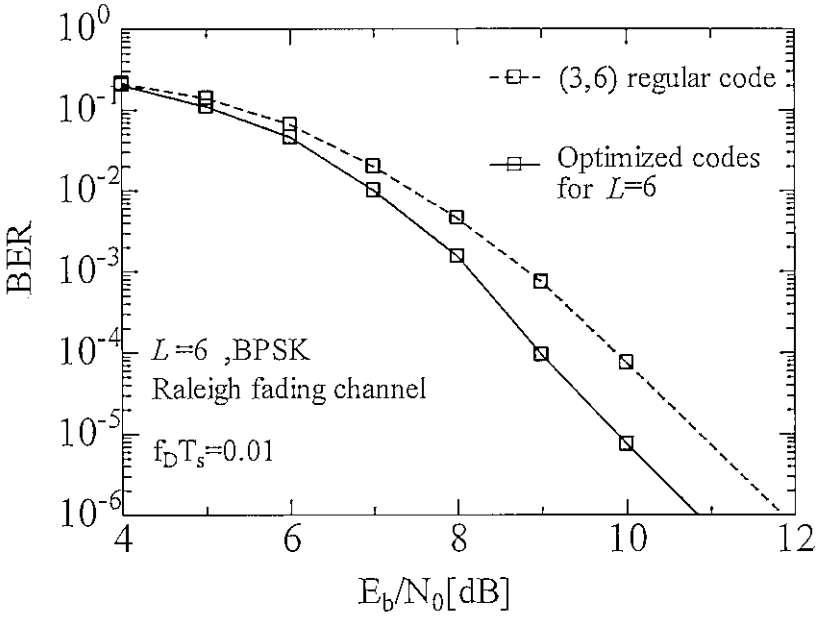

Figure 4.20: BER performance of the systems under consideration with optimized LDPC codes for $L = 6$ with length 1008 over Rayleigh fading channels.

CHAPTER 5

Improved Soft-Output M-Algorithm for Differentially Encoded LDPC Coded Systems with Iterative Multiple-Symbol Differential Detection

In Chapter 4, an iterative MSDD scheme is investigated for DE-LDPC coded systems. It is shown that the system performance can be improved by increasing the OWS of the MSDD SISOD and the iteration number between the MSDD SISOD and the LDPC decoder. However, the MSDD SISOD is a maximum *a posteriori* (MAP) detector, the computational complexity of which increases exponentially with the OWS and the order of the modulator. The high complexity of the MSDD SISOD with a large OWS makes its application infeasible in practical systems especially when the iterative decoding is performed at the receiver. Therefore, it is necessary to develop an approach to reduce the complexity of the MSDD SISOD while maintaining the reliability of its soft-output. In this chapter, we propose an improved soft-output M-algorithm (ISOMA) and use it to reduce the computational complexity of DE-LDPC coded systems with iterative MSDD. Simulation results show that the computational complexity of the MSDD SISOD as well as the iterative decoding complexity of DE-LDPC coded systems with iterative MSDD can be greatly reduced by the proposed ISOMA.

The remainder of this chapter is organized as follows. Section 5.1 introduces the related works. In Section 5.2, after discussing the existing SOMA for the considered systems, we propose the ISOMA. Then, performance analysis by computer simulation is given in Section 5.3.

5.1 Related Works

In [12] and [32], in order to reduce the complexity of MSDD, the M-algorithm [33] was adopted to MSDD for uncoded systems. However, the M-algorithm for the MSDD with soft-output has not been investigated. Moreover, the M-algorithm cannot be directly extended to the coded systems with MSDD, since only M best paths are retained in the M-algorithm, which often leads to these paths having the same binary values in the same bit positions. In such a case, the LLRs of these coded bits cannot be computed. To solve this problem, two existing approaches can be utilized. In [34], an iterative tree search detection based on the M-algorithm (ITS-MA) was proposed for iterative BICM MIMO systems. In this approach, the soft information is computed based on M retained paths. While for those bits whose soft information cannot be computed, the clipping method is performed to assign fixed negative or positive values to these

bits. However, this method reduces the reliability of the soft information of coded bits, especially when M is small, which results in the degradation of the system performance. In [36] and [37], a SOMA was proposed, which obtains the LLRs by using both of the retained and terminated paths. The SOMA can avoid assigning the clipping values, but its performance is close to that of the soft-output Viterbi algorithm (SOVA) [35] but not close to that of the MAP algorithm.

5.2 Complexity of MSDD SISOD

As introduced in Chapter 3 and 4, MSDD makes a joint decision on several symbols simultaneously based on the ML principle by extending the OWS to more than two symbols. For our considered systems, the MSDD SISOD outputs the soft information for the LDPC decoder.

The system model under consideration is the same with Fig. 4.1. Here we assume that the OWS of the MSDD SISOD is L, and the length of LDPC codes is N. Let m denote the number of bits of each MPSK symbol. The received sequence r is divided into sub-blocks r_k, where $k \in \{0, (N/m)/(L-1)\}$.

Let $\mathrm{s}_k = [s_{k,0}, s_{k,1}, ..., s_{k,L-1}]$ denote the differentially encoded sequence corresponding to the kth sub-block of received symbols, and let $\mathrm{c}_k = [c_{k,1}, c_{k,2}, ..., c_{k,m(L-1)}]$ denote the coded bits vector corresponding to the s_k. For convenience, we drop the index k in the following. Based on the principle of the MAP algorithm, as introduced in Section 4.3, the MSDD SISOD generates the *a posteriori probability* of the coded bit c_i, written in terms of LLR

$$L_{M,p}(c_i) = \log \frac{\sum_{\mathrm{s}:c_i=0} p(\mathrm{r}|\mathrm{s}) \prod_{j=1}^{m(L-1)} p(c_j)}{\sum_{\mathrm{s}:c_i=1} p(\mathrm{r}|\mathrm{s}) \prod_{j=1}^{m(L-1)} p(c_j)}, \tag{5.1}$$

where the sums in the numerator and denominator are taken over all sequences c whose bit in position i is the value 0 or 1, respectively. $p(c_j)$ is the *a priori* probability provided by the LDPC decoder. $p(\mathrm{r}|\mathrm{s})$ is the conditional PDF of r given s. In the case of AWGN channels, $p(\mathrm{r}|\mathrm{s})$ is given by

$$p(\mathrm{r}|\mathrm{s}) = \frac{1}{(2\pi\sigma_n^2)^L} \exp[-\frac{1}{2\sigma_n^2} \sum_{i=0}^{L-1} (|r_i|^2 + |s_i|^2)] \times I_0(\frac{1}{\sigma_n^2} \left| \sum_{i=0}^{L-1} r_i s_i^* \right|), \tag{5.2}$$

where $I_0(\cdot)$ is the zero-order modified Bessel function of the first kind, and the superscript '*' denotes the complex conjugation. Since all terms that are not in the Bessel function are common to all s, (5.2) can be simplified to

$$p(\mathrm{r}|\mathrm{s}) \propto I_0(\frac{1}{\sigma_n^2} \left| \sum_{i=0}^{L-1} r_i s_i^* \right|), \tag{5.3}$$

which is used in (5.1) to compute $L_{M,p}(c_i)$.

The complexity of the MSDD SISOD with the MAP algorithm is mainly decided by two aspects:

1) Number of terms in the summation of (5.1): Since all candidate bit sequences need to be taken into account in (5.1), the number of terms in the summation of (5.1) is equal to $2^{m(L-1)}$, which increases exponentially with the OWS and the order of the modulation.

2) Computation of the PDF $p(\mathrm{r}|\mathrm{s})$: Since in the absence of CSI at the receiver, the computation of $p(\mathrm{r}|\mathrm{s})$ cannot be computed recursively as in the coherent detection case. Thus, the computational of $p(\mathrm{r}|\mathrm{s})$ also becomes a large computational cost with the increase of the OWS of the MSDD SISOD.

Therefore, the complexity of (5.1) will become prohibitively high as the OWS and the order of the modulation become large. Especially for iterative decoding systems, this high complexity will result in an unacceptable decoding time delay, which makes difficult to achieve a realistic system. To solve this problem, we employ the M-algorithm to reduce the complexity of the MSDD SISOD, which is specifically explained in the following section.

5.3 SOMA for MSDD SISOD

5.3.1 M-Algorithm for MSDD

M-algorithm is a breadth-first tree search algorithm, which has been proved to be highly efficient for tree decoding problems. The basic principle of the M-algorithm is that starting from the first node of the decoding tree, only M paths which correspond to the best M values of metrics are retained at each tree depth, and the rest paths are discarded. When the M-algorithm reaches the end of the decoding tree, the fist-ranked path in M paths is the most likely candidate.

Since the problem of MSDD also can be viewed as a tree decoding problem, the hard output of MSDD actually is the path with the ML value of the metric. For uncoded system with MSDD, the M-algorithm has been used to reduce the complexity of MSDD [12], [32]. The metric of MSDD with M-algorithm is the PDF $p(\mathrm{r}|\mathrm{s})$.

A tree diagram of MSDD for M_sPSK with M-algorithm is shown in Fig. 5.1. From node $i = 1$, the M-algorithm retains M best states of $\mathrm{S}_1 \ldots \mathrm{S}_{L-1}$ which maximize $p(\mathrm{r}|\mathrm{s})$ at any node of i, and then moves on to the next node of i until the end of the tree is reached. It should be noted that if M is larger than the number of all states of $\mathrm{S}_1 \ldots \mathrm{S}_{L-1}$, the M-algorithm retains all states. From Table 5.1, it is shown that the complexity of MSDD can be greatly by the M-algorithm.

Table 5.1: Number of search paths of MSDD for 16PSK

N	4	5	6
Conventional method (Full path search)	4,096	65,536	1,048,576
M algorithm($M = 16$)	528	784	1,040

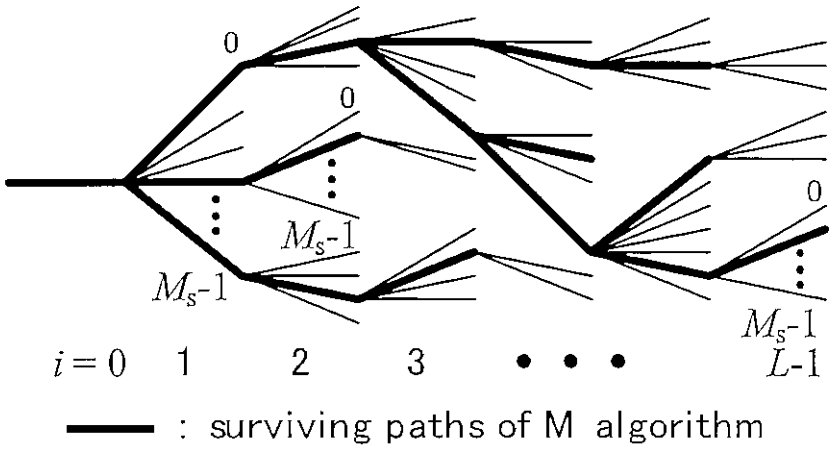

Figure 5.1: Tree diagram of MSDD for M_sPSK with M-algorithm

5.3.2 M-Algorithm for the MSDD SISOD

From the principle of the Max-Log-MAP algorithm [38], the computation of (5.1) can be approximated by two terms which have the best values of the numerator and the denominator of (5.1), respectively. That is, the output of the MSDD SISOD can be approximately computed by only a few candidate bit sequences which have the large values of the metric. Therefore, with small change of the metric of the path, the M-algorithm can be extended to reduce the complexity of the MSDD SISOD. From (5.1), the metric is given as

$$p(\mathrm{r}|\mathrm{s}) \prod_{j=1}^{m(L-1)} p(c_j). \tag{5.4}$$

With this direct extension scheme, the complexity of the MSDD SISOD can be reduced by the M-algorithm. However, this scheme cannot ensure the LLR of each coded bit can be computed, since the best M retained paths often have the same binary values in the same bit positions, which leads to the numerator or denominator of (5.1) equaling to zero. To solve this problem, two existing schemes can be considered to be used, which are briefly introduced as follows.

1) ITS-MA: The basic principle of the ITS-MA approach [34] is identical to the above mentioned direct extension scheme of the M-algorithm for outputting soft information. For those coded bits who cannot be computed by (5.1) using M finally retained paths, the ITS-MA assigns the appropriate clipping values for them. In [34], the clipping value is selected as -3 or +3. More specifically, if M retained paths only have 0 in the same position, the LLR of this coded bit is set to +3 based on (5.1); otherwise, the LLR of this coded bit is set to -3. When the value of M is the total number of tree paths, the ITS-MA is actually equivalent to the MAP algorithm.

2) SOMA: The heart of the SOMA is that not only M retained paths but also the discarded paths at each depth are used to compute the LLRs of coded bits. The compu-

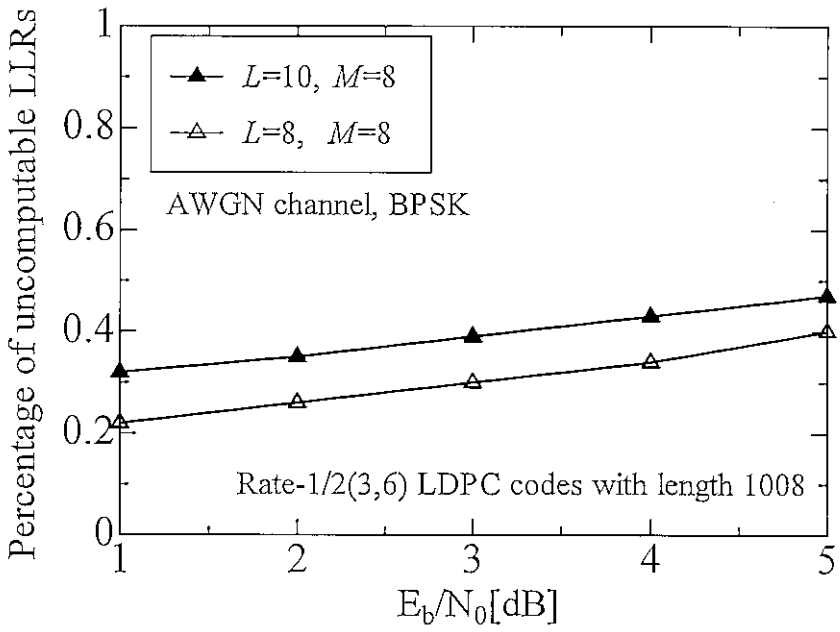

Figure 5.2: Percentage of bits with uncertain LLRs

tation method of the LLR is based on the metric difference between the approximated ML path and the set of the discarded paths, which is similar to the operations of the SOVA algorithm [35]. For more details, we refer to [36] and [37].

The above two approaches can be widely applied to concatenated code systems for complexity reduction. The ITS-MA is easy to perform, however, the reliability of the LLRs is significantly reduced especially when the value of M is small so that many bits need to be assigned a clipping value. As shown in Fig. 5.2, which shows the percentage of bits with uncertain LLRs using ITS-MA, about 30 percent of bits need to be assigned clipping values in the case of the systems under consideration with $L = 8$ and $M = 8$ at SNR = 5dB. Moreover, the percentage of bits with uncertain LLRs increases with the increase of the values of SNR and L. For the SOMA, it can avoid assigning the clipping values. However, the performance of the SOMA is close to that of the SOVA algorithm but not close to that of the MAP algorithm. Therefore, the performance loss still exists even a large value of M is used in the SOMA.

5.3.3 Proposed ISOMA for the MSDD SISOD

To overcome the disadvantages of the ITS-MA and SOMA, we propose an ISOMA approach. The features of the ISOMA are as follows:

1) Different from the ITS-MA computing the LLRs by (5.1) when the M-algorithm reaches the end of the tree, the ISOMA computes the LLRs of coded bits at each depth of the tree. More specifically, when computing the LLRs by (5.1) at each depth, similar to the SOMA, the ISOMA considers both M retained paths and discarded paths. In

other words, the LLRs are computed before discarding the paths at each depth. In this case, the paths at each depth must contain the information of both 0 and 1 of the coded bit corresponding to the current depth. Therefore, LLR of each coded bit can be computed.

2) At each depth, not only the LLR of the coded bit corresponding to the current depth is computed, the LLRs of coded bits corresponding to the positions before the current depth are also recomputed and updated, if (5.1) can be computed for these bits using the current paths. This feature of the ISOMA can guarantee the high reliability of the LLRs of coded bits.

When the value of M is set to the total number of tree paths, in practice, the ISOMA is also equivalent to the MAP algorithm as the ITS-MA. From the above mentioned features, we can find that the proposed ISOMA has features of both the ITS-MA and SOMA. Next, we specifically introduce the application of the ISOMA for the MSDD SISOD.

We introduce some notations that we will use in the following. For one sub-block of s, let W_i be the set of the paths at depth i. Denote the metric of the wth path at depth i by $\Gamma(\mathrm{s}_w^i)$, which is computed by

$$\Gamma(\mathrm{s}_w^i) = p(\mathrm{r}_i|\mathrm{s}_w^i) \prod_{p=1}^{m(i-1)} p(c_{w,p}^i), \tag{5.5}$$

where r_i is the received signals with length i, and $c_{w,p}^i$ is the pth coded bit of the coded bits vector corresponding to s_w^i.

Using the above notations, (5.1) can be rewritten as

$$L_M, p(c_i) = \log \frac{\sum_{\mathrm{s}_w^i : c_i = 0} \Gamma(\mathrm{s}_w^i)}{\sum_{\mathrm{s}_w^i : c_i = 1} \Gamma(\mathrm{s}_w^i)}. \tag{5.6}$$

The specific procedure of the ISOMA for the MSDD SISOD is described as follows:

(1) Initialization. Let W_0 be a set containing only the root node of the tree. The root node is the first symbol of each sub-block of s, and W_0 is set to 1 for the first sub-block.

(2) In one sub-block, for depth i, $1 \leqslant i \leqslant L-1$:

I) Extend each path in W_{i-1} to the next depth and compute the metric of each path in W_i using (5.5).

II) Compute the LLR of bit c_i corresponding to the depth i using all the paths in W_i by (5.6).

III) Recompute and update the LLRs of the bits before c_i using all the paths in W_i by (5.6). In the process of updating, if the LLRs of some bits cannot be recomputed, original LLRs of them are retained.

IV) If the number of paths in W_i is larger than M, go to step V; otherwise return to step I.

V) Sort the paths in W_i according to their values of metrics. Retain M best paths and extend them to the next depth $i+1$, then return to step I until the $(L-1)$th depth is reached.

(3) The last symbol of the best path in W_{L-1} is used as the root node of the tree for the next sub-block. Then return to (2) until the LLR of each LDPC coded bit is obtained.

5.3.4 Scaling Factor for the ISOMA

Due to only small fraction of all possible paths are considered in the computation of the LLRs, similar to the Max-Log-MAP algorithm and SOVA, ISOMA also has the problem of overestimating the LLRs, which is the main reason for degrading the performance of the system, especially when the value of M is small. To improve the decoding quality, the extrinsic LLRs are multiplied by a scaling factor for the Max-Log-MAP algorithm [38]. However, the performance of the scaling factor employed by the soft-output M-algorithm has not been studied in detail yet. The ISOMA with scaling factor (ISOMA-SF) is performed as follows:

The extrinsic LLRs output of the ISOMA is multiplied by a constant SF, $0 < \mathrm{SF} < 1$, that is

$$L'_{M,e}(c_i) = \mathrm{SF} \cdot L_{M,e}(c_i). \tag{5.7}$$

The $L'_{M,e}(c_i)$ is passed to the LDPC decoder as the *a priori* information.

5.4 Simulation Results and Analysis

In this section, the simulation results are presented and discussed. Unless otherwise indicated, the following simulation parameters are used for our simulations. We consider regular rate-1/2 (3, 6) LDPC codes with length of 1008. The Gary mapping rule is used for the modulator. Since we focus on the validity of ISOMA for the reduction of the complexity of the MSDD SISOD, we only consider the system over AWGN channels for the sake of simplicity. The maximum number of iterations of the LDPC decoder is set to 20. The maximum number of outer iterations between the MSDD SISOD and LDPC decoder is set to 8. In order to obtain the optimal performance of the SOMA for comparison purpose, all terminated paths are used by the SOMA for computing the LLRs.

5.4.1 BER Performance

Fig. 5.3 shows the BER performance of the ISOMA for the considered DE-LPDC coded systems with BPSK and the OWS $L = 10$. The performances of the MSDD SISOD using the MAP algorithm for $L = 2, 4$ and 10 are also presented for comparison. We observe that the significant performance improvement can be achieved when L is increased from 2 to 4 and then to 10 along with the growth of the complexity of the MSDD

SISOD. For $L = 10$, when the ISOMA is used in the MSDD SISOD, we observe that when $M = 16$, the performance of the ISOMA can be very close to the performance achieved by the MAP algorithm. In the case of the considered systems with QPSK and $L = 7, M = 16$ is also enough for the performance of ISOMA to be close to that of the MAP algorithm as shown in Fig. 5.4 (a). While for the case of 8PSK and $L = 7$, from Fig. 5.4 (b), we observe that to approach to the performance of the MAP algorithm, a larger M (32) is required by the ISOMA with respect to the case of BPSK, but the retained paths are still a small fraction of the decoding tree.

Fig. 5.5 shows the performance comparison between the ISOMA and ITS-MA for the considered DE-LPDC coded systems with different maximum number of outer iterations for $M = 4$ and $M = 16$, respectively. We can observe that the performance of the ITS-MA is much worse than that of the ISOMA when the maximum outer iteration number is smaller than 4, especially when $M = 4$. This is because the ITS-MA significantly reduces the reliability of the LLRs especially when a small value of M is used, as discussed in Section 5.3.2. In other words, the LLRs output of the MSDD SISOD with the ISOMA is much more reliable than that of the MSDD SISOD with the ITS-MA. In addition, we note that when the maximum outer iteration number is increased to 8, the performances of the two approaches are very close. This implies that the performance loss of the ITS-MA with respect to the ISOMA can be alleviated by increasing the maximum outer iteration number of the considered systems. However, it also implies that for the same BER performance, the ITS-MA requires more average number of outer iterations compared to the ISOMA, which will be supported in the next subsection.

Fig. 5.6 shows the performance comparison between the ISOMA and SOMA similar to Fig. 5.5. For the ISOMA and SOMA with $M = 4$, from Fig. 5.6 (a), it can be seen that the SOMA has the almost same iteration features and performance as that of the ISOMA. While when $M = 16$, the results of Fig. 5.6 (b) show that the SOMA has the almost same performance in the first few iterations but converges to worse performance than the ISOMA with the increase of the outer iteration number. This can be explained by the fact that with the increase of M, the performance of the SOMA is close to that of the SOVA, whereas the performance of the ISOMA is close to that of the MAP algorithm.

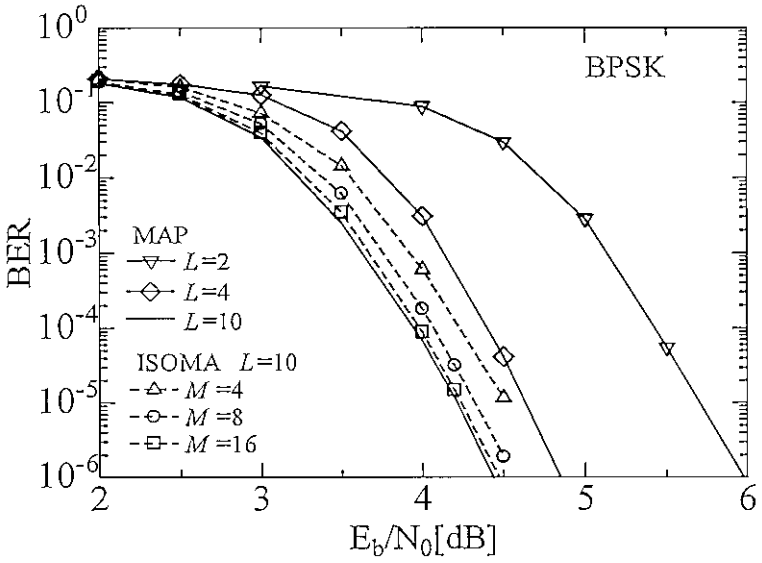

Figure 5.3: Performance comparison between the MAP algorithm and ISOMA used in MSDD SISOD for a DE-LDPC coded system with BPSK.

5.4.2 Decoding Complexity

In the following, decoding complexities of the system using the ISOMA, SOMA, ITS-MA and MAP algorithm are compared and analyzed by mainly considering the multiplication number (MN) and addition number (AN) of them. Due to the irregularity of the reduced tree by the M-algorithm, the number of bits whose LLRs cannot be recomputed/computed based on current paths are random when the ISOMA and ITS-MA are used. Therefore, we cannot give the accurate complexity of them. Thus, we assume that the LLR of each depth can be recomputed/computed in the two approaches. The upper bounds of the complexities of them are approximately given in the following simulation results. In addition, the calculation of the logarithmic term also is ignored.

Table 5.2 shows the MN and AN of the MSDD SISOD computing the LLRs for LDPC coded bits at each outer iteration by using the MAP algorithm and ISOMA, respectively. Based on the above discussion, in order to obtain a small performance loss, 16 is an appropriate value for M. In such a case, for BPSK with $L = 10$, we observe that the complexity of the ISOMA is about 27% of that of the MAP algorithm. In the case of QPSK with $L = 7$ using the ISOMA, about 5% of the complexity of the MAP algorithm is needed. Whereas in the case of 8PSK with $L = 7$ using the ISOMA, only about 0.2% of the complexity of the MAP algorithm is needed. These results confirm that a dramatic reduction of the computational complexity of the MSDD SISOD is realized with the ISOMA, especially when the order of the modulation and the value of L are large.

Next, we compare the complexities of the ISOMA, SOMA and ITS-MA with $M = 16$ at each outer iteration. In the following simulations, BPSK is used. As shown in Table

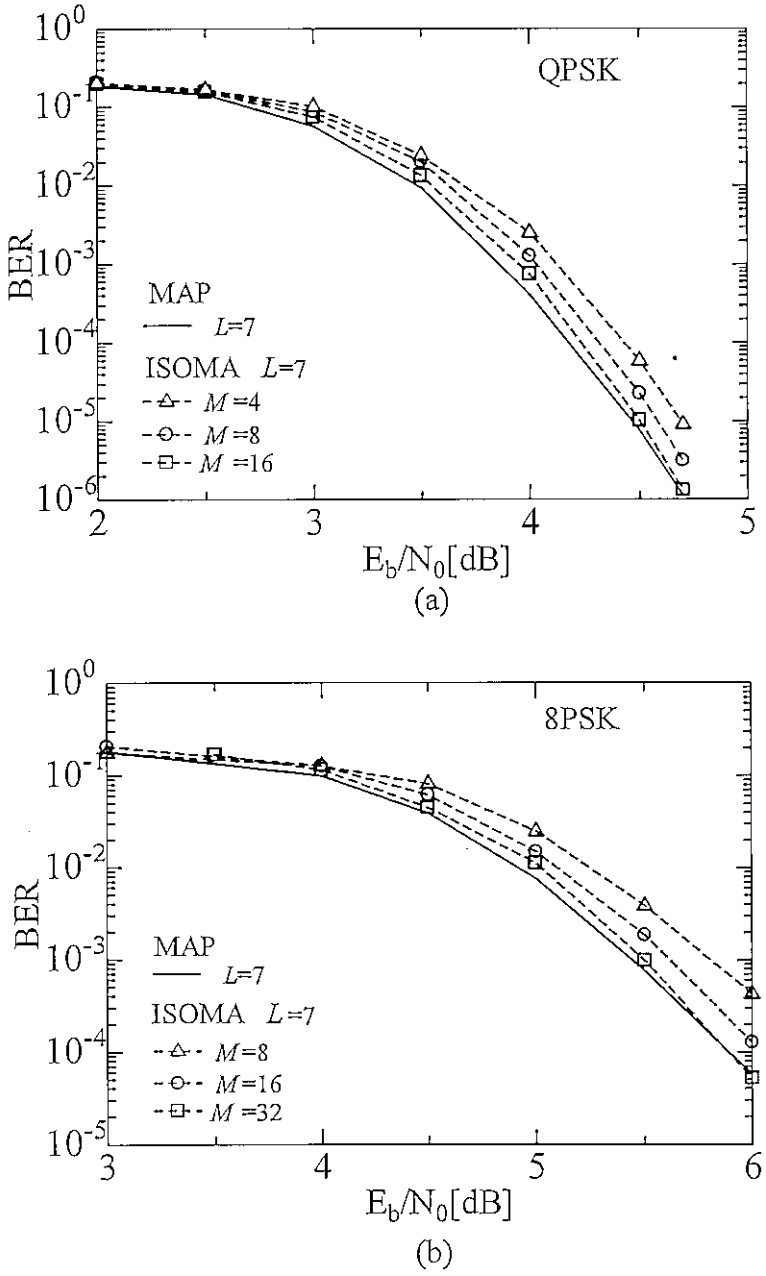

Figure 5.4: Performance comparison between the MAP algorithm and ISOMA used in MSDD SISOD for a DE-LDPC coded system with QPSK and 8PSK.

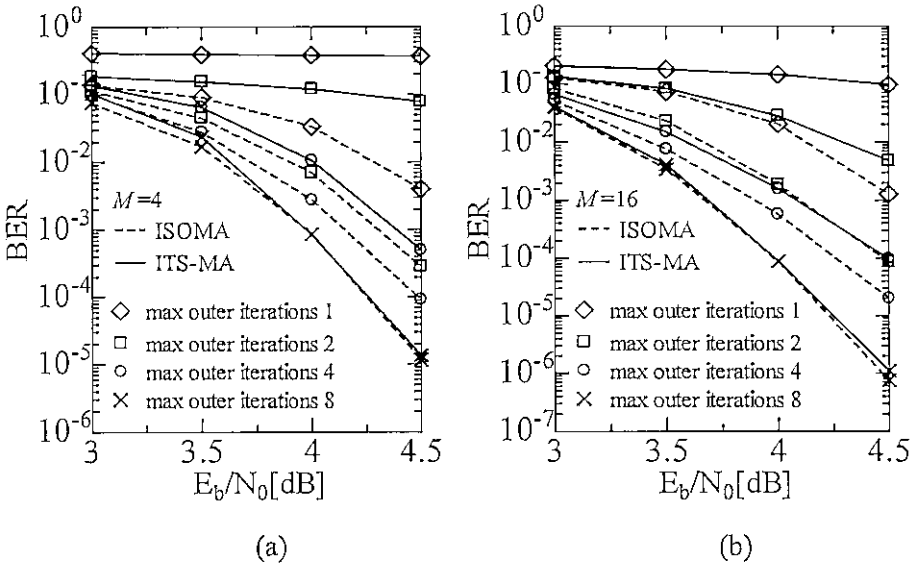

Figure 5.5: Performance comparison between the ISOMA and ITS-MA used in MSDD SISOD for a DE-LDPC coded system with BPSK and $L = 10$.

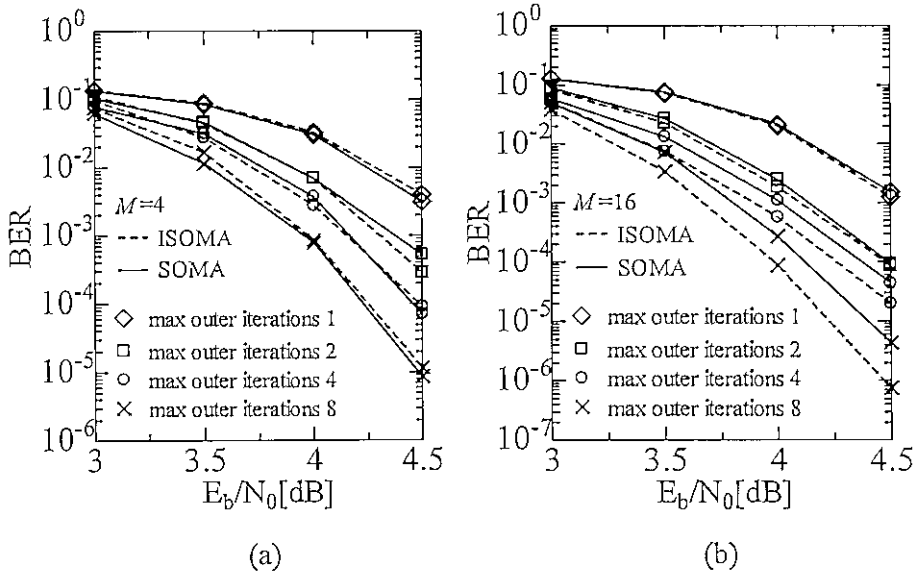

Figure 5.6: Performance comparison between the ISOMA and SOMA used in MSDD SISOD for a DE-LDPC coded system with BPSK and $L = 10$.

5.3, we can observe that the three approaches have almost same complexity in terms of the MN. This is because the three approaches all based on the M-algorithm need to compute the same number of path metrics. On the other hand, due to the different operations of computing the LLRs in the three approaches, we can observe that they have different numbers in terms of the AN. It is shown that the ISOMA needs slightly more number of additions compared to the ITS-MA and SOMA. Thus, from the point of view of one outer iteration, the computation complexity of the ISOMA is slightly more than that of the ITS-MA and SOMA.

In order to evaluate the decoding complexities of the considered DE-LDPC coded systems using the three approaches, we need to further test the average iteration number of the outer iteration and the LDPC decoder, respectively, since the LDPC decoder can detect successful decoding and stop the iteration by checking the parity check constraints of LDPC codes. Let n_{ave}^{O} and n_{ave}^{L} denote the average iteration number of the outer iteration and the LDPC decoder, respectively. The corresponding results for $M = 16$ are shown in Table 5.4 and 5.5. In addition, the results of the MAP algorithm are also given for comparison. We observe that the ISOMA is very close to the MAP algorithm in terms of n_{ave}^{O} and n_{ave}^{L}. This implies that the fraction of the saved complexity of all outer iterations using the ISOMA is similarly identical to that of one outer iteration as discussed in the above. For the SOMA and ITS-MA, it is shown that n_{ave}^{O} and n_{ave}^{L} required by the SOMA are both slight more than that required by the ISOMA, while n_{ave}^{O} and n_{ave}^{L} required by the ITS-MA are both obviously more than that required by the ISOMA. Therefore, we can conclude that the proposed ISOMA requires the least complexity for our considered DE-LDPC coded systems.

Table 5.2: Computational complexity for one outer iteration of the MSDD SISOD with the MAP algorithm and ISOMA (BPSK)

	L	MAP		ISOMA($M = 16$)		ISOMA($M = 4$)	
		MN	AN	MN	AN	MN	AN
BPSK	10	1.66E+6	1.55E+6	4.48E+5	4.04E+5	1.44E+5	1.22E+5
QPSK	7	7.91E+6	9.28E+6	4.30E+5	4.77E+5	1.27E+5	1.34E+5
8PSK	7	3.38E+8	4.84E+8	6.27E+5	8.28E+5	1.76E+5	2.22E+5

Table 5.3: Computational complexity for one outer iteration of the MSDD SISOD with the ISOMA, SOMA and ITS-MA (BPSK, $M = 16$)

	MN	AN
ISOMA	4.48E+5	4.04E+5
SOMA	4.49E+5	3.18E+5
ITS-MA	4.48E+5	3.16E+5

Table 5.4: Average number of outer iterations for the MSDD SISOD with the MAP, ISOMA, SOMA and ITS-MA (BPSK, $M = 16$): n_{ave}^{O}

SNR[dB]	3.0	3.5	4.0	4.5
MAP	4.72	2.35	1.36	1.03
ISOMA	4.86	2.46	1.38	1.03
SOMA	5.18	2.71	1.43	1.04
ITS-MA	5.80	3.61	2.52	2.04

Table 5.5: Average number of inner iterations for the LDPC decoder for the MSDD SISOD with the MAP, ISOMA, SOMA and ITS-MA (BPSK, $M = 16$): n_{ave}^{L}

SNR[dB]	3.0	3.5	4.0	4.5
MAP	18.37	15.37	12.24	7.80
ISOMA	18.44	15.58	12.38	7.92
SOMA	18.72	16.26	12.67	8.04
ITS-MA	18.70	17.09	15.51	13.59

5.4.3 Performance of ISOMA with Scaling Factor

Fig. 5.7(a) shows the performance of the ISOMA with the best evaluated SF compared to the standard ISOMA (SF = 1.0) for $M = 4$ and $M = 16$, respectively. For comparison, the same simulations are also evaluated for the MAP algorithm. For $M = 4$, it is shown that the ISOMA with SF = 0.7 can achieve a significant performance improvement compared to the standard ISOMA. At a BER of 10^{-5}, the performance of the ISOMA with SF = 0.7 is about 0.6 dB better than that of the standard ISOMA. For the ISOMA with $M = 16$ and the MAP algorithm with their best evaluated SFs, we observe that the performance improvement is not as significant as the case of the ISOMA using the best evaluated SF for $M = 4$. In addition, it is worth to note that the performance of the ISOMA with the SF for $M = 4$ is very close to that of the ISOMA with the SF for $M = 16$ and the MAP algorithm with the SF.

In order to understand why the SF can improve the performance of the ISOMA, the performance comparison between the ISOMA with SF and the standard ISOMA for different maximum number of outer iterations with $M = 4$ is examined as shown in Fig. 5.7(b). We observe that they have the same performance in the first outer iteration, but the ISOMA with the SF can achieve much more performance gain at each outer iteration after the first outer iteration. Therefore, a possible explanation for the observation of the performance improvement achieved by the SF is that the SF reduces the correlation of the extrinsic LLRs which are outputted to the LDPC decoder. In addition, from Fig. 5.7(a), it is interesting to see that the performance of the ISOMA with the SF = 0.7 is better than that of the standard MAP algorithm. The reason is that the LDPC code

used in the simulation is a short code (length 1008). Its performance is limited by the correlation of the extrinsic LLRs which is increased with the iterations. When a long LDPC code is used, this observation may not arise, as shown in Fig. 5.8.

Finally, we examine the impact of the value of SF on the performance of the ISOMA, as shown in Fig. 5.9. It is shown that the performance improvement achieved by the proper selected SF is increased with the increase of the SNR. Furthermore, we can observe that the best evaluated SF can be chosen for the whole SNR region for a fixed M. In other words, the best evaluated SF is not changed with the SNR.

Based on the above observations, we can confirm that the ISOMA with a properly selected SF can be used to improve the performance of the ISOMA (also for ITS-MA and SOMA). In addition, the performance gap between the small value and the large value of $M$for the ISOMA with a properly selected SF is significantly reduced compared to the case of the standard ISOMA. That is, in order to approach the performance of the MAP algorithm, only a small value of M is required by the ISOMA when a proper selected SF is employed. Therefore, more significantly, the decoding complexity of the considered systems with the ISOMA can be further reduced by the ISOMA with the proper selected SF.

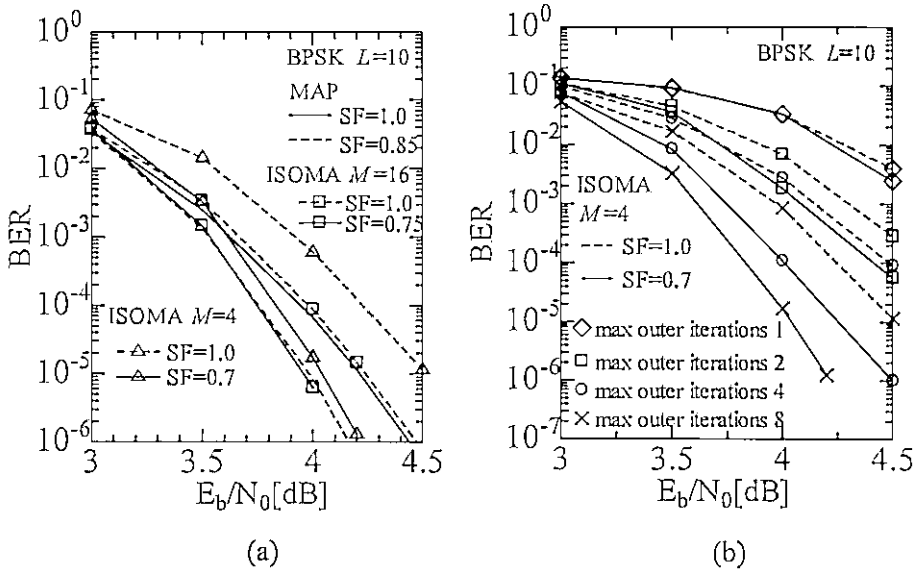

Figure 5.7: Performance comparison between the ISOMA with the best evaluated SF and the standard ISOMA.

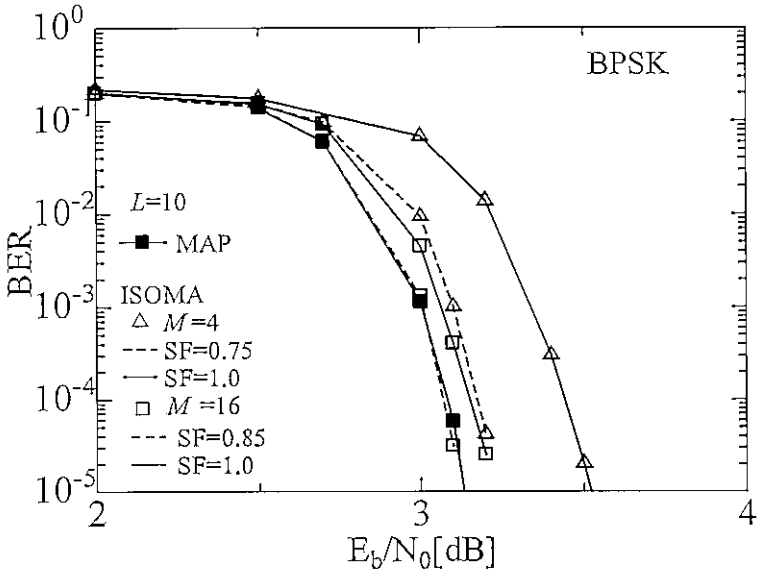

Figure 5.8: Performance comparison between the ISOMA with the best evaluated SF and the standard ISOMA for LDPC codes with length 10080.

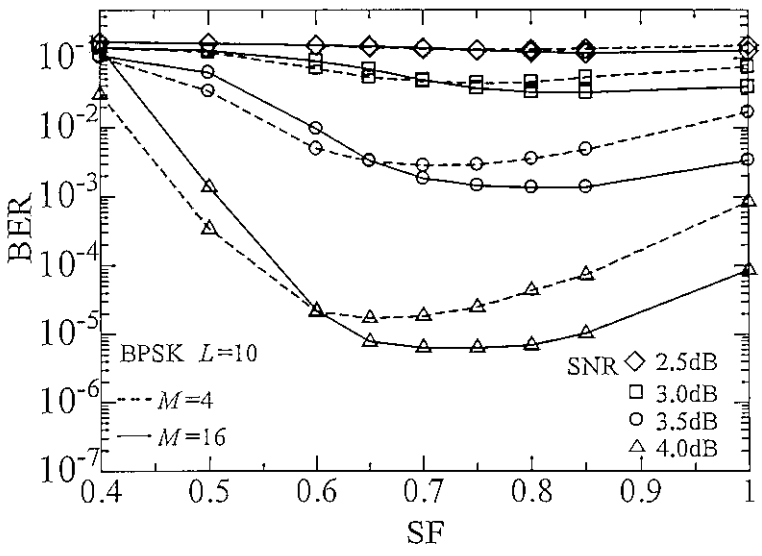

Figure 5.9: Impact of the value of SF on the performance of the ISOMA with the SF.

CHAPTER 6

Adaptive Iterative Decoding of Finite-Length Differentially Encoded LDPC Coded Systems with Iterative Multiple-Symbol Differential Detection

In Chapter 5, we have proposed an ISOMA scheme to greatly reduce the complexity of MSDD SISOD. However, in general, to achieve excellent performance, the iterative MSDD scheme requires the relatively large values of the OWS of MSDD SISOD and the outer iteration number between the MSDD SISOD and the outer decoder. In this case, even ISOMA is employed in MSDD SISOD, the complexity of MSDD SISOD is still relatively high, and a long decoding delay will be produced when the outer iteration number is also set to a large value. On the other hand, at low SNRs, very little performance improvement is achieved by increasing the outer iteration number and using a large OWS. Moreover, at high SNRs, most of the successful decodings can be achieved in the first few iterations with a small OWS. This means that a large outer iteration number and a large OWS are unnecessary in the two SNR regions. In general, due to the fact that the SNR is unknown at the receiver, in order to further reduce the complexity of the iterative decoding of the systems under consideration, it is necessary to propose an adaptive iterative decoding approach (AIDA) to adjust the OWS and the outer iteration number adaptively in a changing SNR environment for the coded systems using iterative MSDD. Moreover, Since LDPC codes are generally limited to finite length blocks in practical systems, we consider finite length DE-LDPC coded systems with iterative MSDD in this chapter.

The remainder of this chapter is organized as follows. Section 6.1 introduces the related works. In Section 6.2, the EXIT band chart analysis of the systems under consideration is described. In Section 6.3, we propose the AIDA scheme, and propose the new stopping criterion (SC) named DMI after discussing the existing SCs for the systems under consideration. Finally, performance analysis by computer simulation is given in Section 6.4.

6.1 Related Works

To adaptively adjust the OWS, an approach based on an analysis of the iterative decoding process using the EXIT chart technique [19] was proposed for "Turbo DPSK" systems [14]. In this approach, the OWS is gradually increased by using a look-up table, which is designed based on a designed iterative decoding trajectory obtained from the EXIT chart. Using this approach, the complexity of iterative MSDD can be reduced.

However, the design of the OWS table is based on assuming that the iterative decoding trajectory is well matched with the EXIT functions of the two component decoders of the iterative decoder, which is valid for using codes with infinite-length but not for using codes with finite-length [39]. Therefore, the approach of [14] is not suitable for using finite-length LDPC codes. Whereas in practical systems, LDPC codes are generally restricted to blocks of a few hundred to a few thousand code bits.

On the other hand, to adjust the outer iteration number adaptively, a SC, which is used to judge whether the iterative decoding should be terminated or not, is needed in AIDA. Up to now, many SCs have been proposed to terminate the iteration early to prevent unnecessary iterations of the iterative decoding, such as the cross entropy (CE) criterion [40], the sign-change-ratio (SCR) criterion [41], the sign difference ratio (SDR) criterion [42] and the mean-estimate (ME) criterion [43]. All these criteria were proposed for turbo coded systems and have been proven to be able to reduce the iteration number significantly with little performance loss by comparing a predefined threshold at high SNRs, where successful decodings are usually achieved in the first a few iterations. In [44], the above criteria were modified to stop the iteration at low and high SNRs for turbo coded systems by designing another stopping threshold for low SNRs based on the decoding threshold obtained from the EXIT chart.

Although so many SCs have been proposed for turbo coded systems, little attention has been paid to the iterative decoding of serially concatenated LDPC coded systems. In contrast to turbo codes, LDPC codes can detect successful decoding to stop the iteration by checking the parity check constraints of LDPC codes. Therefore, serially concatenated LDPC coded systems are more concerned with the performance of the SC for uncorrectable decodings. In order to satisfy this requirement, stopping rules and thresholds of these existing SCs should be redefined. However, the analysis results of Sections 6.3.3 and 6.3.4 will show that the redefined stopping rules and thresholds are only suitable for a certain SNR region, which causes significant performance loss at other SNRs. It is also shown that they need to be redefined when the LDPC code and transmission channel parameters change.

6.2 EXIT Band Chart Analysis of the Finite-Length DE-LDPC Coded Systems with Iterative MSDD

The system model under consideration is shown in Fig. 4.1. The convergence behavior of the iterative decoding of serially concatenated systems can be visualized and predicted by EXIT chart analysis [19]. However, it is known that the analysis results of the EXIT chart are accurate for infinite-length codes, but not for finite-length codes. Whereas in practical systems, LDPC codes are generally limited to short or medium length with a few hundred or thousand bits. In [39], an EXIT band chart, which is a convergence analysis approach using an EXIT curve band instead of a single EXIT curve as in EXIT chart, was proposed for finite-length turbo decoding. In this section, we extend this approach to finite-length DE-LDPC coded systems with iterative MSDD.

In the EXIT band chart, the transfer characteristics of component decoders are characterized by their EXIT functions for each random channel realization. Let [s] denote the seed of the channel realization, and let $I_{L_{M,e}^{[s]}}$ and $I_{L_{M,a}^{[s]}}$ ($I_{L_{D,e}^{[s]}}$ and $I_{L_{D,a}^{[s]}}$) be mutual informations between the transmitted coded bits and the LLRs at the output and the input of MSDD SISOD (LDPC decoder, respectively). For a given [s], the EXIT function of the MSDD SISOD over AWGN channels is defined as

$$I_{L_{M,e}^{[s]}} = T_1^{[s]}(I_{L_{M,a}^{[s]}}, \frac{E_b}{N_0}), \tag{6.1}$$

where $\frac{E_b}{N_0}$ is the SNR of the channel. Similarly, for a given [s], the EXIT function of the LDPC decoder is defined as

$$I_{L_{D,e}^{[s]}} = T_2^{[s]}(I_{L_{D,a}^{[s]}}). \tag{6.2}$$

The mutual information between the transmitted coded bits C and the corresponding LLR values L is calculated as [19]

$$I_L = I(L;C) = \frac{1}{2}\sum_{c=0,1}\int_{-\infty}^{\infty} p_L(l|c) \cdot \log_2 \frac{2p_L(l|c)}{p_L(l|c=0)+p_L(l|c=1)} dl, \tag{6.3}$$

where $p_L(l|c)$ is the conditional PDF of the LLR values L given $c \in \{0,1\}$, and $0 \leqslant I_L \leqslant 1$. Note that we drop the subscript of L in (6.3) to generally represent the LLRs in (6.1) and (6.2). To generate the EXIT band chart, the PDFs of the LLRs corresponding to $I_{L_{M,a}^{[s]}}$ and $I_{L_{D,a}^{[s]}}$ are assumed to be Gaussian distributed. When $I_{L_{M,e}^{[s]}}$ and $I_{L_{D,e}^{[s]}}$ are calculated, the PDFs of $L_{M,e}^{[s]}$ and $L_{D,e}^{[s]}$ are obtained by the histogram method [19].

To obtain the EXIT transfer characteristic of the MSDD SISOD, we repeatedly perform open loop simulations by changing the channel realization seed [s] and the mutual information of the input *priori* information $I_{L_{M,a}^{[s]}}$ at the same SNR. For the considered systems with finite-length LDPC codes, we obtain various values of $I_{L_{M,e}^{[s]}}$ for different channel realizations even with the same values of $I_{L_{M,a}^{[s]}}$ and SNR. Hence, in contrast to the considered systems with infinite-length LDPC codes, whose EXIT transfer characteristic of MSDD SISOD is a single curve, for the considered systems with finite-length LDPC codes, the EXIT transfer characteristic of the MSDD SISOD is a band of curves of $I_{L_{M,e}^{[s]}}$ with respect to $I_{L_{M,a}^{[s]}}$. Similarly, the transfer characteristic of the LDPC decoder for finite-length LDPC codes is also a band of curves of $I_{L_{D,e}^{[s]}}$ with respect to $I_{L_{D,a}^{[s]}}$. Similar to the analysis of [39], the EXIT band of the MSDD SISOD (LDPC decoder) can be represented using the average curves $\text{avg}(I_{L_{M,e}^{[s]}})$ ($\text{avg}(I_{L_{D,e}^{[s]}})$), and the upper and lower bound $\text{avg}(I_{L_{M,e}^{[s]}}) \pm \text{std}(I_{L_{M,e}^{[s]}})$ ($\text{avg}(I_{L_{D,e}^{[s]}}) \pm \text{std}(I_{L_{D,e}^{[s]}})$, respectively), where $\text{avg}(\cdot)$ and $\text{std}(\cdot)$ represent the average and the standard deviation, respectively. The EXIT band chart can be obtained by plotting the EXIT bands of the MSDD SISOD and the LDPC decoder into a signal diagram by switching the x-axis and the y-axis.

Fig. 6.1 shows the EXIT band charts of the considered systems with rate-1/2 (3, 6) regular LDPC codes with different finite code lengths over AWGN channels with BPSK,

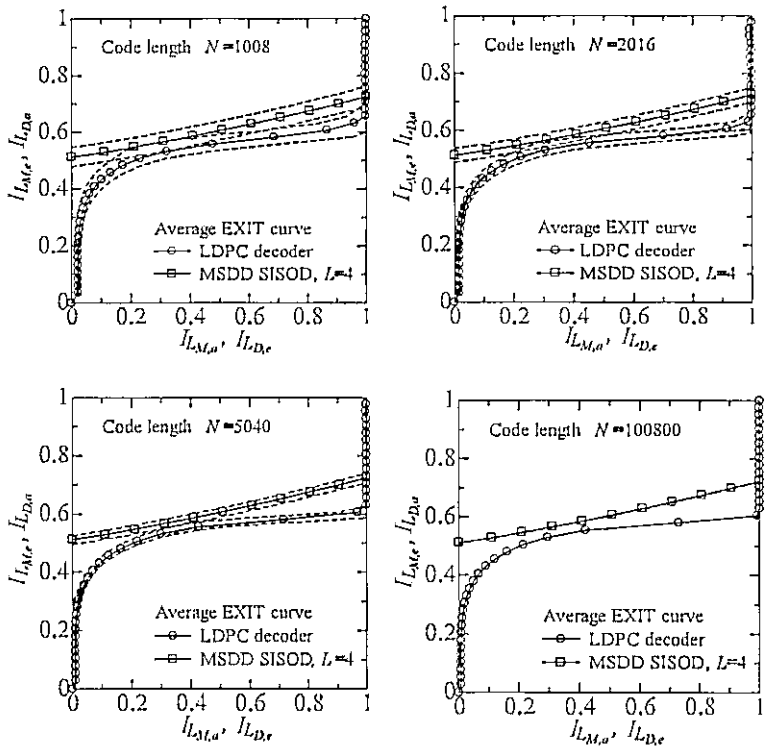

Figure 6.1: EXIT band charts of the considered systems for MSDD SISOD with L = 4 and rate-1/2 (3, 6) regular LDPC codes with different code lengths over AWGN channels with BPSK at SNR = 3.5dB; inner iteration number of the LDPC decoder is 10; repeated 10000 frames.

where the number of inner iterations of the LDPC decoder is 10. For comparison, the EXIT band chart for a very long code length of 100800 (can be viewed as infinite-length) is also presented. We can observe that the shorter the code length is, the wider the EXIT bands become. When the code length is increased to 100800, the widths of the EXIT bands become zero. That is, the EXIT band chart of infinite-length codes is actually equivalent to the ordinary EXIT chart. For infinite-length LDPC codes, it is known that each frame has similar asymptotic performance when the SNR is larger than the asymptotic decoding threshold; thus the uncorrectable frames whose performance can only be improved marginally by increasing the iteration number appear only at low SNRs (those SNRs smaller than the asymptotic decoding threshold). Whereas for the same LDPC code ensemble with finite-length, due to the overlap of the EXIT curve bands, uncorrectable frames still exist at certain SNRs which are above the asymptotic decoding threshold. This means that uncorrectable frames appear at low SNRs and also at medium SNRs. This conclusion can be further supported by Fig. 6.2, which shows three typical simulated snapshot iterative decoding trajectories for the considered systems with rate-1/2 (3, 6) regular LDPC codes of length 1008 and $L = 4$ at SNR = 3.8dB. It is shown that these iterative decoding trajectories exhibit a greater

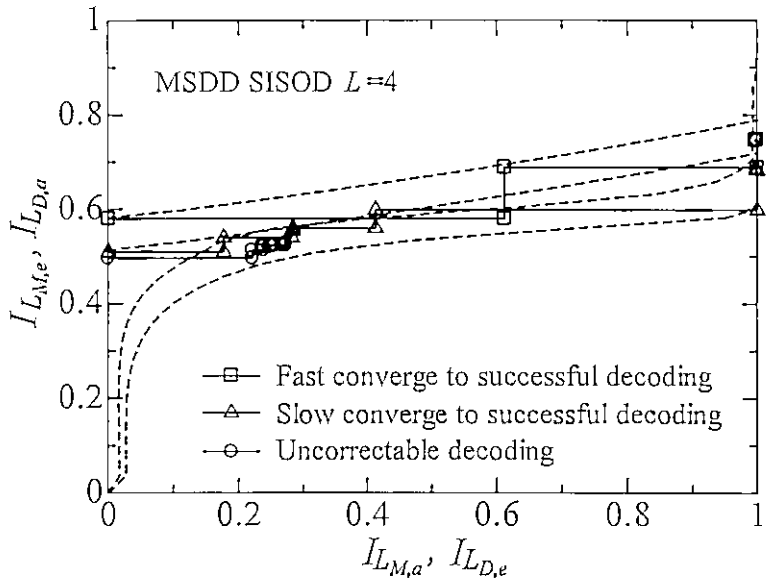

Figure 6.2: Typical simulated snapshot iterative decoding trajectories of the considered systems for MSDD SISOD with $L = 4$ and rate-1/2 (3, 6) regular LDPC codes with length 1008 over AWGN channels with BPSK at SNR = 3.8dB; inner iteration number of the LDPC decoder is 10.

variation from frame to frame when the code length is short. Some frames quickly or slowly converge to successful decoding, while some frames cannot converge to successful decoding.

Next, we analyze the impact of the OWS of the MSDD SISOD on system performance. Fig. 6.3 shows the average EXIT curves of the LDPC decoder and the MSDD SISOD with different OWS at SNR = 3.5dB. It is shown that the slopes of the EXIT curves of the MSDD SISOD increase with an increase in L, which implies that the performance of the systems can be improved by increasing L. However, at low SNRs, e.g. 1.0 dB, it can be expected that the EXIT band of the MSDD SISOD is almost entirely under the EXIT band of the LDPC decoder even with large values of L, which means that increasing L at low SNRs cannot improve the system performance much. On the other hand, at high SNRs, e.g. 5.0 dB, it can be expected that a large tunnel is opened between the EXIT bands of the MSDD SISOD and the LDPC decoder even for $L = 2$, which means that most of the frames can be decoded successfully using a small value of L. While at medium SNRs, as shown in Fig. 6.1 for $N = 1008$ at SNR = 3.5dB, it can be expected that the fraction of the overlap between the two EXIT bands will decrease with an increase in L, and a tunnel will be opened when $L \geqslant 6$. This means that the probability of successful decoding can be increased, and thus a significant improvement in average BER performance can be achieved by increasing L at medium SNRs.

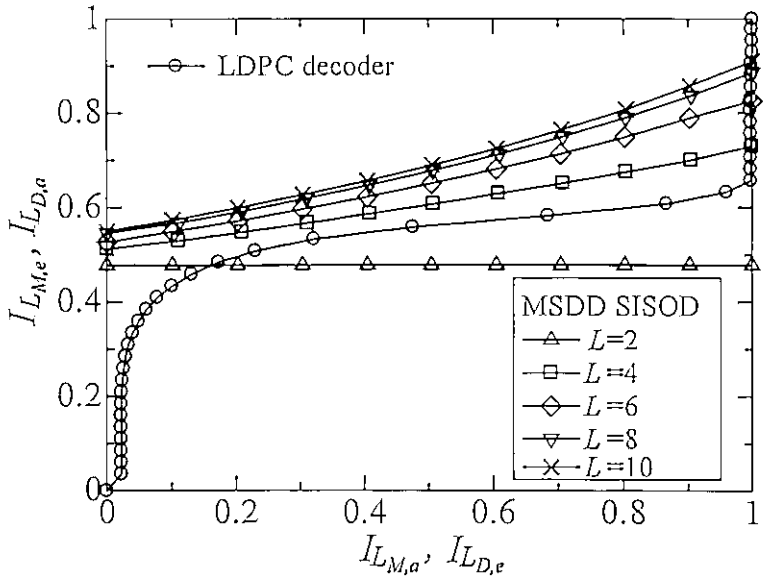

Figure 6.3: Average EXIT curves of the considered systems for MSDD SISOD with different L; rate-1/2 (3, 6) regular LDPC codes with length 1008 over AWGN channels with BPSK at SNR = 3.5dB; inner iteration number of the LDPC decoder is 10; averaged over 10000 frames.

6.3 AIDA

6.3.1 Motivation of AIDA

The OWS of the MSDD SISOD and the outer iteration number are the two key parameters that determine the iterative decoding complexity and delay of the considered systems. It is known that the complexity of MSDD grows exponentially with the OWS. Therefore, the complexity of the MSDD SISOD will become prohibitively high as the OWS becomes large. Especially for iterative decoding systems with large outer iteration numbers, this high complexity will result in an unacceptable decoding delay, which makes it difficult to achieve a realistic system.

In [14], an approach which gradually increases the OWS in accordance with the iteration number of turbo coded systems with MSDD by looking up an OWS table, was proposed based on the analysis of the iterative decoding process using the EXIT chart. It is proved that this approach can reduce the complexity of the iterative MSDD decoding with negligible performance loss compared with the OWS with a fixed large value [14]-[16]. In this approach, the OWS table and the maximum outer iteration number are designed based on a designed iterative decoding trajectory obtained from the EXIT chart. Since the two parameters are designed for a target BER at the expected SNR, the decoding complexity and delay cannot be significantly reduced at all SNRs by using this approach. Moreover, the design of the two parameters is based on the assumption that the iterative decoding trajectory is well matched to the EXIT functions

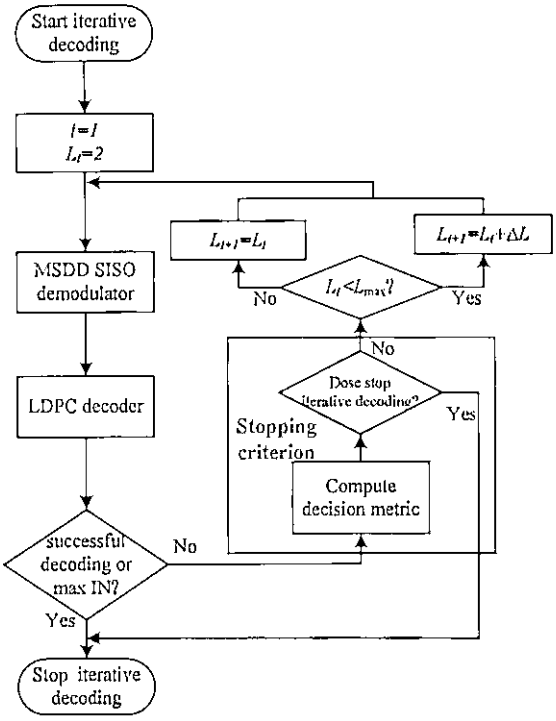

Figure 6.4: Structure of AIDA, where IN is the acronym of iteration number.

of the two component decoders of the iterative decoder, which is valid for codes with very long length, but not for codes with short length.

Based on the analysis results of Section 6.2, a small OWS and a small outer iteration number are sufficient for low SNRs and high SNRs. Furthermore, for medium SNRs, the performance improvement achieved by increasing the OWS and the outer iteration number varies from frame to frame when the code length is finite. Therefore, it is necessary to propose an adaptive iterative decoding approach (AIDA) to adjust the OWS and the outer iteration number adaptively to reduce the iterative decoding complexity and delay of the considered systems with finite-length LDPC codes in a changing SNR environment.

6.3.2 Principle of AIDA

In the proposed AIDA, in contrast to adjusting the OWS of the MSDD SISOD and the outer iteration number according to the predesigned OWS table and the maximum outer iteration number as in [14], the two parameters are adjusted according to the convergence status of the iterative decoding. Fig. 6.4 shows the structure of AIDA. Let L_i and L_{max} denote the OWS at the ith iteration and the predefined maximum OWS of the MSDD SISOD, respectively. In the first iteration, L_1 is set to 2. If the decoding of the LDPC decoder is successful, the iterative decoding will be stopped

automatically; otherwise, the SC will be checked. If the evaluation of the SC indicates that the requirement of stopping iteration is satisfied, the iterative decoding will be stopped; otherwise, the iterative decoding will continue to be executed, and the OWS will be increased by a predefined fixed value ΔL in the next iteration if $L_i < L_{max}$. This process is repeated until the requirement of stopping iteration is satisfied, or until the predefined maximum iteration number is reached or decoding is successful. In the SC, the requirement of stopping iteration is an indicator which can reflect the convergence of the iterative decoding.

The SC is the key part in AIDA. In order for AIDA to be able to reduce the iterative decoding complexity and delay with very little performance loss, a well designed SC should be able to timely and accurately judge the convergence of the iterative decoding. Up to now, many SCs have been proposed to reduce the iteration number for turbo coded systems. In the following, some representative existing SCs are briefly introduced, and these criteria for the considered systems are studied in Section 6.3.3. Then, to circumvent the disadvantages of these criteria for the considered systems, a new SC is proposed in Section 6.3.4.

6.3.3 Existing SCs for the Considered Systems

Let $L_{e1}(c_k)$ and $L_{e2}(c_k)$ denote the extrinsic LLRs of the kth code bit of the two component decoders of the turbo decoder, respectively, and let N denote the length of turbo codes. The existing SCs are introduced as follows.

1) CE criterion: CE is used to measure the closeness of two distributions. At the ith iteration, let $p_1^i(c_k)$ and $p_2^i(c_k)$ denote the *a posteriori* probability distributions of the outputs of the two component decoders of the turbo decoder, respectively. The CE between $p_1^i(c_k)$ and $p_2^i(c_k)$ can be expressed as

$$T(i) = E\left\{\log\frac{p_2^i(c_k)}{p_1^i(c_k)}\right\} \approx \sum_{k=1}^{N}\frac{\left|L_{e2}^i(c_k) - L_{e2}^{i-1}(c_k)\right|^2}{\exp(L_{e1}^i(c_k))}. \tag{6.4}$$

With an increase of the iteration number, the CE becomes smaller and smaller. When $T(i)$ is smaller than a predefined threshold $(10^{-2} \sim 10^{-4})T(1)$, the iterative decoding is terminated.

2) SCR criterion: SCR is based on measuring the sign changes $C(i)$ of $L_{e2}(c_k)$ from iteration (i-1) to iteration i. The principle of SCR is to compute $C(i)$ at each iteration, and the iterative decoding is terminated when $C(i)$ is smaller than the threshold $(0.005 \sim 0.03)N$.

3) SDR criterion: Let $D(i)$ denote the number of sign differences between $L_{e1}(c_k)$ and $L_{e2}(c_k)$ in the same iteration. The principle of SDR is to compute $D(i)$ after each iteration and to terminate the iterative decoding when $D(i)$ is smaller than the threshold $(0.001 \sim 0.01)N$.

4) ME criterion: This approach is based on monitoring $M_{|L|}$, the mean of absolute LLR values of the second decoder over a block after each iteration. Simulation shows

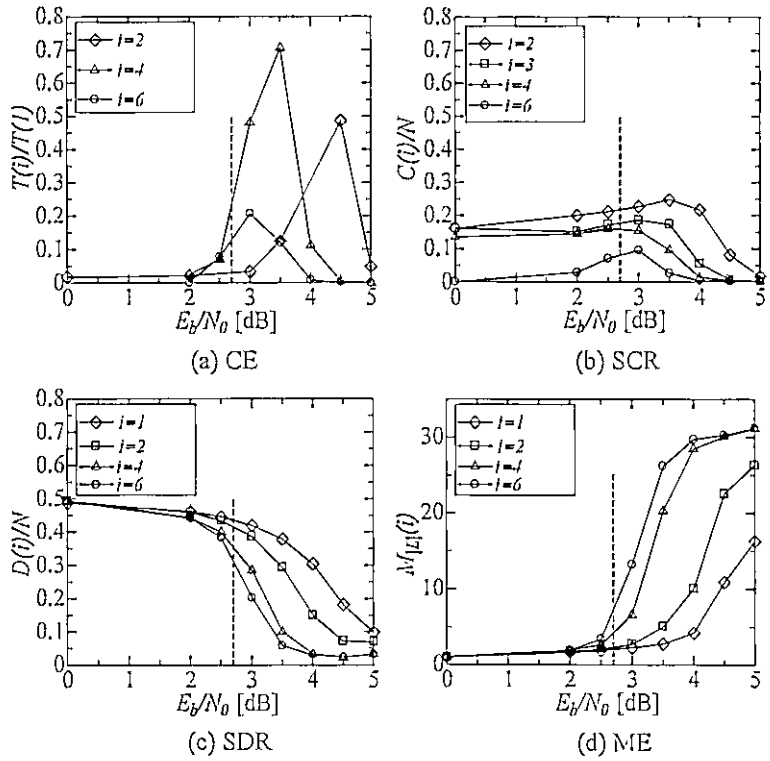

Figure 6.5: Average evaluations of the decision metrics of the existing SCs variation with SNR and outer iteration number; rate-1/2 (3, 6) regular LDPC codes with length 1008 over AWGN channels with BPSK; OWS of MSDD SISOD corresponding to the outer iteration number from 1 to 6 is [2,4,6,8,8,8]; inner iteration number of the LDPC decoder is 10; averaged over 10000 frames.

that $M_{|L|}$ increases as the number of errors decrease. Therefore, the iterative decoding is terminated when $M_{|L|}$ is bigger than a predifined threshold.

Since the iterative decoder of the considered systems, which consists of a demodulator and a LDPC decoder, is different from the turbo decoder, the assumptions which are used to obtain the approximation of CE as expressed in (6.4) [41], do not hold for the iterative decoder of the considered systems. Thus, the CE criterion is modified to be used only for the LDPC decoder, and the decision metric of the CE criterion for the considered systems is turned into

$$T(i) \approx \sum_{k=1}^{N} \frac{\left|L_{D,p}^{i}(c_k) - L_{D,p}^{i-1}(c_k)\right|^2}{\exp(L_{D,p}^{i-1}(c_k))}. \tag{6.5}$$

That is, the CE for the considered systems is used to measure the closeness of the *a posteriori* probability outputted from the LDPC decoder in two consecutive outer iterations. For SCR, SDR and ME criteria, these SCs can be used in the considered systems directly. These SCs have been proved to be the efficient approaches to detect

the successful decoding before reaching the maximum iteration number by comparing a predefined threshold. However, compared with turbo codes, LDPC codes can detect successful decoding to stop the iteration by checking the parity check constraints of LDPC codes. Thus, the SC for the considered systems is more concerned with stopping the uncorrectable decodings which usually occur at low to medium SNRs for our considered systems. In this situation, the stopping rules and the thresholds of the SCs mentioned above should be redesigned to be applicable for this purpose.

To understand how to design the stopping rules and the thresholds of these SCs for the considered systems, the average evaluations of the decision metrics of these SCs variation with SNR and outer iteration number are shown in Fig. 6.5. We can observe that the stopping rules of the CE, SCR and ME criteria for our target are that the evaluation of their decision metrics should be smaller than a threshold. On the contrary, the stopping rule of the SDR criterion is that the evaluation of its decision metric should be bigger than a threshold. However, we can observe that they do not work well at all SNRs based on the redesigned stopping rules. For CE and SCR criteria, the iteration may be stopped prematurely at medium and high SNRs, which results in performance losses at these SNRs, since the evaluations of their decision metrics vary non-monotonically with SNR as shown in Fig. 6.5(a) and 6.5(b). On the other hand, for SDR and ME criteria, it is possible to design proper thresholds for them at low SNRs based on Fig. 6.5(c) and 6.5(d). The low SNR region can be determined with the aid of the asymptotic decoding threshold predicated by the EXIT chart for the considered systems. In this SNR region, the iterative decoding of each frame should be stopped after the first or second iteration also for finite-length LDPC codes. However, it is difficult to design proper thresholds of them also valid for medium SNRs, since the iterative decoding characteristics greatly vary from frame to frame for the considered systems at medium SNRs as discussed in Section 6.2. Moreover, the low SNR region where is no need for increasing iteration number is changed with a change in the LDPC code and transmission channel parameters, which makes the defined thresholds of these SCs need to be redesigned again.

6.3.4 Proposed SC

The basic reason of the disadvantages of these SCs is that these approaches are not good methods to track the convergence status of the iterative decoding. From the decoding trajectories of Fig. 6.2, we can observe that the value of $I_{L_{D,e}}$ increases significantly between two consecutive outer iterations if the iterative decoding can improve the system performance effectively, whereas the value of $I_{L_{D,e}}$ remains almost unchanged as the number of iterations increases when the iterative decoding has converged. Thus, the change of $I_{L_{D,e}}$ between two consecutive outer iterations can reflect the convergence status of the iterative decoding. These observations motivate us to propose a new SC named DMI criterion, which is based on tracking the difference of the output mutual information of the LDPC decoder between two consecutive outer iterations. Based on this idea, at the ith iteration, the decision metric of the DMI criterion can be written

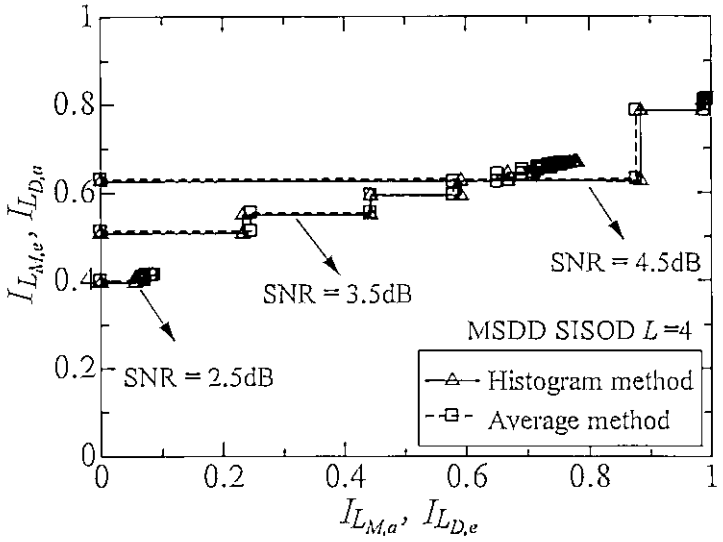

Figure 6.6: Averaged simulated iterative decoding trajectories of the considered systems evaluated by different approaches; MSDD SISOD with $L = 4$ and rate-1/2 (3, 6) regular LDPC codes with length 1008 over AWGN channels with BPSK at different SNRs; inner iteration number of the LDPC decoder is 10; averaged over 10000 frames.

as

$$\Delta I^i_{L_{D,e}} = I^i_{L_{D,e}} - I^{i-1}_{L_{D,e}}. \tag{6.6}$$

If $\Delta I_{L_{D,e}}$ is smaller than a threshold Th, which means that the iterative decoding has converged, the iterative decoding is stopped; otherwise, the next iteration is executed.

The computation of $I_{L_{D,e}}$ using (6.3), also called the histogram method, needs the information about the transmitted coded bits and the PDF of LLR values, which is impractical for realistic systems since the transmitted coded bits are unknown at the receiver. Actually, (6.3) can be computed by the average method shown below without these requirements [27]

$$I_L \approx 1 - \frac{1}{N}\sum_{n=1}^{N} H_b\left(\frac{e^{-|l_n|/2}}{e^{|l_n|/2} + e^{-|l_n|/2}}\right), \tag{6.7}$$

where N is the code sequence length, l_n is the LLR value of the nth bit of the code sequence, and H_b is the binary entropy function

$$H_b(p) = -p\log_2 p - (1-p)\log_2(1-p), 0 \leqslant p \leqslant 1. \tag{6.8}$$

Equation (6.7) is valid when the APP decoder is employed even if the distribution of the LLR outputs of the decoder is non-Gaussian or unknown distributions [27]. Fig. 6.6 shows the comparison of the averaged simulated iterative decoding trajectories of the considered systems at different SNRs obtained by the histogram method and the average method, respectively. It is shown that the average method is a good approximation for

the histogram method. Therefore, it is reasonable to use (6.7) to compute $I_{L_{D,e}}$ after each outer iteration for the considered systems where the LDPC decoder uses the sum-product algorithm.

We must point out that the idea of SC based on (6.6) has been proposed in [45] for turbo HARQ schemes, which is used to reduce the complexity while maintaining a high throughput and a low packet-loss-ratio. However, since the purpose of our work is different from that of [45], the SC based only on (6.6) is not the most effective approach for our considered systems, which will be explained below and be supported in Section 6.4.2. Based on this fact, the principle of our proposed DMI criterion, which is also the difference between our proposed DMI criterion and the SC of [45], is explained from the following two aspects.

On one hand, it is obvious that using the SC with only the decision metric (6.6), the system will execute at least two iterations before successful decoding. However, in fact, in most cases, iterative decoding only needs to be executed once at low SNRs. From Fig. 6.1 and 6.3, we can also observe that the EXIT curves of the MSDD SISOD and the LDPC decoder intersect on the very left side of the EXIT band chart at low SNRs corresponding to very small value of $I_{L_{D,e}}$. Thus, to further reduce unnecessary iterations at low SNRs, before evaluating the decision metric of the DMI criterion, $I^i_{L_{D,e}}$ is compared with another threshold Th_L to decide whether to stop the iterative decoding immediately.

On the other hand, as shown in Fig. 6.2, at medium SNRs, the iterative decoding of some frames corresponding to the typical iterative decoding behavior of slow convergence has the following features: $I^i_{L_{D,e}}$ increases to a relatively large value and seems to be unchanged (that is, $\Delta I^i_{L_{D,e}}$ very small) after the first few iterations, but $I^i_{L_{D,e}}$ can increase to the value of the right side of the EXIT band chart after some iterations, which results in successful decoding at last. Owing to this fact, if the SC only considers (6.6), this type of decoding behavior will be prematurely stopped, resulting in a performance loss. Therefore, the proposed SC should consider how to avoid prematurely stopping this type of decoding behavior. We solve this problem as follows. We define a threshold Th_M which is bigger than the threshold Th_L. At the ith iteration, if $\Delta I^i_{L_{D,e}} < Th$ but $I^i_{L_{D,e}} > Th_M$, which means that the current iterative decoding may be the type of the slow convergence, the iterative decoding is not stopped immediately and let an indicator $SN = SN + 1$. Here SN is used to represent the number of times of $\Delta I^i_{L_{D,e}} < Th$ when $I^i_{L_{D,e}} > Th_M$ during the current iterations. If SN is bigger than a defined number of times Th_N, which means that the current iterative decoding is an uncorrectable decoding with high probability, then decoding is stopped; otherwise, the current iterative decoding may be the type of the slow convergence decoding, thus go to the next iteration.

The proposed DMI criterion is summarized as follows:

At the ith outer iterative decoding($SN = 0$):

Step 1: Compute $I^i_{L_{D,e}}$ using equation (6.7) at the ith iteration.

Step 2: If $I^i_{L_{D,e}} < Th_L$, stop the iterative decoding; otherwise, go to step 3.

Step 3: If $I^i_{L_{D,e}} < Th_M$ and $\Delta I^i_{L_{D,e}} < Th$, stop the iterative decoding; otherwise, go to step 4.

Step 4: If $I^i_{L_{D,e}} > Th_M$ and $\Delta I^i_{L_{D,e}} < Th$, let $SN = SN + 1$. If $SN > Th_N$, stop the iterative decoding; otherwise, execute the $(i+1)$th iteration.

Actually, we can find that the SC of [45] is equivalent to the proposed DMI criterion with $Th_L = 0.0$ and $Th_M = 1.0$. It should be emphasized that the performance of AIDA with the DMI criterion depends on the values of the thresholds Th, Th_L, Th_M and Th_N. The EXIT band chart of the system can simplify the process of choosing of these thresholds. Although, generally speaking, the best choice of these thresholds should be optimized for particular system parameters, the performance of the DMI criterion with a set of determined thresholds is robust when the LDPC code and transmission channel environment parameters are changed, which will be supported by the simulation results presented in the next section.

6.4 Simulation Results and Analysis

The performances of the proposed AIDA with the existing SCs and the proposed SC for the considered systems are evaluated and analyzed using computer simulations. Unless otherwise indicated, the following simulation parameters are used for our simulations. We consider rate-1/2 (3, 6) regular LDPC codes with length 1008 over AWGN channels. The coded bits are modulated using BPSK for simplicity. The maximum number of outer iterations between the MSDD SISOD and the LDPC decoder is set to 6. The maximum number of inner iterations of the LDPC decoder is set to 10. In AIDA, L_{max} and ΔL are set to 8 and 2, respectively.

6.4.1 Selection of the DMI Criterion Thresholds

Based on the EXIT band chart analysis of the considered systems with the above simulation parameters, which is similar to the analysis of Fig. 6.1-6.3 and 6.6, the approximate ranges of the thresholds Th, Th_L, Th_M and Th_N can be first determined as follows: $Th < 0.03$, $Th_L < 0.12$, $0.3 < Th_M < 0.7$ and $Th_N < 3$. Then, we can select appropriate values for them in their corresponding ranges. More specifically, we first evaluate the variation of the average number of outer iterations and the BER performance of the considered systems using AIDA with the DMI criterion with different values of threshold Th as shown in Fig. 6.7. Th_L and Th_M are set to 0.0, 1.0, respectively. In this situation, the performance of the DMI criterion is determined only by the threshold Th. It is shown that the smaller Th is, the smaller performance loss is, but the larger number of outer iterations is required. In order to make a good trade-off between the iterative decoding complexity and the performance loss, the value of Th that is selected is 0.02. After determining the value of Th, we can then determine the value of Th_L, and then determine the values of Th_M and Th_N together at last using simulations similar to the process of the selection of Th. For the sake of conciseness, the specific processes of the selection of Th_L, Th_M and Th_N are not presented in detail here. Finally, the

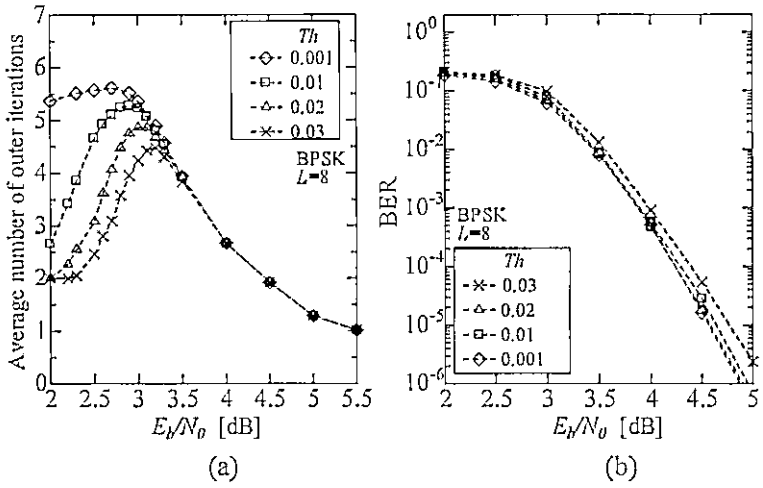

Figure 6.7: Performance of AIDA using the DMI criterion variation with the value of threshold Th; $Th_L = 0.0$ and $Th_M = 1.0$; rate-1/2 (3, 6) regular LDPC codes with length 1008 over AWGN channels with BPSK. (a) Average number of outer iterations. (b) BER performance.

thresholds Th, Th_L, Th_M and Th_N of the proposed DMI criterion are set to 0.02, 0.07, 0.5 and 1, respectively, for the following simulations.

6.4.2 Performance of AIDA with the DMI Criterion

Fig. 6.8(a) shows the average number of outer iterations of the considered systems using AIDA without SC and with SC of the proposed DMI criterion. Here, AIDA without SC is that the early stopping approach is not used in AIDA. In case of AIDA without SC, the OWS of MSDD SISO corresponding to the outer iteration number from 1 to 6 is [2,4,6,8,8,8], and the outer iterations are stopped when the maximum outer iteration number is reached or a legitimate codeword is found. The simulation result of the considered systems with fixed $L = 8$ without AIDA is also presented for comparison. Since the LDPC decoder can detect successful decoding and stop the iteration by checking the parity check constraints of LDPC codes, it is shown that the average number of outer iterations required by AIDA without SC gradually decreases with an increase in SNR. When the DMI criterion is used in AIDA, since the DMI criterion adjusts the number of outer iterations based on the convergence status of the outer iterations, we observe that the average number of outer iterations can be reduced efficiently by AIDA with the DMI criterion at low and medium SNRs. More specifically, the outer iterations are stopped as early as possible at low SNRs, where the performance cannot be improved by increasing the OWS and the number of outer iterations. At medium SNRs, where the performance improvement can be gradually achieved by increasing the OWS and the number of outer iterations, the average iteration number is gradually increased. While at high SNRs, the average numbers of outer iterations required by AIDA with

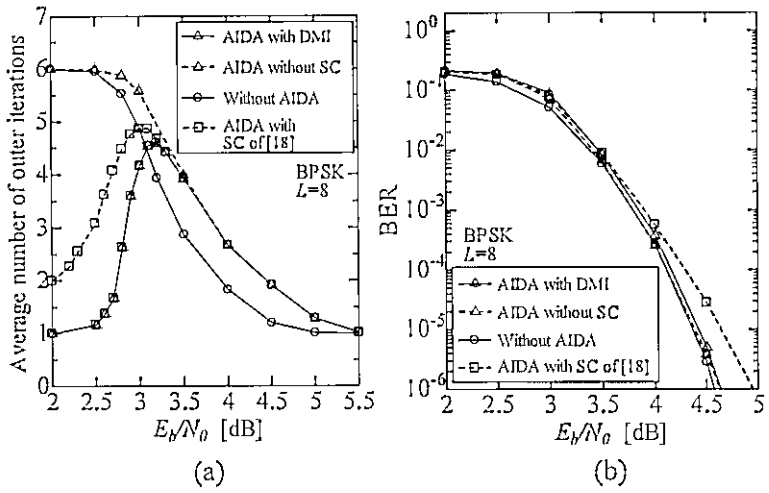

Figure 6.8: Performance of the considered DE-LDPC coded systems with and without AIDA; rate-1/2 (3, 6) regular LDPC codes with length 1008 over AWGN channels with BPSK. (a) Average number of outer iterations. (b) BER performance.

and without the DMI criterion are the same, since almost all of the frames can be successfully decoded in the first few iterations.

Fig. 6.8 also shows the performance comparison between AIDA with the DMI criterion and AIDA with the SC of [45] that is equivalent to setting $Th = 0.02$, $Th_L = 0.0$ and $Th_M = 1.0$ for the DMI criterion. We can observe that for the systems considered here, the proposed DMI criterion performs better than the SC of [45] in terms of the required average number of outer iterations at low SNRs and the performance loss at medium and high SNRs, which supports the discussion of the difference between the proposed DMI criterion and the SC of [45] in Section 6.3.4.

Next, we analyze the reduction of the iterative decoding complexity by using AIDA. The evaluation of the iterative decoding complexity needs to consider the number of outer iterations and the complexity of the MSDD SISOD and the LDPC decoder. Since the LDPC decoder can detect successful decoding and automatically stop the iteration, we evaluate the iterative decoding complexity by mainly considering the average multiplication number (AMN) and the average addition number (AAN) of the iterative decoding of each frame for the considered systems with and without AIDA as shown in Fig. 6.9. The method introduced in [46] is used to evaluate the decoding complexity of the LDPC decoder using the sum-product algorithm. It should be noted that since the number of divisions is far less than the number of multiplications in each iteration, it is not considered in the evaluation of the iterative decoding complexity. In addition, the calculation of the logarithmic term in (4.3) is also ignored.

Although from Fig. 6.8 it can be seen that the average numbers of outer iterations from 2.7dB to 5.25dB required by AIDA without SC are about 0.2 to 0.5 times more than that required without using AIDA for the considered systems, Fig. 6.9 shows that

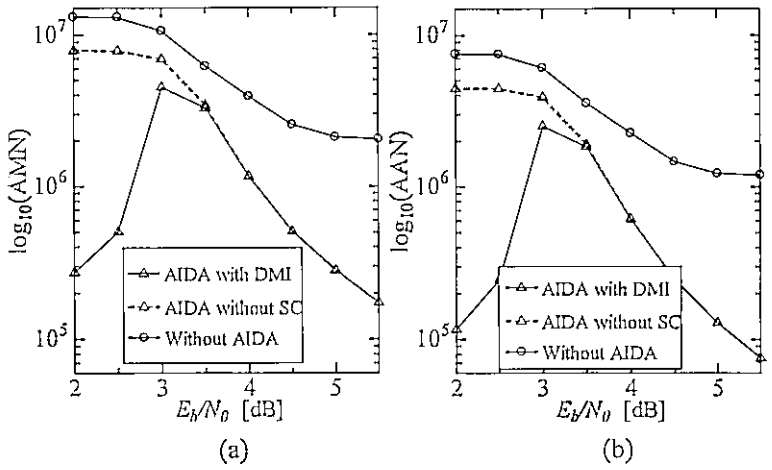

Figure 6.9: Computational complexity of the iterative decoding of each frame of the considered DE-LDPC coded systems with and without AIDA; rate-1/2 (3, 6) regular LDPC codes with length 1008 over AWGN channels with BPSK. (a) AMN. (b) AAN.

both the AMN and AAN are significantly reduced by AIDA without the SC at all SNRs. For example, for SNR = 4.5 dB, about 20% of the complexity of the iterative decoding without AIDA is required by using AIDA. This is because the fraction of successful decoding achieved by the MSDD SISOD with small L increases with an increase in SNR values as shown in Fig. 6.10. For example, at SNR = 5.5 dB, we can observe that about 98% of successful decodings can be achieved by $L = 2$, while only 2% of successful decodings require $L > 2$. From Fig. 6.9, we also observe that AIDA using the DMI criterion can further reduce the decoding complexity at low and medium SNRs. For example, only about 3% of the complexity of the iterative decoding without AIDA is required by using AIDA with the DMI criterion at SNR = 2 dB. Combining the above observations and the results of the BER performance comparison of the considered systems with and without AIDA as shown in Fig. 6.8(b), we can conclude that the proposed AIDA with the DMI criterion can significantly reduce the iterative decoding complexity and delay with negligible performance loss.

6.4.3 Performance of AIDA with Different SCs

In the following, performances of AIDA with different SCs are compared and analyzed. The principle of the selection of the thresholds of the aforementioned existing SCs, which is the same as that of [44], is that the outer iteration should be stopped immediately when the SNR is below the asymptotic decoding threshold of the considered systems. It should be noted that at least two outer iterations are performed before successful decoding when CE and SCR are used. Based on the discussion of Section 6.3.3, according to Fig. 6.5 and the asymptotic decoding threshold of the considered systems with $L = 8$ and the rate-1/2 (3, 6) regular LDPC code ensemble over AWGN channels

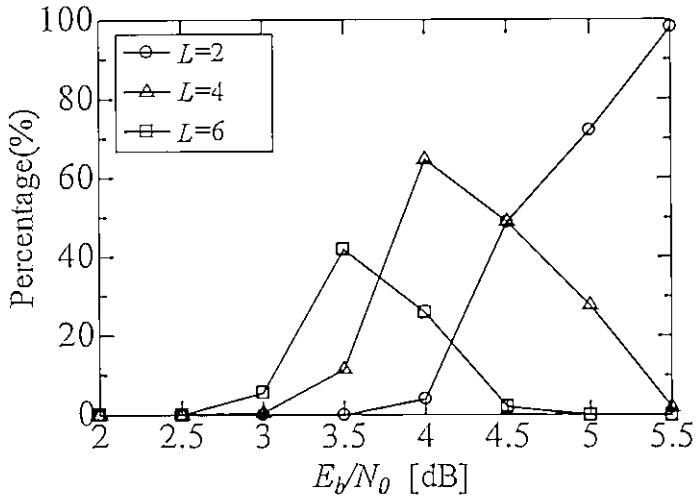

Figure 6.10: Percentage of successful decodings of the considered DE-LDPC coded systems with $L < 8$ when AIDA is used; stopping criterion is not used, iterative decoding is stopped by LDPC decoder; OWS of MSDD SISOD corresponding to the outer iteration number from 1 to 6 is [2,4,6,8,8,8]; rate-1/2 (3, 6) regular LDPC codes with length 1008 over AWGN channels with BPSK; count over 10^6 frames.

with BPSK as shown by the dotted lines in Fig. 6.5, the approximate ranges of their thresholds can be determined from the y-axis of Fig. 6.5. Then, the preferred values of their thresholds can be selected to meet the above principle in their corresponding ranges using simulations similar to Fig. 6.7, the detailed process of which is omitted for the sake of conciseness. For the systems considered here, the thresholds of CE, SCR, SDR and ME criteria are selected as 0.035, 0.24, 0.37 and 2.2, respectively.

The same simulations of Fig. 6.8 are tested for AIDA with different SCs, as shown in Fig. 6.11. For AIDA with CE, SCR and SDR, it is shown that the average numbers of outer iterations are significantly reduced with very little performance loss at low SNRs. However, large BER performance losses are caused at medium and high SNRs, since some correctable decodings are prematurely stopped by the three criteria. This implies that the three existing SCs are not well suited for stopping the iteration of the uncorrectable decodings for our considered systems. On the other hand, for AIDA using the ME criterion, we observe that it has similar performance to AIDA using the proposed DMI criterion.

To further evaluate and compare the performance of the ME criterion and the D-MI criterion for AIDA, the same simulations of Fig. 6.11 are also tested for different code structures and transmission channel environments respectively while keeping other simulation parameters constant. Fig. 6.12 shows the simulation results for rate-3/4 (3, 12) regular LDPC codes with length 1008 over AWGN channels. Fig. 6.13 shows the simulation results for rate-1/2 (3, 6) regular LDPC codes with length 1008 over non-frequency selective Rayleigh fading channels with normalized maximum Doppler

frequency $f_D T_s = 0.01$. It is shown that the ME criterion with the previously defined threshold 2.2 for the two cases is not as effective as it is for rate-1/2 (3, 6) regular LDPC codes over AWGN channels in term of the performance of the reduction of the average number of outer iterations. These observations suggest that the performance of the existing ME criterion is easily affected by changes in the LDPC code and transmission channel parameters, and thus the threshold needs to be redesigned for the new conditions, which supports the discussion in Section 6.3.3. On the contrary, we can find that even if these conditions are changed, the proposed DMI criterion with previously defined thresholds can still effectively reduce the average number of outer iterations with very little performance degradation at all SNRs, which means that the proposed DMI criterion is more robust than the ME criterion. This is due to the fact that the proposed DMI criterion stops the iteration based on the convergence status of the iterative decoding rather than comparing a predefined threshold for a certain SNR region like the existing SCs.

In addition, although the threshold of the ME criterion can be redesigned to suit new conditions, the workload of this process is large, since its threshold needs to be carefully designed based on a large number of simulations as shown in Fig. 6.5. In contrast, the threshold of the proposed DMI criterion can be designed easily, since the EXIT band chart is easily obtained.

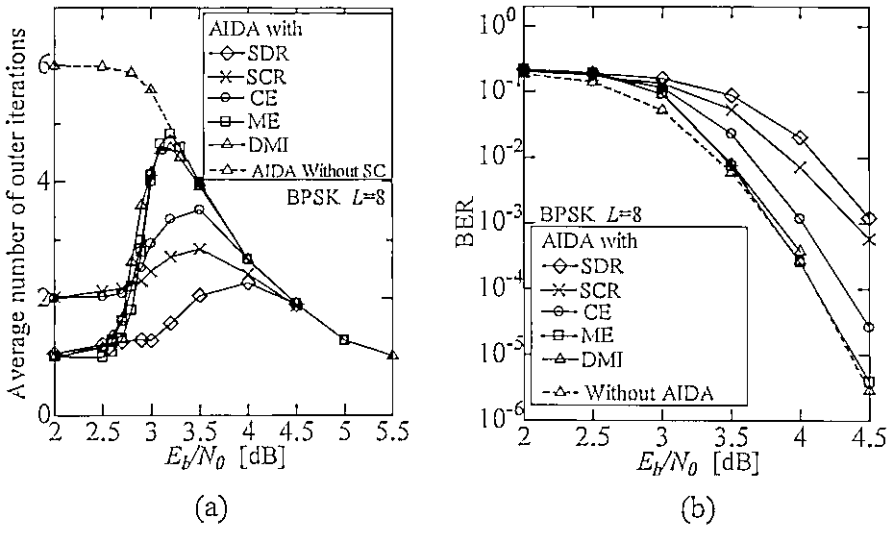

Figure 6.11: Performance of AIDA with different SCs for the considered DE-LDPC coded systems with rate-1/2 (3, 6) regular LDPC codes with length 1008 over AWGN channels with BPSK. (a) Average number of outer iterations. (b) BER performance.

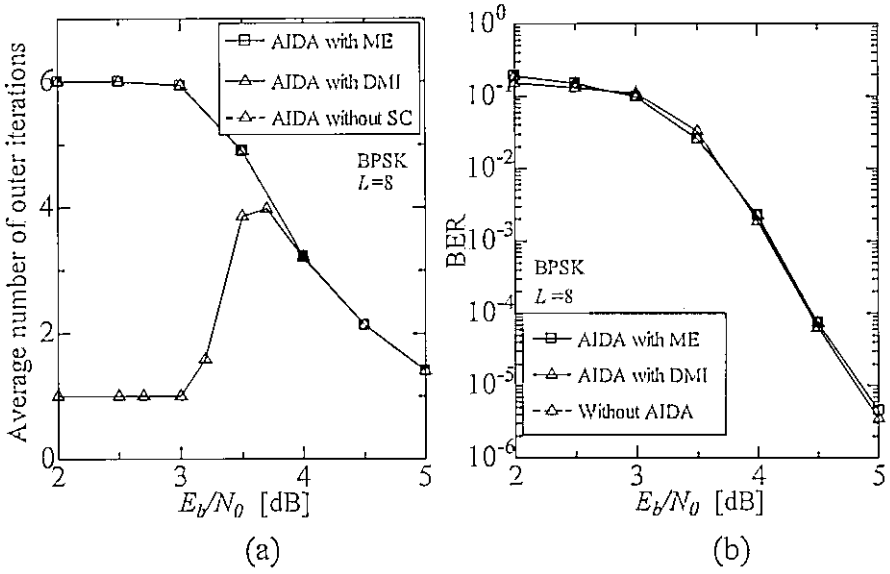

Figure 6.12: Performance of the considered DE-LDPC coded systems with and without AIDA for rate-3/4 (3, 12) regular LDPC codes with length 1008 over AWGN channels with BPSK. (a) Average number of outer iterations. (b) BER performance.

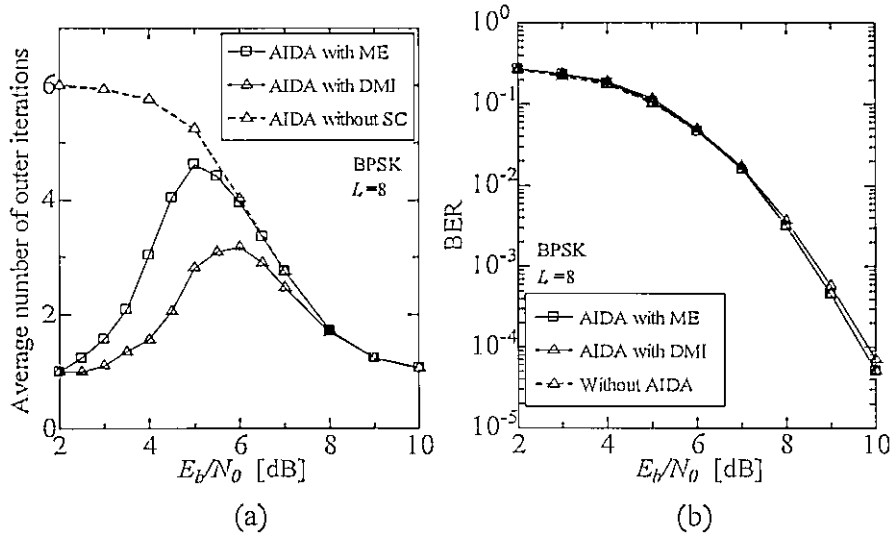

Figure 6.13: Performance of the considered DE-LDPC coded systems with and without AIDA for rate-1/2 (3, 6) regular LDPC codes with length 1008 over non-frequency selective Rayleigh fading channels with normalized maximum Doppler frequency $f_D T_s = 0.01$ and BPSK. (a) Average number of outer iterations. (b) BER performance.

CHAPTER 7

Conclusions and Future Work

7.1 Conclusions

In this thesis, we studied the iterative MSDD scheme for differentially encoded LDPC coded systems. The conclusions are discussed in this chapter.

In this thesis, we consider DE-LDPC coded systems with iterative MSDD. At the transmitter, the LDPC coded sequence is differential encoded by the differential encoder. At the receiver, the MSDD SISOD is viewed as an inner decoder, while the LDPC decoder is viewed as an outer decoder. The iterative decoding is performed between the MSDD SISOD and the LDPC decoder. The transfer characteristics of the MSDD SISOD and the LDPC decoder were analyzed by the EXIT charts. It was shown that the performance can be improved by increasing the OWS of the MSDD SISOD and the number of iterations. Whereas, the performance gain cannot be achieved by iterative decoding, when the inner decoder employs the conventional differential detection. The BER performance of the systems under consideration was evaluated in slow and fast Rayleigh fading channels, respectively. It was shown that these analysis results obtained by the EXIT charts are supported by the computer simulation results. Moreover, the simulation results also showed that in the case of fast fading channels, the systems under consideration has better performance, and more performance gain can be achieved by increasing the OWS and the number of iterations.

When the DE-LDPC coded systems use regular LDPC codes, the system performance can only be improved marginally in the case of the OWS with a large value. In order to solve this problem, we designed the irregular LDPC codes for the MSDD SISOD with large values of OWS by using the EXIT chart to optimize the degree distributions of irregular LDPC codes. The simulation results demonstrated that the performance of the DE-LDPC codes systems with a large OWS can be significantly improved by using the optimized codes. Moreover, when the optimized irregular LDPC codes with very long length, the performance of the systems under consideration can close to the capacity of the noncoherent AWGN channel. Therefore, for the DE-LDPC coded systems with iterative MSDD, it is important to optimize the LDPC codes for different OWS, especially for the large OWS.

On the other hand, the complexity of the MSDD SISOD will become prohibitively high as the OWS and the order of the modulation become large. This high complexity will result in an unacceptable decoding time delay, which makes difficult to achieve a realistic system. To solve this problem, we proposed an ISOMA to reduce the complexity of the MSDD SISOD. The proposed ISOMA combines the features of the existing SOMA

approaches, which can guarantee that the LLR of each coded bit can be computed with high reliability. Through computer simulations, it was shown that the computational complexity of the MSDD SISOD as well as the iterative decoding complexity of DE-LDPC coded systems with iterative MSDD can be greatly reduced by the proposed ISOMA. Furthermore, compared with the existing ITS-MA and SOMA, the proposed ISOMA has better performance in terms of the BER performance and the ability of reducing the decoding complexity of DE-LDPC systems with iterative MSDD.

Since the computation of the metric of the MSDD SISOD cannot be computed recursively as in the coherent detection case, the complexity of the MSDD SISOD is still relatively high even if the proposed ISOMA approach is used. In order to further reduce the iterative decoding complexity and delay of the systems under consideration with finite length LDPC codes, an AIDA scheme which can adaptively adjust the OWS of the MSDD SISOD and the outer iteration number of the iterative decoder was proposed. In AIDA, the OWS and the outer iteration number are adaptively adjusted by using a SC to judge whether the iterative decoding converges or not. To circumvent the disadvantages of the existing SCs, a new SC, which we call DMI criterion, was proposed for tracking the convergence status of the iterative decoding by tracking the difference of the output mutual information of the LDPC decoder between two consecutive outer iterations of the considered systems. Simulation results showed that AIDA with the proposed DMI criterion can significantly reduce the iterative decoding complexity and delay of the considered systems at all SNRs. Moreover, compared with the existing SCs, it is proved that the DMI criterion is more effective for the considered systems in terms of reducing the average number of outer iterations, performance loss and robustness.

7.2 Future Work

In the future research, possible research topics of interest include the following aspects:

1) In this thesis, binary LDPC codes were considered for the studied systems. Compared to binary LDPC codes, non-binary LDPC codes were confirmed that they have a better error correction performance than that of binary LDPC codes [47], since they can eliminate the impact of the four-cycle on the convergence performance of the iterative decoding. Furthermore, non-binary LDPC codes are suit for combining with high order modulations, thereby providing a higher data transfer rate and spectral efficiency [48], [49]. As is well known, wireless transmission systems are required to provide high-speed data transfer in the future. Therefore, the study of DE-non-binary LDPC coded systems with iterative MSDD is desirable.

2) Orthogonal frequency division multiplexing (OFDM) [50] has been widely adopted by many wireless communication standards, such as HyperLAN/2, IEEE802.11a, IEEE802.11g, and IEEE802.16a etc., since it can provide high data-rates with low system complexity and robust against adverse channel effects. Many coded OFDM schemes have been pursued to mitigate the deep channel fading. Up to now, most literatures have focused on the coded OFDM systems with coherent detection. In this case, it is required to use pilot symbols for obtaining CSI, which results in bandwidth consump-

tion [51]. Moreover, the estimation of CSI is sometimes difficult to be realized, such as vehicle-to-vehicle communications [52]. The differential modulation and detection scheme has high spectral efficiency since it does not require the estimation of CSI. On the other hand, the combination of LDPC codes and differentially coded OFDM still has many problems that deserve to study in depth, such as the optimization of LDPC codes for differentially coded OFDM. Therefore, we will further study the characteristic and performance of DE-LDPC coded OFDM systems with iterative MSDD.

REFERENCES

[1] R. G. Gallager, Low-Density Parity-Check Codes. Cambridge, MA: MIT Press, 1963.

[2] RM Tanner, "A recursive approach to low complexity codes," *IEEE Trans. on Inf. Theory,* vol.27, pp.533-5847, Sep.1981.

[3] D. J. C. MacKay,"Good error correcting codes based on very sparse matrices," *IEEE Trans. Inf. Theory,* vol.45, no.2, pp.399-431, Mar. 1999.

[4] MG Luby, M. Mitzenmacher, MAShokollahi, and DA Spielman, "Improved low-density parity-check codes using irregular graphs," *IEEE Trans. Inform. Theory,* vol.47, no.2, pp.585-598, Feb. 2001.

[5] M. Franceschini, G. Ferrari, R. Raheli, and A. Curtoni, "Serial concatenation of LDPC codes and differential modulations," *IEEE J. Select. Areas Commun.,* vol.23, pp.1758-1768, Sept. 2005.

[6] V. T. Nam, P. Y. Kam, and Y. Xin, "LDPC codes with BDPSK and differential detection over flat Rayleigh fading channels," in *Proc. of GLOBECOM f07,* pp.3245-3249, Nov. 2007.

[7] H. Tatsunami, K. Ishibashi, and H. Ochiai, "On the performance of LDPC codes with differential detection over Rayleigh fading channels," in *Proc.of the 63rd IEEE VTC f06,* vol.5, pp. 2388-2392, May 2006.

[8] J. Li, "Differentially-encoded LDPC codes: part I-special case of product accumulate codes," *EURASIP Journal on Wireless Communications and Networking,* vol.2008, Article ID 824673, 14 pages, 2008.

[9] J. Li, "Differentially encoded LDPC codes-part II: general case and code optimization," *EURASIP Journal on Wireless Communications and Networking,* vol. 2008, Article ID 367287, 10 pages, 2008.

[10] D. Divsalar and M. K. Simon, "Multiple-Symbol Differential Detection of MP-SK,"*IEEE Trans. Commun.,* vol. 38, pp. 300-308, Mar. 1990.

[11] D. Divsalar and M. K. Simon, "Maximum-likelihood differential detection of uncoded and trellis coded amplitude phase modulation over AWGN and fading channels—Metrics and perormance,"*IEEE Trans. Commun.,* vol. 42, pp. 76-89, Jan. 1994.

[12] S. Handa, Y. Okano, M. Liu, F. Sasamori, and S. Oshita, "Fast calculation algorithm and error performance of multiple-symbol differential detection over fading channels," *IEICE Trans. Commun.,* vol. E86-B, pp. 1050-1056, Mar. 2003.

[13] M. Peleg and S. Shamai, "Iterative decoding of coded and interleaved noncoherent multiple symbol detected DPSK," *Electron. Lett.*, vol.33, no.12, pp.1018-1020, June 1997.

[14] V. Pauli, L. Lampe, and R. Schober, "Turbo DPSK" using soft multiple-symbol differential sphere decoding," *IEEE Transactions on Information Theory,* vol.52, no.4, pp.1385-1398, April.2006.

[15] L. Wang, L. K. Kong, S. X. Ng and L. Hanzo, "A Near-Capacity Differentially Encoded Non-Coherent Adaptive Multiple-Symbol-Detection Aided Three-Stage Coded Scheme," in *Proc.of the 71rd IEEE VTC'10,* vol.1, pp.1-5, May 2010.

[16] L. Wang, L. Hanzo, "Multiple-Symbol Detection Aided Differential Spatial Division Multiple Access," in *Proc. IEEE ICC 2011*, pp.1-5, Jun. 2011.

[17] Ziyan Jia, Katsunobu Yoshii, Shiro Handa, Fumihito Sasamori, and Shinjiro Oshita, "Noncoherent maximum likelihood detection for differential spatial multiplexing MIMO systems," *IEICE Trans. Commun.,* vol.E93-B, no.2, pp.361-368, Feb. 2010.

[18] L Wang, L Hanzo, "Low-Complexity Near-Optimum Multiple-Symbol Differential Detection of DAPSK Based on Iterative Amplitude/Phase Processing," *IEEE Trans. Veh. Technol.,* vol. 61, no.2, pp.894-900 , Feb. 2012.

[19] S. ten Brink, "Convergence behavior of iteratively decoded parallel concatenated codes," *IEEE Trans. Commun.,* vol.49, no.10, pp.1727-1737, Oct. 2001.

[20] W. C. Jakes, Ed., *Microwave Mobile Communications,* New York: Wiley, 1974.

[21] Y. Satio, Modulation and demodulation of digital wireless communications, P.182, IEICE, Tokyo, 1996.

[22] T.J. Richardson, M. A. Shokrollahi, and R. L. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," *IEEE Trans. Inform. Theory,* Vol.47, pp.619-637, Feb. 2001.

[23] T. J. Richardson and R. L. Urbanke, "Efficient encoding of low-density parity-check codes", *IEEE Trans. Inform. Theory,* Vol.47, no.2, pp.638-656, Feb. 2001.

[24] S. T. Brink, G. Kramer, and A. Ashikhmin, "Design of low-density parity-check codes for modulation and detection," *IEEE Trans. Commun.,* Vol.52, pp.670-678, Apr. 2004.

[25] S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollara, "Serial Concatenation of Interleaved Codes: Performance Analysis, Design, and Iterative Decoding," in *IEEE Trans. Inform. Theory,* vol 44, p.909-926, May 1998.

[26] F. Guo, S. X . Ng and L. Hanzo, "LDPC assisted block coded modulation for transmission over Rayleigh fading channels," in *Proc.of the 57rd IEEE VTC '03,* vol.3, pp.1867-1871, Apr. 2003.

[27] J. Hagenauer, "The EXIT Chart - introduction to extrinsic information transfer in iterative processing," in *Proc. 12th European Signal Processing Conference,* pp.1541-1548, Sep. 2004.

[28] S. ten Brink, G. Kramer, and A. Ashikhmin, "Design of low-density parity-check codes for modulation and detection," *IEEE Trans. Commun.,* vol.52, pp.670-678, Apr. 2004.

[29] K. Price and R. Storn, "Differential evolution-a simple and efficient heuristic for global optimization over continuous spaces," *J. Global Optimization,* vol.11, no.4, pp.341-359, Nov. 1997.

[30] Y. Hu, E. Eleftheriou, and D. M. Arnold, "Regular and irregular progressive edge growth Tanner graphs," *IEEE Trans. Inf. Theory,* vol.51, no.1, pp.386-398, Jan. 2005.

[31] M. Peleg and S. Shamai, "On the capacity of the blockwise incoherent MPSK channel," *IEEE Trans. Comm.,* vol. 46, pp. 603-609, May 1998.

[32] N. Jin, X, p. Jin, Y.G. Ying, S. Wang and Y.F. Lv, "Multiple-symbol M-bound intersection detector for differential unitary space-time modulation ," *Communications, IET,* vol.4, pp.1987-1997, Nov. 2010.

[33] J. B. Anderson and S. Mohan, "Sequential coding algorithms: a survey and cost analysis," *IEEE Trans. Commun.,* vol.com-32, pp.169-176, Feb. 1984.

[34] Y. L. C. de Jong and T. J. Wilink, "Iterative tree search detection for MIMO wireless systems," *IEEE Trans. Commun.,* vol.53, no.6, pp.930-935, Jun. 2005.

[35] MPC Fossorier, F. Burkert, S. Lin and J. Hagenauer, "On the equivalence between SOVA and max-log-MAP decodings," *IEEE Commun. Lett.,* vol.2, pp.137-139, May 1998.

[36] K. K. Y. Wong and P. J. McLane, "Bi-Directional Soft-Output M-Algorithm for iterative Decoding," in *Proc. IEEE Int. Conf. Communications,* vol.2, pp.792-797, Jun. 2004.

[37] K.K.Y. Wong, "The Soft-Output M-Algorithm and its applications," PhD Thesis, Queen's University, Kingston ON, Canada, Aug. 2006.

[38] J Vogt and A Finger, "Improving the max-log-MAP turbo decoder," *IEE Electronics Letters,* vol.36, no,23, pp.1937-1939, Nov. 2000.

[39] J. W. Lee and R. E. Blahut, "Generalized EXIT chart and BER analysis of finite-length turbo codes," in *Proc. GLOBECOM 2003,* vol.4, pp.2067-2072, Dec. 2003.

[40] J. Hagenauer, E. Offer, and L. Papke, "Iterative decoding of binary block and convolutional codes," *IEEE Trans. Inf. Theory,* vol.42, no.2, pp. 429-445, Mar. 1996.

[41] R. Y. Shao, S. Lin, and M. P. C. Fossorier, "Two simple stopping criteria for turbo decoding," *IEEE Trans. Commun.*, vol.47, no.8, pp.1117-1120, Aug. 1999.

[42] Y. Wu, D. Woerner, and J. Ebel, "A simple stopping criteria for turbo decoding," *IEEE Commun. Lett.*, vol.4, no.8, pp.258-260, Aug. 2000.

[43] F. Zhai and I. Fair, "New error detection techniques and stopping criteria for turbo decoding," in *Proc. 2000 Can. Conf. Electr. Comput. Eng.*, pp. 58-62, Mar. 2000.

[44] F. M. Li and A. Y.Wu, "On the new stopping criteria of iterative turbo decoding by using decoding threshold," *IEEE Trans. Signal Process.*, vol.55, no.11, pp.5506-5516, Nov. 2007.

[45] H. Chen, R. G. Maunder and L. Hanzo, "Low-Complexity Multiple-Component Turbo-Decoding-Aided Hybrid ARQ," *IEEE Trans. Veh. Technol.*, vol. 60, no.4, pp.1571-1577, May. 2011.

[46] M. Fossorier, M. Mihaljevic, and H. Imai, "Reduced complexity iterative decoding of low density parity check codes based on belief propagation," *IEEE Trans. Commun.*, vol.47, pp.673-680, May 1999.

[47] Davey M C and MacKay D J C, "Low-density parity check codes over GF(q)," *IEEE Commun. Lett.*, vol.2, no.6, pp.165-167, Jun. 1998.

[48] Hu X Y and Eleftheriou E, "Binary representation of cycle tanner-graph GF(2^b) codes," in *Proc. IEEE ICC 2004*, pp.528-532, Jun. 2004.

[49] Sanae El Hassani, Marie-Helene Hamon and Pierre Penard, "A comparison study of binary and non-binary LDPC codes decoding," in *Proc. International Conference on SoftCOM 2010*, pp.355-359, Sept. 2010.

[50] Leonard J. Cimini, Jr., "Analysis and simulation of a digital mobile channel using orthogonal frequency-division multiplexing", *IEEE Trans. Comm.*, vol.33, no.7, pp.665-675, July 1985.

[51] H. Zamiri-Jafarian, M. J. Omidi, and S. Pasupathy, "Improved channel estimation using noise reduction for OFDM systems", in *Proc. Veh. Technol. Conf.*, vol.2, pp.1308-1312, Apr. 2003.

[52] Lin Cheng. Cimini, Henty, B.E., Stancil, D.D., Fan Bai and Mudalige, P. "Mobile Vehicle-to-Vehicle Narrow-Band Channel Measurement and Characterization of the 5.9 GHz Dedicated Short Range Communication (DSRC) Frequency Band ", *IEEE J. Select. Areas Commun.*, vol.25, no.8, pp.1501-1516 , Oct. 2007.

APPENDIX A

Abbreviations and Acronyms

AIDA	Adaptive Iterative Decoding Approach
APSK	Amplitude/Phase-Shift Keying
AWGN	Additive White Gaussian Noise
BER	Bit Error Rate
BPSK	Binary Phase Shift Keying
BF	Bit-flipping
BP	Belief-propagation
CE	Cross Entropy
CN	Check Node
CSI	Channel State Information
DE-LDPC	Differential Encoded Low-density parity-check
DMI	Differential Mutual information
DBPSK	Differential Binary Phase Shift Keying
ECCs	Error Correction Codes
EXIT	Extrinsic Information Transfer
LDPC	Low-density parity-check
LLR	Log Likelihood Ratio
LLRs	Log Likelihood Ratios
MAP	Maximum A Posteriori
ME	Mean-estimate
MI	Mutual information
ML	Maximum Likelihood
MLD	Maximum Likelihood Detection
MPSK	Multiple-Phase Shift Keying
MSDD	Multiple-Symbol Differential Detection
MSDD SISOD	MSDD SISO Demodulator
OFDM	Orthogonal Frequency Division Multiplexing
OWS	Observation Window Size

PDF	Probability Density Function
PEG	Progressive-edge-growth
QPSK	Quadrature Phase Shift Keying
SC	Stopping Criterion
SDR	Sign-change-ratio
SCR	Sign Difference Ratio
SF	Scaling Factor
SISO	Soft-Input Soft-output
SNR	Signal-to-Noise Ratio
VN	Variable Node

APPENDIX B

Notations

A_{ij}	the i,jth element of matrix A
A^*	complex conjugate of matrix A
A^T	transpose of matrix A
det	determinant
diag	diagonal matrix
$E\{\cdot\}$	(statistical) mean value or expected value
$\exp\{\cdot\}$	exponential function
f_D	the maximum Doppler frequency
$f_D T_s$	the normalized maximum Doppler frequency
$I_0(\cdot)$	zeroth order modified Bessel function of the first kind
j	$\sqrt{-1}$
$J_0(\cdot)$	zeroth order Bessel function of the first kind
L	the length of observation window size of MSDD
N	the length of LDPC codes
T_s	the symbol period