Title	Evaluating the Block Error Probability of Trellis Codes and Designing a Recursive Maximum Likelihood Decoding Algorithm of Linear Block Codes Using Their Trellis Structure
Author(s)	Yamamoto, Hiroshi
Citation	大阪大学, 1996, 博士論文
Version Type	VoR
URL	https://doi.org/10.11501/3110102
rights	
Note	

Evaluating the Block Error Probability of Trellis Codes and Designing a Recursive Maximum Likelihood Decoding Algorithm of Linear Block Codes Using Their Trellis Structure

Yamamoto Hiroshi

January 1996

Evaluating the Block Error Probability of Trellis Codes and Designing a Recursive Maximum Likelihood Decoding Algorithm of Linear Block Codes Using Their Trellis Structure

Yamamoto Hiroshi

January 1996

Evaluating the Block Error Probability of Trellis Codes and Designing a Recursive Maximum Likelihood Decoding Algorithm of Linear Block Codes Using Their Trellis Structure

by

Yamamoto Hiroshi

January 1996

Dissertation submitted to Graduate School of Engineering Science of
Osaka University in partial fulfillment of the requirements for
the degree of Doctor of Philosophy in Engineering

Abstract

Recently, digital communication systems are widely used. To transmit such as binary data file, command and control sequences of channel, high reliability and high speed communication system is required. For this purpose, it is important to research codes with strong error correcting capability and efficient decoding algorithms. In this dissertation an efficient method to estimate the block error probability of trellis codes and a maximum likelihood decoding (MLD) algorithm with low computational complexity for linear block codes using characteristics of trellis diagrams of codes are proposed.

To analyze the error correcting capability provided by trellis codes, an efficient method to evaluate the block error probability of trellis codes is proposed in Chapter 2. The notion of correct state on the most likely paths in Viterbi decoding on AWGN channels is introduced. The probability that a state is correct and the probability that the state following a correct state is also correct are evaluated by simulation. These probabilities do not depend on block length. Then, the block error probability for any given block length is estimated by simple calculation from the above mentioned probabilities.

The proposed method is useful in flexibility of adapting the analysis for given block length. This method enables us to get precise estimation for various block length efficiently.

This method is applied for some specific 8-PSK Ungerboeck codes with $2^4, 2^5$ and 2^6 states. The results show that the values obtained by this method are very close to those by exhaustive simulation. It is concluded that our method is very effective.

To simplify decoding of a code, a new trellis-based MLD algorithm for a linear block code using a sectionalized trellis diagram is proposed and its computational complexity is analyzed in Chapter 3.

In general, a sectionalized minimal trellis diagram for a linear block code is loosely

connected provided that the section length is not too small. Within a section, there are parallel branches between two adjacent states, these parallel branches may form a subtrellis, and many adjacent state pairs have the same label set for the parallel branches. These structural properties can be used to reduce the computational complexity of a trellis-based maximum likelihood decoding (MLD) using the Viterbi algorithm. In Chapter 3, a new trellis-based MLD algorithm for a linear block code using a sectionalized trellis diagram is proposed and its computational complexity is analyzed in terms of the number of addition equivalent operations. In the proposed algorithm, to process the label sets of parallel branches in a section of the trellis, a maximum likelihood decoding procedure is used. The overall computational complexity depends on how to choose recursive sectionalization in the algorithm. A method for finding the optimum sectionalization of a trellis in terms of computation complexity using a dynamic programming approach is given. Numerical results show that a proper sectionalization of a trellis considerably reduces the decoding complexity.

It is concluded that characteristics of trellis diagrams are of great use to estimate the block error probability and to design an MLD algorithm.

Acknowledgment

I am deeply indebted to many people for the advice, feedback and support they gave to me in the course of this work. I would especially like to thank Professor emeritus Tadao Kasami, currently Professor of Nara Institute of Science and Technology for his invaluable support, discussions and encouragement throughout the work.

I am grateful to my supervisor Professor Nobuki Tokura for his invaluable suggestions and discussions on the work. I am also obliged to Professor Hideo Miyahara and Professor Toshinobu Kashiwabara for their helpful comments and suggestions.

I am extremely thankful to Associate Professor Toru Fujiwara for his invaluable discussions and great support throughout the work. I would like to thank Associate Professor Toyoo Takata of Nara Institute of Science and Technology. I also thank Mr. Yuji Yamashita of Yomiuri Television for his helpful discussions.

I would like to thank Associate Professor Hiroyuki Seki of Nara Institute of Science and Technology, Dr. Robert H. Morelos-Zaragoza of Tokyo University, Dr. Masahiro Higuchi and Dr. Ryuichi Nakanishi for their kind and helpful support. I am thankful to Dr. Yuich Kaji of Nara Institute of Science and Technology and Mr. Hajime Watanabe of Nara Institute of Science and Technology for their valuable support. I am also grateful to Ms. Yukiko Tanobe and Ms. Machiko Uehara for their kind support.

Lastly, I would like to thank all the students of information theory and logics laboratory of Osaka University.

List of Publications

Journal Papers

(1) Hiroshi Yamamoto, Toyoo Takata, Toru Fujiwara Tadao Kasami, "On Evaluation of Block Error Probability of Trellis codes," (in Japanese), IEICE, Vol.J78-A, No.6, pp.716-721 (1995-06).

(2) Hiroshi Yamamoto, Toru Fujiwara, Tadao Kasami, and Shu Lin, "A Recursive Maximum Likelihood Decoding Algorithm for a Linear Block Code Using a Sectionalized Trellis Diagram and Its Optimization," IEICE Transactions on Fundamentals, (submitted 1995).

International Conferences

(3) Toru Fujiwara, Hiroshi Yamamoto, Tadao Kasami, and Shu Lin, "A Recursive Maximum Likelihood Decoding Procedure for a Linear Block Code Using an Optimum Sectionalized Trellis Diagram," The 33rd Annual Allerton Conference on Communication, Control, and Computing, (1995-10).

Others

(4) Hiroshi Yamamoto, Toyoo Takata, Toru Fujiwara Tadao Kasami, "Approximation To the Block Error Probability fo Convolutional Codes," (in Japanese), IEICE Technical Report, **IT 92**-102 (1993-01).

(5) Hiroshi Yamamoto, Toru Fujiwara, Tadao Kasami, and Shu Lin, "On the Complexity of Maximum Likelihood Decoding for a Linear Block Code Using a Section Trellis Diagram," IEICE Technical Report, **IT95**-26 (1995-07).

(6) Hiroshi Yamamoto, Toru Fujiwara, Tadao Kasami, and Shu Lin, "A Recursive Maximum Likelihood Decoding Procedure for a Linear Block Code Using an Optimum Sectionalized Trellis Diagram," Proc. of 18th Symposium on Information Theory and Its Applications, pp. 445-448, Hanamaki, Japan (1995-10).

Contents

Chapter 1

Introduction

Digital communication technology has made remarkable advances lately, and demand for high reliability and high speed communication system has risen. Research about such codes as guarantee low decoding error and can be decoded efficiently has been carried out earnestly.

A trellis diagram of a code is a directed graph with the initial state (node) and the final state, whose labels of paths from the initial node to the final node correspond to the code words one-to-one. In evaluating the error probability of codes and designing simpler decoder, trellis diagrams of codes have a lot of useful information. Therefore, trellis diagrams of codes has been studied very actively in recent years. In this dissertation, a method to estimate the block error probability of trellis codes and a maximum likelihood decoding (MLD) algorithm for linear block codes are proposed using characteristics of trellis diagrams of codes.

For example, we can tell a specific bit of a trellis code must be decoded incorrectly under some conditions from the connection of the trellis diagram of the code. In Chapter 2, such characteristics of trellis diagram to evaluate the block error probability of trellis codes are used.

The error performance of a trellis code is usually evaluated in terms of bit error probability. However, block error probability is more adequate measure for an error control system whose unit of message is block.

It is assumed that code bits are transmitted through an additive white Gaussian noise (AWGN) channel, and they are decoded by Viterbi decoder[1]. Viterbi decoding is an algorithm of MLD, which is a dynamic programming algorithm to find the shortest path from the initial state to the final state of a trellis diagram. The block error probability of trellis codes was evaluated by simulation[2, 3]. When block error probabilities of a code are required for various block lengths, simulation must be executed for each block length. It takes enormous time for the computation when the block error probability is small, since examination must be repeated until the unlikely phenomenon occurs enough times to estimate a reliable value.

In Chapter 2, a two-step method for evaluating the block error probability is proposed. We consider if a state on the path with largest metric in Viterbi decoding is correct or not. In step 1, the probability that a state is correct and the probability that the state following a correct state is also correct are evaluated by simulation. In step 2, the block error probability for any given block length is estimated by simple calculation from the above mentioned probabilities.

The evaluation in step 1 does not depend on the block length. If the evaluation of the values in step 1 has carried out, the block error probability for any block length can be estimated by the simple calculation of step 2.

When the performance of an error control system from the block error probability is analyzed, the analysis for various required block length is adapted flexibly by proposed method.

This method is applied for the specific 8-PSK Ungerboeck codes[4] with $2^4, 2^5$ and 2^6 states. For the example code with 2^5 states, the ratio between the value for the block error probability obtained by this method and one by simulation is less than 1.1 when the block error probability is less than 10^{-3}. These results show that our method is very effective.

A trellis diagram of a linear block code often has parts of subgraphs with identical structure. Such characteristic of trellis codes can be used to reduce the computational complexity of a trellis-based MLD using the Viterbi algorithm. In Chapter 3 such characteristics of trellis diagrams are used to design an efficient decoding algorithm.

A trellis diagram for a linear block code is generally expressed and constructed in terms of sections [1-7]. A binary (N, K) block code C has an N-section trellis diagram [10] in which each branch represents a code bit. An l-section trellis diagram with $l < N$ can be constructed from the N-section trellis diagram. In this l-section trellis diagram, each branch represents multiple code bits. In many cases, the trellis diagrams for linear block codes, such as Reed-Muller codes, are loosely connected and have simple parallel structure. The loose connectivity and parallel structure simplify the implementation of Viterbi decoding. In fact, smaller Viterbi decoders can be built to process the parallel sub-trellis diagrams in parallel. This speeds up the decoding process which is important in high-speed data communication systems.

The parallel structure of the l-section minimal trellis diagram for a binary linear block code was first investigated in [10], in which the number of parallel and structurally identical (except branch labels) sub-trellis diagrams without cross connections, called parallel components, in each section was expressed in terms of the dimensions of specific linear codes related to the given code.

It is shown in [11] that sectionalization of a trellis diagram does not reduce the branch complexity. However, if the section length is not too small, there are parallel branches between two adjacent states. These parallel branches may form a sub-trellis, and many adjacent state pairs have the same label set for the parallel branches. These structural properties can be used to reduce the computational complexity of a trellis-based MLD using the Viterbi algorithm. Furthermore, an MLD procedure can be used recursively for finding the branches with the largest metrics among the sets of parallel branches in a section.

In Chapter 3, a new trellis-based MLD algorithm for a linear block code using a sectionalized trellis diagram is proposed and its computational complexity is analyzed in terms of the number of addition equivalent operations. The overall computational complexity depends on how to choose recursive sectionalization in the algorithm. A method for finding the optimum sectionalization of a trellis diagram in terms of computational complexity is given using a dynamic programming approach. Numerical results show that a proper sectionalization of a trellis diagram considerably reduces the decoding complexity.

Chapter 3 is organized as follow. Section 2 gives a brief review of the structural properties of a sectionalized minimal trellis diagram of a linear block code. A new trellis-based MLD algorithm is proposed and its computational complexity is analyzed in Section 3. An MLD procedure is presented to process the sets of parallel branches in a section to find the largest branch metrics among the sets of parallel branches. Also presented in Section 3 is a method for finding the optimum sectionalization of a trellis diagram for a linear block code. This method is based on a dynamic programming approach. In Section 4, specific examples are given. Conclusion is given in Section 5.

It is concluded that characteristics of trellis diagrams are of great use to estimate the block error probability and to design an MLD algorithm. These results contribute large advance in designing high reliability and high speed digital communication technology. Chapter 4 presents a summary of the ideas and results presented in this dissertation and discusses future research goals.

Chapter 2

A Method to Evaluate the Block Error Probability of Trellis Codes

2.1 Definitions

Consider a trellis code C such that the encoder accepts k-bit block and outputs n-bit block. The n-tuple $(y_1, y_2, \ldots, y_n)$ with $y_i \in \{0, 1\}$ which is the encoder output in each time is called **symbol**. The set of all the symbol of the encoder output is denoted Σ.

Let G be a trellis diagram of C. Assume that G does not have any parallel edges. Let S be the set of the states of G. For a path $\mathbf{p}$ on G which starts from the initial state of G, we define the state of time t and the edge of time t as follows: The initial state is called the state of time 0. The edge on path $\mathbf{p}$ from the state of time $t-1$ is called the edge of time t, with $t \geq 1$ and the state which the edge of time t enters is called the state of time t. For path $\mathbf{p}$, the symbol labeled to the edge of time t is called the symbol of time t for $\mathbf{p}$.

Let $\mathbf{u} = u_1 u_2 \ldots$ with $u_t \in \Sigma$ be the encoded symbol sequence of a trellis code C. Let u_t^s be the state of time t on the path corresponds to $\mathbf{u}$. $\mathbf{u}^s = u_0^s u_1^s \ldots$ with $u_t^s \in S$ is called **encoded state sequence.**

Encoded symbol sequence $\mathbf{u}$ is transmitted through a channel, then it is decoded by a decoder. The longer the path which is stored for the maximum likelihood decoding with Viterbi algorithm is, the harder the implementation of decoder is. In this chapter, the pseudo-maximum likelihood decoding using Viterbi algorithm is considered, where the length of the path stored in memory is limited by given positive integer w[5].

The decoder outputs **decoded symbol sequence** and **decoded state sequence** which is defined as follows: The path with largest metric among the paths from the initial state to a state of time t is called the maximum likelihood path of time t. Suppose that a positive integer w is given. We call w **window size.** Let the symbol of time t on the maximum likelihood path of time $t+w-1$ be v_t, and the state of time $t-1$ on the path be v_{t-1}^s with $t \geq 1$. The output sequence of the decoder is $\mathbf{v} = v_1 v_2 \dots$ with $v_t \in \Sigma$, and $\mathbf{v}$ is called the decoded symbol sequence. Also, $\mathbf{v}^s = v_0^s v_1^s \dots$ with $v_t^s \in S$ is called the decoded state sequence.

At time t, when $u_t = v_t$ holds we say that the symbol of time t was decoded correctly, otherwise, we say it was decoded incorrectly. Similarly, we say that the state of time t was decoded correctly if $u_t^s = v_t^s$, otherwise, we say it was decoded incorrectly.

For a given positive integer b, the symbol sequence from time t to time $t+b-1$ is called the block of length b from time t. When at least one symbol in the block was decoded incorrectly, we say that the block was decoded incorrectly.

2.2 Formula of Approximation of Block error Probability

Assume that input of an encoder of the trellis code C is independent of time and each input k-tuple occurs with the same probability. The encoded sequence for the input is transmitted through a noisy channel. Then they are decoded with the method described in Sect. 2.1.

Let b be the length of a block. The probability that a block starting from time t is decoded incorrectly is denoted $p_{b,t}$. Let $p'_{b,t}$ be the probability that at least one state from time $t-1$ to $t+b-1$ is decoded incorrectly. Let $q_{m,t}$ be the probability that the state of time t is decoded correctly under the condition that for a positive integer m, every state from time $t-m$ to $t-1$ is decoded correctly. $q_{0,t}$ is defined as the probability that the state of time t is decoded correctly. The probability $1-p'_{b,t}$ that every state from time $t-1$ to $t+b-1$ is decoded correctly is given by

$$1 - p'_{b,t} = q_{0,t-1} \prod_{m=1}^{b} q_{m,t+m-1}. \tag{2.1}$$

If every state from time $t-1$ to $t+b-1$ is decoded correctly, every symbol from time t to $t+b-1$ is decoded correctly, since the trellis diagram does not have any parallel edges. Therefore, we have that

$$p_{b,t} \leq p'_{b,t}. \tag{2.2}$$

When at least one state from time $t-1$ to time $t+b-1$ is decoded incorrectly, at least one of the following conditions i), ii), iii) holds:

i) There exists an integer i with $t \leq i \leq t+b-1$ such that the state of time $i-1$ is decoded correctly and the state of time i is decoded incorrectly.

ii) There exists an integer i with $t \leq i \leq t+b-1$ such that the state of time $i-1$ is decoded incorrectly and the state of time i is decoded incorrectly.

iii) All the states from time $t-1$ to $t+b-1$ are decoded incorrectly.

The decoding described in Sect. 2.1 can be regarded as a maximum likelihood decoding when the window size w is sufficiently large[5, 6]. If a maximum likelihood decoding is used, a decoded symbol sequence and a decoded state sequence are one of the encoded symbol sequences and one of the encoded state sequences, respectively. If any state of a trellis diagram does not have edges diverging from the state with the same label, condition i) implies that this block is decoded incorrectly. Similarly, If any state of a trellis diagram does not have edges entering to the state with the same label, condition ii) implies that this block is decoded incorrectly. All the example codes in Sect. 2.4 satisfy the both properties described above. When the condition iii) holds and the length of a block is sufficiently large, the probability that all the states of a block are decoded incorrectly and that all the symbols of the block are decoded correctly must be very small. Thus, when at least one state from time $t-1$ to $t+b-1$ is decoded incorrectly, it can be considered that the block with length b from time t is decoded incorrectly.

From these observation, when the window size w is sufficiently large and the block length b is not too small, the following approximation holds:

$$p_{b,t} \approx p'_{b,t}. \tag{2.3}$$

For example codes in Sect. 2.4, the outputs of the encoder is not periodic when inputs of it is not periodic. Assuming that the channel noise is AWGN, a state of decoder is independent of time when time t is sufficient large. We call this steady state. $q_{m,t}$ with $m \geq 0$ does not depend on t in a steady state. We denote this probability q_m. Also, $p_{b,t}$ does not depend on t in a steady state. We denote this probability p_b and we call this **brock error probability in a steady state.**

Considering practical use of trellis codes, states decoded correctly continue during sufficiently long time for b. In such situation, $q_1, q_2, \dots, q_b$ have almost equal values. We denote this value simply q.

From (2.1) and (2.3), brock error probability p_b in a steady state can be approximated by the following simple formula:

$$1 - q_0 q^b. \tag{2.4}$$

2.3 Two-Step Method of Evaluation

We propose the following two-step method for evaluating the block error probability; In step 1, the values of q_0, q are evaluated. In step 2, the value of $1 - q_0 q^b$ is calculated. Simulation to evaluate q_0, q in Step 1 requires a lot of computation time especially when the block error probability is small. The values of q_0, q are independent of the block length b. If we already have the evaluation of the values in step 1, we can estimate the block error probability for any block length by the simple calculation of step 2.

2.4 Examples Using 0-Tail Trellis Codes

2.4.1 Step 1 for 0-Tail Trellis Codes

We evaluate the values q_0, q by simulation in the following manner: Let 2^ν be the number of states of C, where ν is called the memory order. For sufficiently large N for ν, we consider 0-tail trellis code of C with length N. If N is sufficiently large, the decoder of 0-tail trellis code which is not periodic is supposed in a steady state except states close to the initial state or the final state. Simulation is executed for this 0-tail code in m times, then the number of states decoded correctly is observed within the state from time ν to $N - \nu - 1$. The total number of states which are subjects to be observed is $(N - 2\nu)m$. Let c be the number of the states decoded correctly in the states observed. We evaluate the value of q_0 to be $c/\{(N - 2\nu)m\}$. Let c_c be the number of the states which are decoded correctly and whose preceding states are also decoded correctly. We evaluate the value of q to be c_c/c.

In Fig. 2.1, the 8-PSK Ungerboeck code[4] with $2^\nu = 2^5$ states is used as the 0-tail code with code length N, and the window size of the decoder is $w = 32$. We show p_1, the error probability of symbol (the block error probability with block length $b = 1$), $1-q_0$, the error probability of state, and $1-q$ evaluated by simulation of the probability that a state following a correct state is incorrect. The fact that $1-q_0 > 1-q$ shows that state errors are actually not independent events. The fact that $1 - q_0 > p_1$ shows that the symbol is not always decoded incorrectly even if the state is decoded incorrectly.

2.4.2 Comparison Between the Values of the Approximation and the Values Obtained from Simulation

In the following, we shall show that the block error probability p_b can be approximated by $1 - q_0 q^b$, a function of block length b, as described in Sect. 2.2 through analysis of example codes.

In Fig. 2.2, we compared the simulation results of $1 - p_b$ when S/N ratio of the channel is 8.0(dB), the values obtained by the proposed approximation $q_0 q^b$ and approximation $(1 - p_1)^b$ simply from the symbol error probability p_1 assuming that each symbol error is independent of each other for the above described example code with the horizontal axis to be block length b. While b is small for N and b is not too small (b is between about 10 and $N/2$), the line of our proposed approximation gives good agreement with the results of simulation.

In the domain where the block length b is close to N, the curve of simulation result becomes close to horizontal and the gap between the curve and the approximation becomes larger. Because the error probabilities of states close to the initial state or the final state are smaller than ones for the other states, so the closer the value of b is to N, the less dependant the brock error probability on block length. This special characteristics is not considered in the approximation formula proposed in this chapter.

The formula is the approximation of the block error probability for a steady state. The symbol error probability in a steady state is almost constant, as described above, we can approximate by the simple formula. In the domain where the value of b is small, there is a gap between the approximation and the value obtained by simulation. It can be considered that the probability that a block is decoded correctly in under the condition iii) is not negligible.

We compare the evaluation by proposed method of block error probability and the value obtained by simulation for some example codes with the horizontal axis to be S/N ratio par symbol. Figures 2.3 to 2.5 are results for 0-tail Ungerboeck codes[4] with $2^4, 2^5, 2^6$ states respectively.

We use 0-tail Ungerboeck codes with code length $N = 128$ in simulations. It is shown in [2] that this code length is sufficient for approximation of the code length to be infinite.

In these figures, the parameter b is the block length and w is the window size for decoding. Generally, the window size which is $4-5$ times of memory order is considered sufficient[6, 5]. We take the window size as such value for the decoder of these example codes.

For example, the ratio of the approximation to the value obtained by simulation is less than 1.1 for the code with 2^5 states when the block error probability is less than about 10^{-3}. From these result, we conclude that the difference between the proposed approximation and the value obtained by simulation is small.

In Figures 2.3 to 2.5, the curve of the proposed approximation is above the curve of simulation when the block error probability is large. The proposed approximation exploits the consideration in Sect. 2.2 that each state error is independent of each other when the error probability is small because each interval between a erroneous state and the next erroneous state. Each state error is generally not independent of each other when the error probability is large. It is known that the errors tend to gather, so it is considered that the value evaluated by simulation tends to become lower.

The proposed approximation gives good agreement with the results of simulation when the block length b is not too small and the block error probability is not large. It is confirmed that the proposed method is effective in practical condition.

2.5 Conclusions

In this chapter, for a code whose trellis diagram satisfies the following condition,

- The code does not have any parallel edges.
- Any state of the diagram does not have edges diverging from the state with the same label.
- Any state of the diagram does not have edges entering to the state with the same label.

and the output encoder is not periodic, we proposed a method to approximate the block error probability when the state of the decoder is independent of time. This method is effective when S/N ratio is in practical range and the block length is not too small.

Consider that we attempt to compare using a block code with block length b and using a trellis code for a error control communication system whose unit is block length $B = hb$ for given integer h. The block error probability for block length B can be calculated briefly from its block error probability for block length b for a block code. However, any good method was not known other than executing simulation of the block error probability for each value of B for a trellis code. The proposed method enables us to estimate the block error probability for any block length B from the evaluated values q_0 and q about a trellis code too.

The step to evaluate the values q_0 and q and the step to calculate the approximation of the block error probability from these values are separated. If a method to calculate an upper-bound or precise value of q_0 or q becomes available, the proposed method can be modified to evaluate the block error probability more efficiently.

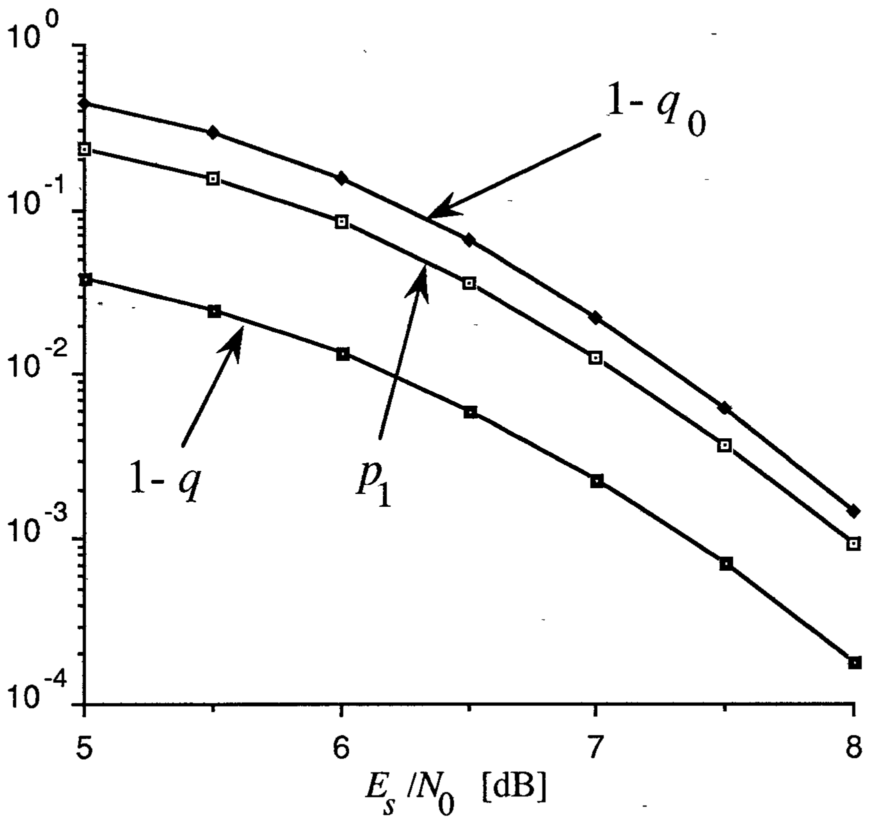

Fig. 2.1 The symbol error probability p_1, the state error probability $1 - q_0$ and the conditional state error probability $1 - q$ for $\nu = 5$, $N = 128$ and $w = 32$.

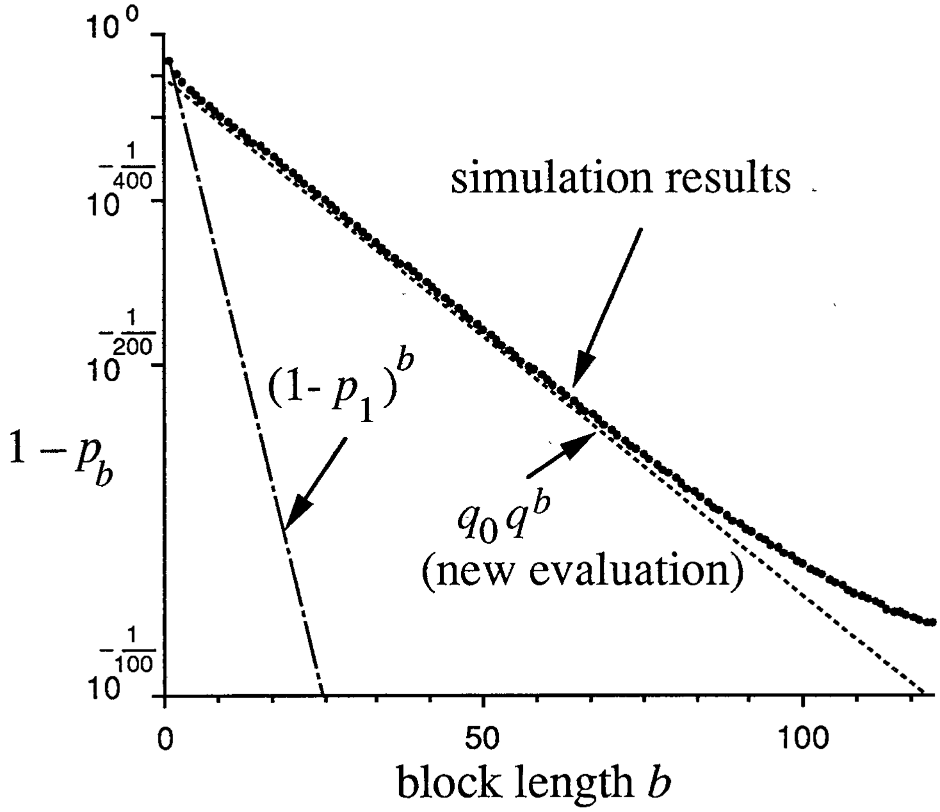

Fig. 2.2 The probability $1-p_b$ of correct block decoding for $\nu = 5$, $N = 128$, $w = 32$ and SNR= 8.0(dB). The simulation results, new evaluation q_0q^b and the value $(1-p_1)^b$ obtained from the symbol error probability p_1.

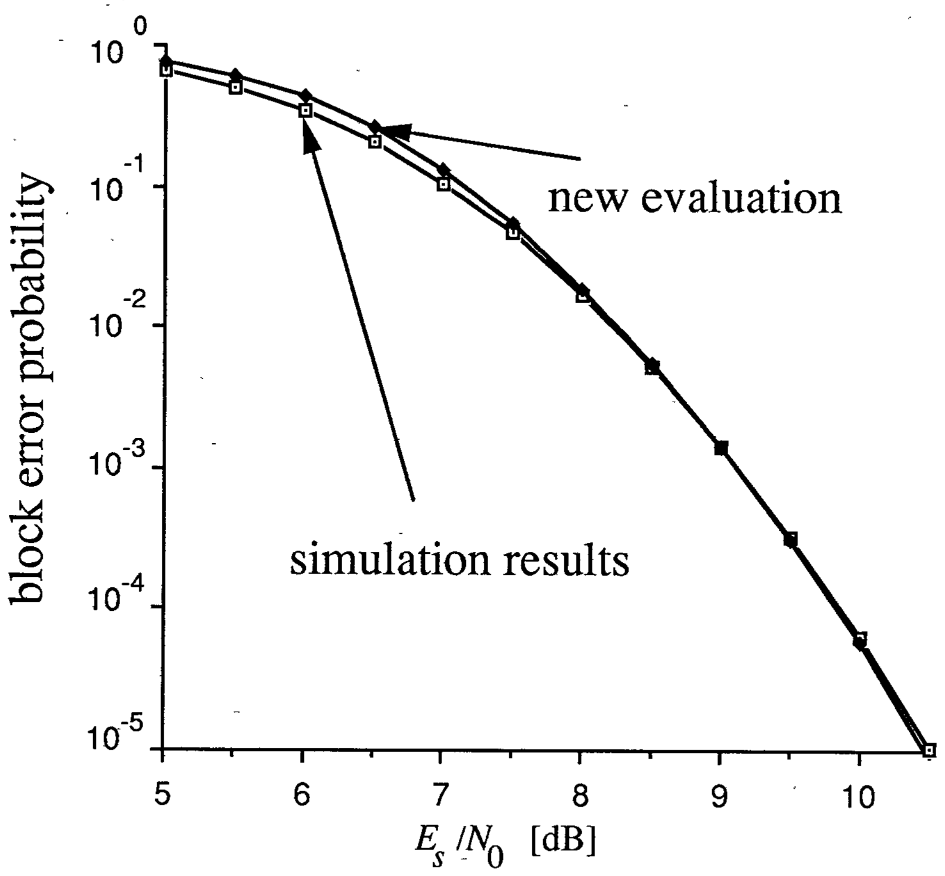

Fig. 2.3 The block error probability for $\nu = 4$, $N = 128$ and $b = w = 16$.

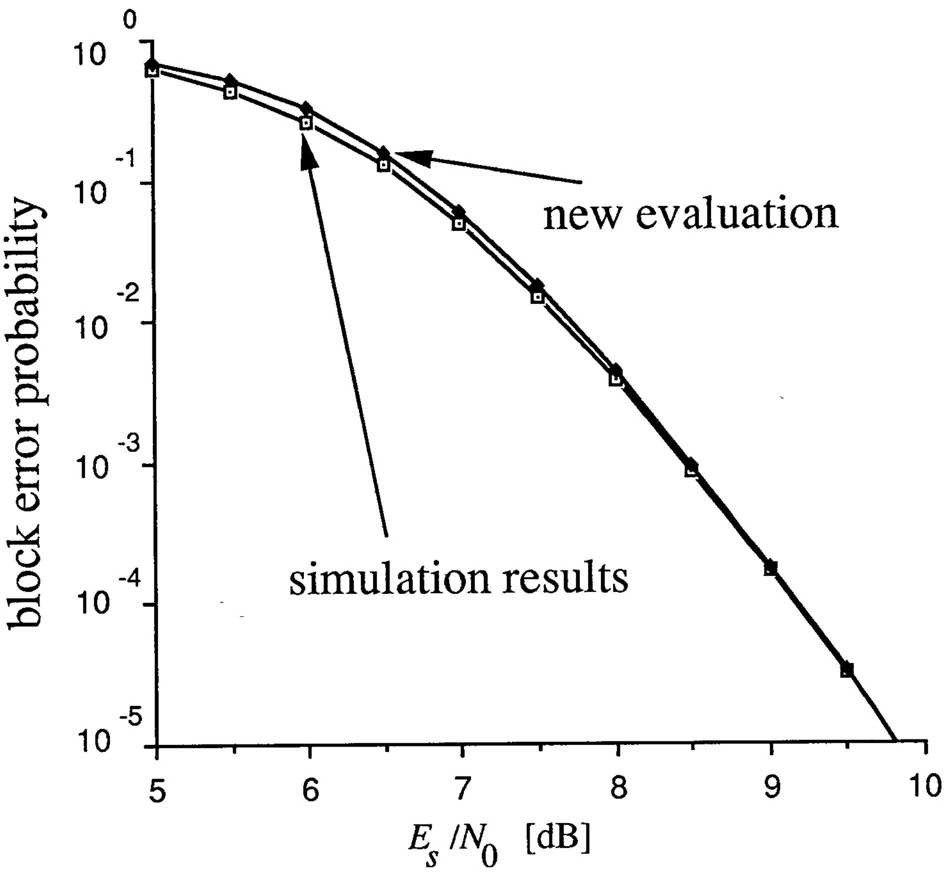

Fig. 2.4 The block error probability for $\nu = 5$, $N = 128$ and $b = w = 32$.

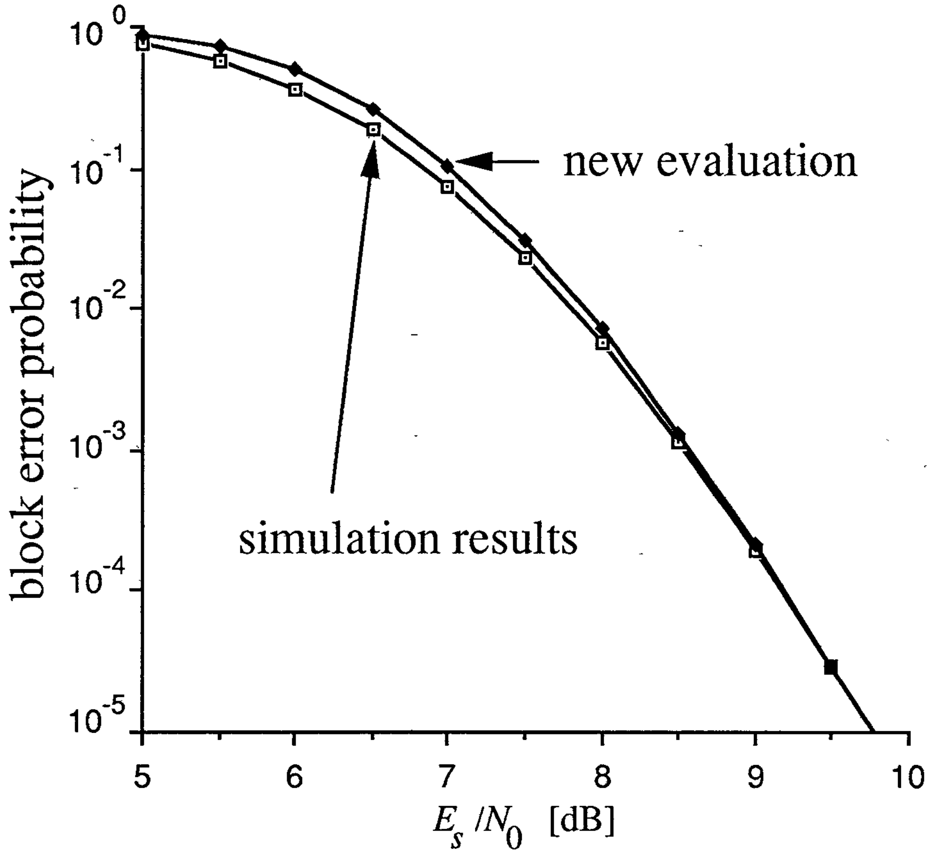

Fig. 2.5 The block error probability for $\nu = 6$, $N = 128$ and $b = w = 64$.

Chapter 3

A Recursive Maximum Likelihood Decoding Algorithm for a Linear Block Code Using a Sectionalized Trellis Diagram and Its Optimization

3.1 Structure of Minimal Trellis Diagram

For simplicity, we only consider the binary linear block codes in this chapter. The extension to nonbinary linear codes is straightforward.

Consider a binary linear (N, K) block code C. By an N-section trellis diagram for C [8, 9], we mean a directed graph, denoted T, such that (1) T has an initial state s_0 and a final state s_f (a state is simply a node in the graph), (2) each branch (an edge in the graph) has a label and two branches diverging from the same state have different labels, and (3) there is a directed path from s_0 to s_f with label sequence $u_1 u_2 \cdots u_N$ if and only if $(u_1, u_2, \dots, u_N)$ is a codeword in C. In the following, a binary sequence of length ℓ is regarded as a binary ℓ-tuple, and vice versa. A trellis diagram for C with the minimum number of states is said to be minimal, and a minimal trellis is unique within graph isomorphism [12]. A subgraph of a trellis diagram is called a sub-trellis diagram.

For a nonnegative integer h not greater than N, let $S_h[C]$ denote the set of states of T just after the h-th bit position, where $S_0[C]$ consists of the initial state s_0 only and $S_N[C]$ consists of the final state s_f only.

For a set of integers $U \triangleq \{h_0, h_1, h_2, \ldots, h_l\}$ with $0 = h_0 < h_1 < \cdots < h_l = N$, a sectionalized minimal trellis diagram for C, denoted $T_C(U)$ or simply $T(U)$, can be obtained from the N-section minimal trellis diagram T by deleting every state in $S_h[C]$ for $h \in \{0, 1, \ldots, N\} - U$ and every branch to or from a deleted state and by writing a branch with label α from a state $s \in S_{h_{j-1}}[C]$ to a state $s' \in S_{h_j}[C]$ for $1 \leq j \leq l$, if and only if there is a path with label α from s to s' in T [10]. $T(\{0, 1, \ldots, N\})$ is T itself. This sectionalized minimal trellis diagram $T(U)$ may have parallel branches between two adjacent states with different labels. Every branch from a state in $S_{h_{j-1}}[C]$ to a state in $S_{h_j}[C]$ for $1 \leq j \leq l$ represents $(h_j - h_{j-1})$ code bits.

Structural properties of a sectionalized trellis diagram of a linear block code have been analyzed in [10, 13]. Here, we briefly review them. Let h and h' be two integers such that $0 \leq h < h' \leq N$. For a binary N-tuple $\mathbf{v} = (v_1, v_2, \ldots, v_N)$, let $p_{h,h'}\mathbf{v}$ denote the binary $(h' - h)$-tuple $(v_{h+1}, v_{h+2}, \ldots, v_{h'})$ and let $p_{h,h'}[C]$ be defined as

$$p_{h,h'}[C] \triangleq \{p_{h,h'}\mathbf{v} : \mathbf{v} \in C\}. \tag{3.1}$$

Let $C_{h,h'}$ be the linear subcode of C consisting of all codewords whose components are all zero except for the $h' - h$ components from the $(h+1)$-th bit position to the h'-th bit position. Let $K_{h,h'}[C]$ be the dimension of $C_{h,h'}$, i.e.,

$$K_{h,h'}[C] = \log_2 |C_{h,h'}|,$$

where for a set S, $|S|$ denotes the number of elements in S. For convenience, $K_{h,h}[C]$ is defined as zero. For simplicity, we write $C^{\mathrm{tr}}_{h,h'}$ for $p_{h,h'}[C_{h,h'}]$, the truncation of $C_{h,h'}$. Then, $C^{\mathrm{tr}}_{h,h'}$ is a linear subcode of $p_{h,h'}[C]$. For integers h, h' and h'' such that $0 \leq h < h' < h'' \leq N$, let $K_{h,h',h''}[C]$ be defined as

$$K_{h,h',h''}[C] \triangleq K_{h,h''}[C] - K_{h,h'}[C] - K_{h',h''}[C]. \tag{3.2}$$

For simplicity, we write $K_h[C]$ for $K_{0,h,N}[C]$.

For a linear code A and its linear subcode B, let A/B denote the set of cosets in A with respect to B. For two states $s \in S_h[C]$ and $s' \in S_{h'}[C]$ in the minimal trellis diagram T of C, let $L(s, s')$ denote the set of all label sequences for paths from s to s'.

Then,

$$\{L(s,s') : s \in S_h[C], s' \in S_{h'}[C]\} = p_{h,h'}[C]/C^{\text{tr}}_{h,h'}, \tag{3.3}$$

and it has been shown in [9, Appendix A] that

$$|S_h[C]| = 2^{K_h[C]}. \tag{3.4}$$

For a state $s' \in S_{h'}[C]$, the number of those states s in $S_h[C]$ that $L(s,s') \neq \emptyset$ is given by $2^{Q_{h,h'}[C]}$ where

$$\begin{aligned} Q_{h,h'}[C] &\triangleq \log_2 |p_{0,h}[C_{0,h'}]| - \log_2 |C^{\text{tr}}_{0,h}| \\ &= K_{0,h'}[C] - K_{h,h'}[C] - K_{0,h}[C] \\ &= K_{0,h,h'}[C]. \end{aligned} \tag{3.5}$$

For a sectionalized trellis diagram $T(U)$ with $h, h' \in U$ and $h < h'$, consider the sub-trellis diagram obtained by truncating $T(U)$ except for the h-th to the h'-th bit positions. This truncated sub-trellis diagram, denoted $p_{h,h'}[T(U)]$, consists of parallel and structurally identical (except branch labels) sub-trellis diagrams without cross connections, called parallel components. The number of parallel components is given by $2^{K_h[C]-Q_{h,h'}[C]}$[10].

Let $C_{\overline{h,h'}}$ denote the linear subcode of C consisting of all codewords whose components from the $(h+1)$-th bit position to the h'-th bit position are all zero, and let $K_{\overline{h,h'}}[C]$ denote $\log_2 |C_{\overline{h,h'}}|$, the dimension of $C_{\overline{h,h'}}$. Then, the number of different label sequences in the truncated trellis diagram $p_{h,h'}[T]$ is given by [9, Appendix A]

$$|p_{h,h'}[C]| = 2^{K-K_{\overline{h,h'}}[C]}. \tag{3.6}$$

The number of different $L(s,s')$'s with $s \in S_h$ and $s' \in S_{h'}$ is given by

$$|p_{h,h'}[C]| 2^{-K_{h,h'}} = 2^{K-K_{h,h'}[C]-K_{\overline{h,h'}}[C]}. \tag{3.7}$$

The number of state pairs (s,s') such that $L(s,s') \neq \emptyset$ is given by $2^{K_{h'}[C]+K_{0,h,h'}[C]}$. Any coset in $p_{h,h'}[C]/C^{\text{tr}}_{h,h'}$ appears $2^{K_h[C]+K_{\overline{h,h'}}[C]-K}$ times as the label set of parallel branches between state pairs in the section. For example, consider the 4-section minimal trellis diagram $T(\{0,16,32,48,64\})$ for the third order Reed-Muller code of length

64. In the second (or third) section, each coset in $p_{16,32}[C]/C^{\text{tr}}_{16,32}$ (or $p_{32,48}[C]/C^{\text{tr}}_{32,48}$) appears 64 times as the label set of parallel branches.

Let $K''_{h,h'}[C]$ be defined as

$$K''_{h,h'}[C] \triangleq \log_2 |p_{0,h}[C_{0,h'}] \cap p_{0,h}[C_{\overline{h,h'}}]|. \tag{3.8}$$

Define $\lambda_{h,h'}[C]$ as

$$\lambda_{h,h'}[C] \triangleq K_{\overline{h,h'}}[C] - K''_{h,h'}[C] - K_{h',N}[C]. \tag{3.9}$$

Then, the set of parallel components can be partitioned into $2^{K_h[C]-Q_{h,h'}[C]-\lambda_{h,h'}[C]}$ blocks of size $2^{\lambda_{h,h'}[C]}$ in such a way that (1) two parallel components in the same block are identical up to the path labeling and (2) if there is a common label sequence in two parallel components, they are in the same block[12].

3.2 New Maximum Likelihood Decoding

3.2.1 Decoding Procedure

By using an l-section minimal trellis diagram $T(\{h_0, h_1, \ldots, h_l\})$ for C, a maximum likelihood decoding for C can be done as follows: Choose an integer m with $0 \le m \le l$. For every state in $S_{h_j}[C]$ with $1 \le j \le m$, find the survivor path from the initial state s_0 to the state with the largest metric together with its metric from the metrics of states in $S_{h_{j-1}}[C]$ in the following way: For every parallel branch set from a state in $S_{h_{j-1}}[C]$ to one in $S_{h_j}[C]$, find the branch with the largest metric. Then, find the survivor path by using the largest branch metrics of the parallel branches between every state pair in each parallel component. Similarly, for every state in $S_{h_j}[C]$ with $l-1 \ge j \ge m$, find the survivor path from the final state s_f to the state with the largest metric together with its metric from the metrics of states in $S_{h_{j+1}}[C]$. Then, for every state in $S_{h_m}[C]$, by concatenating survivor paths to the state, we obtain $|S_{h_m}[C]|$ paths from the initial state to the final state. Finally, find the path with the largest metric among these paths. This method is called the double-ended trellis decoding [15], and is denoted DETD($T(\{h_0, h_1, \ldots, h_l\}), h_m$). When m is chosen to l, the method corresponds to one-way (left to right) Viterbi decoding for $T(\{h_0, h_1, \ldots, h_l\})$.

As shown in Section 2 (refer to Eq.(3.3)), the label set of parallel branches between a state in $S_{h_{j-1}}[C]$ and one in $S_{h_j}[C]$ is a coset in $p_{h_{j-1},h_j}[C]/C^{\mathrm{tr}}_{h_{j-1},h_j}$. In case that the same label set appears many times in a section as stated in Section 3.1, the decoding complexity can be reduced by constructing a table for the metrics of parallel branches, denoted $\mathrm{PBT}_{h_{j-1},h_j}$, which, for each coset in $p_{h_{j-1},h_j}[C]/C^{\mathrm{tr}}_{h_{j-1},h_j}$, stores the largest metric in the coset and a label with the metric.

Now, we devise the procedure to construct the table $\mathrm{PBT}_{x,y}$ for integers x, y with $0 \leq x < y \leq N$. Since the set of cosets $p_{0,N}[C]/C_{0,N} = C/C$ consists of C only, the table $\mathrm{PBT}_{0,N}$ stores the decoded codeword as well as its metric. Hereafter, the bit positions for $p_{x,y}[C]$ and its subcodes are numbered $x, x+1, x+2, \ldots, y$ instead of $0, 1, 2, \ldots, y-x$.

(A) A straightforward method to construct the table $\mathrm{PBT}_{x,y}$ is to compute the metrics for all vectors in $p_{x,y}[C]$, and then find the vector with the largest metric for every coset by comparing the metrics of vectors in the coset. In general, this method is efficient only when $y-x$ is small.

(B) When $y-x$ is large, we can reduce the computational complexity by using a trellis diagram for each coset.

(B-a) Let l' be an integer with $2 \leq l' \leq y-x$ and $U' = \{h'_0, h'_1, \ldots, h'_{l'}\}$ be a set of integers with $x = h'_0 < h'_1 < \cdots < h'_{l'} = y$. Let $T_{C^{\mathrm{tr}}_{x,y}}(U')$ be an l'-section trellis diagram for $C^{\mathrm{tr}}_{x,y}$. For a coset $D = \mathbf{u} + C^{\mathrm{tr}}_{x,y}$ with $\mathbf{u} \in p_{x,y}[C]$, an l'-section trellis diagram for D, denoted $T_D(U')$ can be obtained from $T_{C^{\mathrm{tr}}_{x,y}}(U')$ by adding $p_{h'_{j-1},h'_j}\mathbf{u}$ to the label of each branch from a state in $S_{h'_{j-1}}[C^{\mathrm{tr}}_{x,y}]$ to a state in $S_{h'_j}[C^{\mathrm{tr}}_{x,y}]$ for $1 \leq j \leq l'$. $T_D(U')$ is minimal if and only if $T_{C^{\mathrm{tr}}_{x,y}}(U')$ is minimal. The set of states of $T_D(U')$ just after the h'_j-th bit position is also denoted $S_{h'_j}[D]$. The label set of parallel branches between a state in $S_{h'_{j-1}}[D]$ and one in $S_{h'_j}[D]$ in $T_D(U')$ is also a coset in $p_{h'_{j-1},h'_j}[C]/C^{\mathrm{tr}}_{h'_{j-1},h'_j}$. The tables $\mathrm{PBT}_{h'_{j-1},h'_j}$ with $1 \leq j \leq l'$ are used for the decoding. These tables are constructed recursively.

(B-b) Instead of performing one-way (left to right) Viterbi decoding for each coset separately, we can execute the one-way (left to right) Viterbi decoding with the following one trellis diagram for $p_{x,y}[C]/C^{\mathrm{tr}}_{x,y}$: In the trellis diagram, there is a common initial state $s_{x,y,0}$ and for each coset $D \in p_{x,y}[C]/C^{\mathrm{tr}}_{x,y}$, there is a final state $s_{x,y,f}[D]$ to which the set of label sequences of paths from $s_{x,y,0}$ is D. The trellis diagram can be constructed as follows: Choose a basis $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_{K-K_{\overline{x,y}}[C]}$ of $p_{x,y}[C]$ such that

$$\mathbf{v}_{K-K_{\overline{x,y}}[C]-K_{x,y}[C]+1}, \mathbf{v}_{K-K_{\overline{x,y}}[C]-K_{x,y}[C]+2}, \ldots, \mathbf{v}_{K-K_{\overline{x,y}}[C]} \in C^{\mathrm{tr}}_{x,y}.$$

Let

$$\mathbf{u}_i \triangleq \begin{cases} \mathbf{v}_i \cdot \mathbf{e}_{i,K-K_{\overline{x,y}}[C]-K_{x,y}[C]}, & \text{for } 1 \le i \le K - K_{\overline{x,y}}[C] - K_{x,y}[C] \\ \mathbf{v}_i \cdot \mathbf{0}_{K-K_{\overline{x,y}}[C]-K_{x,y}[C]}, & \text{for } K - K_{\overline{x,y}}[C] - K_{x,y}[C] < i \le K - K_{\overline{x,y}}[C], \end{cases}$$

where

$$\begin{aligned} \mathbf{e}_{i,n} &= (\overbrace{\underbrace{0,0,\ldots,0,1}_{i},0,0,\ldots,0}^{n}), \\ \mathbf{0}_n &= (\overbrace{0,0,\ldots,0}^{n}), \end{aligned}$$

and '$\cdot$' denotes the concatenation of two vectors. Let $C_{\mathrm{L}}(x,y)$ be the binary linear code of length $n_{x,y} \triangleq y-x+K-K_{\overline{x,y}}[C]-K_{x,y}[C]$ spanned by $\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_{K-K_{\overline{x,y}}[C]}$. The bit positions for $C_{\mathrm{L}}(x,y)$ are numbered $x, x+1, \ldots, y, \ldots, x+n_{x,y}$ instead of $0, 1, \ldots, y-x, \ldots, n_{x,y}$. Then, the sub-trellis diagram (sub-graph) obtained by truncating an $(l'+1)$-section minimal trellis diagram of $C_{\mathrm{L}}(x,y)$, $T_{C_{\mathrm{L}}(x,y)}(U' \cup \{x+n_{x,y}\})$, except for the x-th to the y-th bit positions is the desired l'-section trellis diagram. This truncated trellis diagram $p_{x,y}[T_{C_{\mathrm{L}}(x,y)}(U' \cup \{x+n_{x,y}\})]$ is denoted $p_{x,y}[T_{C_{\mathrm{L}}(x,y)}(U')]$ for simplicity.

Define $z = h'_{l'-1}$. Consider the one-way (left to right) Viterbi decoding using the trellis diagram $T_{C_{\mathrm{L}}(x,y)}(\{x,z,y\})$ together with the two tables $\mathrm{PBT}_{x,y}$ and $\mathrm{PBT}_{z,y}$. We compare the decoding complexity of this method with that using $p_{x,y}[T_{C_{\mathrm{L}}(x,y)}(U')]$. It follows from (3.3), the definition of $C_{\mathrm{L}}(x,y)$ that for any u and z with $x \le u < z \le y$,

$$\begin{aligned} \{L(s,s') : s \in S_u[C_{\mathrm{L}}(x,y)]\}, s' \in S_z[C_{\mathrm{L}}(x,y)]\} &= p_{u,z}[C_{\mathrm{L}}(x,y)]/(C_{\mathrm{L}}(x,y))^{\mathrm{tr}}_{u,z} \\ &= p_{u,z}[C]/C^{\mathrm{tr}}_{u,z}. \end{aligned} \qquad (3.10)$$

From this fact and that the minimality of $p_{x,y}[T_{C_{\mathrm{L}}(x,y)}(U')]$, we have that $p_{x,z}[p_{x,y}[T_{C_{\mathrm{L}}(x,y)}(U')]] = p_{x,z}[T_{C_{\mathrm{L}}(x,y)}(U')]$ is the $(l'-1)$-section minimal trellis diagram for $p_{x,z}[C]/C^{\mathrm{tr}}_{x,z}$, $p_{x,z}[T_{C_{\mathrm{L}}(x,z)}(\{h'_0, h'_1, \ldots, z\})]$. Therefore, the complexity of the decoding with $p_{x,y}[T_{C_{\mathrm{L}}(x,y)}(\{x,z,y\})]$ is not worse than that with $p_{x,y}[T_{C_{\mathrm{L}}(x,y)}(U')]$, and it is enough to consider the two-section trellis diagram $T_{C_{\mathrm{L}}(x,y)}(\{x,z,y\})$ only at each recursion level, for finding a decoding procedure with the smallest complexity, as far as the complexity depends on the used trellis diagram only.

(B-c) The above method using the two-section trellis diagram $T_{C_{\mathrm{L}}(x,y)}(\{x,z,y\})$ can be viewed as an algorithm to solve the following problem: how to obtain $\mathrm{PBT}_{x,y}$ from $\mathrm{PBT}_{x,z}$ and $\mathrm{PBT}_{z,y}$ efficiently, where $x < z < y$. Since $|C^{\mathrm{tr}}_{x,y}/(C^{\mathrm{tr}}_{x,z} + C^{\mathrm{tr}}_{z,y})| = 2^{K_{x,y}[C]-K_{x,z}[C]-K_{z,y}[C]} = 2^{K_{x,z,y}[C]}$, for $D \in p_{x,y}[C]/C^{\mathrm{tr}}_{x,y}$, we can find the set of $2^{K_{x,z,y}[C]}$ coset pairs in $p_{x,z}[C]/C^{\mathrm{tr}}_{x,z} \times p_{z,y}[C]/C^{\mathrm{tr}}_{z,y}$, denoted $\mathrm{CP}[D]$, such that

$$D = \bigcup_{(D_{\mathrm{L}},D_{\mathrm{R}})\in\mathrm{CP}[D]} D_{\mathrm{L}} \cdot D_{\mathrm{R}} \triangleq \bigcup_{(D_{\mathrm{L}},D_{\mathrm{R}})\in\mathrm{CP}[D]} \{\mathbf{u}\cdot\mathbf{v} : \mathbf{u} \in D_{\mathrm{L}}, \mathbf{v} \in D_{\mathrm{R}}\}. \tag{3.11}$$

For a coset X, let $m[X]$ denote the largest metrics of X. From (3.11),

$$m[D] = \max_{(D_{\mathrm{L}},D_{\mathrm{R}})\in\mathrm{CP}[D]} \{m[D_{\mathrm{L}}] + m[D_{\mathrm{R}}]\}. \tag{3.12}$$

Let $N(x,y;z)$ be the number of addition-equivalent operations to find $m[D]$, when the tables $\mathrm{PBT}_{x,z}$ and $\mathrm{PBT}_{z,y}$ are given. $N(x,y;z)$ depends on the used algorithm. If we simply follow the right-hand side of (3.12), the number of additions is $|p_{x,y}[C]/C^{\mathrm{tr}}_{x,y}| \times 2^{K_{x,z,y}[C]} = 2^{K-K_{\overline{x,y}}[C]-K_{x,y}[C]+K_{x,z,y}[C]}$, that of comparisons is $2^{K-K_{\overline{x,y}}[C]-K_{x,y}[C]}(2^{K_{x,z,y}[C]} - 1)$ and the sum, denoted $N^{(0)}(x,y;z)$, is given by

$$N^{(0)}(x,y;z) = 2^{K-K_{\overline{x,y}}[C]-K_{x,y}[C]}(2^{K_{x,z,y}[C]+1} - 1). \tag{3.13}$$

We review the method using the two-section trellis diagram $T_{C_{\mathrm{L}}(x,y)}(\{x,z,y\})$. Let s' be the state in $S_y[C_{\mathrm{L}}(x,y)]$ corresponding to D, that is, $L(s_{x,y,0}, s') = D$. Then, from (3.10), $\mathrm{CP}[D]$ is given by

$$\{(L(s_{x,y,0}, s), L(s,s')) : s \in S_z[C_{\mathrm{L}}(x,y)] \text{ and } L(s,s') \neq \emptyset\}.$$

Therefore, a standard processing of the trellis diagram $T_{C_{\mathrm{L}}(x,y)}(\{x,z,y\})$ to find $m[D]$ is essentially the same as the method which simply follows the right-hand side of (3.12).

We can also use a trellis diagram for $p_{x,y}[C]/C^{\mathrm{tr}}_{x,y}$ such that there is a common final state $s_{x,y,f}$ and for each coset $D \in p_{x,y}[C]/C^{\mathrm{tr}}_{x,y}$, there is a initial state $s_{x,y,0}[D]$. We can also show that this method is also the essentially same as the above method.

There is another implementation [13, 17] based on the detailed structure of each section that is a union of the same simple regular sub-trellis diagrams. The implementation yields a smaller $N(x, y; z)$ for the worst case than $N^{(0)}(x, y; z)$ and its average computational complexity is considerably smaller than the worst case one.

By summarizing the above, we devise the following procedure $NewMLD(x, y)$ to construct the table $\mathrm{PBT}_{x,y}$ for a received word.

[Procedure $NewMLD(x, y)$**]**

Input: Integers x, y.

Output: $\mathrm{PBT}_{x,y}$.

(Step 1) Choose an integer z. If $x + 1 = y$, let $z = -1$. Otherwise, choose an integer z with $z = -1$ or $x < z < y$.

(Step 2) Execute one of the following (Case A) or (Case B) depending on the value of z. When $z = -1$, execute (Case A). Otherwise, execute (Case B). In both cases, we obtain the table $\mathrm{PBT}_{x,y}$ by storing the results.

<u>(Case A: $z = -1$)</u> Compute the metrics for all vectors in $p_{x,y}[C]$. The method of this metric computation is discussed in 3.2. For every coset in $p_{x,y}[C]/C^{\mathrm{tr}}_{x,y}$, find the vector with the largest metric by comparing the metrics of vectors in the coset.

<u>(Case B: $x < z < y$)</u>

(Step B-1) Construct the tables $\mathrm{PBT}_{x,z}$ and $\mathrm{PBT}_{z,y}$ by executing $NewMLD(x, z)$ and $NewMLD(z, y)$, respectively.

(Step B-2) For every coset $D \in p_{x,y}[C]/C^{\mathrm{tr}}_{x,y}$, find the vector with the largest metric in the coset by using (3.12). $\Delta\Delta$

3.2.2 Complexity Analysis

In the following, we evaluate the computational complexity, denoted $\psi(x, y)$, of the above procedure $NewMLD(x, y)$ in terms of the number of addition-equivalent operations. This number includes those for all the recursive executions of $NewMLD$. The chosen integer z in (Step 1) for $NewMLD(x, y)$ is denoted by $z_{x,y}$. When it is necessary to specify the value of parameter $z_{x,y}$, notation $\psi(x, y; z_{x,y})$ is used.

(A) First, we derive a formula for $\psi(x, y; -1)$. We assume that the metrics of the individual bits are given.

1) Consider the case where $x+1=y$. Assume for the simplicity that the minimum distance of C is greater than 1. Then, no operation is needed and hence

$$\psi(x, x+1; -1) = 0, \tag{3.14}$$

since the metrics are those of the individual bits and $|C^{\mathrm{tr}}_{x,x+1}| = 1$.

2) When $x+1<y$, the number $\psi(x,y;-1)$ depends on the implementation.

2.0) For example, if the branch metrics for all branch labels are computed independently, the value $\psi(x,y;-1)$ for this implementation, denoted $\psi^{(0)}(x,y;-1)$, is given by

$$\begin{aligned}\psi^{(0)}(x,y;-1) &= (y-x-1)|p_{x,y}[C]| + |p_{x,y}[C]/C^{\mathrm{tr}}_{x,y}|(|C^{\mathrm{tr}}_{x,y}|-1)\\ &= (y-x-1)2^{K-K_{\overline{x,y}}[C]} + 2^{K-K_{\overline{x,y}}[C]-K_{x,y}[C]}(2^{K_{x,y}[C]}-1),\\ &\qquad \text{for } x+1<y.\end{aligned} \tag{3.15}$$

The first term is the number of additions to compute all the metrics for the label set $p_{x,y}[C]$, and the second term is the number of comparisons for finding the vectors with the largest metrics by comparing the metrics of vectors in a coset for all the cosets in $p_{x,y}[C]/C^{\mathrm{tr}}_{x,y}$.

2.1) We may also use the following method to compute the branch metrics for labels in $p_{x,y}[C]$. Consider the case $|p_{x,y}[C]| = 2^{y-x}$. First, compute the metric of a label with $y-x-1$ additions. Then, we compute the metric of each branch label by one addition operation in the order of the Gray code [16]. This method can also be used for the case where $|p_{x,y}[C]| < 2^{y-x}$ by discarding the metrics for the labels which are not in $p_{x,y}[C]$.

Let $\psi^{(1)}(x,y;-1)$ be the value of $\psi(x,y;-1)$ for this implementation. Then, we have that

$$\psi^{(1)}(x,y;-1) = 2^{y-x}+y-x-2+2^{K-K_{\overline{x,y}}[C]-K_{x,y}[C]}(2^{K_{x,y}[C]}-1), \quad \text{for } x+1<y. \tag{3.16}$$

(B) Next, consider the case with $z_{x,y} \neq -1$. We have the following recursive formula:

$$\psi(x,y;z_{x,y}) = \psi(x,z_{x,y}) + \psi(z_{x,y},y) + N(x,y;z_{x,y}), \qquad \text{for } x<z_{x,y}<y. \tag{3.17}$$

Given formulas for $\psi(x,y;-1)$, $N(x,y;z)$ and the choices of z, we can compute $\psi(0,N)$ easily by using (3.17).

3.2.3 Optimum Sectionalization

We need choose z in (Step 1) properly to obtain a decoder with the small number of operations. A sectionalization of the trellis diagrams (choices of z) which gives the smallest $\psi(0, N)$ for a given code is called the optimum sectionalization for the code.

Assume that the formulas for $\psi(x, y; -1)$ and $N(x, y; z)$ are given explicitly. In this section, we present the method for finding the smallest computational complexity for $NewMLD(0, N)$ as well as the optimum sectionalization.

Define $\psi_{\min}(x, y)$ as follows:

(1) If $x + 1 = y$,

$$\psi_{\min}(x, x+1) = \psi(x, x+1; -1) = 0. \tag{3.18}$$

(2) Otherwise

$$\psi_{\min}(x, y) = \min\{\psi(x, y; -1), \min_{x<z<y}\{\psi_{\min}(x, z) + \psi_{\min}(z, y) + N(x, y; z)\}\}. \tag{3.19}$$

$\psi_{\min}(x, y)$ is the smallest number of operations to construct the table $\mathrm{PBT}_{x,y}$, and therefore $\psi_{\min}(0, N)$ is that for decoding a received word.

By using (3.18) and (3.19) together with formulas for $\psi(x, y; -1)$ and $N(x, y; z)$, we can compute $\psi_{\min}(x, y)$ for every (x, y) with $0 \le x < y \le N$, efficiently. By storing the information when the minimum value occurs in the right-hand side of (3.19), it is easy to find an optimum sectionalization.

Finally, we present two theorems on the choice of (Case A) or (Case B).

Theorem 1: Consider a binary linear code C of length N such that a generator matrix of C does not contain any all-zero column. Suppose that $\psi^{(0)}(x, y; -1)$ is used for $\psi(x, y; -1)$ and $N^{(0)}(x, y; z)$ is used for $N(x, y; z)$.

(1) The complexity for (Case B) is smaller than that for (Case A) except for $y - x \le 2$. When $y - x = 2$, they are the same, if the minimum distance of C is greater than 1.

(2) If $y - x > 2$, $|p_{x,y}[C]| = 2^{y-x}$ and $C^{\mathrm{tr}}_{x,y} = \{\mathbf{0}_{y-x}\}$ or $\{\mathbf{0}_{y-x}, (1, *, \ldots, *, 1)\}$, where $(1, *, \ldots, *, 1)$ denotes a vector $\mathbf{v}$ such that $p_{x,x+1}\mathbf{v} = p_{y-1,y}\mathbf{v} = 1$, then the right-hand side of (3.19) takes its minimum for both $z = \lfloor (x+y)/2 \rfloor$ and $z = \lceil (x+y)/2 \rceil$.

(Proof) See Appendix. $\Delta\Delta$

Theorem 2: Suppose that $\psi^{(1)}(x, y; -1)$ is used for $\psi(x, y; -1)$ and $N^{(0)}(x, y; z)$ is used for $N(x, y; z)$.

(1) If $|p_{x,y}[C]| = 2^{y-x}$ and $C^{\text{tr}}_{x,y} = \{\mathbf{0}_{y-x}\}$ or $\{\mathbf{0}_{y-x}, (1, *, \ldots, *, 1)\}$, then the complexity for (Case A) is smaller than that for (Case B) except for $y - x \leq 2$. When $y - x = 2$, they are the same.

(2) If the condition in (1) does not hold, then the complexity for (Case B) is smaller than that for (Case A), and if $|p_{x,y}[C]| < 2^{y-x}$ and $C^{\text{tr}}_{x,y} = \{\mathbf{0}_{y-x}\}$ or $\{\mathbf{0}_{y-x}, (1, *, \ldots, *, 1)\}$, then the right-hand side of (3.19) takes its minimum for both $z = \lfloor (x + y)/2 \rfloor$ and $z = \lceil (x + y)/2 \rceil$.

(Proof) See Appendix. $\Delta\Delta$

3.3 Examples

Let RM(64, 22), RM(64, 42) and RM(64, 57) be the second, third and fourth order Reed-Muller code of length 64, respectively. Let exBCH(63, k) denote the extended code of the binary primitive (63, k) BCH code. We computed the decoding complexity of RM(64, 22), RM(64, 42), RM(64, 57) and permuted codes of exBCH(63, k) with $16 \leq k \leq 51$.

A cyclic (or shortened cyclic) code or its extended code has the worst trellis state complexity among linear block codes of the same length and dimension [14]. In order to obtain a trellis diagram with a smaller number of states for these codes, the order of symbol positions must be permuted. The natural symbol ordering of a Reed-Muller code is optimal for the state complexity [18]. For extended BCH codes, we consider the following permutations [14, 19]: Let α be a primitive element of GF(2^6) and let $\{\beta_1, \ldots, \beta_6\}$ be a basis of GF(2^6) over GF(2). For a positive integer i less than 2^6, let α^{i-1} be expressed as

$$\alpha^{i-1} = \sum_{j=1}^{6} b_{i,j}\beta_j, \tag{3.20}$$

with $b_{i,j} \in$ GF(2). For $i = 0$, let $b_{0,j} \triangleq 0$ for $1 \leq j \leq 6$. Let π denote the following permutations on $\{1, 2, \ldots, 2^6\}$,

$$\pi(i) \triangleq 1 + \sum_{j=1}^{6} b_{i-1,j} 2^{6-j}, \quad \text{for } 1 \leq i \leq 2^6. \tag{3.21}$$

In general, the state complexity of the minimal trellis diagram for a permuted code depends on the choice of the basis $\{\beta_1, \beta_2, \ldots, \beta_6\}$. We consider the following three bases for codes of length 64: (1) Basis A is a polynomial basis, $\{1, \alpha, \alpha^2, \ldots, \alpha^5\}$. (2) Basis B is $\{1, \alpha, \alpha^2, \alpha^{21}, \alpha^{22}, \alpha^{23}\}$, which is obtained by combining a basis GF(2^6) over GF(2^2), $\{1, \alpha, \alpha^2\}$. and a basis GF(2^2) over GF(2), $\{1, \alpha^{21}\}$. (3) Basis C is $\{1, \alpha, \alpha^9, \alpha^{10}, \alpha^{18}, \alpha^{19}\}$, which is obtained by combining a basis of GF(2^6) over GF(2^3), $\{1, \alpha\}$, and a basis of GF(2^3) over GF(2), $\{1, \alpha^9, \alpha^{18}\}$.

In Table 3.1, the number of addition-equivalent operations are listed for the above example codes. We use $N^{(0)}(x, y; z)$ for $N(x, y, z)$. The column labeled 64-section gives the numbers of operations required in conventional Viterbi decoding with 64-section trellis diagram, which is equivalent to the proposed method with the sectionalization $z_{0,i} = i - 1$, with $2 \leq i \leq 64$. The column labeled Optimum ($\psi^{(0)}_{\min}$) gives the number of operations for the optimum sectionalization when $\psi^{(0)}(x, y; -1)$ is used for $\psi(x, y; -1)$. The column labeled Optimum ($\psi^{(1)}_{\min}$) is that when $\psi^{(1)}(x, y; -1)$ is used for $\psi(x, y; -1)$. Both optimum values for each code are almost the same. For every BCH code, the two symbol permutations other than one indicated in Table 3.1 among the three symbol permutations above mentioned, the optimum values are larger than those in the table. It is worth while to mention that the numbers of operations required in conventional Viterbi decoding with 64-section trellis diagram for exBCH(63,51) takes its minimum value 340,217 for Basis B among the three permutations. This shows that a good bit ordering for the N-section trellis diagram is not always good for our decoding procedure. The last column shows the numbers of addition-equivalent operations given by Vardy and Be'ery [16]. Note that the bit permutations for BCH codes and the decoding algorithm in [16] are different from those in this chapter.

An optimum sectionalization of several example codes when $\psi^{(1)}(x, y; -1)$ is used for $\psi(x, y; -1)$ are presented in Table 3.2. The optimum sectionalization is chosen in the following manner: If $\psi_{\min}(x, y) = \psi(x, y; -1)$, then $z_{x,y} = -1$. Otherwise, $z_{x,y}$ is the largest integer z such that $\psi_{\min}(x, y) = \psi_{\min}(x, z) + \psi_{\min}(z, y) + N(x, y; z)$. In the table, we present the values of $z_{x,y}$ for every (x, y) such that $NewMLD(x, y)$ is invoked during the execution of $NewMLD$ except for (x, y) with $z_{x,y} = -1$. The sectionalization (the structure of recursive executions of $NewMLD$ during the execution of $NewMLD(0, N)$) can be expressed as the following ordered binary tree R: A node of R represents each invocation of $NewMLD$. The node corresponds to the invocation $NewMLD(x, y)$ is labeled (x, y). The label of the root of R is $(0, N)$. The node labeled (x, y) has the left child labeled $(x, z_{x,y})$ and the right child labeled $(z_{x,y}, y)$, if and only if $z_{x,y} \neq -1$. This represents that $NewMLD(x, y)$ calls $NewMLD(x, z_{x,y})$ and $NewMLD(z_{x,y}, y)$. A node (x, y) in R is a leaf if $z_{x,y} = -1$. The sectionalization R can be reconstructed from the values of $z_{x,y}$ in the following recursive way: Construct a binary tree R with the root only. The root of R is labeled $(0, N)$. For a node in R labeled (x, y), if $z_{x,y} \neq -1$, then add the child nodes of the node, the left child labeled $(x, z_{x,y})$ and the right child labeled $(z_{x,y}, y)$. For RM(64, 42), the tree R is depicted in Figure 3.1.

In the procedure $NewMLD$, we have considered only two-section trellis diagrams. As discussed in (B) of Section 3, the following reformulation of $NewMLD$ is essentially identical to the original procedure. In this reformulation, l-section trellis diagrams with $l \geq 2$ are used.

(Step 1') Choose an integer l with $l \geq 1$ and a set of $l+1$ integers $U = \{h_0, h_1, \ldots, h_l\}$ with $x = h_0 < h_1 < \cdots < h_l = y$.

(Step 2') Execute one of the following (Case A) or (Case B) depending on U. When $U = \{x, y\}$, execute (Case A). Otherwise, execute (Case B).

(Case A: $U_{x,y} = \{x, y\}$) Execute the (Case A) in the original $NewMLD$.

(Case B: $U_{x,y} \neq \{x, y\}$)

(Step B-1') Construct the tables $\mathrm{PBT}_{h_{i-1},h_i}$ by executing the variation of $NewMLD(h_{i-1}, h_i)$ recursively for $1 \leq i \leq l$.

(Step B-2') Choose an integer z with $z \in U$. For every coset $D \in p_{x,y}[C]/C^{\mathrm{tr}}_{x,y}$, find the vector with the largest metric in the coset by $\mathrm{DETD}(T_{\mathrm{CL}(x,y)}(U), z)$. $\Delta\Delta$

For a given sectionalization (binary tree) R for the original $NewMLD$, we can choose the parameters of the reformulated procedure as follows:

(1) For a given sectionalization (binary tree) R, define $U_{0,N} \triangleq \{0, N\} \cup \{i|$ a node labeled $(0, i)$ is in $R\} \cup \{i|$ a node labeled (i, N) is in $R\}$(refer to Fig. 3.1). Then, execute the above variation of $NewMLD$ with $x = 0$, $y = N$, $U = U_{0,N}$ and $z = z_{0,N}$.

Table 3.1 The number of addition-equivalent operations for the maximum likelihood decoding.

Code	64-section	Opt. ($\psi^{(0)}_{\min}$)	Opt. ($\psi^{(1)}_{\min}$)	Vardy-Be'ery[16]
RM(64,22)	425,209	78,209	78,119	—
RM(64,42)	773,881	326,017	325,717	—
RM(64,57)	7,529	5,281	5,209	7.52×10^3
exBCH(63,16), Basis B	764,153	120,193	120,103	1.6929×10^5
exBCH(63,18), Basis B	2,865,401	468,353	468,263	6.0629×10^5
exBCH(63,24), Basis B	1,327,353	271,745	271,655	4.7148×10^5
exBCH(63,30), Basis C	35,028,985	16,091,009	16,090,739	1.82×10^7
exBCH(63,36), Basis C	18,710,521	9,995,617	9,995,419	1.36×10^7
exBCH(63,39), Basis C	38,436,857	24,741,161	24,740,975	3.44×10^7
exBCH(63,45), Basis C	1,082,105	893,489	893,237	9.8514×10^5
exBCH(63,51), Basis A	418,553	312,721	312,673	3.448×10^5

(2) For each execution $NewMLD(h_{i-1}, h_i)$ in (Step B-1'), we choose $\{h_{i-1}, h_i\} \cup \{j|$ a node labeled (h_{i-1}, j) is in $R\} \cup \{j|$ a node labeled (j, h_i) is in $R\}$ as U and $z_{x,y}$ as z. $\Delta\Delta$

Figure 3.2 illustrates this sectionalization for RM(64, 42).

Table 3.2 Optimum sectionalization of the several example codes when $\psi^{(1)}(x,y;-1)$ is used for $\psi(x,y;-1)$.

Code	An $\psi^{(1)}_{\min}$ Optimum Sectionalization
RM(64,22)	$z_{0,64}=63, z_{0,63}=62, z_{0,62}=60, z_{0,60}=56, z_{0,56}=48, z_{0,48}=32,$ $z_{0,32}=16, z_{0,16}=8, z_{0,8}=4, z_{8,16}=12, z_{16,32}=24, z_{16,24}=20,$ $z_{24,32}=28, z_{32,48}=40, z_{32,40}=36, z_{40,48}=44, z_{48,56}=52$
RM(64,42)	$z_{0,64}=63, z_{0,63}=62, z_{0,62}=60, z_{0,60}=56, z_{0,56}=48, z_{0,48}=32,$ $zxy03216, z_{0,16}=8, z_{16,32}=24, z_{32,48}=40$
exBCH(63,24) Basis B	$z_{0,64}=63, z_{0,63}=62, z_{0,62}=60, z_{0,60}=56, z_{0,56}=48, z_{0,48}=32,$ $z_{0,32}=16, z_{0,16}=8, z_{0,8}=4, z_{8,16}=12, z_{16,32}=24, z_{16,24}=20,$ $z_{24,32}=28, z_{32,48}=40, z_{32,40}=36, z_{40,48}=44, z_{48,56}=52$
exBCH(63,30) Basis C	$z_{0,64}=63, z_{0,63}=62, z_{0,62}=60, z_{0,60}=32, z_{0,32}=28, z_{0,28}=24,$ $z_{0,24}=16, z_{0,16}=8, z_{32,60}=56, z_{32,56}=48, z_{32,48}=40$
exBCH(63,45) Basis C	$z_{0,64}=63, z_{0,63}=62, z_{0,62}=32, z_{0,32}=30, z_{0,30}=29, z_{0,29}=28,$ $z_{0,28}=27, z_{0,27}=26, z_{0,26}=24, z_{0,24}=16, z_{0,16}=8, z_{32,62}=61,$ $z_{32,61}=60, z_{32,60}=59, z_{32,59}=58, z_{32,58}=56, z_{32,56}=48, z_{32,48}=40$

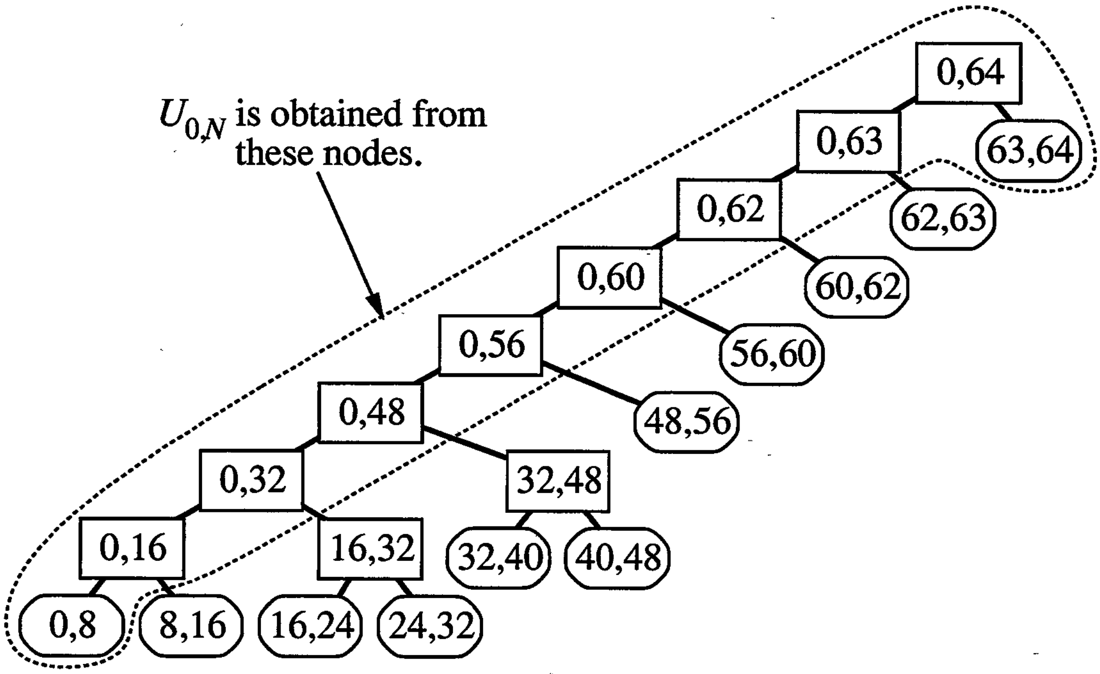

Fig. 3.1 An optimum sectionalization for RM(64,42) when $\psi^{(1)}(x,y;-1)$ is used for $\psi(x,y;-1)$.

$$\mathrm{DETD}(T(\{0,8,16,32,48,56,60,62,63,64\}),63)$$

$$\mathrm{DETD}(T_{C_L(16,32)}(\{16,24,32\}),24) \qquad \mathrm{DETD}(T_{C_L(32,48)}(\{32,40,48\}),40)$$

Fig. 3.2 An implementation by using DETD for RM(64,42).

3.4 Conclusions

In this chapter, a new efficient trellis-based maximum likelihood decoding algorithm for a linear block code using a sectionalized trellis diagram is proposed and its computational complexity is analyzed in terms of the number of addition-equivalent operations. The optimum sectionalizations of trellis diagram for some well known codes are found.

Chapter 4

Conclusions

4.1 Conclusions

In this dissertation, we studied about utilization of characteristics of trellis diagram of codes.

To analyze the error correcting capability provided by trellis codes, an efficient method to evaluate the block error probability of trellis codes is proposed in Chapter 2.

The proposed method consists of two-step. The step to evaluate the values q_0 and q and the step to calculate the approximation of the block error probability from these values are separated. Therefore the proposed method is useful in flexibility of adapting the analysis for given block length. This method enables us to get precise estimation for various block length efficiently. Furthermore, if a method to calculate an upper-bound or precise value of q_0 or q becomes available, the proposed method can be modified to evaluate the block error probability more efficiently.

This method is applied for some specific 8-PSK Ungerboeck codes with $2^4, 2^5$ and 2^6 states. The results show that the values obtained by this method are very close to those by exhaustive simulation. It is concluded that our method is very effective.

To simplify decoding of a code, a new trellis-based MLD algorithm for a linear block code using a sectionalized trellis diagram is proposed and its computational complexity is analyzed in terms of the number of addition-equivalent operations in Chapter 3. The optimum sectionalizations of trellis diagram for some well known codes are found. From these result shows a possibility of construction of much efficient decoders for linear block codes.

It is concluded that characteristics of trellis diagrams are of great use to estimate the block error probability and to design an MLD algorithm. These results contribute large advance in designing high reliability and high speed digital communication technology.

4.2 Future Works

To make the evaluation of the block error probability of trellis codes more efficient, we are investigating a method to calculate an upper-bound or precise value of q_0 or q in Chapter 2.

As mentioned in Chapter 3, there is another implementation [13, 17] based on the detailed structure of each section that is a union of the same simple regular subtrellis diagrams. The implementation yields a smaller $N(x, y; z)$ for the worst case than $N^{(0)}(x, y; z)$ and its average computational complexity is considerably smaller than the worst case one. Modification of proposed decoding algorithm with this implementation of $N(x, y; z)$ is investigated.

A long range objective is a proposition of codes which have good trellis diagrams in terms of usefulness for evaluating error correcting capability and decoding efficiently.

References

[1] A. J. Viterbi and J. K. Omura, *Principles of Digital Communication and Coding*, McGraw-Hill (1979).

[2] H. Yamamoto, Y. Yamashita, T. Takata, and T. Kasami, "Evaluation of block error probabilities for trellis coded modulation," (in Japanese) IEICE Technical Report, **IT 91**-29 (1991-07).

[3] Y. Yamamoto, Y. Yamashita, T. Takata, and T. Kasami, "Evaluation of block error probabilities for multidimensional MPSK trellis codes," (in Japanese) Proc. of the 14th Symposium on Information Theory and Its Applications, Ibusuki, Japan, pp. 85–88 (1991-12).

[4] G. Ungerboeck, "Channel Coding with Multilevel/Phase Signals," *IEEE Trans. Inf. Theory*, **IT-28**, pp. 55–67 (Jan. 1982).

[5] S. Lin and D.J. Costello, Jr. , *Error Control Coding: Fundamentals and Applications*, pp. 337–338, Prentice-Hall, Englewood Cliffs, New Jersey (1983).

[6] G. D. Forney, Jr. , "Convolutional Codes II : Maximum Likelihood Decoding," *Inf. & Control*, **25**, pp. 222–266 (Jul. 1974).

[7] H. Yamamoto, T. Takata, T. Fujiwara, and T. Kasami, "Approximation to the block error probability for convolutional codes," (in Japanese), IEICE Technical Report, **IT 92**-102 (1993-01).

[8] J. Wolf, "Efficient maximum likelihood decoding of linear block codes using a trellis," IEEE Trans. Inf. Theory, vol.IT-24, no.1, pp.76–80, Jan. 1978.

[9] G.D. Forney Jr., "Coset codes—part II: binary lattices and related codes," IEEE Trans. Inf. Theory, vol.34, no.5, pp.1152–1187, Sep. 1988.

[10] T. Kasami, T. Takata, T. Fujiwara, and S. Lin, "On structural complexity of the l-section minimal trellis diagrams for binary linear block codes," IEICE Trans. Fundamentals, vol.E76-A, no.9, pp.1411–1421, Sep.1993.

[11] G.D. Forney Jr., "Dimension/length profiles and trellis complexity of linear block codes," IEEE Trans. Inf. Theory, vol.40, no.6, pp.1741–1752, Nov. 1994.

[12] D.J. Muder, "Minimal trellises for block codes," IEEE Trans. Inf. Theory, vol.34, no.5, pp.1049–1053, Sept. 1988.

[13] T. Kasami, T. Fujiwara, Y. Desaki, and S. Lin, "On branch labels of parallel components of the L-section minimal trellis diagrams for binary linear block codes, " IEICE Trans. Fundamentals, vol.E77-A, no.6, pp.1058–1068, Jun. 1994.

[14] T. Kasami, T. Takata, T. Fujiwara, and S. Lin, "On complexity of trellis structure of linear block codes," IEEE Trans. Inf. Theory, vol.39, no.3, pp.1057–1064, May 1993.

[15] B. Honary, G. Markarian, and S. Marple, "Trellis decoding of the Reed-Solomon codes: practical approach," Int'l Sympo. Commun. Theory & Applications, pp.12–16, Jul. 1995.

[16] A. Vardy and Y. Be'ery, "Maximum-likelihood soft decoding of BCH codes," IEEE Trans. Inf. theory, vol.40, no.2, pp.546–554, Mar. 1994.

[17] H. Nagano, T. Fujiwara, and T. Kasami, "Average complexity evaluation of an MLD algorithm using the trellis structure for a linear block code," IEICE Trans. Fundamentals, vol.E78-A, no.9, Sep. 1995.

[18] T. Kasami, T. Takata, T. Fujiwara, and S. Lin, "On the optimum bit orders with respect to the state complexity of trellis diagrams for binary linear codes," IEEE Trans. Inf. Theory, vol.39, no.1, pp.242–245, Jan. 1993.

[19] T. Fujiwara, T. Kasami, R. Morelos-Zaragoza, and S. Lin, "The state complexity of trellis diagram for a class of generalized concatenated codes," Proc. 16th Sympo. Inf. Theory and Its Applications, pp.21–24, Kanazawa, Japan, Oct. 1993.

[20] S. Lin and D.J. Costello Jr., "Error Control Coding: Fundamentals and Applications," Prentice-Hall, 1983.

Appendix

(Proofs of Theorems 1 and 2)

Before proving the theorems, we compare the number of operations for $NewMLD(x, y)$ with $z_{x,y} = -1$, $\psi(x, y; -1)$, with that for $NewMLD(x, y)$ such that $z_{x,y} = z$ with $x < z < y$ and $z_{x,z} = z_{z,y} = -1$. The number of operations for the latter case is given by

$$\psi(x, z; -1) + \psi(x, z; -1) + N(x, y; z).$$

Define

$$\mathrm{Add}_A^{(0)} \triangleq (y - x - 1)2^{K - K_{\overline{x,y}}[C]}, \qquad \text{A·1}$$

$$\mathrm{Add}_B^{(0)}(z) \triangleq (z - x - 1)2^{K - K_{\overline{x,z}}[C]} + (y - z - 1)2^{K - K_{\overline{z,y}}[C]}$$

$$+2^{K - K_{\overline{x,y}}[C] - K_{x,y}[C] + K_{x,z,y}[C]}, \qquad \text{A·2}$$

$$\mathrm{Cmp}_A^{(0)} \triangleq 2^{K - K_{\overline{x,y}}[C] - K_{x,y}[C]}(2^{K_{x,y}[C]} - 1), \qquad \text{A·3}$$

$$\mathrm{Cmp}_B^{(0)}(z) \triangleq 2^{K - K_{\overline{x,z}}[C] - K_{x,z}[C]}(2^{K_{x,z}[C]} - 1) + 2^{K - K_{\overline{z,y}}[C] - K_{z,y}[C]}(2^{K_{z,y}[C]} - 1)$$

$$+2^{K - K_{\overline{x,y}}[C] - K_{x,y}[C]}(2^{K_{x,z,y}[C]} - 1). \qquad \text{A·4}$$

Then,

$$\psi^{(0)}(x, y; -1) = \mathrm{Add}_A^{(0)} + \mathrm{Cmp}_A^{(0)}, \qquad \text{A·5}$$

$$\psi^{(0)}(x, z; -1) + \psi^{(0)}(z, y; -1) + N(x, y; z) = \mathrm{Add}_B^{(0)}(z) + \mathrm{Cmp}_B^{(0)}(z). \qquad \text{A·6}$$

Lemma A.1: For any integers x, y and z with $0 \le x < z < y \le N$,

$$\mathrm{Cmp}_A^{(0)} \ge \mathrm{Cmp}_B^{(0)}(z). \qquad \text{A·7}$$

The equality holds if and only if

$$K_{x,z}[C] = K_{z,y}[C] = 0. \qquad \text{A·8}$$

(Proof) Since it follows from the definitions of $K_{h,h'}[C]$ and $K_{\overline{h,h'}}[C]$ that

$$K_{\overline{x,z}}[C] \geq K_{\overline{x,y}}[C] + K_{z,y}[C] \quad \text{and} \quad K_{\overline{z,y}}[C] \geq K_{\overline{x,y}}[C] + K_{x,z}[C], \qquad \text{A·9}$$

we have that

$$\begin{aligned}
\mathrm{Cmp}_B^{(0)}(z) &= 2^{K-K_{\overline{x,z}}[C]-K_{x,z}[C]}(2^{K_{x,z}[C]}-1) + 2^{K-K_{\overline{z,y}}[C]-K_{z,y}[C]}(2^{K_{z,y}[C]}-1) \\
&\quad +2^{K-K_{\overline{x,y}}[C]-K_{x,y}[C]}(2^{K_{x,z,y}[C]}-1) \\
&\leq 2^{K-K_{\overline{x,y}}[C]-K_{x,z}[C]-K_{z,y}[C]}(2^{K_{x,z}[C]}-1) \\
&\quad +2^{K-K_{\overline{x,y}}[C]-K_{x,z}[C]-K_{z,y}[C]}(2^{K_{z,y}[C]}-1) \\
&\quad +2^{K-K_{\overline{x,y}}[C]-K_{x,y}[C]}(2^{K_{x,z,y}[C]}-1) \\
&= 2^{K-K_{\overline{x,y}}[C]-K_{x,y}[C]}(2^{K_{x,y}[C]}(2^{-K_{z,y}[C]}+2^{-K_{x,z}[C]}-2^{-K_{x,z}[C]-K_{z,y}[C]})-1) \\
&= 2^{K-K_{\overline{x,y}}[C]-K_{x,y}[C]}(2^{K_{x,y}[C]}(1-(1-2^{-K_{x,z}[C]})(1-2^{-K_{z,y}[C]}))-1) \\
&\leq 2^{K-K_{\overline{x,y}}[C]-K_{x,y}[C]}(2^{K_{x,y}[C]}-1) \\
&= \mathrm{Cmp}_A^{(0)}. \qquad \text{A·10}
\end{aligned}$$

In the second inequality of A·10, the equality holds if and only if A·8 holds. Assume that A·8 holds. Then, the equality in the first inequality of A·10 also holds. ΔΔ

Lemma A.2: For any integers x, y and z with $0 \leq x < z < y \leq N$,

$$\mathrm{Add}_A^{(0)} \geq \mathrm{Add}_B^{(0)}(z). \qquad \text{A·11}$$

The equality holds if and only if all of the following three conditions hold:

(i) $K_{x,z}[C] = K_{z,y}[C] = 0$.

(ii) Either $K_{\overline{x,y}}[C] = K_{\overline{x,z}}[C]$ or $x = z-1$.

(iii) Either $K_{\overline{x,y}}[C] = K_{\overline{z,y}}[C]$ or $z = y-1$.

(Proof)

$$\begin{aligned}
\mathrm{Add}_B^{(0)}(z) &= (z-x-1)2^{K-K_{\overline{x,z}}[C]} + (y-z-1)2^{K-K_{\overline{z,y}}[C]} \\
&\quad +2^{K-K_{\overline{x,y}}[C]-K_{x,y}[C]+K_{x,z,y}[C]} \\
&\leq (z-x-1)2^{K-K_{\overline{x,y}}[C]} + (y-z-1)2^{K-K_{\overline{x,y}}[C]} \\
&\quad +2^{K-K_{\overline{x,y}}[C]-K_{x,z}[C]-K_{z,y}[C]} \\
&= (y-x-2+2^{-K_{x,z}[C]-K_{z,y}[C]})2^{K-K_{\overline{x,y}}[C]} \\
&\leq \mathrm{Add}_A^{(0)}. \qquad \text{A·12}
\end{aligned}$$

The equality in the first inequality of A·12 holds iff (ii) and (iii) hold, and the equality in the second inequality holds iff (i) holds. $\Delta\Delta$

Lemma A.3: Suppose that $|p_{x,y}[C]| = 2^{y-x}$ and that $K_{x,z}[C] = K_{z,y}[C] = 0$ for any z with $x < z < y$. Then,

$$\mathrm{Add}_B^{(0)}(z) \geq \mathrm{Add}_B^{(0)}(\left\lfloor \frac{x+y}{2} \right\rfloor) = \mathrm{Add}_B^{(0)}(\left\lceil \frac{x+y}{2} \right\rceil), \qquad \text{for } x < z < y. \qquad \text{A·13}$$

(Proof) Since $|p_{x,y}[C]| = 2^{y-x}$, we have that $|p_{x,z}[C]| = 2^{z-x}$ and $|p_{z,y}[C]| = 2^{y-z}$. Hence,

$$\mathrm{Add}_B^{(0)}(z) = (z-x-1)2^{z-x} + (y-z-1)2^{y-z} + 2^{y-x}. \qquad \text{A·14}$$

By defining that $n \triangleq y - x$ and $u = z - x$, we have that

$$\begin{aligned} \mathrm{Add}_B^{(0)}(z+1) - \mathrm{Add}_B^{(0)}(z) &= u2^{u+1} + (n-u-2)2^{n-u-1} - (u-1)2^u \\ &\quad -(n-u-1)2^{n-u} \\ &= (u+1)2^u - (n-u)2^{n-u-1}. \end{aligned} \qquad \text{A·15}$$

The function of X, $(X+1)2^X$ increases monotonically when X increases. Hence,

$$\mathrm{Add}_B^{(0)}(z+1) - \mathrm{Add}_B^{(0)}(z) \begin{cases} > 0, & \text{if } 2z > x+y-1, \\ = 0, & \text{if } 2z = x+y-1, \\ < 0, & \text{if } 2z < x+y-1. \end{cases} \qquad \text{A·16}$$

$\mathrm{Add}_B^{(0)}(z)$ takes its minimum when $z = \left\lfloor \frac{x+y}{2} \right\rfloor$ and $z = \left\lceil \frac{x+y}{2} \right\rceil$. $\Delta\Delta$

Lemma A.4: The equation A·8 holds for any z with $x < z < y$ if and only if $C_{x,y}^{\mathrm{tr}} = \{(0,0,\ldots,0)\}$ or $\{(0,0,\ldots,0),(1,*,\ldots,*,1)\}$.
(Proof) If part: trivial. Only if part: If $C_{x,y}^{\mathrm{tr}}$ does not satisfy the condition, $C_{x,y}^{\mathrm{tr}}$ contains a nonzero codeword of the form $(0,\ldots)$ or $(\ldots,0)$. Then, we have that $K_{x+1,y} \neq 0$ or $K_{x,y-1} \neq 0$. $\Delta\Delta$

(Proof of Theorem 1) Proof of (1): From the condition on the generator matrix, we have that

$$K_{\overline{x,z}}[C] > K_{\overline{x,y}}[C] \qquad \text{and} \qquad K_{\overline{z,y}}[C] > K_{\overline{x,y}}[C]. \qquad \text{A·17}$$

Then, the equality does not hold in A·11 for $y-x>2$. From this fact and Lemma A.1, we have the first statement of (1) in the theorem. When $y-x=2$, we have $y-2=z-1=x$. Also, we have that $K_{x,z}[C]=K_{z,y}[C]=0$ from the assumption on the minimum distance of C. Then, the equalities hold in A·7 and A·11. Hence, the complexities for the both cases are the same.

Proof of (2): From the condition on $C^{\mathrm{tr}}_{x,y}$ and Lemma A.4, we have that $K_{x,z}[C]=K_{z,y}[C]=0$. Part (2) of the theorem follows from part (1) of the theorem and Lemma A.3. $\triangle\triangle$

Define

$$\mathrm{Add}_A^{(1)} \triangleq f(x,y), \qquad \text{A·18}$$

$$\mathrm{Add}_B^{(1)}(z) \triangleq f(x,z)+f(z,y)+2^{K-K_{\overline{x,y}}[C]-K_{x,y}[C]+K_{x,z,y}[C]}, \qquad \text{A·19}$$

$$\mathrm{Cmp}_A^{(1)} \triangleq \mathrm{Cmp}_A^{(0)}, \qquad \text{A·20}$$

$$\mathrm{Cmp}_B^{(1)}(z) \triangleq \mathrm{Cmp}_B^{(0)}(z), \qquad \text{A·21}$$

where

$$f(i,j) \triangleq \begin{cases} 2^{j-i}+(j-i-2), & \text{for } j-i>1, \\ 0, & \text{for } j-i=1. \end{cases} \qquad \text{A·22}$$

Then,

$$\psi^{(1)}(x,y;-1) = \mathrm{Add}_A^{(1)}+\mathrm{Cmp}_A^{(1)}, \qquad \text{A·23}$$

$$\psi^{(1)}(x,z;-1)+\psi^{(1)}(x,z;-1)+N(x,y;z) = \mathrm{Add}_B^{(1)}(z)+\mathrm{Cmp}_B^{(1)}(z). \qquad \text{A·24}$$

Lemma A.5: (1) If $|p_{x,y}[C]| = 2^{K-K_{\overline{x,y}}[C]} = 2^{y-x}$ and $K_{x,z}[C]=K_{z,y}[C]=0$ for any z with $x<z<y$, then

$$\mathrm{Add}_A^{(1)} \le \mathrm{Add}_B^{(1)}(z), \qquad \text{for } x<z<y. \qquad \text{A·25}$$

The equality holds if and only if $x+2=y$.

(2) Otherwise,

$$\mathrm{Add}_A^{(1)} > \mathrm{Add}_B^{(1)}(z). \qquad \text{A·26}$$

(Proof) For $x+1<z<y-1$, we have that

$$\begin{aligned}\mathrm{Add}_B^{(1)}(z) &= 2^{z-x}+(z-x-2)+2^{y-z}+(y-z-2)+2^{K-K_{\overline{x,y}}[C]-K_{x,y}[C]+K_{x,z,y}[C]} \\ &= 2^{z-x}+2^{y-z}+2^{K-K_{\overline{x,y}}[C]-K_{x,z}[C]-K_{y,z}[C]}-2+(y-x-2),\end{aligned} \qquad \text{A·27}$$

and otherwise we have that

$$\mathrm{Add}_B^{(1)}(z) \geq 2^{z-x} + 2^{y-z} + 2^{K-K_{\overline{x,y}}[C]-K_{x,z}[C]-K_{y,z}[C]} - 4 + (y-x-2). \qquad \text{A·28}$$

The equality in A·28 holds if and only if $x+2=y$.

Proof of (1):

$$\mathrm{Add}_B^{(1)}(z) - (y-x-2) \geq 2^{z-x} + 2^{y-z} + 2^{y-x} - 4 \geq 2^{y-x}, \qquad \text{A·29}$$

and hence we have A·25.

Proof of (2): (a) If $x+1 < z < y-1$,

$$\begin{aligned} \mathrm{Add}_B^{(1)}(z) - (y-x-2) &= 2^{z-x} + 2^{y-z} + 2^{K-K_{\overline{x,y}}[C]-K_{x,z}[C]-K_{y,z}[C]} - 2 \\ &\leq 2^{z-x} + 2^{y-z} + 2^{y-x-1} - 2 \\ &< 2^{y-x}, \end{aligned} \qquad \text{A·30}$$

and hence we have A·26.

(b) Otherwise,

$$\mathrm{Add}_B^{(1)}(z) - (y-x-2) < 2^{z-x} + 2^{y-z} + 2^{y-x-1} - 2 \leq 2^{y-x}, \qquad \text{A·31}$$

and hence we have A·26. ΔΔ

Lemma A.6: If $|p_{x,y}[C]| < 2^{y-x}$ and $K_{x,z}[C] = K_{z,y}[C] = 0$ for any z with $x < z < y$, then

$$\mathrm{Add}_B^{(1)}(z) \geq \mathrm{Add}_B^{(1)}\left(\left\lfloor \frac{x+y}{2} \right\rfloor\right) = \mathrm{Add}_B^{(1)}\left(\left\lceil \frac{x+y}{2} \right\rceil\right). \qquad \text{A·32}$$

(Proof) When $y-x \leq 3$, the integers z such that $x < z < y$ is only $\left\lfloor \frac{x+y}{2} \right\rfloor$ and $\left\lceil \frac{x+y}{2} \right\rceil$. When $y-x=2$, these two integers are the same. When $y-x=3$,

$$\mathrm{Add}_B^{(1)}\left(\left\lfloor \frac{x+y}{2} \right\rfloor\right) = \mathrm{Add}_B^{(1)}\left(\left\lceil \frac{x+y}{2} \right\rceil\right) = 2^{y-x-1} + 2^{K-K_{\overline{x,y}}[C]} + y - x - 3. \qquad \text{A·33}$$

Consider the case with $y-x \geq 4$. Then,

$$\begin{aligned} &\min_{x<z<y}\{\mathrm{Add}_B^{(1)}(z)\} - (y-x-2) - 2^{K-K_{\overline{x,y}}[C]} \\ &= \min\{2^{y-x-1} - 1, \min_{x+1<z<y-1}\{2^{z-x} + 2^{y-z} - 2\}\} \\ &= \min_{x+1<z<y-1}\{2^{z-x} + 2^{y-z} - 2\}. \end{aligned} \qquad \text{A·34}$$

ΔΔ

(Proof of Theorem 2) Proof of (1): When $y - x = 2$, the statement follows from Lemma A.1 and part (1) of Lemma A.5. Consider the case with $y - x > 2$. From Lemma A.1 and part (1) of Lemma A.5, we have that

$$\psi^{(1)}(x, y; -1) < \psi^{(1)}(x, z; -1) + \psi^{(1)}(z, y; -1) + N(x, y; z).$$

Suppose that the statement is not true. Then, there is a pair of integers, (x, y), such that $y - x > 2$ and the smallest complexity of (Case B) is not greater than $\psi^{(1)}(x, y; -1)$. Consider such a pair with the smallest $y - x$. From the minimality of $y - x$, the smallest complexity of (Case B) is given by

$$\min_{x<z<y} \{\psi^{(1)}(x, z; -1) + \psi^{(1)}(z, y; -1) + N(x, y; z)\}.$$

A contradiction.

Proof of (2): The first part follows from Lemma A.1 and (2) of Lemma A.5. The last part follows from the first part of (2) of this theorem and Lemma A.6. $\Delta\Delta$