Title

Multihead Multitrack Detection for Next Generation Magnetic Recording Systems

Author

Fan, Bing

Publication Date

2017

UNIVERSITY OF CALIFORNIA, SAN DIEGO

Multihead Multitrack Detection for Next Generation Magnetic Recording Systems

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

in

Electrical Engineering
(Communication Theory and Systems)

by

Bing Fan

Committee in charge:

Professor Paul H. Siegel, Chair
Professor Laurence B. Milstein
Professor Bhaskar D. Rao
Professor Steven J. Swanson
Professor Alexander Vardy

2017

The dissertation of Bing Fan is approved, and it is acceptable in quality and form for publication on microfilm and electronically:

Chair

University of California, San Diego

2017

DEDICATION

Dedicated to my family.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

It is my fortune to be admitted into UCSD in 2011. I appreciate UCSD for providing me a wonderful learning experience and a pleasant working environment. All these past years will always be a previous memory for me in my life.

I would like to acknowledge many people for their support and encouragement during my doctoral research. This dissertation would not have been finished without their help.

First, I would like to express my deepest gratitude to my advisor, Prof. Paul H. Siegel, for his guidance and endless patience throughout my research progress. It is Paul who introduced me to the world of coding theory. I learned a lot from his vast knowledge, clear explanation, and insightful comments. He is very rigorous in research, kind and patient to students. His encouragement and belief in me help me build confidence to overcome difficulties in research and pursue higher goals.

I would like to thank Prof. Laurence Milstein, Prof. Bhaskar Rao, Prof. Paul Siegel, Prof. Alon Orlitsky, and Prof. Ken Zeger for their introductory courses on signal processing, communication theory, information theory, and coding theory. This dissertation benefits a lot from the knowledge I have learned in these classes. I would like to thank Prof. Paul Siegel, Prof. Laurence Milstein, Prof. Bhaskar Rao, Prof. Steven Swanson and Prof. Alexander Vardy, for their time and effort to serve on my committee.

I would like to express my special thanks to my collaborators, Dr. Hemant Thapar, and Dr. Minghai Qin, for their great contributions to my dissertation. Hemant has vast knowledge in the area of magnetic recording and signal processing. The brainstorm with him often stimulates elegant ideas that improve my research. His enthusiasm and rigorousness in work will always be a model for me to follow in my future career. I also learned a lot from the discussions with Minghai, for he always came up with interesting questions and brilliant ideas.

Many thanks to Derrick Burton, my manager during the internship at Western Digital in 2012, and Xiaojie Zhang, my manager during the internship at Cnexlabs, and Jim Fitzpatrick, my manager during the internship at SanDisk. I would also like to thank Jonas Goode, Daniel

Helmick, Bruce Kaufman, Majid Nemati, and Scott Kayser for their help during my internships.

I would like to thank the current members and alumni of STAR group: Aman Bhatia, Brian Butler, Sarit Buzaglo, Pengfei Huang, Vasu Kanekal, Seyhan Karakulak, Scott Kayser, Lingjun Kong, Yonglong Li, Yi Liu, Minghai Qin, Veeresh Taranalli, Osamu Torii, Karthik Tunuguntul, Hironori Uchikawa, Wei Wu, Eitan Yaakobi, and Xiaojie Zhang. Their kindness and friendship make the STAR group a delighted place to work in. A special thanks to Aman Bhatia and Wei Wu, for being my great office-mates. I enjoyed our conversations and numerous interactions at work.

I would like to thank all CMRR staff especially Ray Descoteaux, Marina Robenko, Gabby Tshamjyan, and Iris Villanueva for their support during my Ph.D. program. I would like to thank my friends in other CMRR research groups, Sidi Fu, Youyi Fu, Katcher Morris, Alex Phan, Haowen Ren, Young Seo, and Robert Tolley, for their valuable friendship. I would like to thank all my friends in San Diego especially Yacong Ding, Qi Huang, Ying Huang, Yao Liu, Qing Song, Tiantian Zhang, and Yuchun Zhou for their support and help during my six years in San Diego.

Last but not least, I would like to thank my parents, Ms. Caiju Zhan and Mr. Dengfeng Fan, for giving me life and wisdom. Whenever I meet challenges, I know they will always be there to support me. I also wish to thank my fiance, Mr. Hao Shi, since without his love and company, I would never have gone so far.

This dissertation was supported in part by the National Science Foundation under Grant CCF-1405119, and the Center for Memory and Recording Research (formerly, Center for Magnetic Recording Research) at UC San Diego.

Chapter 2 contains material from the papers: Bing Fan, Hemant K. Thapar and Paul H. Siegel, “Multihead multitrack detection in shingled magnetic recording with ITI estimation,” in *IEEE Int. Conf. on Commun. (ICC)*, London, UK, Jun. 2015, pp. 425-430, and Bing Fan, Hemant K. Thapar and Paul H. Siegel, “Multihead multitrack detection for next generation magnetic recording, part I: weighted sum subtract joint detection with ITI estimation,” *IEEE Trans. Commun.*, vol. 65, no. 4, pp. 1635-1648, April 2017. The dissertation author was the primary investigator and author of these papers, and co-authors have approved the use of the material for this dissertation.

Chapter 3 contains material from the papers: Bing Fan, Hemant K. Thapar and Paul H. Siegel, “Multihead multitrack detection with reduced-state sequence estimation in shingled magnetic recording,” *IEEE Trans. Magn.*, vol. 51, no. 11, pp. 1-4, Nov. 2015, and Bing Fan, Hemant K. Thapar and Paul H. Siegel, “Multihead multitrack detection for next generation magnetic recording, part II: complexity reduction - algorithms and performance analysis,” *IEEE Trans. Commun.*, vol. 65, no. 4, pp. 1649-1661, April 2017. The dissertation author was the primary investigator and author of these papers, and co-authors have approved the use of the material for this dissertation.

Chapter 4 is in part a reprint of the material in the paper: Bing Fan, Paul H. Siegel and Hemant K. Thapar, “Generalized weighted sum subtract joint detection,” which is being prepared for submission to *IEEE Transactions on Magnetics*. The dissertation author was the primary investigator and author of these papers, and co-authors have approved the use of the material for this dissertation.

Chapter 5 is in part a reprint of the material in the paper: Bing Fan, Minghai Qin and Paul H. Siegel, “Enhancing the average lifetime of flash memory by lattice-based WOM codes,” which is being prepared for submission to *IEEE Communications Letters*. The dissertation author was the primary investigator and author of these papers, and co-authors have approved the use of the material for this dissertation.

VITA

2011	B.S. in Electrical and Computer Engineering, UM-SJTU Joint Institute, Shanghai Jiao Tong University, China
2013	M.S. in Electrical Engineering (Communication Theory and Systems), University of California, San Diego
2017	Ph.D. in Electrical Engineering (Communication Theory and Systems), University of California, San Diego

PUBLICATIONS

Bing Fan, Hemant K. Thapar and Paul H. Siegel, “Multihead multitrack detection in shingled magnetic recording with ITI estimation,” in *IEEE Int. Conf. on Commun. (ICC)*, London, UK, Jun. 2015, pp. 425-430.

Bing Fan, Hemant K. Thapar and Paul H. Siegel, “Multihead multitrack detection with reduced-state sequence estimation in shingled magnetic recording,” in *IEEE International Magnetics Conference (INTERMAG)*, Beijing, China, May 2015, pp. 1-1.

Bing Fan, Hemant K. Thapar and Paul H. Siegel, “Multihead multitrack detection with reduced-state sequence estimation in shingled magnetic recording,” *IEEE Trans. Magn.*, vol. 51, no. 11, pp. 1-4, Nov. 2015.

Bing Fan, Hemant K. Thapar and Paul H. Siegel, “Multihead multitrack detection for next generation magnetic recording, part I: weighted sum subtract joint detection with ITI estimation,” *IEEE Trans. Commun.*, vol. 65, no. 4, pp. 1635-1648, April 2017.

Bing Fan, Hemant K. Thapar and Paul H. Siegel, “Multihead multitrack detection for next generation magnetic recording, part II: complexity reduction - algorithms and performance analysis,” *IEEE Trans. Commun.*, vol. 65, no. 4, pp. 1649-1661, April 2017.

Bing Fan, Paul H. Siegel and Hemant K. Thapar, “Generalized weighted sum subtract joint detection,” *to be submitted to IEEE Transactions on Magnetics*.

Bing Fan, Minghai Qin and Paul H. Siegel, “Enhancing the average lifetime of flash memory by lattice-based WOM codes,” *to be submitted to IEEE Communications Letters*.

ABSTRACT OF THE DISSERTATION

Multihead Multitrack Detection for Next Generation Magnetic Recording Systems

by

Bing Fan

Doctor of Philosophy in Electrical Engineering
(Communication Theory and Systems)

University of California, San Diego, 2017

Professor Paul H. Siegel, Chair

Two dimensional magnetic recording (TDMR) is one of the leading technologies proposed to achieve ultra high storage density in the next generation hard disk drives. A typical read channel of TDMR can be approximated by a multihead multitrack (MHMT) model which is characterized by the intersymbol interference (ISI) in the downtrack direction and intertrack interference (ITI) in the crosstrack direction. In this dissertation, we aim to address two challenging problems that can potentially make the maximum likelihood (ML) detection for the MHMT channel impractical.

One problem associated with the conventional ML detector is the ITI dependency of the ML trellis, which makes it inefficient to adapt to a time-varying ITI environment. We propose a novel detection method, called the weighted sum-subtract joint detection (WSSJD), to solve this issue. We show that the ITI interfering channels in MHMT system can be transformed to several parallel subchannels, whose joint trellis is independent of ITI. For the case when the receiver lacks the knowledge of ITI, we propose a gain loop structure which can be incorporated into WSSJD to efficiently track the ITI estimates. We present the implementation of WSSJD and simulation results for several widely-studied MHMT models including the 2H2T channel, the 3H3T channel, and the 3H2T channel.

Another challenging problem is the exponentially increased computational complexity of the ML detector due to its multitrack processing scheme. We propose to use the reduced-state sequence estimation (RSSE) to mitigate this issue. The underlying idea is to drop less likely paths at early state during the detection by using a heavily reduced subset trellis. To apply RSSE to the MHMT channel, we find that the channel transformation developed in WSSJD offers a natural set partition principle on the input constellation that is necessary to successfully implement RSSE. The theoretical error event analysis shows a good consistency with the simulation results.

Flash memory has faster access speed, lower power consumption, and better data integrity, which make it more attractive than the magnetic recording in many applications. However, the lifetime of flash memory is limited by the program/erase cycles. Write-once memory (WOM) codes are proposed to extend the flash memory lifetime by constraining the unidirectional changes of cell levels. In this work, we study the expected performance of WOM codes. Dynamic programming based algorithm is proposed to construct the optimal WOM updating function of a given labeling function.

Chapter 1

Introduction

1.1 Magnetic Recording System

The first commercially available hard disk drive, the IBM 350 Disk Drive, was introduced by IBM as a part of the random access method of accounting and control (RAMAC) system. This huge storage device consisted of fifty disks, each of which was 24-inch in diameter. It weighted over 500 lb, but could only store 5MB, with a data density of 2000 bits/in^2 [50].The signal processing on the first drive was very simple. Data bits were detected by an amplitude detector which registered a bit-1 whenever the readback voltage exceeded a specified level. Neither constrained codes nor error correction codes (ECCs) were used.

After sixty years of development, today's hard disk drives (HDDs) have much higher storage capacity but smaller volume. In 2016, Seagate Technology announced a 5TB consumer drive at a record-high areal density of 1307 Gb/in^2 [42], about 600 million times that of the first drive. The growth in areal density has benefited from progress in head and media design, and the adoption of more sophisticated signal processing techniques. One radical innovation is the use of perpendicular recording. Compared to longitudinal recording, perpendicular recording can significantly increase the linear density. Another improvement is the use of partial response (PR) signaling and Viterbi detection. They can better combat the intersymbol interference (ISI). Constrained codes and ECCs are also incorporated to better protect the user data.

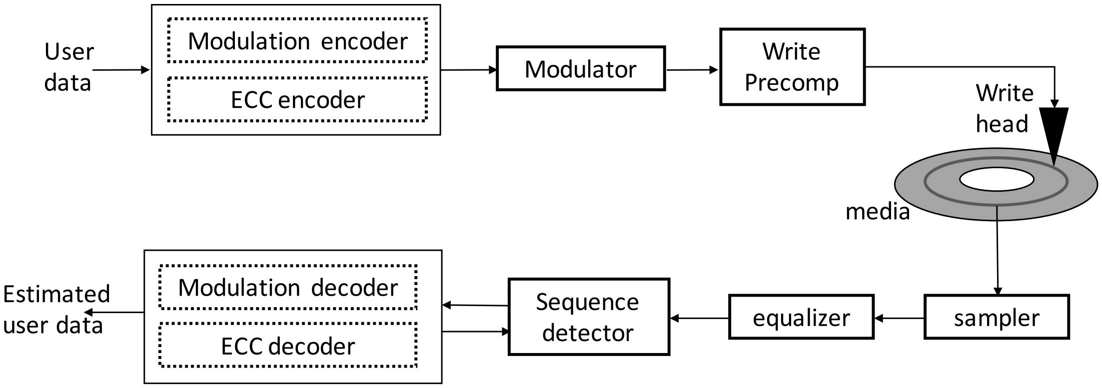


Figure 1.1: A typical magnetic recording channel model.

A typical write/read channel model for a magnetic recording system is depicted in Fig. 1.1 [50]. The write channel consists of modulation/ECC encoder, modulator, and write precompensation. User data bits are encoded by a modulation code and ECC before being written to the disk. A commonly used modulation code is the maximum transition run (MTR) code [34]. Low density parity check (LDPC) codes are widely used in today's drives as the ECC to protect user data from channel noise. On early drives, user data bits are first encoded by ECC before being passed to the modulation encoder. However, this scheme can lead to severe error propagation when the modulation code rate is high. To solve this problem, a reverse concatenation (RC) is proposed, where the ECC is implemented after the modulation encoder. The encoded bit sequence is then fed into the modulator to generate a rectangular write current waveform. A write precompensation block is incorporated to reduce the nonlinear transition shift effect.

The data is stored on the disk as binary digits. Controlled by the modulated write current, a write head generates tiny areas of positive and negative magnetization on the rotating disk. Each polarized area represents a "bit" of information. The recorded bits are organized into data tracks, which are closely-spaced concentric bands on the disk surface. A data track is further divided into many sectors, each of which contains one codeword. The storage capacity of a HDD depends on two factors: the number of bits on each track, and the number of tracks on the disk.

In the read process, the read head can only sense magnetic transitions. For longitudinal recording, the transition response is a bell-shaped waveform approximated by a Lorentzian function,

$$h(t) = \frac{A}{1 + (2t/\text{PW50})^2}, \tag{1.1}$$

where A is the peak amplitude, and PW50 is the width of the pulse at half the peak amplitude. For perpendicular recording, the transition response is approximated by

$$h(t) = A \cdot \text{erf}\left(\frac{0.954t}{T_{50}}\right), \tag{1.2}$$

where T_{50} is the width when $h(t)$ changes from $-A/2$ to $A/2$. The error function erf$(\cdot)$ is defined as

$$\text{erf}(t) = \frac{2}{\sqrt{\pi}} \int_0^t e^{-x^2} dx. \tag{1.3}$$

Let T be the time to read one channel bit. The ratios PW$50/T$ and T_{50}/T measure the linear density: the larger the ratio is, the higher linear density will be. The read back signal is roughly a linear combination of the shifted transition responses. The analog waveform is then sampled and equalized to a PR target. The PR target characterizes the intersymbol interference (ISI) in the downtrack direction. It is usually represented by a polynomial with a finite degree. For example, a widely known PR target is the extended class-IV PR (EPR4) channel given by

$$h(D) = (1 - D)(1 + D)^n = 1 + D - D^2 - D^3. \tag{1.4}$$

The output sequence of a PR channel is corrupted by electronic Gaussian noise. The bipolar input sequence $x(D)$, the discretized output sequence $y(D)$, and the noise sequence $n(D)$ are mathematically related by

$$y(D) = x(D)h(D) + n(D). \tag{1.5}$$

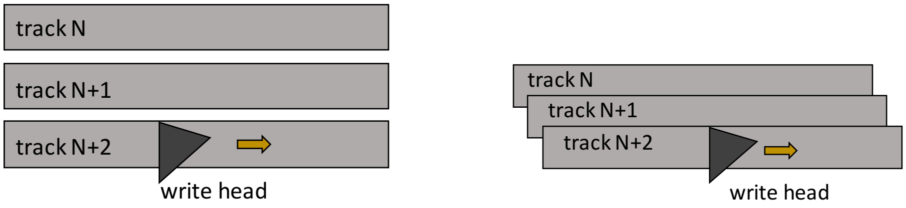
Given the noisy channel output, a sequence detector is used to estimate each input bit. Based on the type of outputs, the sequence detector can be classified into two categories. A hard decision detector predicts an input bit to be either 1 or 0, while a soft decision detector generates the likelihood of input bits. The likelihood is also called the soft information of the bit. In high-performance drives, the soft-output detector is preferable since it provides more reliable estimates to the ECC decoder. A turbo-equalization is implemented so that the soft-decision detector and the ECC decoder can iteratively share soft information of input bits. After several iterations (typically less than 3 iterations), the system outputs the estimated user bits, based on their accumulated soft information.

1.2 Two Dimensional Magnetic Recording

With the development of information networks and data centers, the demand for ultra-high capacity storage devices is continually increasing. For HDD storage, several promising technologies have been proposed to increase the areal density. Among these options, bit-patterned magnetic recording (BPMR), heat-assisted magnetic recording (HAMR), and microwave-assisted magnetic recording (MAMR) are technologies that require radical modification of the recording media and/or the read-write transducer to push beyond the superparamagnetic limit [44]. In contrast, two dimensional magnetic recording (TDMR) combined with shingled writing, has been proposed as an approach to advancing areal density using conventional media and recording head technology, relying instead upon more sophisticated signal processing techniques to decode data under severe noise and interference conditions. It is expected that TDMR on a conventional recording medium with 20 Teragrains/in^2 can achieve an areal density of 10 Tb/in^2 [56].

1.2.1 Shingled Writing

TDMR increases the areal density by aggressively shrinking the tiny area storing one bit. A specially designed “corner writer” is used to maintain high write fields. To increase the tracks density, the write head sequentially writes data tracks, and each time overwrites a small portion of the previous track. The resulting tracks are heavily squeezed and overlapped. This process is



(a) traditional writing (b) shingled writing

Figure 1.2: A comparison between traditional writing and shingled writing on HDD.

called shingled magnetic recording (SMR). Figs. 1.2 illustrate the difference between traditional writing and shingled writing.

SMR introduces a challenge in the data updating process. Since the dimension of the write head is larger than the width of one track, updating a portion of one track requires first recovering the data on sequentially written tracks. After rewriting the target track, the recovered data are rewritten back to the disk. This process could incur a huge amount of delay and a lot of background work. Several data management methods are proposed to solve this issue [3] [21].

1.2.2 Detection Methods

One particular challenge faced by TDMR is how to deal with the cross-talk between densely-packed data tracks. It is conceivable that in the near term, the width and separation of recording tracks will be significantly reduced, with a relatively smaller reduction in read head size [56] [24] [38]. Therefore, the read head will sense signals from neighboring tracks when reading from a target track, causing substantial intertrack interference (ITI) [36]. The performance of a single-head single-track (SHST) detector in such a scenario with ITI was studied in [40]. The results show that the additional signal distortion caused by ITI can severely degrade the performance of disk drives using conventional SHST detection methods.

2D-ISI Channel

When the ITI is as severe as ISI, the magnetic recording channel can be well approximated by a two-dimensional ISI (2D-ISI) model, in which the impulse response of the read head is

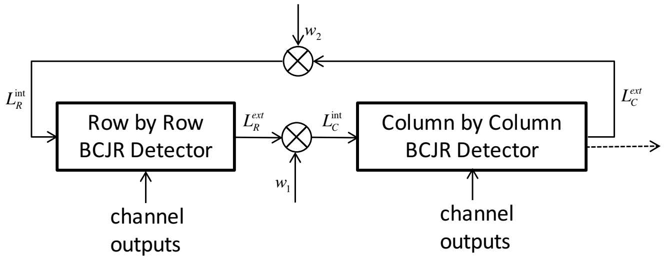


Figure 1.3: A Turbo-like detector consisting of two component BCJR detectors. The component detectors iteratively share soft information, represented by L_R^{int}, L_R^{ext}, L_C^{int}, and L_C^{ext}. Weights w_1 and w_2 are used to improve the reliability of the soft information.

represented by a matrix [22]. Assume $\mathbf{H}$ is an $m \times n$ 2D-ISI channel matrix whose main tap is $h_{0,0}$.

$$\mathbf{H} = \begin{bmatrix} h_{-D_M,-D_N} & \cdots & h_{-D_M,0} & \cdots & h_{-D_M,N-D_N-1} \\ \vdots & \ddots & \vdots & \iddots & \vdots \\ h_{0,-D_N} & \cdots & h_{0,0} & \cdots & h_{0,N-D_N-1} \\ \vdots & \iddots & \vdots & \ddots & \vdots \\ h_{M-D_M-1,-D_N} & \cdots & h_{M-D_M-1,0} & \cdots & h_{M-D_M-1,N-D_N-1} \end{bmatrix} \tag{1.6}$$

Given a bipolar input matrix $\mathbf{X} = \{x_{i,j}\}$, $x_{i,j} \in +1, -1$, the noiseless channel output is a matrix $\mathbf{R}$ where

$$r_{i,j} = \sum_m \sum_n h_{m,n} x_{i-m,j-n}. \tag{1.7}$$

It has been proved that maximum likelihood (ML) detection on a general 2D-ISI channel is NP-hard [37]. Considerable effort has been spent on designing practical suboptimal detectors. For example, the authors in [31] propose a one-dimensional (1D) multi-strip Bahl-Cocke-Jelinek-Raviv (BCJR) detector for a 2×2 channel matrix. Their algorithm, which passes soft information from column to column, achieves performance that is approximately $\frac{2}{3}$dB from ML. The detector proposed in [10] adopts a turbo structure, where the soft information is iteratively shared between a row-by-row BCJR detector and another column-by-column BCJR detector. An illustration of this type of detector is shown in Fig. 1.3. Simulation results show that the performance of the proposed detector is near optimal. In [33], the performance of a similar turbo detector was then examined on a more realistic magnetic recording channel derived from a Voronoi grain model. Several other turbo-structured detectors have also been explored in [57], [11], [9]. Although the row-column turbo detector achieves promising performance on the 2D-ISI channel, it generally suffers from high computational complexity and long delay. In [26] the authors construct a factor graph for the 2D-ISI channel and design a general belief propagation (GBP) detector to combat both the 2D-ISI and the "overwriting" effect in TDMR. Although GBP has higher complexity, its parallelized information processing can lead to shorter detection delay.

MHMT Channel

Another approach to approximating the magnetic recording channel is the multihead multitrack (MHMT) model, where the ISI is generally more severe than ITI. In an MHMT system, multiple read heads scan a group of data tracks simultaneously. Guard gaps are generally added between neighboring groups to prevent cross-talk [2], [54].

Magnetic recording schemes that jointly process multiple tracks were recognized in the early 1990s as a technique to potentially improve the areal density of HDDs [6]. Since then, they have been extensively studied [46, 51–53]. In [6], Barbosa implemented a maximum likelihood (ML) detector for an ideal 2H2T symmetric interfering magnetic recording system. Theoretical bounds on achievable performance of an ML detector for several MHMT channels were derived in [47]. Recently, MHMT detectors have been simulated on more complex channels which better approximate real HDDs. In [24], the ML detector along with a joint-track equalizer was simulated on a channel model for bit patterned media storage (BPMR). In [54], the authors derive a symbol-based maximum a posteriori (MAP) detector for a BPMR channel consisting of three heads and two tracks. A pattern-dependent Viterbi detector is given in [58] for an array-reader system. In [61], the implementation cost for realizing the optimal MHMT detection is studied. Many efforts have been devoted to finding low-cost suboptimal detection methods. In [13], the readback signals from multiple heads are equalized to a 1D target to reduce the detection complexity. ITI cancellation [20, 27, 41] is another effective technique to combat ITI with low implementation cost.

Compared to the 2D-ISI detector in Fig. 1.3, the MHMT detector has relatively lower complexity and smaller delay that could allow its adoption in the first generation TDMR drives. In this dissertation, we assume a rather simplified MHMT model, and address two fundamental problems with the ML detector. First, it is generally inefficient to implement ML detection in the conventional manner on MHMT models with varying ITI. Since the construction of ML trellis depends on ITI, a change in ITI estimates requires to recalculate the output labels, which could lead to intolerable delay. Second, as the number of tracks and channel memory increase, the computational complexity of ML detection exponentially grows so that implementing ML detector could be infeasible. Searching for suboptimal solutions that can significantly reduce the complexity without degrading performance too much is pressing. To solve the first problem, we propose an alternative way to implement the ML detection, called weighted sum subtract joint detector (WSSJD). We show that by applying a simple channel transformation, the ITI interfering channels in can be transformed to parallel subchannels, whose joint trellis is independent of ITI. We also propose an architecture with gain loops to efficiently track the ITI estimates. To mitigate the high complexity issue, we redefine the distance between input symbols, and apply the reduced-state sequence estimation (RSSE) [14] to the MHMT channel. Simulation results show that the reduced trellis with much fewer states could achieve near optimal performance. We also develop tools to analyze the error event probability of the RSSE algorithm, based on the work in [43].

1.3 Flash Memory

Although HDD has dominated the storage market for many decades, recent progress in solid state drive (SSD) technology has offered many advantages that is attracting more and more attention. An SSD is a non-volatile storage device that stores data by integrated circuits. Most SSDs use NAND-based flash memory as the storage media. For flash memories, data is recorded by charging memory cells with different amounts of electrons. The electrons are trapped in the memory cells even if power is interrupted. Data can be read back by thresholding the voltage of a memory cell. For single-level cell (SLC) flash, the cell charge is two-level. If the readback voltage is higher than the threshold, or "programmed", for short, the input bit is said to be 1; otherwise, it is output as 0. Thus, SLC stores a single bit in one cell. Recently, MLC (4-level) and TLC (8-level) flash memories have been commercialized. They can store multiple bits in one memory cell, thus offering higher storage densities.

Compared to the traditional HDD, flash memory has a lot of advantages, such as fast read/write speed, low power consumption, and better data integrity. It is more durable and quieter than HDD since there are no mechanical moving parts in flash memory. However, the limited lifetime and the high cost per bit are two main properties of flash drives that are less appealing than HDD. One of the most prominent features of flash memories is the asymmetry in programming and erasing. Namely, the level of flash memory cells can be easily increased by injecting electrons to their floating gates; however, to decrease the level of a single cell, the whole block of cells ($\sim 10^6$ cells) has to be erased and reprogrammed. The block erasure operation is not only time-consuming, but degrades the performance of flash memory cells as well. A typical 4-level flash cell can tolerate approximately 10^3 to 10^4 erasures before it no longer can be used due to the heavy programming noise.

Write-once memory (WOM) codes have been extensively studied as a coding technique for data storage with one-time programming properties, e.g., punch cards, optical disks, and flash memories. Considerable progress has been made to construct binary and non-binary WOM codes since the pioneering work in [39] and [55]. Different constructions based on algebraic codes [25], [59], and with error correction capability [60] were studied. For multi-level representation, WOM codes based on lattices were studied in [28], [7] and [8]. Some of the above codes have very good rate (close to capacity in [55]) and low encoding and decoding complexity. In all works mentioned, it is assumed that the WOM codes should guarantee t writes (i.e. information updating cycles) in the worst case.

In this dissertation, we study an alternate performance criterion for WOM codes, that is, the average number of writes before an erasure is required. This criterion is reasonable because the worst case does not represent a typical sequence of information stored. If we assume the information message follows a distribution over a finite set Σ, then the average number of writes would be linearly related to the expected lifetime of flash memory cells. Some initial results have been presented in [12] on the average-case performance of floating codes. In our work, we focus on developing tools to evaluate the average number of writes of different WOM codes.

1.4 Dissertation Overview

This dissertation is organized as follows.

In Chapter 2, we introduce the nHnT model and propose a modified ML detector, called weighted sum-subtract joint detector (WSSJD). We show that, by taking the eigenvalue decomposition of the channel interference matrix, and applying coordinate transformations in both the input and the output spaces, the cross-interfering channels are transformed into n separate sub-channels. The parameter that indicates the ITI level appears in the gain factor of each resulting sub-channel, and thus can be estimated by gain loops. The trellis of the transformed system remains the same under varying ITI levels, so adaptive estimates of ITI level can be easily incorporated into the detection architecture. Simulation results on 2H2T and 3H3T systems show that under time-varying ITI conditions, WSSJD with adaptively estimated ITI outperforms the traditional ML detector for which practical considerations dictate the use of a static ITI estimate. We also analyze the performance advantage of the nHnT detection system over ITI-free single-track detection by examining the distance properties of each system.

In Chapter 3, we show that the use of reduced-state sequence estimation (RSSE) algorithm can significantly reduce the complexity of the MHMT ML detector while achieving near optimal performance. We briefly review the original RSSE algorithm as developed for QAM transmission systems. Next, we show how to construct a reduced-state trellis for the symmetric 2H2T channel by redefining the distance measure in the input constellation and designing proper set partitioning trees. Performance simulation results for RSSE on 2H2T systems with different channel models and trellis configurations are provided. We define the early-merging condition and provide error event analysis. The dominant error events for several reduced-state trellises on different channels are also tabulated. The applicability of RSSE to an asymmetric 2H2T model is also considered. Finally, set partition trees of the 3H3T system are proposed and their performance is analyzed and simulated.

In Chapter 4, we generalize WSSJD to channels that satisfy an algebraic property called the WSSJD property. We first consider a 3H2T system, and analyze the achievable performance of the ML detector. Then, we derive the WSSJD algorithm for the 3H2T channel. We discuss three techniques associated with WSSJD: channel decomposition, reduced-complexity implementation, and ITI estimation. We define the WSSJD property, which is required for an MHMT system to be amenable to the application of WSSJD. Several examples of MHMT channels that satisfy the WSSJD property are given. We also discuss the application of WSSJD to other potential 3H2T models based on the work in [49].

In Chapter 5, we study the expected performance of WOM codes. We model the code rewriting process as a Markov process, and formally define the average number of writes of a given WOM code. Recursive formulas are derived to calculate their average number of writes. We propose a greedy algorithm to search for the optimal updating function for a given WOM labeling. The “optimality” is in the sense of maximizing the expected number of successful writes. We also prove the optimality of the tile-labeling construction [8] when the alphabet size is 3.

Chapter 2

Weighted Sum Subtract Joint Detection with ITI Estimation

Multitrack detection with array-head reading is a promising technique proposed for next generation magnetic storage systems. The multihead multitrack (MHMT) system is characterized by intersymbol interference (ISI) in the downtrack direction and intertrack interference (ITI) in the crosstrack direction. Constructing the trellis of a MHMT maximum likelihood (ML) detector requires knowledge of the ITI, which is generally unknown at the receiver. Furthermore, in a time-varying ITI environment, updating ML trellis labels using adaptively-generated ITI estimates could incur significant delay. In this chapter, we propose one approach to solve these issues. The proposed detector uses a different trellis structure whose output labels are independent of the ITI level, with ITI-dependence appearing only in a scale factor used to suitably weight the computed path metrics in order to retain ML optimality. The detector formulation facilitates the design of a gain loop structure that can track the time-varying ITI and provide ITI estimates to adaptively adjust the weights in the path metric evaluation. Simulation results show that the proposed detector architecture with ITI estimation offers a substantial performance advantage over ML detection using a static ITI estimate.

2.1 Symmetric n-Head n-Track Channel Model

In next-generation magnetic recording using shingled magnetic recording, the data tracks will be organized into bands, each consisting of a number of narrow, closely space tracks, with a small gap between bands to prevent interference [2].

Consider a band of n tracks. Let $x^i(D)$ denote the bipolar data sequence recorded on the i-th track, $x^i(D) = \Sigma_{k=0}^{N} x_k^i D^k$, with $x_k^i \in \{-1, +1\}$. We assume that x_k^i is i.i.d. and equiprobable, and $x^i(D), i \in \{1, ..., n\}$ are independent sequences. We also assume that there is no phase offset during the writing, i.e., the data sequences are perfectly aligned.

During the readback process, n heads are evenly placed over n tracks, with one head designated per track. The signal from each head is passed through a matched filter, a sampler, and then equalized to the target dipulse response represented by a polynomial $h(D) = h_0 + h_1 D + \cdots + h_\nu D^\nu$ that reflects the intersymbol interference (ISI). The interference from neighboring tracks is additive, and the interfering by a read head from a neighboring track is assumed to be a scaled version of the read signal that would be sensed by the head directly over the neighboring track in the absence of any intertrack interference. Let $r^i(D)$ denote the sampled readback samples corresponding to the signal from the head corresponding to the i-th track, $i = 1, ..., n$. The resulting n-head n-track (nHnT) system is described by

$$\mathbf{R}(D) = \mathbf{A}_n \mathbf{X}(D) h(D) + \mathbf{\Omega}(D), \tag{2.1}$$

where

$$\mathbf{X}(D) = [x^1(D), \cdots, x^n(D)]^\top \tag{2.2}$$

is the input vector, and

$$\mathbf{R}(D) = [r^1(D), \cdots, r^n(D)]^\top \tag{2.3}$$

is the output vector. The vector of electronic noise components is denoted by

$$\mathbf{\Omega}(D) = [\omega^1(D), \cdots, \omega^n(D)]^\top. \tag{2.4}$$

We assume that the noise samples are independent and Gaussian distributed, with zero mean and variance σ^2. The term $\mathbf{X}(D)h(D) = [x^1(D)h(D), \cdots, x^n(D)h(D)]^\top$ denotes the vector of noiseless ISI channel outputs, and $\mathbf{A}_n$ is an $n \times n$ interference matrix. In our model, we assume that only adjacent tracks interfere. This assumption is reasonable since in most cases the ITI from the adjacent tracks is the dominant one. Additionally, by assuming the physical uniformity and symmetry of read heads, $\mathbf{A}_n$ can be modeled as a symmetric tridiagonal Toeplitz matrix

$$\mathbf{A}_n = \begin{bmatrix} 1 & \epsilon & & & \\ \epsilon & 1 & \ddots & \mathbf{0} & \\ & \ddots & \ddots & \ddots & \\ & \mathbf{0} & \ddots & 1 & \epsilon \\ & & & \epsilon & 1 \end{bmatrix}, \tag{2.5}$$

where $\epsilon \in [0, 0.5]$ represents the ITI level determined by the distance between the head and the adjacent track.

Given the received sequences $\mathbf{R}(D)$, the ML detector chooses $\mathbf{X}^*(D)$ that satisfies

$$\begin{aligned} \mathbf{X}^*(D) &= \arg\max_{\mathbf{X}(D)} \Pr(\mathbf{R}(D)|\mathbf{X}(D)) \\ &= \arg\min_{\mathbf{X}(D)} \|\mathbf{R}(D) - A_n\mathbf{X}(D)h(D)\|^2, \end{aligned} \tag{2.6}$$

where $\|\cdot\|^2$ denotes the squared Euclidean norm. For instance, $\|\mathbf{X}(D)\|^2$ is calculated by

$$\|\mathbf{X}(D)\|^2 = \sum_i \|x^i(D)\|^2 = \sum_{i,j} (x^i_j)^2. \tag{2.7}$$

The trellis constructed in the ML detector contains $2^{n\nu}$ states, each of which is associated with 2^n

edges.

Let $x^i(D)$ and $\hat{x}^i(D)$ be the correct and estimated input sequences corresponding to track i, respectively. An error event occurs if for some track i, $e^i(D) = x^i(D) - \hat{x}^i(D)$ is not zero. For an error event

$$\boldsymbol{e}(D) = [e^1(D), \cdots, e^n(D)]^\top, \tag{2.8}$$

the distance parameter is calculated by

$$d^2(\boldsymbol{e}(D)) = \|\mathbf{A}_n \boldsymbol{e}(D) h(D)\|^2 = \sum_{i=1}^{n} \|y^i(D)\|^2, \tag{2.9}$$

where

$$y^1(D) = [e^1(D) + \epsilon e^2(D)]h(D), \tag{2.10}$$

$$y^n(D) = [e^n(D) + \epsilon e^{n-1}(D)]h(D), \tag{2.11}$$

$$y^i(D) = [e^i(D) + \epsilon e^{i-1}(D) + \epsilon e^{i+1}(D)]h(D), i \in [2, n-1]. \tag{2.12}$$

The minimum distance parameter of the channel is obtained by minimizing $d^2(\boldsymbol{e}(D))$ over all possible $\boldsymbol{e}(D)$. In Fig. 2.1, we plot the minimum distances of the nHnT ML detector for $n = 2, 3, 4, 5$, assuming $h(D) = 1 + D$. It can be observed that in a large region of ϵ, the nHnT ML detectors have a greater minimum distance parameter than the ITI-free SHST ML detector. The distance parameters associated with the other detectors plotted in Fig. 2.1 will be discussed in Section 2.2.3.

It is well known that the error event probability of a trellis-based detector can be approximated as $\mathrm{P_e} \propto Q(\frac{d_{\text{min}}}{2\sigma})$, where $Q(\cdot)$ is the tail probability of the standard Gaussian distribution, d_{min} is the minimum distance, and σ is the standard deviation of the additive Gaussian noise. The performance of the detector can be accurately predicted by analyzing its minimum distance.

We notice that the calculation of trellis output labels, $\mathbf{Y}(D) = \mathbf{A}_n\mathbf{X}(D)h(D)$, requires knowledge of the ITI level ϵ. This can introduce disadvantages in the hardware realization

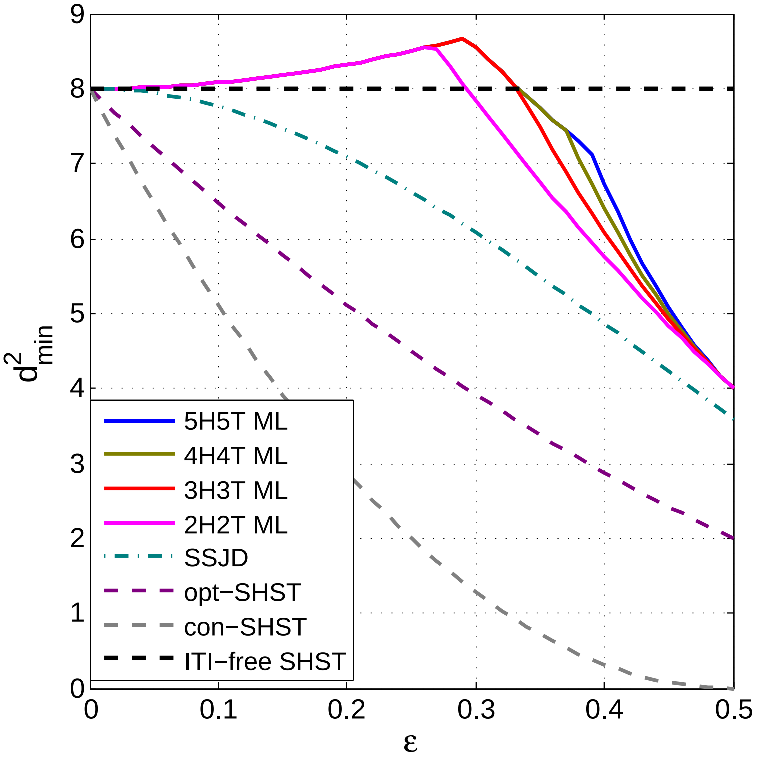


Figure 2.1: Minimum squared-distance comparison of different detectors on channel $h(D) = 1 + D$ with $d_0^2 = 8$.

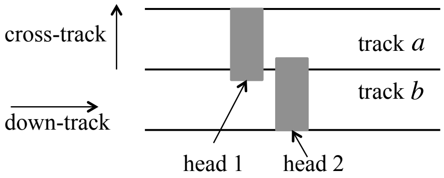


Figure 2.2: Schematic of a two-head two-track recording system.

of the conventional ML detector. With time-varying ITI levels, the detector in effect has to recalculate and update the branch output labels whenever ϵ changes. If ϵ changes frequently, this process could incur a substantial increase in hardware complexity and delay. Therefore, using a conventional approach to implementation, MHMT ML detection is essentially limited to a static value of ϵ. Indeed, in the prior literature, studies of MHMT ML detection have always assumed a static ITI environment [46], [6], [30]. On a hard disk drive, however, ϵ generally varies spatially due to mechanical effects such as head skew and flying height variation. Thus, adaptive estimation of ϵ will be necessary, potentially introducing the complexity and delay issues just mentioned if conventional ML detection is used.

In the following sections, we propose a novel detection architecture that is amenable to incorporating adaptive ITI estimation while retaining the optimality of ML detection. The proposed detector uses a different trellis diagram from the conventional ML detector. For convenience, we refer to the latter as the "ML trellis" even though both detectors produce the same ML output sequences.

2.2 WSSJD on 2H2T system

The weighted sum-subtract joint detection (WSSJD) algorithm differs from the conventional ML detector in two respects. First, it has a "sum-subtract" preprocessor before the Viterbi detector. Second, it uses weighted branch metrics in detection. In this section, we assume ϵ to be known. This condition will be relaxed in Section 2.3 where we show that ϵ acts as a gain factor that can be estimated by means of a first-order gain loop.

We begin the presentation of WSSJD by first analyzing the 2H2T case. A schematic of the

2H2T system is shown in Fig. 2.2. Let $x^a(D)$ and $x^b(D)$ be the input sequences corresponding to track a and track b, respectively. The outputs of the corresponding read heads are

$$\begin{aligned} r^a(D) &= y^a(D) + \omega^a(D), \\ r^b(D) &= y^b(D) + \omega^b(D), \end{aligned} \tag{2.13}$$

where

$$\begin{aligned} y^a(D) &= x^a(D)h(D) + \epsilon\, x^b(D)h(D), \\ y^b(D) &= \epsilon\, x^a(D)h(D) + x^b(D)h(D), \end{aligned} \tag{2.14}$$

are the noiseless outputs. As given in [46], the minimum distance parameter of the 2H2T ML detector is

$$d^2_{\text{min, ML}} = \begin{cases} (1+\epsilon^2)d_0^2, & \text{if } 0 \le \epsilon \le 2-\sqrt{3} \\ 2(1-\epsilon)^2 d_0^2, & \text{if } 2-\sqrt{3} \le \epsilon \le 1/2 \end{cases} \tag{2.15}$$

where d_0 is the minimum distance of the single-track detector on channel $h(D)$ when there is no ITI. The single track error events are the dominant error patterns at low ITI, while the double track error events are the dominant ones at high ITI. The operating point that gives the highest minimum distance, or the best performance of the ML detector, is at $\epsilon = 2-\sqrt{3}$.

2.2.1 Sum-subtract preprocessing

Instead of directly processing $r^a(D)$ and $r^b(D)$, the WSSJD method first calculates the weighted sum and difference, $r^+(D)$ and $r^-(D)$, given by

$$\begin{aligned} r^+(D) &= \frac{1}{1+\epsilon}\,[\, r^a(D) + r^b(D)\,], \\ r^-(D) &= \frac{1}{1-\epsilon}\,[\, r^a(D) - r^b(D)\,], \end{aligned} \tag{2.16}$$

respectively. Defining new input signals by

$$z^{+}(D) = x^{a}(D) + x^{b}(D), \quad z^{-}(D) = x^{a}(D) - x^{b}(D), \tag{2.17}$$

and the corresponding noiseless output signals by

$$y^{+}(D) = z^{+}(D)h(D), \quad y^{-}(D) = z^{-}(D)h(D), \tag{2.18}$$

we can rewrite (**??**) as

$$r^{+}(D) = y^{+}(D) + \omega^{+}(D), \quad r^{-}(D) = y^{-}(D) + \omega^{-}(D). \tag{2.19}$$

The new noise components,

$$\omega^{+}(D) = \frac{1}{1+\epsilon}\left(\omega^{a}(D) + \omega^{b}(D)\right), \quad \omega^{-}(D) = \frac{1}{1-\epsilon}\left(\omega^{a}(D) - \omega^{b}(D)\right) \tag{2.20}$$

satisfy $\omega_k^{+} \sim \mathcal{N}(0, \frac{2\sigma^2}{(1+\epsilon)^2})$, $\omega_k^{-} \sim \mathcal{N}(0, \frac{2\sigma^2}{(1-\epsilon)^2})$. Furthermore,

$$E[\,\omega_k^{+}\,\omega_k^{-}\,] = \frac{1}{1-\epsilon^2}(E[\,\omega_k^{a2}\,] - E[\,\omega_k^{b^2}\,]\,) = 0, \tag{2.21}$$

which implies that $\omega^{+}(D)$ and $\omega^{-}(D)$ are uncorrelated and, therefore, independent.

We can think of $r^{+}(D)$ and $r^{-}(D)$ as the noisy outputs obtained by passing each of $z^{+}(D)$ and $z^{-}(D)$ through a channel $h(D)$, but with different noise powers. These two channels are called the "sum channel" and the "subtract channel", respectively. Notice that the new inputs z_k^{+} and z_k^{-} have a three-level alphabet, $\mathcal{B} = \{-2, 0, 2\}$. There is a one-to-one mapping between (z_k^{+}, z_k^{-}) and (x_k^{a}, x_k^{b}), as shown in Table 2.1.

Since $r^{+}(D)$ and $r^{-}(D)$ are obtained from separate channels, one can independently detect $z^{+}(D)$ and $z^{-}(D)$, and then map (z_k^{+}, z_k^{-}) to (x_k^{a}, x_k^{b}) according to Table 2.1. This

Table 2.1: Mapping between (x_k^a, x_k^b) and (z_k^+, z_k^-).

x_k^a	x_k^b	z_k^+	z_k^-
1	1	2	0
1	-1	0	2
-1	1	0	-2
-1	-1	-2	0

corresponds to solving for

$$\begin{aligned}\hat{z}^+(D) &= \underset{z^+(D)}{\arg\max} \log \Pr(r^+(D) \mid z^+(D)) = \underset{z^+(D)}{\arg\min} \|r^+(D) - z^+(D)\|^2 , \\ \hat{z}^-(D) &= \underset{z^-(D)}{\arg\max} \log \Pr(r^-(D) \mid z^-(D)) = \underset{z^-(D)}{\arg\min} \|r^-(D) - z^-(D)\|^2 . \end{aligned} \tag{2.22}$$

This computation has complexity $O(2 \cdot 3^\nu)$. However, it is suboptimal. From Table 2.1 we see that $z^+(D)$ and $z^-(D)$ are not independent, e.g., $z_k^+ = 2$ forces z_k^- to be 0. Independent detection ignores this correlation and produces some undecodable $(\hat{z}_k^+, \hat{z}_k^-)$ pairs. Optimal detection must jointly consider both the sum channel and the subtract channel, determining

$$\hat{z}^+(D), \hat{z}^-(D) = \underset{z^+(D), z^-(D)}{\arg\max} \log \Pr(r^+(D), r^-(D) \mid z^+(D), z^-(D)). \tag{2.23}$$

The Viterbi algorithm (VA) is applied to solve (2.23). The WSSJD trellis has the same number of states as the ML trellis. Each branch connects an initial state

$$s(k) = [z_{k-\nu}^+ \ldots z_{k-1}^+, z_{k-\nu}^- \ldots z_{k-1}^-]$$

to a terminal state

$$s(k+1) = [z_{k-\nu+1}^+ \ldots z_k^+, z_{k-\nu+1}^- \ldots z_k^-]$$

with input label $\mathcal{L}_{\text{in}} = (z_k^+, z_k^-)$ and output label $\mathcal{L}_{\text{out}} = (y_k^+, y_k^-)$. Fig. 2.3 shows a WSSJD trellis for the channel $h(D) = 1 + D$. The text to the left of each state lists the branch labels in the form of input/output. Notice that, unlike the ML trellis, the WSSJD trellis labels are independent

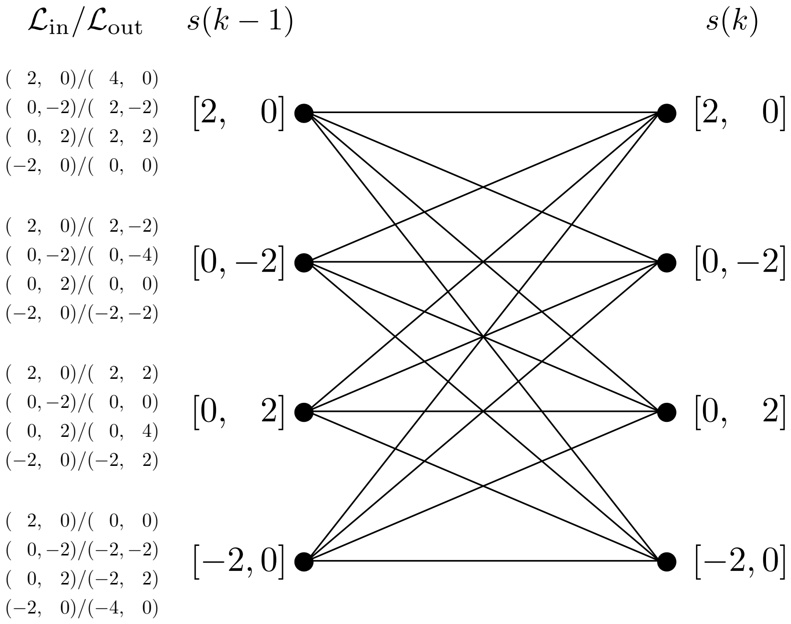


Figure 2.3: WSSJD trellis for channel $h(D) = 1 + D$

of ϵ.

2.2.2 Weighted branch metric

Since the sum and the subtract channel have different noise powers, WSSJD computes a weighted sum of their individual distance metrics, $\|r^{+}(D) - y^{+}(D)\|^2$ and $\|r^{-}(D) - y^{-}(D)\|^2$. The optimal choice of the weights is found by evaluating (2.23),

$$\begin{aligned}
\hat{z}^{+}(D), \hat{z}^{-}(D) &= \underset{z^{+}(D), z^{-}(D)}{\arg\max} \; \log \Pr(r^{+}(D), r^{-}(D) | z^{+}(D), z^{-}(D)) \\
&= \underset{z^{+}(D), z^{-}(D)}{\arg\max} \; \log \Pr(r^{+}(D) | z^{+}(D)) + \log \Pr(r^{-}(D) | z^{-}(D)) \\
&= \underset{z^{+}(D), z^{-}(D)}{\arg\min} \; \frac{\|r^{+}(D) - y^{+}(D)\|^2}{2\sigma^2/(1+\epsilon)^2} + \frac{\|r^{-}(D) - y^{-}(D)\|^2}{2\sigma^2/(1-\epsilon)^2} \\
&= \underset{z^{+}(D), z^{-}(D)}{\arg\min} \; (1+\epsilon)^2 \|r^{+}(D) - y^{+}(D)\|^2 + (1-\epsilon)^2 \|r^{-}(D) - y^{-}(D)\|^2.
\end{aligned} \tag{2.24}$$

Let $M_k(s)$ denote the survivor path metric for state s at time k. The path metric corresponding to the extension along a branch from state s to s' is

$$M_{k+1}(s') = M_k(s) + (1+\epsilon)^2(r_k^+ - y_k^+)^2 + (1-\epsilon)^2(r_k^- - y_k^-)^2. \tag{2.25}$$

The term $m_k(s,s') = (1+\epsilon)^2(r_k^+ - y_k^+)^2 + (1-\epsilon)^2(r_k^- - y_k^-)^2$ is the weighted branch metric.

Since the sum-subtract transformation is bijective, we have

$$\Pr(r^+(D), r^-(D)|z^+(D), z^-(D)) = \Pr(r^a(D), r^b(D)|x^a(D), x^b(D)). \tag{2.26}$$

Therefore, WSSJD gives the ML solution.

Assume $(z^+(D), z^-(D))$ are the correct input sequences. WSSJD outputs wrong estimates $(\hat{z}^+(D), \hat{z}^-(D))$ if

$$\Pr(r^+(D), r^-(D)|z^+(D), z^-(D)) < \Pr(r^+(D), r^-(D)|\hat{z}^+(D), \hat{z}^-(D)). \tag{2.27}$$

Let $e^+(D) = z^+(D) - \hat{z}^+(D)$ and $e^-(D) = z^-(D) - \hat{z}^-(D)$ be the error event. Notice that the alphabet of e_k^+ and e_k^- is $\{\pm 4, \pm 2, 0\}$, and e_k^+ and e_k^- are not independent, e.g., $e_k^+ = 4$ implies $e^- = 0$. The probability of having $(e^+(D), e^-(D))$ is approximated by $Q(\frac{d_{\text{WSSJD}}(e^+(D), e^-(D))}{2\sigma})$, where

$$\begin{aligned} &d^2_{\text{WSSJD}}(e^+(D), e^-(D)) \\ &\quad = \frac{(1+\epsilon)^2\|e^+(D)h(D)\|^2 + (1-\epsilon)^2\|e^-(D)h(D)\|^2}{2} \end{aligned} \tag{2.28}$$

is the effective distance parameter defined for WSSJD. Although the labels on the WSSJD trellis are independent of ϵ, the distance metric used to estimate the WSSJD performance has to incorporate the effect of the signal-to-noise ratio (SNR) differences in the sum and subtract channels. Evaluating (2.28) for all possible error events shows that WSSJD has the same minimum distance parameter as the ML detector.

2.2.3 Performance loss from neglecting branch metric weighting factors

We refer to the detector that ignores the weighting factors, i.e., that uses

$$m_k(s,s') = (r_k^+ - y_k^+)^2 + (r_k^- - y_k^-)^2 \tag{2.29}$$

as the branch metric, as the sum-subtract joint detector. The corresponding sum-subtract joint detection (SSJD) method is suboptimal. The performance loss of SSJD is reflected in its minimum distance parameter. Let $(z^+(D), z^-(D))$ and $(\hat{z}^+(D), \hat{z}^-(D))$ be the correct and estimated sequences. The error event probability is

$$\begin{aligned}&\Pr(\|r^+(D) - z^+(D)h(D)\|^2 + \|r^-(D) - z^-(D)h(D)\|^2 \\ &\quad > \|r^+(D) - \hat{z}^+(D)h(D)\|^2 + \|r^-(D) - \hat{z}^-(D)h(D)\|^2) \\ &= Q\left(\frac{d_{\text{SSJD}}(e^+(D), e^-(D))}{2\sigma}\right),\end{aligned} \tag{2.30}$$

where

$$d_{\text{SSJD}}(e^+(D), e^-(D)) = \frac{\|e^+(D)h(D)\|^2 + \|e^-(D)h(D)\|^2}{\sqrt{\frac{2\|e^+(D)h(D)\|^2}{(1+\epsilon)^2} + \frac{2\|e^-(D)h(D)\|^2}{(1-\epsilon)^2}}}. \tag{2.31}$$

Since $e^+(D)$ and $e^-(D)$ are not independent, we express them as $e^+(D) = e^a(D) + e^b(D)$ and $e^-(D) = e^a(D) - e^b(D)$ to find $d^2_{\text{min, SSJD}}$. To simplify the notation, let $A(D) = e^a(D)h(D)$ and $B(D) = e^b(D)h(D)$. We have

$$\begin{aligned}&d^2_{\text{SSJD}}(e^a(D), e^b(D)) \\ &\quad = \frac{(1+\epsilon)^2(1-\epsilon)^2(\|A(D)\|^2 + \|B(D)\|^2)^2}{(1+\epsilon^2)(\|A(D)\|^2 + \|B(D)\|^2) - 4\epsilon\langle A(D), B(D)\rangle}.\end{aligned} \tag{2.32}$$

Consider the case of a single-track error event, e.g., assume $e^b(D) = 0$. Then

$$\begin{aligned} d^2_{\text{SSJD}}(e^a(D), 0) &= \frac{(1+\epsilon)^2(1-\epsilon)^2}{1+\epsilon^2}\|A(D)\|^2 \\ &\geq \frac{(1+\epsilon)^2(1-\epsilon)^2}{1+\epsilon^2} d_0^2 \end{aligned} \tag{2.33}$$

with equality achieved when $e^a(D)$ gives the minimum distance d_0^2 on channel $h(D)$.

For the case of a double-track error event we have

$$\begin{aligned} d^2_{\text{SSJD}}(e^a(D), e^b(D)) &\geq (1-\epsilon)^2(\|A(D)\|^2 + \|B(D)\|^2) \\ &\geq 2(1-\epsilon)^2 d_0^2, \end{aligned} \tag{2.34}$$

where we have used the fact that

$$-\langle A(D), B(D)\rangle \leq \|A(D)\|\,\|B(D)\| \leq \frac{1}{2}(\|A(D)\|^2 + \|B(D)\|^2). \tag{2.35}$$

Equality is achieved in (2.34) when $e^a(D) = -e^b(D)$ and both $e^a(D)$ and $e^b(D)$ lead to the minimum distance d_0 on channel $h(D)$. Comparison between (2.33) and (2.34) shows that, in contrast to WSSJD, the minimum distance of SSJD is always dominated by single-track error events. Therefore

$$d^2_{\text{min, SSJD}} = \frac{(1+\epsilon)^2(1-\epsilon)^2}{1+\epsilon^2} d_0^2. \tag{2.36}$$

In Fig. 2.1 we plot the squared minimum distance parameters for several detectors as a function of ϵ. Recall that WSSJD on the 2H2T model has the same d^2_{min} as the 2H2T ML detector. Two SHST detectors [47] are included for comparison purposes.The optimal SHST detector estimates the data on a track using the model for channel ISI and the interference induced by the side track. Its minimum distance is dominated by double-track error events, leading to

$$d^2_{\text{min, opt-SHST}} = (1-\epsilon)^2 d_0^2. \tag{2.37}$$

The conventional SHST detector considers only the channel ISI and treats the side track interference as additional electronic noise, and is therefore suboptimal. For the channel with $h(D) = 1 + D$, the minimum distance has a closed form expression given by

$$d^2_{\text{min, con-SHST}} = (1 - 2\epsilon)^2 d_0^2. \tag{2.38}$$

Finally, the ITI-free SHST corresponds to the SHST channel model with no ITI. The performance of the ITI-free SHST detector can be viewed as the performance of an ideal ITI cancellation scheme in which the side track response is perfectly removed.

We summarize the proposed algorithms as follows. The branch labels of the trellis constructed for WSSJD or SSJD are independent of the ITI level ϵ. This independence is the key property underlying the proposed architecture for combining WSSJD or SSJD with adaptive estimation of ϵ. In WSSJD, the ITI level ϵ is used to weight the branch metrics, thereby ensuring that it achieves ML performance. The SSJD algorithm does not weight the branch metrics, thus producing suboptimal decisions.

2.3 Adaptive ITI Level Estimation

2.3.1 ITI Sensitivity

To evaluate the sensitivity of the various detectors to a small change in the ITI level, we introduce a small offset into our performance simulations. Suppose the nominal ITI level is ϵ_0, while the true ITI level is adjusted by an offset $\Delta\epsilon$. The new noiseless channel outputs become

$$\begin{aligned} y^a(D) &= x^a(D)h(D) + (\epsilon_0 + \Delta\epsilon)\, x^b(D)h(D), \\ y^b(D) &= x^b(D)h(D) + (\epsilon_0 + \Delta\epsilon)\, x^a(D)h(D). \end{aligned} \tag{2.39}$$

If the detectors use the nominal level ϵ_0, rather than the true level $\epsilon = \epsilon_0 + \Delta\epsilon$, mismatch will lead to a degradation in performance.

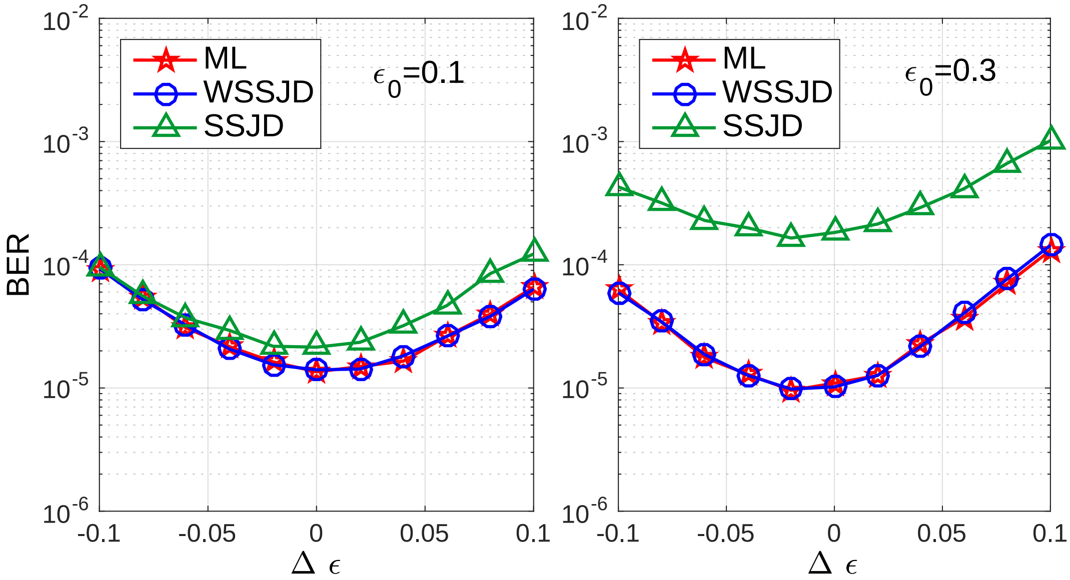


Figure 2.4: ITI sensitivity of different detectors on channel $h(D) = 1 + D$ at $\epsilon_0 = 0.1$ (left) and $\epsilon_0 = 0.3$ (right). SNR=10dB.

Fig. 2.4 shows the simulated bit error rate (BER) as a function of the mismatch $\Delta\epsilon$ for WSSJD, SSJD, and the conventional ML detector on channel the channel with $h(D) = 1 + D$ at SNR $= 10$dB, with $\epsilon_0 = 0.1$ and $\epsilon_0 = 0.3$, respectively. We see that the minimum BER occurs at $\Delta\epsilon \approx 0$ for $\epsilon_0 = 0.1$ and at $\Delta\epsilon = -0.02$ for $\epsilon_0 = 0.3$. However, the BER performance does not vary significantly in the interval $|\Delta\epsilon| \leq 0.02$ in either case. It is evident that the BER curves are not symmetric about $\Delta\epsilon = 0$. When $\epsilon_0 = 0.1$, the slope of the BER curve in the region $\Delta\epsilon < 0$ is slightly higher than that in the region $\Delta\epsilon > 0$. On the other hand, when $\epsilon_0 = 0.3$, the asymmetry is reversed, and the difference between the slopes is more significant.

Fig. 2.1 suggests that the observed behaviors are due to minimum distance properties of the mismatched detectors. To see this, consider the ML detector as an example. The probability of having an error event $(e^a(D), e^b(D))$ when the system has an offset $\Delta\epsilon$ is

$$\mathrm{P_e} = Q\left(\frac{1}{2\sigma} d(e^a, e^b, x^a, x^b)\right) = Q\left(\frac{1}{2\sigma}(d_{\text{ideal}} + d_{\text{mism}})\right), \tag{2.40}$$

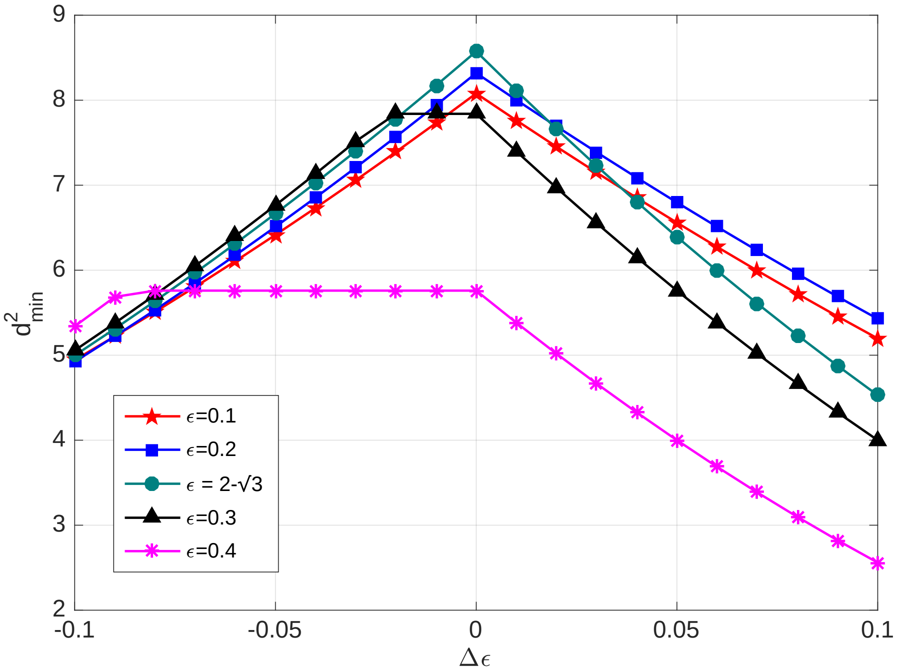


Figure 2.5: Minimum distance parameter of the ML detector at different levels of mismatch for channel $1 + D$.

where

$$d_{\text{ideal}} = \sqrt{\|\mathcal{A}(D)\|^2 + \|\mathcal{B}(D)\|^2}, \tag{2.41}$$

$$d_{\text{mism}} = 2\Delta\epsilon \frac{\langle \mathcal{A}(D), x^b(D)h(D)\rangle + \langle \mathcal{B}(D), x^a(D)h(D)\rangle}{\sqrt{\|\mathcal{A}(D)\|^2 + \|\mathcal{B}(D)\|^2}}, \tag{2.42}$$

$$\mathcal{A}(D) = e^a(D)h(D) + \epsilon \cdot e^b(D)h(D), \quad \mathcal{B}(D) = e^b(D)h(D) + \epsilon \cdot e^a(D)h(D). \tag{2.43}$$

Compared to the ideal case, d_{mism} is the additional effect caused by the mismatch. Notice that with the existence of mismatch, the distance parameter is now dependent on the input sequence $(x^a(D), x^b(D))$. In addition, having mismatch does not always decrease the distance. Some sequence combinations could lead to larger distance than the ideal case. The error event probability is dominated by the sequence combination (e^a, e^b, x^a, x^b) that leads to the smallest value of $d_{\text{ideal}} + d_{\text{mism}}$. Finding such a combination is not an easy task because $(e^a(D), e^b(D))$ and $(x^a(D), x^b(D))$ are not independent. For example, $e^a_k = 2$ forces x^a_k to be 1. Due to this correlation, it is hard to obtain an explicit expression for the minimum distance of a general channel polynomial. But for the channel with $h(D) = 1 + D$, we show in the Appendix that the minimum distance of the single-track error events is

$$d_{\text{s}}^2 = \begin{cases} \frac{8(1+\epsilon_0^2-2\Delta\epsilon)^2}{1+\epsilon_0^2} & \text{if } \Delta\epsilon > 0 \\ \frac{8[1+\epsilon_0^2+(2+2\epsilon_0)\Delta\epsilon]^2}{1+\epsilon_0^2} & \text{if } \Delta\epsilon < 0. \end{cases} \tag{2.44}$$

With the additional assistance of computer search, we also show that the minimum distance produced by double-track error events is

$$d_{\text{d}}^2 = \begin{cases} 16[(1-\epsilon_0) - 2\Delta\epsilon]^2 & \text{if } \Delta\epsilon > 0 \\ 16(1-\epsilon_0)^2 & \text{if } \Delta\epsilon < 0. \end{cases} \tag{2.45}$$

The distance values, d_{s}^2 and d_{d}^2, are achieved by the single track error events and double track error events that minimize d_{ideal}, respectively. Table 2.2 gives examples of sequence combinations that achieve d_{s}^2 and d_{d}^2. The overall minimum distance of the system is

$$d_{\text{min}}^2 = \min\{d_{\text{s}}^2,\, d_{\text{d}}^2\}. \tag{2.46}$$

In summary, the asymmetry of the BER curve about $\Delta\epsilon = 0$ is due to the correlation between $(e^a(D), e^b(D))$ and $(x^a(D), x^b(D))$. The reason that minimum BER points for $\epsilon_0 = 0.1$ and $\epsilon_0 = 0.3$ occur at different values of the mismatch $\Delta\epsilon$ is because at $\epsilon_0 = 0.1$ the system performance is largely dominated by the single track error events, while at $\epsilon_0 = 0.3$ the double track error events are dominant. Fig. 2.5 shows the overall minimum distance d_{min}^2 as a function of the mismatch $|\Delta\epsilon| \leq 1$ for several values of ϵ between 0.1 and 0.4. Comparing these results to the BER curves in Fig. 2.4, we find that for $\epsilon_0 = 0.1$, a positive mismatch produces a higher d_{min}^2 than a negative offset of the same magnitude. For $\epsilon_0 = 0.3$, this situation is reversed, and in a small range of negative offsets, $-0.02 \leq \Delta\epsilon \leq 0$, the mismatch does not reduce the minimum

Table 2.2: Sequences achieving d_{min} in (2.46) under positive/negative offsets for (a) single-track error events, and (b) double-track error events.

(a) single track error events

$\Delta\epsilon < 0$	$\boldsymbol{e}^a = \cdots, \quad 0, \quad 0, \quad 2, \quad 0, \quad 0, \cdots$ $\boldsymbol{x}^a = \cdots, x^a_{k-2}, -1, +1, -1, x^a_{k+2}, \cdots$ $\boldsymbol{x}^b = \cdots, x^b_{k-2}, -1, -1, -1, x^b_{k+2}, \cdots$
$\Delta\epsilon > 0$	$\boldsymbol{e}^a = \cdots, \quad 0, \quad 0, \quad 2, \quad 0, \quad 0, \cdots$ $\boldsymbol{x}^a = \cdots, x^a_{k-2}, +1, +1, +1, x^a_{k+2}, \cdots$ $\boldsymbol{x}^b = \cdots, x^b_{k-2}, +1, +1, +1, x^b_{k+2}, \cdots$

(b) double track error events

$\Delta\epsilon < 0$	$\boldsymbol{e}^a = \cdots, \quad 0, \quad 0, \quad 2, \quad 0, \quad 0, \cdots$ $\boldsymbol{e}^b = \cdots, \quad 0, \quad 0, -2, \quad 0, \quad 0, \cdots$ $\boldsymbol{x}^a = \cdots, x^a_{k-2}, -1, +1, -1, x^a_{k+2}, \cdots$ $\boldsymbol{x}^b = \cdots, x^b_{k-2}, +1, -1, +1, x^b_{k+2}, \cdots$
$\Delta\epsilon > 0$	$\boldsymbol{e}^a = \cdots, \quad 0, \quad 0, \quad 2, \quad 0, \quad 0, \cdots$ $\boldsymbol{e}^b = \cdots, \quad 0, \quad 0, -2, \quad 0, \quad 0, \cdots$ $\boldsymbol{x}^a = \cdots, x^a_{k-2}, +1, +1, +1, x^a_{k+2}, \cdots$ $\boldsymbol{x}^b = \cdots, x^b_{k-2}, -1, -1, -1, x^b_{k+2}, \cdots$

distance of the system. In this case, the mismatch also reduces the probability of worst case error events, leading to a shift of the minimum BER to the negative side.

2.3.2 Gain Loop

Recall that in the sum-subtract preprocessing, ϵ appears in the gain factors that normalize $r^+(D)$ and $r^-(D)$. We rewrite (2.16) as

$$r^+(D) = g^+ \left[\, r^a(D) + r^b(D) \,\right], r^-(D) = g^- \left[\, r^a(D) - r^b(D) \,\right], \tag{2.47}$$

where g^+, g^- are the gain factors with true values $\frac{1}{1+\epsilon}$ and $\frac{1}{1-\epsilon}$, respectively. There are several well-known ways to estimate g^+ and g^-[29]. In our work, we adopt the LMS adaptive algorithm

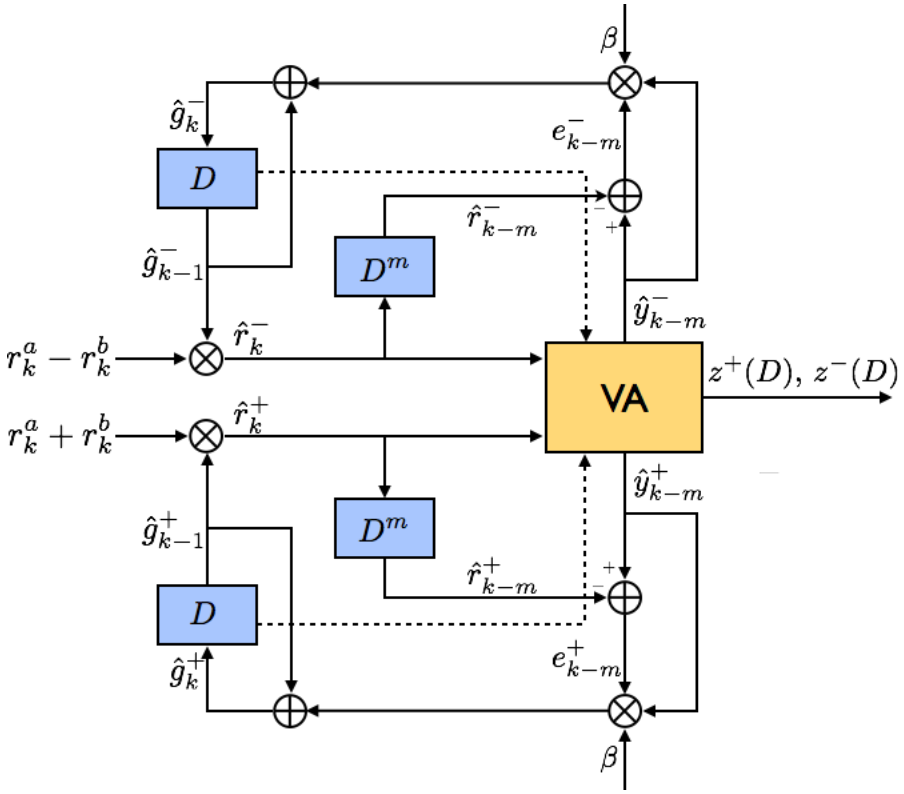


Figure 2.6: WSSJD with gain loops to adaptively estimate the ITI level

due to its simplicity and good convergence properties. For $\hat{g}^+$, the updating rule is given by

$$\hat{r}_k^+ = \hat{g}_{k-1}^+ \,(r_k^a + r_k^b), \tag{2.48}$$

$$e_k = \hat{y}_k^+ \,-\, \hat{r}_k^+, \tag{2.49}$$

$$\hat{g}_k^+ = \hat{g}_{k-1}^+ + \beta\, \hat{y}_k^+ \, e_k. \tag{2.50}$$

The step-size parameter β controls the convergence speed. Larger β leads to faster convergence, but also results in larger error variance. Note that $\hat{y}_k^+$ represents the instantaneous hard decision made by the Viterbi detector. The use of hard decisions can potentially lead to growing estimation error at low SNR. To mitigate this, one can introduce a small delay $m \geq 1$ to get more accurate tentative decisions[1]. In this case, (2.49) and (2.50) become

$$e_{k-m} = \hat{y}_{k-m}^+ \,-\, \hat{r}_{k-m}^+, \tag{2.51}$$

$$\hat{g}_k^+ \quad = \hat{g}_{k-1}^+ + \beta\, \hat{y}_{k-m}^+ \, e_{k-m}. \tag{2.52}$$

Similarly, $\hat{g}_k^-$ can be estimated in the same manner. The estimates $\hat{g}_k^+$ and $\hat{g}_k^-$ will be fed back to the Viterbi detector to evaluate path metrics, i.e.,

$$M_{k+1}(s') = M_k(s) + {\hat{g}_{k-1}^+}^2 (r_k^+ - y_k^+)^2 + {\hat{g}_{k-1}^-}^2 (r_k^- - y_k^-)^2. \tag{2.53}$$

Fig. 2.6 shows a complete block diagram for WSSJD with adaptive gain estimation. The system contains two separate gain loops for $\hat{g}_k^+$ and $\hat{g}_k^-$. While a combined loop for estimating $\hat{g}_k^+$ and $\hat{g}_k^-$ can provide a better estimate for ϵ, using separate loops achieves similar performance in a more efficient way.

In our simulations, $\hat{g}_0^+$ and $\hat{g}_0^-$ are initially set to 1. At time k, $r_k^a + r_k^b$ and $r_k^a - r_k^b$ are normalized by the previously estimated $\hat{g}_{k-1}^+$ and $\hat{g}_{k-1}^-$, respectively. The resulting signals $\hat{r}_k^+$ and $\hat{r}_k^-$ are sent to the Viterbi detector. The path metric of each trellis state is calculated by

---

[1] The use of soft decisions would lead to even further improvement in the gain estimation, at the cost of increased complexity.

using weights $\hat{g}_{k-1}^{+}$ and $\hat{g}_{k-1}^{-}$. The Viterbi detector picks the most likely path, and makes an instantaneous decision on $\hat{y}_{k-m}^{+}$ and $\hat{y}_{k-m}^{-}$. The error signal is calculated to update $\hat{g}_{k}^{+}$ and $\hat{g}_{k}^{-}$. Note that SSJD can also work with these gain loops, without feeding $\hat{g}_{k}^{+}$ and $\hat{g}_{k}^{-}$ to the path metric evaluation.

In Fig. 2.7, we track the values of g_k^+ and g_k^- estimated by gain loops in one sector of $N = 4096$ bits on the channel with $h(D) = 1 + D$ at high and low SNRs. The true values of the gain factors, $g^+ = \frac{1}{1+\epsilon}$ and $g^- = \frac{1}{1-\epsilon}$, are also plotted for comparison. The step-size β is set to 0.005, and the delay unit $m = 5$. In Fig. 2.7(a), ϵ has the fixed value 0.1, while in Fig. 2.7(a) ϵ slowly varies around the value 0.1. We see that gain factors are well tracked by the gain loops. The estimates at high SNR (10dB) show better convergence than those in a low SNR (7dB) environment.

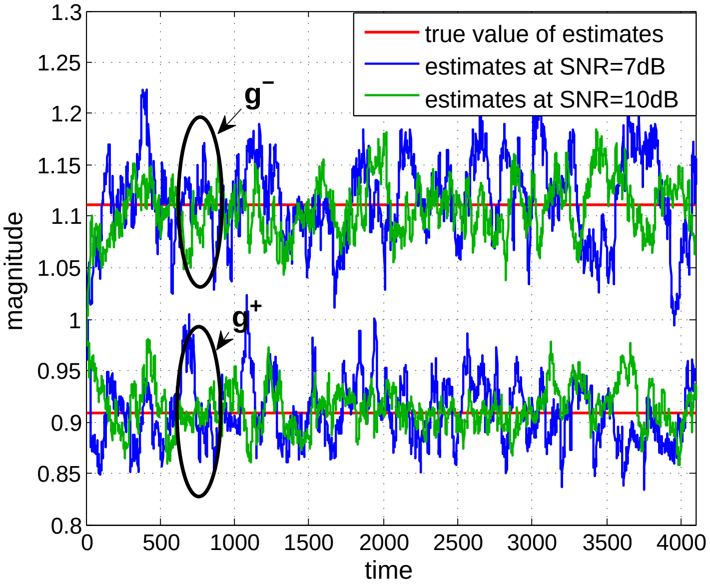
2.4 WSSJD on General ITI Channel

2.4.1 Decomposition of Interference Matrix

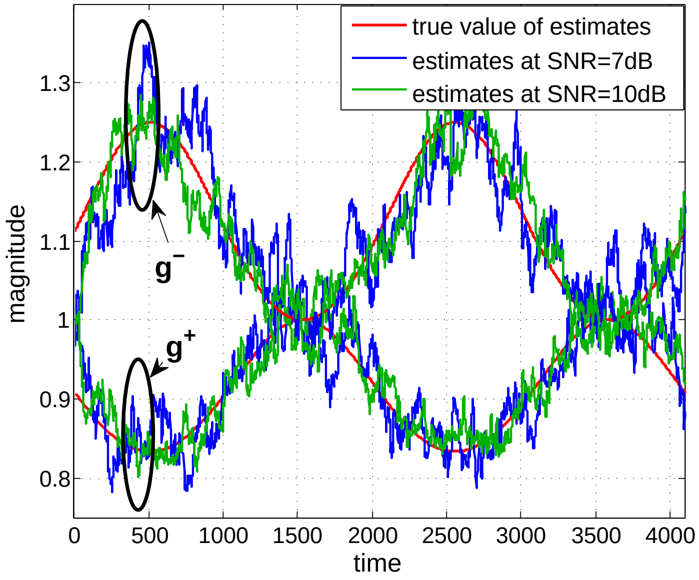
To generalize WSSJD to the nHnT model, consider the eigen-decomposition of $\mathbf{A}_n$ in (2.5),

$$\mathbf{A}_n = \mathbf{V}_n \mathbf{\Lambda}_n \mathbf{V}_n^{\top}, \tag{2.54}$$

where $\mathbf{V}_n$ is an $n \times n$ matrix whose columns are the eigenvectors of $\mathbf{A}_n$, and $\mathbf{\Lambda}_n$ is a diagonal matrix whose diagonal elements are the corresponding eigenvalues. The eigenvalues and eigenvectors of the symmetric tridiagonal Toeplitz matrix have a known closed form [45] [35]. If we



(a) $\epsilon = 0.1$



(b) $\epsilon(k) = 0.1 + 0.1\sin(4\pi k/N)$

Figure 2.7: Adaptive estimation of g^+ and g^- over one sector of $N = 4096$ bits for the channel $h(D) = 1 + D$ at different SNRs. In (a), $\epsilon = 0.1$. In (b), ϵ is a sinusoidal function of time.

define

$$\hat{\mathbf{T}}_n = \begin{bmatrix} 0 & 1 & & \mathcal{O} \\ 1 & 0 & \ddots & \\ & \ddots & \ddots & 1 \\ \mathcal{O} & & 1 & 0 \end{bmatrix}, \tag{2.55}$$

then

$$\mathbf{A}_n = \mathbf{I}_n + \epsilon\hat{\mathbf{T}}_n = \mathbf{V}_n(\mathbf{I}_n + \epsilon\hat{\mathbf{\Lambda}}_n)\mathbf{V}_n^\top, \tag{2.56}$$

where $\mathbf{I}_n$ is an $n \times n$ identity matrix, and $\hat{\mathbf{\Lambda}}_n$ is the diagonal matrix containing the eigenvalues of $\hat{\mathbf{T}}_n$. Therefore, the columns of $\mathbf{V}_n$ are also the eigenvectors of $\hat{\mathbf{T}}_n$, and $\mathbf{\Lambda}_n = \mathbf{I}_n + \epsilon\hat{\mathbf{\Lambda}}_n$. In fact, both $\hat{\mathbf{\Lambda}}_n$ and $\mathbf{V}_n$ have closed forms: the k^{th} eigenvalue of $\hat{\mathbf{T}}_n$ is

$$\hat{\lambda}_k = 2\cos\left(\frac{k\pi}{n+1}\right), \tag{2.57}$$

and the j^{th} element in the k^{th} eigenvector $\boldsymbol{v}_k$ is

$$v_{jk} = \sqrt{\frac{2}{n+1}}\sin\left(\frac{kj\pi}{n+1}\right). \tag{2.58}$$

Note that $\mathbf{V}_n$ is independent of ϵ.

Example 1. *For the case* $n = 2$,

$$\mathbf{\Lambda}_2 = \begin{bmatrix} 1+\epsilon & 0 \\ 0 & 1-\epsilon \end{bmatrix}, \qquad \mathbf{V}_2 = \begin{bmatrix} \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2} \end{bmatrix}.$$

Example 2. *For the case* $n = 3$,

$$\mathbf{\Lambda}_3 = \begin{bmatrix} 1+\sqrt{2}\epsilon & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1-\sqrt{2}\epsilon \end{bmatrix}, \quad \mathbf{V}_3 = \begin{bmatrix} \frac{1}{2} & \frac{\sqrt{2}}{2} & \frac{1}{2} \\ \frac{\sqrt{2}}{2} & 0 & -\frac{\sqrt{2}}{2} \\ \frac{1}{2} & -\frac{\sqrt{2}}{2} & \frac{1}{2} \end{bmatrix}.$$

2.4.2 Channel Decomposition and Generalized WSSJD

The $n\mathrm{H}n\mathrm{T}$ model is given by (2.1). Using the decomposition of $\mathbf{A}_n$ in (2.54), we can express the channel output as

$$\mathbf{R}(D) = \mathbf{V}_n \mathbf{\Lambda}_n \mathbf{V}_n^\top \mathbf{X}(D) h(D) + \mathbf{\Omega}(D). \tag{2.59}$$

Reorganizing (2.59) gives

$$\mathbf{\Lambda}_n^{-1} \mathbf{V}_n^\top \mathbf{R}(D) = \mathbf{V}_n^\top \mathbf{X}(D)\, h(D) + \mathbf{\Lambda}_n^{-1} \mathbf{V}_n^\top \mathbf{\Omega}(D). \tag{2.60}$$

Let $\bar{\mathbf{X}}(D) = \mathbf{V}_n^\top \mathbf{X}(D)$, $\bar{\mathbf{R}}(D) = \mathbf{\Lambda}_n^{-1} \mathbf{V}_n^\top \mathbf{R}(D)$ and $\bar{\mathbf{\Omega}}(D) = \mathbf{\Lambda}_n^{-1} \mathbf{V}_n^\top \mathbf{\Omega}(D)$ be the vectors of transformed input sequences, received sequences and noises, respectively. The new channel model becomes

$$\bar{\mathbf{R}}(D) = \bar{\mathbf{X}}(D) h(D) + \bar{\mathbf{\Omega}}(D), \tag{2.61}$$

which is composed of n parallel channels. The j^{th} channel is obtained by considering the j^{th} row of both sides of (2.61), which gives

$$\bar{r}^j(D) = \bar{x}^j(D) h(D) + \bar{\omega}^j(D), \tag{2.62}$$

Algorithm 1 WSSJD with gain loop on nHnT

```
function $\hat{\mathbf{X}}(D)$ =WSSJD($\mathbf{R}(D), \epsilon_0$)
Initialize:
$M(0) = 0$,
$M(p) = \infty$ for $p = 1, \cdots, 2^{n\nu} - 1$ {path metric}
$\Psi$ is a $2^{n\nu} \times N$ zero matrix {path history}
$\mathbf{G} = (\mathbf{I}_n + \epsilon_0 \hat{\mathbf{\Lambda}}_n)^{-1}${gain factors}
Begin: $k = 1$ to $N$
$\bar{\boldsymbol{r}}_k = \mathbf{G}\mathbf{V}_n^\top \boldsymbol{r}_k$ $p = 0$ to $2^{n\nu} - 1$
for each predecessor state $q_i$ of $p$
$M_i = M(q_i) + (\bar{\boldsymbol{r}}_k - \boldsymbol{y}^{(q_i,p)})^\top \mathbf{G}^{-2} (\bar{\boldsymbol{r}}_k - \boldsymbol{y}^{(q_i,p)})$
update $M(p) = \min_i M_i$
$\Psi(p, k) = q_i$ {extend survivor path} $k > m$
$p^* = \arg\min_p M(p)$ $j = 1$ to $\delta$
$p^* = \Psi(p^*, k - j + 1)$; {trace back path history}
$\boldsymbol{e} = \boldsymbol{y}^{(\Psi(p^*,k-\delta),p^*)} - \bar{\boldsymbol{r}}_{k-\delta}$
$\mathbf{G} = \mathbf{G} + \beta \mathrm{diag}(\boldsymbol{y}^{(\Psi(p^*,k-\delta),p^*)}) \mathrm{diag}(\boldsymbol{e})$
end
```

where

$$\bar{r}^j(D) = \frac{1}{1 + \epsilon \hat{\lambda}_j} \sum_{i=1}^{n} v_{ij}\, r^i(D), \tag{2.63}$$

$$\bar{x}^j(D) = \sum_{i=1}^{n} v_{ij}\, x^i(D), \tag{2.64}$$

$$\bar{w}^j(D) = \frac{1}{1 + \epsilon \hat{\lambda}_i} \sum_{i=1}^{n} v_{ij}\, \omega^i(D). \tag{2.65}$$

Several properties of the transformed channel model can be observed:

1. The noise components in $\bar{\mathbf{\Omega}}(D)$ are independent. Let $\boldsymbol{\omega}_k$ and $\bar{\boldsymbol{\omega}}_k$ be length-n vectors of the original and transformed noise samples at time k, i.e., the coefficients of D^k in the sequences $\boldsymbol{\omega}(D)$ and $\bar{\boldsymbol{\omega}}(D)$, respectively. Then

$$E[\bar{\boldsymbol{\omega}}_k \bar{\boldsymbol{\omega}}_k^\top] = E[\mathbf{\Lambda}_n^{-1} \mathbf{V}_n^\top \boldsymbol{\omega}_k \boldsymbol{\omega}_k^\top \mathbf{V}_n \mathbf{\Lambda}_n^{-1}] = (\sigma \mathbf{\Lambda}_n^{-1})^2, \tag{2.66}$$

which is a diagonal matrix. So the components of $\bar{\boldsymbol{\omega}}_k$ are uncorrelated and Gaussian, therefore independent. Furthermore, the noise power of the j^{th} channel is σ^2/λ_j^2.

2. After the transformation, the inputs of different component channels have different alphabets. For the j^{th} component channel, the alphabet Σ_j is

$$\Sigma_j = \left\{ \sum_{i=1}^{n} v_{ij} x_i | x_i \in \{+1, -1\} \right\}. \tag{2.67}$$

3. The j^{th} component channel corresponds to transmitting $\bar{x}^j(D)$ through the ISI channel $h(D)$ and adding electronic noise of power σ^2/λ_j^2. Since the inputs to different channels are correlated, a joint trellis is needed to search for the optimal decision. The new trellis states can be found by applying the one-to-one mapping to the conventional ML states. Therefore, the WSSJD trellis has $2^{n\nu}$ states.

4. Since V_n is determined once n is given, the WSSJD trellis is well-defined, and the branch labels are also independent of ϵ.

The optimal decision $\bar{\mathbf{X}}^*(D)$ satisfies

$$\begin{aligned} \bar{\mathbf{X}}^*(D) &= \arg \max_{\bar{\mathbf{X}}(D)} \log \Pr(\bar{\mathbf{R}}(D) | \bar{\mathbf{X}}(D)) \\ &= \arg \min_{\bar{\mathbf{X}}(D)} \sum_{j=1}^{n} \lambda_j^2 \, \|\bar{r}^j(D) - \bar{x}^j(D) h(D)\|^2. \end{aligned} \tag{2.68}$$

It is easy to see that WSSJD gives the optimal ML solution.

For a given error event $\bar{\boldsymbol{e}}(D) = [\bar{e}^1(D), ..., \bar{e}^n(D)]$, where $\bar{e}^j(D)$ is the error sequence on the j^{th} component channel, the distance parameter is given by

$$d^2(\bar{\boldsymbol{e}}(D)) = \sum_j \lambda_j^2 \, \|\bar{e}^j(D) h(D)\|^2. \tag{2.69}$$

2.4.3 Gain loops

As shown in (2.62)-(2.65), for each channel the ITI level ϵ appears in a gain factor normalizing $\sum_{i=1}^{n} v_{ij} \, r^i(D)$ such that its expectation is $\bar{x}(D)h(D)$. Gain loops can be used to adaptively estimate these gain factors.

Let g_k^j denote the gain factor estimated for the j^{th} channel at time k. Then $E[g_k^j] = \frac{1}{1+\epsilon\hat{\lambda}_j}$. Again, we use the LMS adaptive algorithm to update g_k^j, according to the equations

$$\hat{r}_k^j = g_{k-1}^j \sum_i v_{ij} r_k^i, \tag{2.70}$$

$$\hat{e}_{k-m}^j = \hat{y}_{k-m}^j - \hat{r}_{k-m}^j, \tag{2.71}$$

$$g_k^j = g_{k-1}^j + \beta \hat{y}_{k-m}^j \hat{e}_{k-m}^j, \tag{2.72}$$

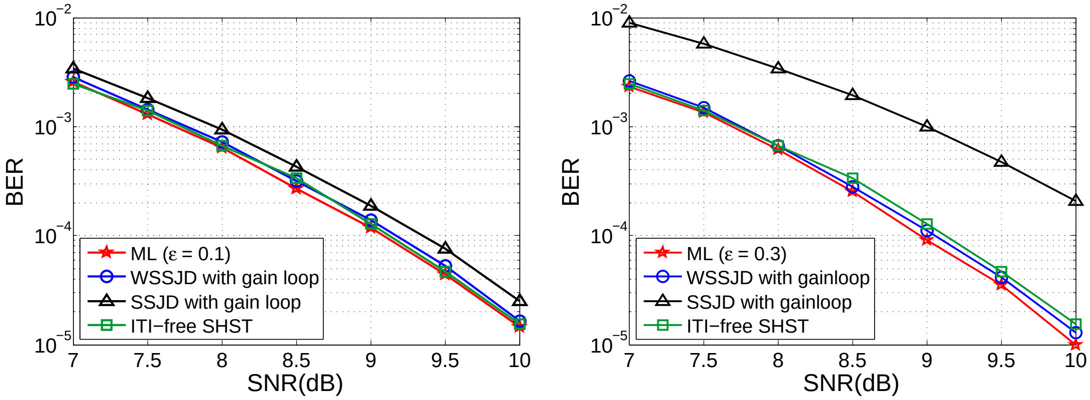
where $\hat{y}_{k-m}^j$ is the instantaneous hard decision on the noiseless output of the j^{th} ISI channel at time $k-m$. To find $\hat{y}_{k-m}^j$, we identify the trellis state which currently has the smallest path metric, and trace back the path history for m time slots to obtain the corresponding channel output. The gain factors g_k^j are also used in weighting the path metric, to ensure the ML performance of the detector.

Algorithm 1 summarizes the procedures to implement WSSJD with gain loops on the nHnT channel. The algorithm makes use of the following notation:

1. N is the length of one frame; ϵ_0 is the initial value of ϵ.
2. $\mathbf{G}$ is a diagonal matrix with g_k^j as the j^{th} diagonal element.
3. $\boldsymbol{r}_k$ is a column vector of the received signals from $nHnT$ channel at time k; $\bar{\boldsymbol{r}}_k$ is the vector of outputs from the transformed channel.
4. $M(p)$ is the accumulated path metric at state p.
5. $\boldsymbol{y}^{(q,p)}$ is a column vector of the trellis output label from state q to p.
6. diag($\boldsymbol{v}$) is a diagonal matrix with the elements of the vector $\boldsymbol{v}$ along the diagonal.

2.5 Simulation Results

In this section, we present BER performance simulation results for WSSJD and SSJD on the nHnT channel model for several values of n. We plot the BER as a function of channel SNR,



(a) $\epsilon = 0.1$

(b) $\epsilon = 0.3$

Figure 2.8: BER vs. SNR of different detectors with static ITI level (a) $\epsilon = 0.1$ and (b) $\epsilon = 0.3$.

where we define SNR as

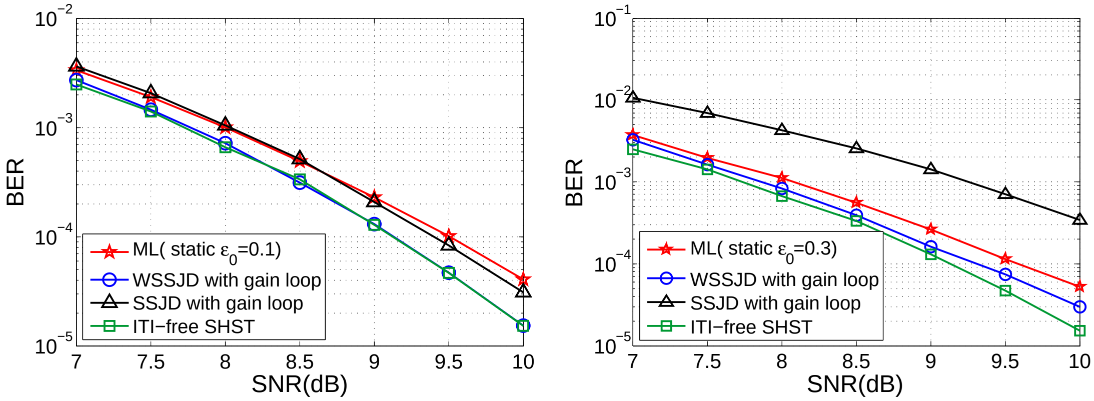
$$\text{SNR(dB)} = 10 \log \frac{\|h(D)\|^2}{2\sigma^2}. \tag{2.73}$$

Note that, in contrast to some studies such as [57], our definition of SNR does not take the energy $\epsilon^2 \|h(D)\|^2$ in the cross-track signal into account. Since we use separate figures for the results corresponding to different static ITI values, this definition should not lead to any confusion. It is also better suited for presenting the results obtained when the ITI is characterized by a small variation around a fixed nominal ITI level.

2.5.1 2H2T System

We simulate WSSJD and SSJD with gain loops on the 2H2T system with channel polynomial $h(D) = 1 + D$. We set $\beta = 0.008$ and $m = 5$. The initial values of gain factors g_0^+ and g_0^- are obtained by passing training samples through the system.

We first test the proposed detectors and gain loop structure when ϵ is fixed. In Fig. 2.8, we compare the BER performances of WSSJD, SSJD, and the conventional 2H2T ML detector, for $\epsilon = 0.1$ and $\epsilon = 0.3$. The frame size is $N = 4096$ bits. In our setup, the ML detector knows the value of ϵ, while WSSJD and SSJD adaptively estimate gain factors using the gain



(a) (b)

Figure 2.9: BER vs. SNR of different detectors with ϵ slowly varying about the mean value (a) $\epsilon_0 = 0.1$ and (b) $\epsilon_0 = 0.3$.

loop structure shown in Fig. 2.6. Therefore, the static ML detector provides the optimal BER performance. It is observed that the BER curve of WSSJD with a gain loop almost coincides with that of the static ML detector. This indicates that the LMS adaptive algorithm provides sufficiently accurate estimates of gain factors. The recursive least squares (RLS) algorithm can speed up the convergence of the gain loop at the expense of higher complexity, but the BER performance improvement would be negligible. As expected from the minimum distance plots in Fig. 2.1, the performance loss suffered by SSJD relative to WSSJD is more severe at $\epsilon = 0.3$ than at $\epsilon = 0.1$. Although not shown here, simulation results for the frame error rate vs. SNR correlate well with the BER results.

Next, we test the performance of the detectors by assuming a dynamic ITI model in which ϵ changes slowly with respect to the location k. Specifically, we set

$$\epsilon(k) = \epsilon_0 + 0.1 \sin(4\pi k/N), \tag{2.74}$$

where $N = 4096$ and ϵ_0 is the mean value. In this case, we compare the adaptive WSSJD algorithm, in which gain loops are used to track the value of ϵ, with the ML detector which uses the static value ϵ_0. The simulation results, shown in Figs. 2.9, show that the adaptive WSSJD outperforms the static ML detectiion by about 0.3-0.5dB at high SNR.

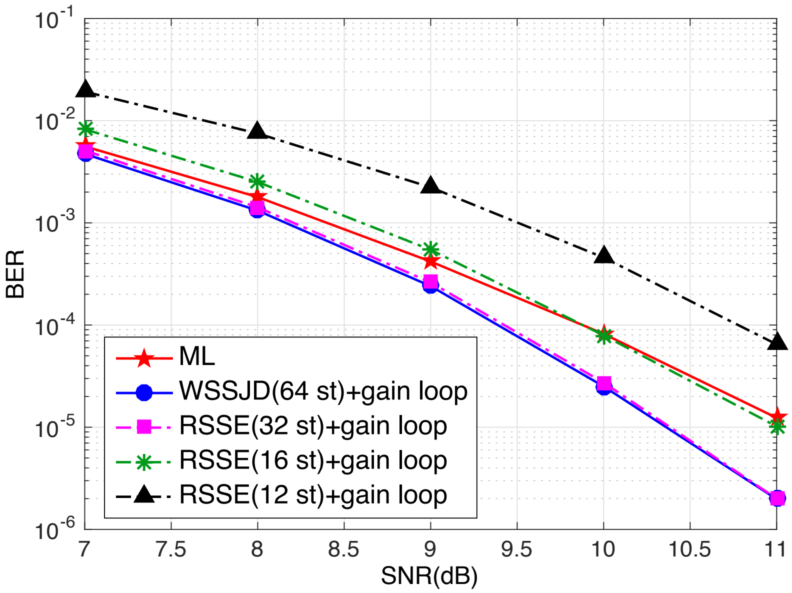


Figure 2.10: Performance of reduced complexity implementations of WSSJD with gain loop on 2H2T EPR4 channel. ϵ is sinusoidally varying with mean value $\epsilon_0 = 0.1$

In both cases, the performance of the ITI-free SHST detector is plotted for comparison. It is interpreted as the performance of the ideal ITI cancellation scheme, where the ITI is completely removed from the readback signals of each track.

The trellis complexity of the MHMT ML detector can be prohibitively complex for practical purposes. We address this problem in [15] and [16] by using a reduced-state trellis with the RSSE detection algorithm. Moreover, the gain loop structure for ITI estimation in WSSJD can be directly applied in the RSSE setting. Fig. 2.10 shows BER simulation results for WSSJD with several RSSE implementations of various trellis-complexities on the extended class-IV partial response (EPR4) channel, with channel polynomial $h(D) = 1 + D - D^2 - D^3$. We see that, as expected, WSSJD with a 64-state trellis and adaptive ITI estimation outperforms the static ML detector with the same number of states. Results are also shown for RSSE using 12, 16, and 32 trellis states, along with gain estimation loops. We can see that RSSE can achieve the same performance as WSSJD using only one-half the number of trellis states (32 vs. 64).

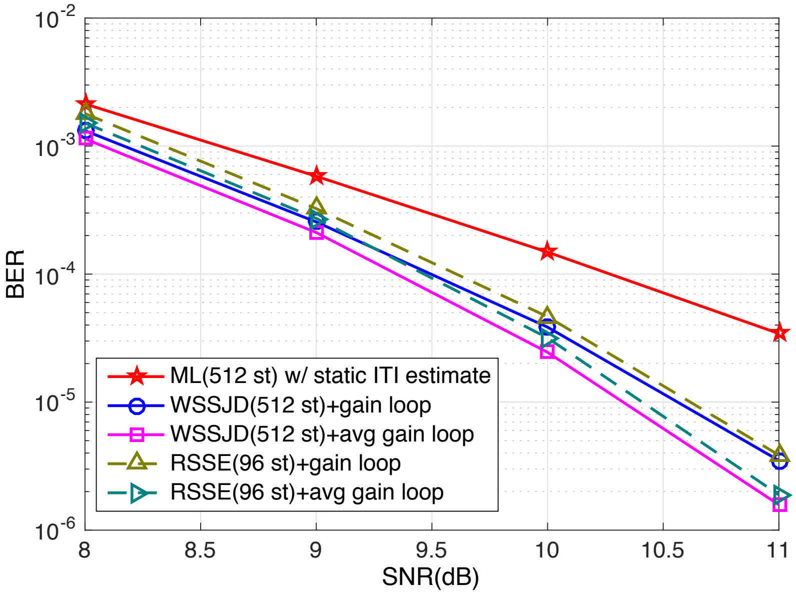


Figure 2.11: BER performance of WSSJD on 3H3T EPR4 channel ($h(D) = 1 + D - D^2 - D^3$) under sinusoidally varying ITI with mean value $\epsilon_0 = 0.1$

2.5.2 3H3T System

In Fig. 2.11, we plot the BER performance of several detectors on the 3H3T channel. We assume that each component channel is equalized to the EPR4 target. The WSSJD detector therefore requires 512 trellis states. As the matrix decomposition described in Example 2 indicates, the ITI estimation for the transformed 3H3T system requires only 2 gain loops, one for each of the first and the third component channels. We observe in the figure that WSSJD with ITI estimation outperforms the static ML algorithm by about 1dB when the BER is in the range $[10^{-5}, 10^{-4}]$. The performance is further improved by averaging the two gain factors to get a better estimate of ϵ. Finally, we see that an RSSE implementation of WSSJD with only 96 states achieves performance comparable to that of the full 512-state WSSJD implementation.

2.6 Conclusion

We study MHMT detection as a potential candidate for next-generation magnetic recording. We assume a simplified symmetric nHnT model in which only adjacent tracks interfere. The

ITI is assumed to be linear and additive, and its amplitude is controlled by a scalar. In this chapter, we consider how to estimate the ITI and how to efficiently use the new estimates to improve the detector performance. By means of a channel transformation, we decompose the original nHnT system into n separate subchannels, each of which has a channel gain factor that depends on the ITI level. Based on this transformed system, we propose a novel detection method which we call weighted sum-subtract joint detection (WSSJD). The new method achieves ML performance. However, in contrast to the conventional MHMT ML detector trellis, the branch labels of the WSSJD trellis do not depend on the ITI level. Instead, the value of ITI level is incorporated into a weighting factor used to compute the path metrics. A simple gain loop structure is described, permitting efficient adaptive estimation of the gain factors and their straightforward incorporation into the WSSJD algorithm. Simulation results show that when ITI is time-varying, WSSJD with adaptive ITI estimation outperforms the ML detection using static ITI estimates. Minimum distance analysis is presented, providing a theoretical basis for the comparison of different MHMT detectors. The WSSJD technique is also amenable to a reduced complexity implementation based upon RSSE, which is the focus of Part II of this work.

2.7 Appendix

In this section we give the derivation of (2.44) and (2.45).

Single track error events

Assume $e^b(D) = 0$. The distance components reduce to

$$\begin{aligned}
d_{\text{ideal}} &= \sqrt{(1+\epsilon_0^2)\|e^a(D)h(D)\|^2}, \\
d_{\text{mism}} &= 2\Delta\epsilon \frac{\left\langle e^a(D)h(D), x^b(D)h(D)\right\rangle + \epsilon_0 \left\langle e^a(D)h(D), x^a(D)h(D)\right\rangle}{\sqrt{(1+\epsilon_0^2)\|e^a(D)h(D)\|^2}}.
\end{aligned}$$

We bound d_{mism} as follows:

$$\begin{aligned}
&\left\langle e^a(D)h(D), x^b(D)h(D)\right\rangle \\
&\quad = \sum\nolimits_n \left(\sum\nolimits_m x^b_{n-m}h_m\right)\left(\sum\nolimits_m e^a_{n-m}h_m\right) \\
&\quad \leq \left|\sum\nolimits_n \left(\sum\nolimits_m x^b_{n-m}h_m\right)\left(\sum\nolimits_m e^a_{n-m}h_m\right)\right| \\
&\quad \leq \sum\nolimits_n \left|\sum\nolimits_m x^b_{n-m}h_m\right| \cdot \left|\sum\nolimits_m e^a_{n-m}h_m\right| \\
&\quad \leq M_h \sum\nolimits_n \left|\sum\nolimits_m e^a_{n-m}h_m\right| \\
&\quad = 2M_h \sum\nolimits_n \left|\sum\nolimits_m \frac{e^a_{n-m}}{2}h_m\right| \\
&\quad \leq 2M_h \sum\nolimits_n \left(\sum\nolimits_m \frac{e^a_{n-m}}{2}h_m\right)^2 \\
&\quad = \frac{M_h}{2}\|e^a(D)h(D)\|^2, \qquad (2.75)
\end{aligned}$$

where $M_h = \sum_m |h_m| = 2$ for the channel $h(D) = 1 + D$. Using a similar derivation, we can show

$$\left\langle e^a(D)h(D), x^b(D)h(D)\right\rangle \geq -\frac{M_h}{2}\|e^a(D)h(D)\|^2. \qquad (2.76)$$

To find the bounds for $\langle e^a(D)h(D), x^a(D)h(D)\rangle$, note that

$$\begin{aligned}
&\langle e^a(D)h(D), x^a(D)h(D)\rangle \\
&\quad = \sum\nolimits_k (e^a_{k-1} + e^a_k)(x^a_{k-1} + x^a_k) \\
&\quad = \sum_{k=k_1}^{k_2+1} (e^a_{k-1}x^a_{k-1} + e^a_{k-1}x^a_k + e^a_k x^a_{k-1} + e^a_k x^a_k) \qquad (2.77) \\
&\quad \geq \sum_{k=k_1}^{k_2+1} \left(|e^a_{k-1}| - |e^a_{k-1}| - |e^a_k| + |e^a_k|\right) \qquad (2.78) \\
&\quad = 0.
\end{aligned}$$

The inequality in (2.78) follows from the fact that x^a_k always has the same sign as e^a_k, so $e^a_k x^a_k =$

$|e_k^a|$. Choosing x_k^a to have the opposite sign to e_{k-1}^a leads to the lower bound $e_{k-1}^a x_k^a \geq -|e_{k-1}^a|$.

The upper bound derived in (2.75) is also applicable to $\langle e^a(D)h(D), x^a(D)h(D)\rangle$. Therefore,

$$0 \leq \langle e^a(D)h(D), x^a(D)h(D)\rangle \leq \|e^a(D)h(D)\|^2. \tag{2.79}$$

Combining (2.75) and (2.79), and using $\|e^a(D)h(D)\|^2 \geq 8$ for the channel $h(D) = 1 + D$, we can bound the single-track minimum distance in the two cases of $\Delta\epsilon > 0$ and $\Delta\epsilon < 0$ as follows:

$$\begin{aligned} d_{\text{s}} &= d_{\text{ideal}} + d_{\text{mism}} \\ &\geq \left(\sqrt{1+\epsilon_0^2} - \frac{2\Delta\epsilon}{\sqrt{1+\epsilon_0^2}}\right)\|e^a(D)h(D)\| \\ &\geq \frac{2\sqrt{2}(1+\epsilon_0^2 - 2\Delta\epsilon)}{\sqrt{1+\epsilon_0^2}}, \quad \text{if } \Delta\epsilon > 0, \end{aligned}$$

and

$$\begin{aligned} d_{\text{s}} &= d_{\text{ideal}} + d_{\text{mism}} \\ &\geq \left(\sqrt{1+\epsilon_0^2} + \frac{2\Delta\epsilon(1+\epsilon_0)}{\sqrt{1+\epsilon_0^2}}\right)\|e^a(D)h(D)\| \\ &\geq \frac{2\sqrt{2}(1+\epsilon_0^2 + 2(1+\epsilon_0)\Delta\epsilon)}{\sqrt{1+\epsilon_0^2}}, \quad \text{if } \Delta\epsilon < 0. \end{aligned}$$

Table 2.2(a) gives examples of error events that achieve these bounds.

Double track error events

In this case, both $e^a(D)$ and $e^b(D)$ are non-zero at some locations. To find an achievable bound on $d_{\text{ideal}} + d_{\text{mism}}$, we assume $\Delta\epsilon \ll 1$. Therefore the distance increment/decrement caused by the mismatch will not be as significant as the distance in the ideal case. The minimum value of

d_{ideal} given by (2.41) is $4(1-\epsilon_0)$, achieved by the error sequences of the form

$$\begin{aligned}\boldsymbol{e}^a &= [0,\cdots,0,e^a_{k_1},\cdots,e^a_{k_2},\cdots,0],\\ \boldsymbol{e}^b &= [0,\cdots,0,e^b_{k_1},\cdots,e^b_{k_2},\cdots,0],\end{aligned}$$

with $e^a_{k+1} = -e^a_k$ for $k_1 \le k \le k_2 - 1$, and $e^b_k = -e^a_k$ for $k_1 \le k \le k_2$. The assumption on $\Delta\epsilon$ suggests that we focus on these error events. We use d^*_{mism} to denote the minimum distance parameter attained by this subset of double track error events.

We can express $\langle \mathcal{A}(D), x^b(D)h(D)\rangle$ by

$$\begin{aligned}&\langle \mathcal{A}(D), x^b(D)h(D)\rangle\\ &= \sum_{k=k_1}^{k_2+1}[e^a_k + e^a_{k-1} + \epsilon_0(e^b_k + e^b_{k-1})](x^b_k + x^b_{k-1})\\ &= (e^a_{k_1} + \epsilon_0 e^b_{k_1})(x^b_{k_1} + x^b_{k_1-1}) + (e^a_{k_2} + \epsilon_0 e^b_{k_2})(x^b_{k_2+1} + x^b_{k_2})\\ &= -|e^a_{k_1}| + \epsilon_0|e^b_{k_1}| + (e^a_{k_1} + \epsilon_0 e^b_{k_1})x^b_{k_1-1}\\ &\quad - |e^a_{k_2}| + \epsilon_0|e^b_{k_2}| + (e^a_{k_2} + \epsilon_0 e^b_{k_2})x^b_{k_2+1}.\end{aligned} \tag{2.80}$$

Upper and lower bounds for (2.80) can be found by carefully choosing values for $x^b_{k_1-1}$ and $x^b_{k_2+1}$. If $x^b_{k_1-1}$ and $x^b_{k_2+1}$ have the same sign as $e^a_{k_1}$ and $e^a_{k_2}$, respectively, (2.80) achieves the maximum value 0. If $x^b_{k_1-1}$ and $x^b_{k_2+1}$ have the same sign as $e^b_{k_1}$ and $e^b_{k_2}$, respectively, (2.80) achieves the minimum value $8(\epsilon_0 - 1)$. Similarly, we have

$$8(\epsilon_0 - 1) \le \langle \mathcal{B}(D), x^a(D)h(D)\rangle \le 0. \tag{2.81}$$

We conclude that in the case of $\Delta\epsilon > 0$ and $\Delta\epsilon < 0$,

$$\begin{aligned}d^*_{\text{d}} \overset{\text{def}}{=} d_{\text{ideal}} + d^*_{\text{mism}} &\ge 4(1-\epsilon_0) + \frac{2\Delta\epsilon}{4(1-\epsilon_0)}\cdot 16(\epsilon_0 - 1)\\ &= 4(1-\epsilon_0-2\Delta\epsilon), \quad \text{if } \Delta\epsilon > 0,\end{aligned}$$

and

$$d_{\mathrm{d}}^{*} \overset{\mathrm{def}}{=} d_{\mathrm{ideal}} + d_{\mathrm{mism}}^{*} \geq 4(1-\epsilon_0), \quad \text{if } \Delta\epsilon < 0.$$

Notice that these bounds are derived for a subset of double track error events which achieve $\min d_{\mathrm{ideal}}$, examples of which are given in Table 2.2(b).

We compared the values of $d_{\mathrm{min}}^2 = \min\{d_{\mathrm{s}}^2, d_{\mathrm{d}}^2\}$ obtained by exhaustive computer search with $\min\{(d_{\mathrm{s}}^*)^2, (d_{\mathrm{d}}^*)^2\}$, and they agreed at all points plotted in Fig. 2.5. This supports the assumption that the simplification in our analysis of double track error events does not affect the d_{min}^2 computation.

Acknowledgement

This chapter contains material from the papers: Bing Fan, Hemant K. Thapar and Paul H. Siegel, “Multihead multitrack detection in shingled magnetic recording with ITI estimation,” published in *IEEE Int. Conf. on Commun. (ICC)*, London, UK, Jun. 2015, pp. 425-430, and Bing Fan, Hemant K. Thapar and Paul H. Siegel, “Multihead multitrack detection for next generation magnetic recording, part I: weighted sum subtract joint detection with ITI estimation,” published in *IEEE Trans. Commun.*, vol. 65, no. 4, pp. 1635-1648, April 2017. The dissertation author was the primary investigator and author of these papers, and co-authors have approved the use of the material for this dissertation.

Chapter 3

Reduced-State Sequence Estimation for Multihead Multitrack Channel

To achieve large storage capacity on magnetic hard disk drives, very high track density is required, causing severe intertrack interference (ITI). Multihead multitrack (MHMT) detection has been proposed to better combat the effects of ITI. Such detection, however, has prohibitive implementation complexity. Reduced-state sequence estimation (RSSE) is a promising technique for significantly reducing the complexity, while retaining good performance. In this chapter, several different MHMT models are considered, including symmetric and asymmetric 2H2T systems, and a symmetric 3H3T system. By carefully evaluating the effective distance between two input symbols, we propose optimized set partition trees for each channel model. Different trellis configurations for RSSE are constructed based on the desired performance/complexity tradeoff. Simulation results show that the reduced MHMT detector can achieve near maximum-likelihood (ML) performance with a small fraction of the original number of trellis states. We also use error event analysis to explain the behavior of RSSE. The proposed algorithm could be potentially applied to next generation magnetic recording systems, especially when the ML detector is infeasible due to the high computational complexity.

3.1 Introduction

Intertrack interference (ITI), caused by aggressively shrinking the track pitch, is one of the more severe impairments in next generation hard disk drives (HDDs) [56] [23]. The use of an array reader to simultaneously read and process multiple tracks has recently drawn intensive interest because of its capability to handle ITI as well as electronic noise [32] [58]. The associated maximum likelihood (ML) detector complexity is, however, drastically increased.

Let $\boldsymbol{x}_i = [x_i^1, x_i^2, ..., x_i^n]^\top$, $x_i^j \in \{-1, +1\}$, be a column vector of the input symbols written on n adjacent tracks at time i. Let $\boldsymbol{x}(D) = [x^1(D), x^2(D), ..., x^n(D)]$ denote the collection of sequences recorded on n tracks, where D is the delay unit. Assume all the tracks are equalized to the same channel polynomial $h(D) = h_0 + h_1 D + \cdots + h_\nu D^\nu$. An n-head, n-track (nHnT) system is generally modeled as

$$\boldsymbol{r}_i = A_n \boldsymbol{y}_i + \boldsymbol{\omega}_i, \tag{3.1}$$

where $\boldsymbol{r}_i = [r_i^1, r_i^2, ..., r_i^n]^\top$ is the vector of received signals from n heads, $\boldsymbol{y}_i = [y_i^1, y_i^2, ..., y_i^n]^\top$ is a vector of noiseless channel outputs, $y_i^j = \sum_{k=0}^{\nu} h_k x_{i-k}^j$, and $\boldsymbol{\omega}_i = [\omega_i^1, \omega_i^2, ..., \omega_i^n]^\top$, $\omega_i \sim \mathcal{N}(0, \sigma^2)$ is a vector of independent Gaussian electronic noise samples. We assume the noise samples are uncorrelated across tracks, i.e. $E[\boldsymbol{\omega}_i \boldsymbol{\omega}_i^\top] = \sigma^2 I_n$, where I_n is the $n \times n$ identity matrix. The ITI effect is characterized by an $n \times n$ matrix A_n. We will be primarily interested in the situation when

$$A_n = \begin{bmatrix} 1 & \epsilon & & & \\ \epsilon & 1 & \ddots & \mathbf{0} & \\ & \ddots & \ddots & \ddots & \\ & \mathbf{0} & \ddots & 1 & \epsilon \\ & & & \epsilon & 1 \end{bmatrix}, \tag{3.2}$$

where $\epsilon \in [0, 0.5]$ represents the ITI level. Such a system is symmetric, and the ITI only comes from the immediately adjacent tracks.

The ML detector decodes n tracks by simultaneously processing readback signals from n heads [46] [6]. The resulting joint trellis is composed of $2^{n\nu}$ states, each associated with 2^n incoming and outgoing edges. Constructing the trellis requires knowledge of ϵ, which is generally time varying, and unknown to the receiver. This problem is resolved in Part I. More specifically, in Part I we propose a novel ML-equivalent detection method - weighted sum subtract joint detector (WSSJD) - along with a gain loop structure that can estimate the ITI as well as adapt itself to the new estimates.

In Part II, we explore ways to reduce the complexity of MHMT detection, which is another challenging problem that needs to be solved to make MHMT practical. Compared to the traditional single-head single-track (SHST) detector with complexity $O(2^\nu)$, the nHnT ML detector has complexity $O(2^{n\nu})$. For $\nu > 3$, which is typical in practical recording channels, direct implementation of the nHnT ML detector could become infeasible even for small n. On the other hand, reduced-state sequence estimation (RSSE) [14], which was first proposed for transmitting quadrature amplitude modulation (QAM) symbols through a partial response channel with long memory, is a good candidate to mitigate the complexity issue of MHMT detector. The RSSE trellis, originally constructed based on the Ungerboeck set partition tree, has fewer states. We note that the M-Viterbi algorithm [5] offers similar performance and complexity to RSSE, but it uses a larger trellis which complicates its analysis and hardware implementation.

Efforts have been made to develop similar algorithms for use in MHMT detection. In particular, the authors of [29] presented a way to apply RSSE to MHMT system, but their construction generally suffers from high performance loss.

We propose a different approach, which is based on our work in Part I. We find that the channel transformation in WSSJD decomposes the system in (3.1) into n parallel sub-channels, which naturally leads to a set partition rule on the MHMT input constellation. For the simplest 2H2T system, the channel after transformation becomes a QAM-type model, and the reduced-state trellis can be constructed by redefining the distance measure on the transformed input constellation. The resulting four-level set partition tree provides better flexibility in performance/complexity tradeoffs. Our simulation results show that, with fewer than half the number of the full ML trellis states, RSSE can achieve near-ML performance on many channels. The concept of using RSSE in 2H2T case was partially presented in [15] [16], and some of that discussion is briefly restated in this paper for the sake of completeness and better understanding. Further details about implementation issues and more thorough performance evaluations are also provided here.

We further show that the evaluation of RSSE performance is tractable through error event analysis. In contrast to ML detection, some error events in RSSE merge early due to the reduced-state trellis structure. We introduce an early-merging condition to identify these error events, and a modified error state diagram is used to search for the dominant early-merged error events. The search results for several reduced-state trellis configurations at different ITI levels are presented. When the minimum distance parameter of the early-merged error events is larger than that of the ML detector, the performance loss of the RSSE trellis is almost negligible. An asymmetric 2H2T system is also considered because of its potential practical interest. The error event analysis shows that the proposed set partitioning rule is also applicable in the asymmetric case.

Finally, we investigate a more complex 3H3T model. The effective distance between symbols shows different monotonicity behavior as the ITI level changes. Therefore, we propose two types of set partition trees, one suitable for the low ITI environment, and the other better suited for high ITI levels. Simulation results are provided for both cases, and they show that RSSE can significantly reduce the computational complexity of 3H3T detection while retaining acceptable performance.

3.2 Background

3.2.1 Review of RSSE

The traditional RSSE is designed for transmitting QAM symbols through an ISI channel with channel memory ν. Recall that in the ML detector, the trellis state is represented as a length

ν vector,

$$\boldsymbol{p}_n = [\,\boldsymbol{x}_{n-1}, \boldsymbol{x}_{n-2}, \ldots, \boldsymbol{x}_{n-\nu}\,], \tag{3.3}$$

where each symbol $\boldsymbol{x}_{n-k}$ is complex-valued, and selected from a two-dimensional signal set $\mathcal{C}$ whose size is M. In RSSE, to reduce the number of trellis states, several ML states are grouped into a *s*ubset state. To do this, for the kth element $\boldsymbol{x}_{n-k}$ in $\boldsymbol{p}_n$, a set partition $\Omega(k)$ of $\mathcal{C}$ is defined, and $\boldsymbol{x}_{n-k}$ is represented by its subset index $a_{n-k}(k)$ in $\Omega(k)$. Notice that $\Omega(k)$ can be different for $k = 1, \cdots, \nu$. Let $J_k = |\Omega(k)|$ be the number of subsets in partition $\Omega(k)$, $1 \le J_k \le M$. Then the subset index $a_{n-k}(k)$ can take its value from $0, 1, \cdots, J_k - 1$. The corresponding subset state of $\boldsymbol{p}_n$ is denoted by

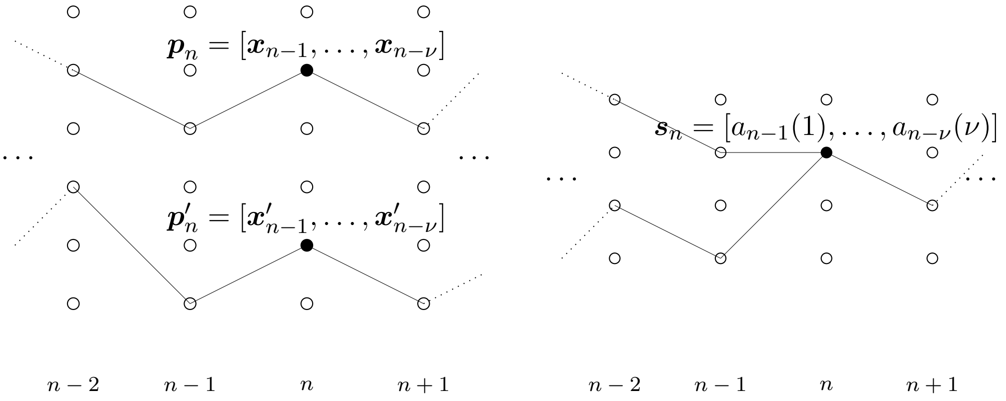
$$\boldsymbol{s}_n = [\,a_{n-1}(1), a_{n-2}(2), \ldots, a_{n-\nu}(\nu)\,]. \tag{3.4}$$

The trellis constructed from all possible $\boldsymbol{s}_n$ is called the *s*ubset trellis. To obtain a well-defined trellis structure, the partition $\Omega(k)$ is restricted to be a further partition of the subsets in $\Omega(k+1)$, for $1 \le k \le \nu - 1$. This condition guarantees that for a given state $\boldsymbol{s}_n$ and current input $\boldsymbol{x}_n$, the next subset state is uniquely determined and represented as

$$\boldsymbol{s}_{n+1} = [\,a_n(1), a_{n-1}(2), \ldots, a_{n-\nu+1}(\nu)\,], \tag{3.5}$$

where $a_n(1)$ is the subset index of $\boldsymbol{x}_n$ in $\Omega(1)$, $a_{n-1}(2)$ is the index of $\boldsymbol{x}_{n-1}$ in $\Omega(2)$, and so on. The number of states in the subset trellis is $\prod_{k=1}^{\nu} J_k$. The complexity of a RSSE trellis can be controlled by specifying ν parameters, J_k for $1 \le k \le \nu$. We define the *c*onfiguration of a subset trellis to be a vector $\boldsymbol{J} = [J_1, J_2, \ldots, J_\nu]$. A valid configuration satisfies $J_1 \ge J_2 \ge \cdots \ge J_\nu$.

To apply the Viterbi algorithm (VA) on a subset trellis, a decision feedback scheme is introduced to calculate the branch metric, since the subset state $\boldsymbol{s}_n$ does not uniquely specify the most recent ν symbols. During the detection process, a modified path history is used to store the surviving symbol $\hat{\boldsymbol{x}}_{n-1}$ that leads to state $\boldsymbol{s}_n$. The actual surviving ML state $\hat{\boldsymbol{p}}_n$ is obtained by



(a) ML trellis

(b) Subset trellis

Figure 3.1: Comparison between the decoding paths on (a) ML trellis and (b) subset trellis.

tracing back ν steps in the path history. Error propagation may occur in this process, but its effect is negligible [14] [43].

The underlying idea of RSSE is to drop less likely paths early in the detection process. Since each subset state contains multiple ML states, certain paths will merge earlier in the subset trellis than in the ML trellis, as shown in Figs. 3.1. If at time n two paths ending at ML states $\boldsymbol{p}_n$ and $\boldsymbol{p}'_n$ satisfy $\boldsymbol{x}_{n-k} \in a_{n-k}(k)$ and $\boldsymbol{x}'_{n-k} \in a_{n-k}(k)$ for all $k = 1, \ldots, \nu$, then they will merge early at subset state $\boldsymbol{s}_n$ in the subset trellis. If $J_k = M$ for $1 \leq k \leq \nu$, RSSE becomes MLSE. Otherwise it is suboptimal. To minimize the performance loss, proper set partitions $\Omega(k)$ should be selected carefully to guarantee that enough distance differences have been accumulated to reliably distinguish between merging paths. For the M-QAM system, it is suggested that good performance can generally be obtained by maximizing the minimum intrasubset Euclidean distance for each partition $\Omega(k)$, $k = 1, \cdots, \nu$ [14]. The Ungerboeck set partition tree [48] is shown to have this property and is adopted to make the selection of $\Omega(k)$. For more details about the subset trellis construction for the M-QAM system, the reader is referred to [14].

The use of the Ungerboeck set partition tree is key to obtaining good performance of the RSSE algorithm on the QAM system. However, such a set partition tree cannot be directly applied to the 2H2T system because of the ITI. In the next subsection we will show that a simple transformation can decompose the original 2H2T system into two independent channels, resulting in a QAM-like structure. Then, instead of using the Euclidean distance, we define a new distance measure between the input symbols, based on which we construct a more suitable set partition tree for the 2H2T system.

3.2.2 Weighted Sum Subtract Joint Detector (WSSJD)

WSSJD is proposed to resolve the problem of ITI estimation. It works on the transformed model of (3.1), given by

$$\Lambda_n^{-1} V_n^\top \boldsymbol{r}_i = V_n^\top \boldsymbol{y}_i + \Lambda_n^{-1} V_n^\top \boldsymbol{\omega}_i, \tag{3.6}$$

where V_n and Λ_n are the induced matrices from the eigen-decomposition of A_n, $A_n = V_n \Lambda_n V_n^\top$. Since A_n is a symmetric tridiagonal Toeplitz matrix, it has the property that, V_n is a constant matrix, and Λ_n is a diagonal matrix whose diagonal elements, λ_j, $j = 1, ..., n$, are functions of ϵ. For example, the 2H2T system has $A_2 = V_2 \Lambda_2 V_2^\top$, where

$$\Lambda_2 = \begin{bmatrix} 1+\epsilon & 0 \\ 0 & 1-\epsilon \end{bmatrix}, \quad V_2 = \begin{bmatrix} \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2} \end{bmatrix}. \tag{3.7}$$

Let $\boldsymbol{z}_i = V_n^\top \boldsymbol{x}_i$, $\bar{\boldsymbol{r}}_i = \Lambda_n^{-1} V_n^\top \boldsymbol{r}_i$ and $\bar{\boldsymbol{\omega}}_i = \Lambda_n^{-1} V_n^\top \boldsymbol{\omega}_i$ be the input, received sample and noise of the transformed system given by (3.6). Then the system consists of n parallel sub-channels, each of which has the input-output relationship

$$\bar{r}_i^j = \sum_{k=0}^{\nu} h_k z_{i-k}^j + \bar{\omega}_i^j, \quad j = 1, ..., n. \tag{3.8}$$

In the joint trellis constructed for (3.6), a state $(\boldsymbol{z}_{i-\nu}, ..., \boldsymbol{z}_{i-1})$ associated with the input $\boldsymbol{z}_i$ will have output $\bar{y}_i^j = \sum_{k=0}^{\nu} h_k z_{i-k}^j$. Notice that the trellis labels are independent of ϵ. Moreover, the noise samples, $\bar{\omega}_i^j$, are independent and have different powers, $E[\bar{\omega}_i \bar{\omega}_i^\top] = \sigma^2 (\Lambda_n^{-1})^2$.

The use of WSSJD is summarized as follows:

1. Calculate $\bar{\boldsymbol{r}}_i$ by $\bar{\boldsymbol{r}}_i = \Lambda_n^{-1} V_n^\top \boldsymbol{r}_i$.

2. To retain the ML property, the branch metrics are weighted when applying the Viterbi algorithm (VA),

$$m(\boldsymbol{s}_{i-1}, \boldsymbol{s}_i) = \sum_{j=1}^{n} \lambda_j^2 (\bar{r}_i^j - \bar{y}_i^j)^2. \tag{3.9}$$

WSSJD can work with a gain loop structure which adaptively estimates ϵ. The new estimate is then fed back to WSSJD to update the weights in calculating the branch metric. The discussion of ITI estimation is beyond the scope of this paper; therefore, we assume that ϵ is known. For more details on WSSJD, the reader is referred to Part I.

Throughout this paper, the complexity reduction techniques are developed based on WSSJD. This is motivated by the fact that WSSJD has ML-equivalent performance. Moreover, the coordinate transformations in WSSJD lead to a better measure of the distance between input symbols, which plays an important role in designing the set partition tree. Further, as we will see, the structure of parallel channels can provide additional complexity reduction in selecting survivor paths. We emphasize, however, that the RSSE techniques described here are also applicable to the standard MHMT ML detector trellis. Henceforth, with a slight abuse of terminology, when we say the "ML trellis", we refer to the WSSJD trellis.

3.3 Set Partition Tree for 2H2T System

We first consider the symmetric 2H2T system

$$\begin{bmatrix} r_i^1 \\ r_i^2 \end{bmatrix} = \begin{bmatrix} 1 & \epsilon \\ \epsilon & 1 \end{bmatrix} \begin{bmatrix} y_i^1 \\ y_i^2 \end{bmatrix} + \begin{bmatrix} \omega_i^1 \\ \omega_i^2 \end{bmatrix}. \tag{3.10}$$

which is also studied in [46] [6]. After the WSSJD transformation, two parallel sub-channels are formed, given by

$$\bar{r}_i^1 = \sum_{k=0}^{\nu} h_k z_{i-k}^1 + \bar{\omega}_i^1, \quad \bar{r}_i^2 = \sum_{k=0}^{\nu} h_k z_{i-k}^2 + \bar{\omega}_i^2 \tag{3.11}$$

where

$$\begin{bmatrix} z_i^1 \\ z_i^2 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} x_i^1 \\ x_i^2 \end{bmatrix} \tag{3.12}$$

$$\begin{bmatrix} \bar{r}_i^1 \\ \bar{r}_i^2 \end{bmatrix} = \begin{bmatrix} \frac{1}{1+\epsilon} & 0 \\ 0 & \frac{1}{1-\epsilon} \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} r_i^1 \\ r_i^2 \end{bmatrix} \tag{3.13}$$

$$\begin{bmatrix} \bar{\omega}_i^1 \\ \bar{\omega}_i^2 \end{bmatrix} = \begin{bmatrix} \frac{1}{1+\epsilon} & 0 \\ 0 & \frac{1}{1-\epsilon} \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} \omega_i^1 \\ \omega_i^2 \end{bmatrix}. \tag{3.14}$$

are the new input symbol, received symbol, and noise component of the transformed system (3.11), respectively. The noise samples satisfy $\bar{\omega}_i^1 \sim \mathcal{N}(0, \frac{2\sigma^2}{(1+\epsilon)^2})$, $\bar{\omega}_i^2 \sim \mathcal{N}(0, \frac{2\sigma^2}{(1-\epsilon)^2})$.

In this new system, $z^1(D)$ and $z^2(D)$ are transmitted separately through $h(D)$. If we treat z_i^1 and z_i^2 as the real and imaginary components of a complex symbol, the resulting system is QAM-like, where the only difference is that the two subchannels have different signal-to-noise ratios (SNRs). Considering this dimensional asymmetry, we define the **effective symbol pair distance** (ESPD) between two input symbols $\boldsymbol{z}_i$ and $\tilde{\boldsymbol{z}}_i$ as

$$d_{\mathrm{e}}^2(\boldsymbol{z}_i, \tilde{\boldsymbol{z}}_i) = \frac{(1+\epsilon)^2}{2}(z_i^1 - \tilde{z}_i^1)^2 + \frac{(1-\epsilon)^2}{2}(z_i^2 - \tilde{z}_i^2)^2. \tag{3.15}$$

The ESPDs between different pairs of inputs are listed in Table 3.1, and plotted in 3.3. Notice that the ESPDs, Δ_1^2, Δ_2^2, and Δ_3^2, show different monotonicity behavior when ϵ changes. For $\epsilon \in [0, 0.5]$, Δ_1^2 and Δ_3^2 are increasing functions, while Δ_2^2 decreases. The changes in ESPDs affect the performance of a reduced-state trellis. Therefore, even with the same subset trellis configuration, the RSSE performs differently at various ITI levels.

Based on Table 3.1, we propose a set partition tree shown in Fig. 3.2. The horizontal and vertical axes in the constellation correspond to z_i^1 and z_i^2 dimension, respectively. This tree contains 4 levels, $\{L_1, L_2, L_3, L_4\}$, each of which is a set partition of the constellation. The minimum ESPD on each level is specified on the right side. The number labeled on each branch is the index of the subset in the corresponding set partition. Compared to the Ungerboeck set

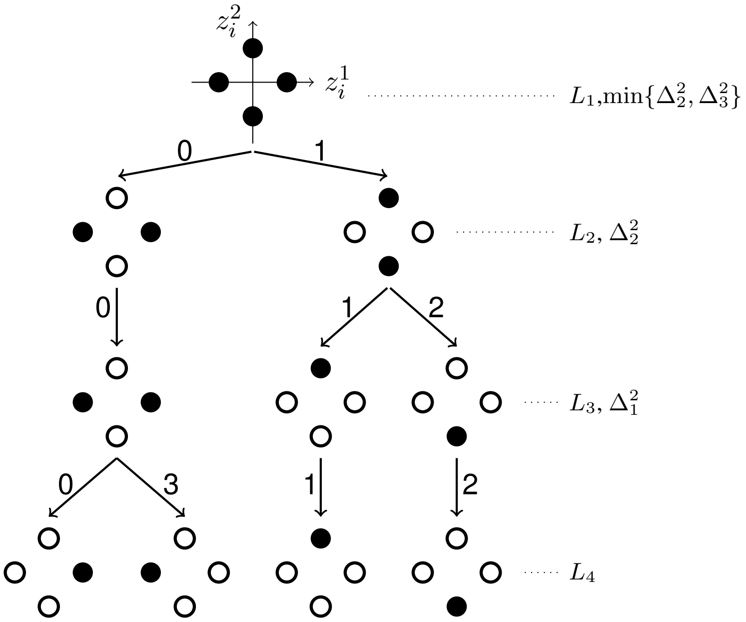


Figure 3.2: The set partition tree constructed for 2H2T system.

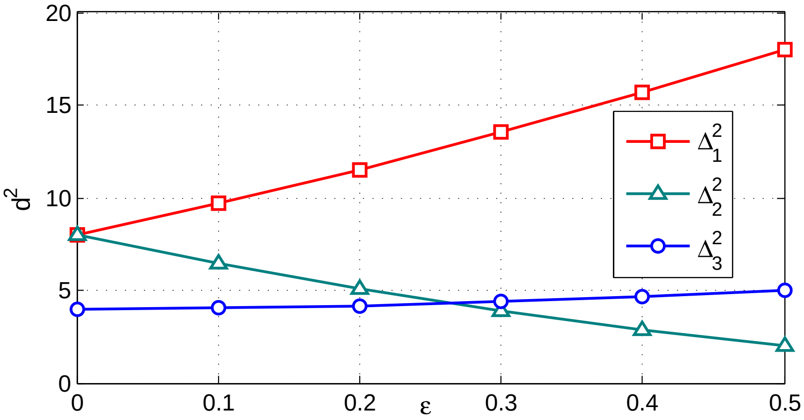


Figure 3.3: ESPDs as functions of ϵ

Table 3.1: The ESPDs between different input symbols.

$(\boldsymbol{z}_i, \tilde{\boldsymbol{z}}_i)$	$d_{\mathrm{e}}(\boldsymbol{z}_i, \tilde{\boldsymbol{z}}_i)$
$([+2,0],[-2,0])$	$\Delta_1^2 = 8(1+\epsilon)^2$
$([0,+2],[0,-2])$	$\Delta_2^2 = 8(1-\epsilon)^2$
$([+2,0],[0,+2])$	$\Delta_3^2 = 4(1+\epsilon^2)$
$([+2,0],[0,-2])$	
$([-2,0],[0,+2])$	
$([-2,0],[0,-2])$	

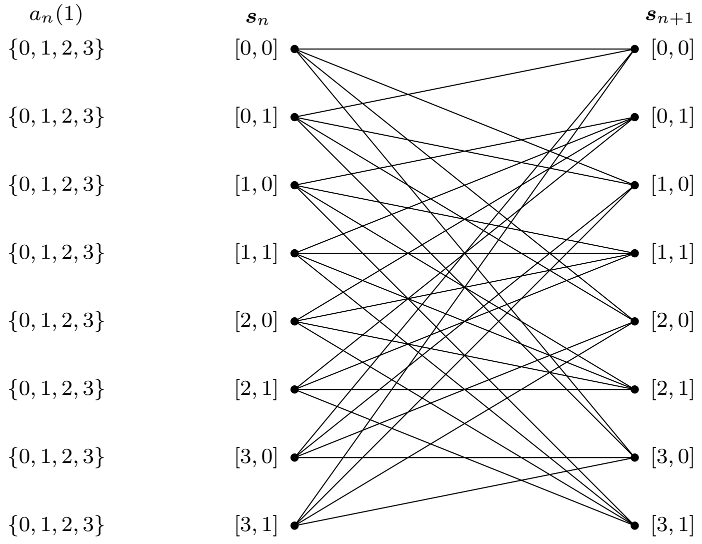


Figure 3.4: Subset trellis with configuration [4,2] on memory-2 channel.

partition tree, the additional level L_3 comes from the asymmetric distance measure in the z^1 and z^2 dimensions, and it provides better flexibility in performance/complexity tradeoff. We emphasize that, although the proposed set partition tree is motivated by the WSSJD transformation, it can be implemented with the standard ML detectors.

The subset trellis is constructed by choosing $\Omega(k)$ from the levels of the set partition tree for each $k = 1, \cdots, \nu$, and to guarantee a well-defined trellis structure, $\Omega(k)$ should always be at the same level or at a higher level than $\Omega(k-1)$. During the detection process, only one ML state can survive inside each subset state at each time slot. Consider the example given in Fig. 3.5. The subset trellis is constructed for PR2 channel $1 + 2D + D^2$, has configuration $[4, 2]$. The left columns list the subset states $\boldsymbol{s}_i$ and their associated survivor ML states $\boldsymbol{p}_i$. The label on each branch is the channel input$|$output. All of the branches terminate at subset state $\boldsymbol{s}_{i+1} = [0, 0]$.

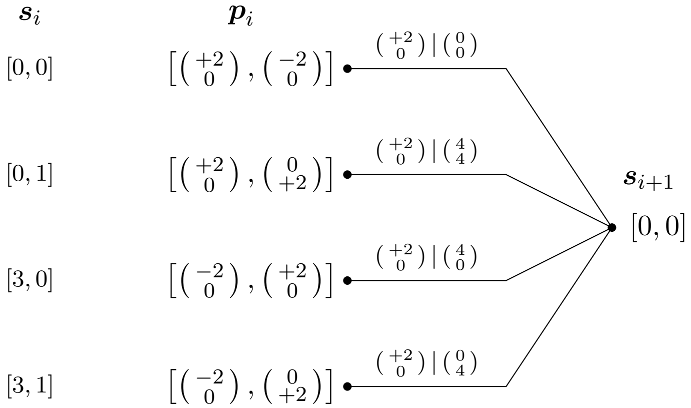
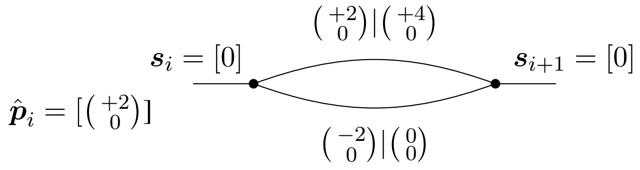
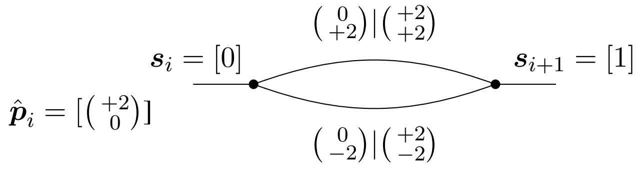


Figure 3.5: An illustration of detection on subset trellis.



(a) parallel branches from subset state 0 to 0



(b) parallel branches from subset state 0 to 1

Figure 3.6: Sample parallel branches for subset trellis with 2 states constructed for channel $1 + D$.

Assume $\epsilon = 0.1$. Once the survivor state $\boldsymbol{p}_i$ is decided, the output labels are also determined. A look-up table can be stored to facilitate the process of finding the corresponding output labels given the survivor ML states. Assume the transformed received signals are $\bar{\boldsymbol{r}} = (\bar{r}^1, \bar{r}^2) = (5, 3)$. The metric comparison shows that the subset state $[0, 1]$ with survivor ML state $\left[\binom{+2}{0}, \binom{0}{+2}\right]$ gives the smallest path metric, so the survivor ML state of $\boldsymbol{s}_{n+1} = [0, 0]$ is updated to be $\left[\binom{+2}{0}, \binom{+2}{0}\right]$, which will be used in the next time slot.

For a configuration with $J_1 < 4$, the subset trellis contains parallel branches. A pre-selection between the parallel branches is required during detection. Due to the symmetric property of WSSJD trellis labels, this pre-selection can be done without explicitly calculating the branch metric, for $J_1 = 2$ or 3. For instance, consider the two scenarios illustrated in Fig. 3.6. In both cases, the survivor ML state at the starting stage is assumed to be $\hat{\boldsymbol{p}}_i = [(+2, 0)]$. The input and output labels are marked on the branches. In Fig. 3.6(a), both the input symbols $(+2, 0)$ and $(-2, 0)$ lead the paths to subset state 0. Notice that the input symbols $(+2, 0)$ and $(-2, 0)$ have the same value in the z^2 dimension, producing the same output on the subtract channel. Instead of calculating metrics from (3.9), the pre-selection performs a thresholding on the sum channel output and makes the decision. In this example, the threshold is $+2$, obtained by averaging $+4$ and 0. If $\bar{r}_i^1 > +2$, the strategy is to pick $(+2, 0)$ as the survivor symbol, while for the case $\bar{r}_i^1 < +2$, $(-2, 0)$ should be the survivor. Similarly for another case shown in Fig. 3.6(b), the thresholding is conducted on the subtract channel output, since the two input symbols produce the same output in the sum channel. By comparing $\bar{r}_i^2$ with the threshold 0, the detector picks $(0, +2)$ if $\bar{r}_i^1 > 0$, or $(0, -2)$ if $\bar{r}_i^2 < 0$. This symmetry property renders the WSSJD formulation preferable over the traditional ML detector.

3.4 Performance of RSSE on 2H2T System

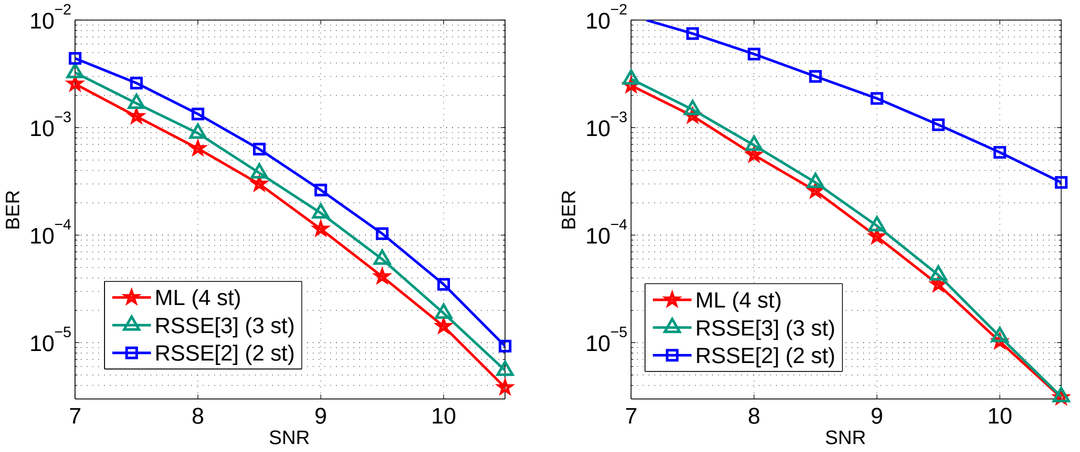
We examine the RSSE performance on various subset trellises constructed from the proposed set partition tree. Several types of channels at different ITI levels are considered. The SNR is defined as

$$\text{SNR(dB)} = 10 \log \frac{\|h(D)\|^2}{2\sigma^2} \tag{3.16}$$

where $\|h(D)\|^2 = \sum_i h_i^2$.

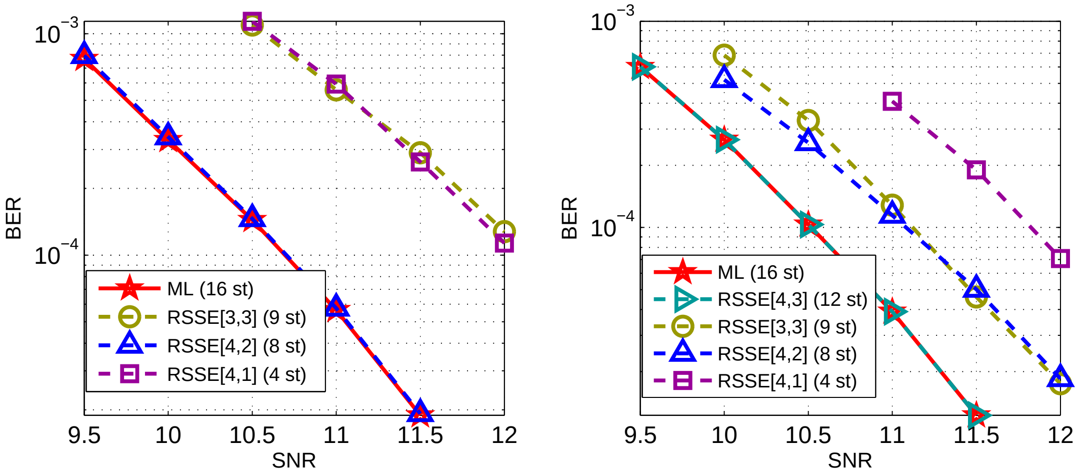
3.4.1 Dicode Channel

This simple example helps us understand how the pre-selection between parallel branches affects the system. Although early-merging happens at every time step, it does not seriously degrade the performance. From Fig. 3.7 we see that the performance loss of the 3-state subset trellis is less than 0.1dB. Moreover, the 3-state RSSE has better performance at the higher ITI



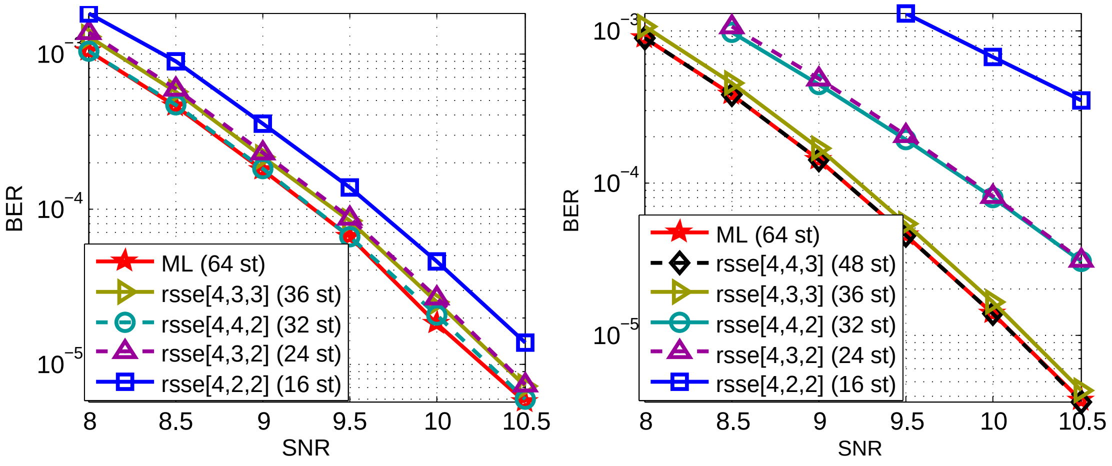
(a) $\epsilon = 0.1$ (b) $\epsilon = 0.3$

Figure 3.7: Performance comparison between RSSE and ML detector on dicode channel at different ITI levels. The legend shows the RSSE subset trellis configuration and the corresponding number of trellis states.



(a) $\epsilon = 0.1$ (b) $\epsilon = 0.3$

Figure 3.8: Performance comparison between RSSE and ML detector on PR2 channel at different ITI levels. The legend shows the RSSE subset trellis configuration and the corresponding number of trellis states.



(a) $\epsilon = 0.1$

(b) $\epsilon = 0.3$

Figure 3.9: Performance comparison between RSSE and ML detector on EPR4 channel at different ITI levels. The legend shows the RSSE subset trellis configuration and the corresponding number of trellis states.

level ($\epsilon = 0.3$), while the 2-state RSSE performs better at the lower ITI level ($\epsilon = 0.1$). In Section 3.5.1, we explain this observation by analyzing the length-1 error events.

3.4.2 Channel with Higher Memory

Higher channel memory provides more flexibility in constructing the subset trellis. PR2 and EPR4 are two commonly used PR targets to approximate magnetic recording channels. For the PR2 channel, $h(D) = 1 + 2D + D^2$, the bit error rate (BER) performance as a function of SNR at different ITI levels is plotted in Fig. 3.8. The comparison between Fig. 3.8(a) and Fig. 3.8(b) shows that even using the same subset trellis, RSSE performs differently at different ITI levels, and its performance correlates with the minimum intrasubset ESPDs of the set partitions configured in the subset trellis. At a low ITI level ($\epsilon = 0.1$), the performance of RSSE on the $[4, 2]$ subset trellis coincides with that of the ML detector. The performance of trellis $[4, 3]$ is not plotted, but can be predicted to be close to the ML curve. The other two trellises, $[4, 1]$ and $[3, 3]$, lose approximately 1.25dB. When the ITI level becomes higher ($\epsilon = 0.3$), the subset trellis $[4, 2]$ cannot provide reliable early path merging because the minimum intrasubset ESPD Δ_2^2 in $\Omega(3) = L_2$ is substantially reduced. However, $[4, 3]$ can achieve near-ML performance.

Table 3.2: The SNR loss of different subset trellis configurations to achieve BER= 10^{-4}.

(a) PR2 channel

Trellises	ϵ			
	0.1	0.2	0.3	0.4
RSSE $[4,1]$	1.25dB	1.3dB	1.35dB	1.2dB
RSSE $[4,2]$	$\ll 0.1$dB	0.15dB	0.6dB	1.1dB
RSSE $[3,3]$	1.4dB	0.9dB	0.6dB	0.2dB
RSSE $[4,3]$	$\ll 0.1$dB			

(b) EPR4 channel

Trellises	ϵ			
	0.1	0.2	0.3	0.4
RSSE $[4,3,3]$	0.1dB	0.1dB	0.05dB	$\ll 0.1$dB
RSSE $[4,4,2]$	$\ll 0.1$dB	0.15dB	0.7dB	$>$ 1dB
RSSE $[4,3,2]$	0.1dB	0.25dB	0.7dB	$>$ 1dB
RSSE $[4,2,2]$	0.3dB	$>$ 1dB		
RSSE $[3,3,3]$	$>$ 1dB	0.7dB	0.4dB	0.05dB
RSSE $[4,3,1]$	$>$ 1dB			

The trellis $[4,1]$ still has a 1.35dB loss, while the increase of Δ_1^2 brings $[3,3]$ closer to the ML performance.

The simulation results for the EPR4 channel $h(D) = 1 + D - D^2 - D^3$ are shown in Figs. 3.9. The comparison between Figs. 3.9(a) and Figs. 3.9(b) shows that even using the same subset trellis, RSSE performs differently under different ITI levels, and its performance correlates with the minimum intrasubset ESPDs of the set partitions configured in the subset trellis. At a low ITI level ($\epsilon = 0.1$), the performance of RSSE on subset trellis $[4,4,2]$ coincides with that of the ML detector. The BER curves of $[4,3,3]$ and $[4,3,2]$ overlap, and are both within 0.1dB away from the ML curve. Subset trellis $[4,2,2]$ further reduces the number of states to 16, but incurs a 0.3dB loss. When the ITI level becomes higher ($\epsilon = 0.3$), the subset trellis $[4,4,2]$ cannot provide reliable early path merging because the minimum intrasubset ESPD Δ_2^2 in $\Omega(3) = L_2$ is substantially reduced. However, a less aggressive construction using configuration $[4,4,3]$ achieves near-ML performance. The decrease in Δ_2^2 at this ITI level also degrades the performance of RSSE$[4,2,2]$ and $[4,3,2]$. Their BER curves overlap in Figs. 3.9(b). In contrast, the increase of Δ_1^2 brings $[4,3,3]$ closer to the ML performance, compared to the case $\epsilon = 0.1$.

Tables 3.2 summarize the performance loss in dB for several subset trellis configurations compared to an ML detector at BER= 10^{-4} on the PR2 and EPR4 channels, respectively. Several conclusions can be drawn from these tables. First, a trellis with fewer states does not necessarily have worse performance than one with more states. For example, in Table 3.2(b) for the EPR4 channel, when $\epsilon = 0.1$, the $[4, 4, 2]$ configuration with 32 states outperforms the$[4, 3, 3]$ configuration with 36 states. Second, the performance of a configuration may change drastically at different ITI levels. One example is the $[4, 4, 2]$ trellis, which essentially achieves ML performance at $\epsilon = 0.1$, but loses over 1dB for $\epsilon = 0.4$. Finally, not all configurations suffer further performance losses at higher ITI. It is interesting to observe that the RSSE$[3, 3, 3]$ tellis with parallel branches can have near-optimal performance at $\epsilon = 0.4$. Therefore, the pre-selection between parallel branches at every stage is quite reliable. In Section 3.5 we give an explanation of these observations from the point of view of error event analysis.

3.4.3 Minimum phase channels

Minimum phase channels can better model the real channel on a disk drive. Assume the transition response of a perpendicular magnetic recording (PMR) disk is $s(t) = V_{\max} \tanh(\frac{2t}{0.579\pi\delta})$, where $V_{\max}$ is the writing voltage and δ indicates the linear density on one data track. Using the whitened matched filter structure in [19], we derive two minimum phase channel polynomials: channel 1, $h(D) = 1 + 1.6D + 1.1D^2 + 0.4D^3$ for $\delta = 1.3$, and channel 2, $h(D) = 1 + 1.9D + 1.6D^2 + 0.8D^3 + 0.3D^4$ for $\delta = 1.5$. These are two commonly used densities in current commercial HDDs. Since the minimum phase condition implies that most of the channel energy is distributed over the most recent samples, the early merge in RSSE can be more reliable for these channels compared to linear phase channels, such as PR2 and EPR4. It is interesting to compare channel 1 and EPR4, both of which have memory $\nu = 3$. As shown in Fig. 3.10(a), ML, RSSE$[4, 3, 2]$, and RSSE$[4, 2, 2]$ have essentially identical performance. Therefore, RSSE can achieve near-ML performance with only 16 states, if the $[4, 2, 2]$ trellis is used, whereas the ML detector requires 64 states. The performance of other, more aggressive configurations is also plotted. As can be seen, RSSE with only 8 states can achieve performance that is within

0.3dB of ML detection.

The simulation results for channel 2 are plotted in Fig. 3.10(b). They show that RSSE$[4, 2, 2, 2]$ with 32 states can achieve near-ML performance. In constrast, the ML trellis requires 256 states. If 0.1dB loss is permissible, the RSSE $[4, 2, 2, 1]$ trellis can be used, reducing the number of states to only 16.

In summary, the simulation results on linear phase channels and minimum phase channels show that RSSE can achieve near optimal performance with significantly reduced number of states. It could potentially substitute the ML detector when the channel interference becomes much more severe.

3.5 Error Event Analysis

We will use error event analysis to study the performance-complexity tradeoff among different subset trellis configurations.

The detector makes errors if the survivor path diverges from the correct one. Let

$$\boldsymbol{e}(D) = [e^1(D), e^2(D)] = [x^1(D) - \hat{x}^1(D), x^2(D) - \hat{x}^2(D)] \tag{3.17}$$

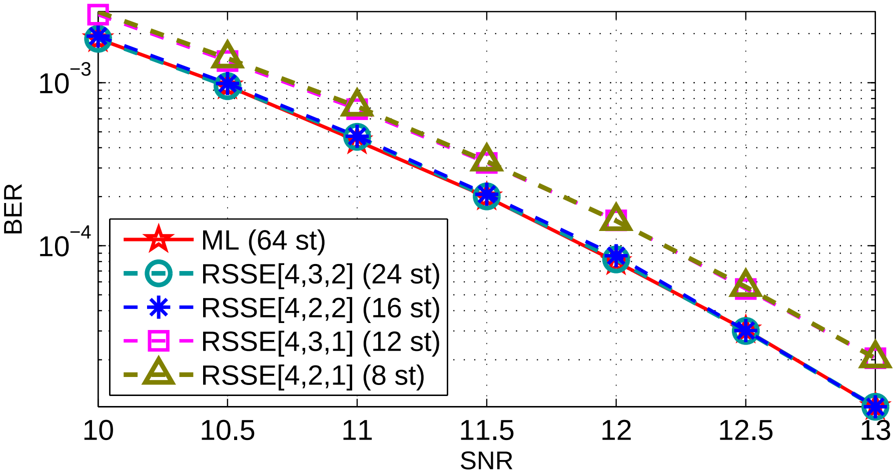
denote an error event of the original 2H2T system, and let

$$v\bar{c}te(D) = [\bar{e}^1(D), \bar{e}^2(D)] = [z^1(D) - \hat{z}^1(D), z^2(D) - \hat{z}^2(D)] \tag{3.18}$$

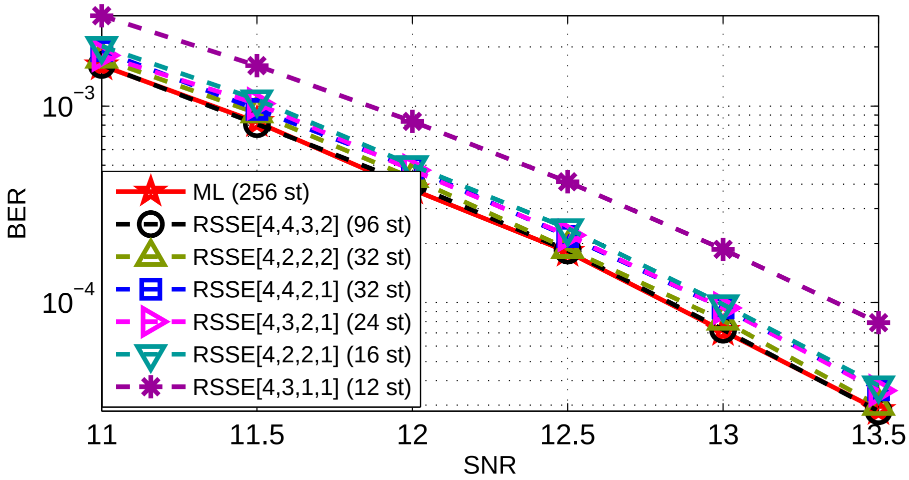
be the transformed error event of the WSSJD system. We also use $\boldsymbol{e}_i = [e_i^1, e_i^2]^\top$ and $\bar{\boldsymbol{e}}_i = [\bar{e}_i^1, \bar{e}_i^2]^\top$ to represent the original and transformed error symbols at time i, respectively. It is easy to see that

$$\begin{bmatrix} \bar{e}_i^1 \\ \bar{e}_i^2 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} e_i^1 \\ e_i^2 \end{bmatrix}. \tag{3.19}$$

It is well-known [19] that at high SNR, the error event probability of a trellis-based



(a) $\delta = 1.3$



(b) $\delta = 1.5$

Figure 3.10: Performance comparison between RSSE and ML detector on minimum phase channels at $\epsilon = 0.1$. The polynomials are (a) $h(D) = 1 + 1.6D + 1.1D^2 + 0.4D^3$, (b) $h(D) = 1 + 1.9D + 1.6D^2 + 0.8D^3 + 0.3D^4$.

detector can be approximated by $P_e \approx c \cdot Q(\frac{d_{\text{min}}}{2\sigma})$, where $Q(\cdot)$ is the area under the tail of the standard Gaussian distribution,

$$d^2_{\text{min}} = \min_{\boldsymbol{e}(D)} d^2(\boldsymbol{e}(D)) \tag{3.20}$$

is the **minimum distance parameter**, and c is a coefficient indicating the average number of error events at distance d^2_{min}. Due to the exponential nature of the Q function, the performance comparison between two detectors can be easily conducted by considering their minimum distance parameter. The error events that lead to d^2_{min} are the **dominant error events**.

For the WSSJD detector, an effective measure of the distance associated with $\bar{\boldsymbol{e}}(D)$ is defined by

$$d^2_{\text{W}}(\bar{\boldsymbol{e}}(D)) = \frac{(1+\epsilon)^2}{2}\|\bar{e}^1(D)h(D)\|^2 + \frac{(1-\epsilon)^2}{2}\|\bar{e}^2(D)h(D)\|^2. \tag{3.21}$$

Comparing (3.15) to (3.21), we can see that ESPD is proportional to the distance associated with a single error symbol $\bar{\boldsymbol{e}}_i$. Recall that the set partition tree is constructed based on ESPDs, and $\Delta^2_1, \Delta^2_2, \Delta^2_3$ are varying with respect to ϵ. Therefore, the minimum distance parameter of the reduced-state trellis configuration also changes with ϵ, and its trend can be roughly predicted by analyzing the change of minimum ESPD in each $\Omega(k)$. We will give more detailed insights into this behavior in the following subsections. The minimum value of $d^2_{\text{W}}(\bar{\boldsymbol{e}}(D))$ is abbreviated to d^2_{min}, which is the minimum distance parameter of the 2H2T ML detector. It serves as a benchmark for evaluating the performance of the RSSE algorithm.

3.5.1 Parallel Branches

For the subset trellis with parallel branches, early merge happens at every time instant. Ignoring the error propagation effect, we assume that at time i, both the correct and the estimated sequences are at state $\boldsymbol{s}_i$, and $\boldsymbol{z}_{i-k} = \hat{\boldsymbol{z}}_{i-k}$ for all $k = 1, ..., \nu$. At time $i+1$, if $\boldsymbol{z}_i, \hat{\boldsymbol{z}}_i \in a_i(1)$, the detector needs to decide a survivor symbol, and discard the other one. Once the correct symbol is discarded, this wrong decision can not be reversed in the remaining steps. The probability of making a wrong decision in the parallel branch selection is $Q(\frac{h_0 d_e(\boldsymbol{z}_i,\hat{\boldsymbol{z}}_i)}{2\sigma})$, where $d(\boldsymbol{z}_i,\hat{\boldsymbol{z}}_i)$ is the square-root of ESPD. Let E_1 be the set of all such length-1 error events due to the parallel branches. Then, $\bar{\boldsymbol{e}}_i \in E_1$ if and only if there exist two inputs $\boldsymbol{z}_i, \hat{\boldsymbol{z}}_i \in a_i(1)$ such that $\bar{\boldsymbol{e}}_i = \boldsymbol{z}_i - \hat{\boldsymbol{z}}_i$. It can be shown that

$$\begin{aligned} d^2_{\text{min}}(E_1) &= \min_{\bar{\boldsymbol{e}}_i \in E_1} \frac{(1+\epsilon)^2}{2}(h_0 \bar{e}_i^1)^2 + \frac{(1-\epsilon)^2}{2}(h_0 \bar{e}_i^2)^2 \\ &= \begin{cases} 8h_0^2(1+\epsilon)^2 = h_0^2 \Delta_1^2 & J_1 = 3 \\ 8h_0^2(1-\epsilon)^2 = h_0^2 \Delta_2^2 & J_1 = 2. \end{cases} \end{aligned} \tag{3.22}$$

The existence of parallel branches will not significantly degrade the performance if it can achieve the same minimum distance as ML detection, i.e., $d^2_{\text{min}}(E_1) \geq d^2_{\text{min}}$. For the dicode channel,

$$d^2_{\text{min}} = \begin{cases} 8(1+\epsilon^2) & \text{if } 0 \leq \epsilon \leq 2-\sqrt{3} \\ 16(1-\epsilon)^2 & \text{if } 2-\sqrt{3} \leq \epsilon \leq 1/2. \end{cases} \tag{3.23}$$

Therefore, for the 3-state subset trellis, $d^2_{\text{min}}(E_1) \geq d^2_{\text{min}}$ for all ϵ, leading to performance close to the ML detector, as shown in Fig. 3.7. Moreover, as ϵ increases, $d^2_{\text{min}}(E_1)$ becomes much larger than d_{min}, making the effect of these length-1 error events negligible. So we observe that the BER curve of the 3-state RSSE trellis almost overlaps with that of the ML detector at $\epsilon = 0.3$. In contrast to the 3-state trellis, in the 2-state trellis $d^2_{\text{min}}(E_1) < d^2_{\text{min}}$ for all ϵ, resulting in worse performance.

As for the PR2 and EPR4 channels, their d^2_{min} is given by

$$d^2_{\text{min}} = \begin{cases} 16(1+\epsilon^2) & \text{if } 0 \leq \epsilon \leq 2-\sqrt{3} \\ 32(1-\epsilon)^2 & \text{if } 2-\sqrt{3} \leq \epsilon \leq 1/2. \end{cases} \tag{3.24}$$

Consider a subset trellis with $J_1 = 3$. For $\epsilon > \frac{1}{3}$, $d^2_{\text{min}}(E_1)$ is strictly larger than d^2_{min}. Therefore the error events in E_1 are not the dominant ones. As shown in Tables 3.2(a) and 3.2(b), at $\epsilon = 0.4$, the RSSE $[3,3]$ and RSSE $[3,3,3]$ configurations perform very close to their corresponding ML

Table 3.3: Error symbols by index.

index	$[\bar{e}_i^1, \bar{e}_i^2]$	$[e_i^1, e_i^2]$
0	$[0,0]$	$[0,0]$
1	$[4,0]$	$[2,2]$
2	$[-4,0]$	$[-2,-2]$
3	$[0,4]$	$[2,-2]$
4	$[0,-4]$	$[-2,2]$
5	$[2,2]$	$[2,0]$
6	$[-2,-2]$	$[-2,0]$
7	$[2,-2]$	$[0,2]$
8	$[-2,2]$	$[0,-2]$

detectors, respectively.

3.5.2 Early Merging Condition

We next try to identify longer RSSE error events. Suppose the decoding paths of $[z^1(D), z^2(D)]$ and $[\hat{z}^1(D), \hat{z}^2(D)]$ are merged at times i_1 and i_2 and unmerged in between. Let E denote the set of all error events ending at time i_2, where the starting position i_1 is arbitrary. According to [14], an error event $\bar{\boldsymbol{e}}(D) \in E$ if and only if the following hold.

1. $\bar{\boldsymbol{e}}_{i_1}$ is non-zero.
2. The last ν elements, $[\bar{\boldsymbol{e}}_{i_2-\nu}, \cdots, \bar{\boldsymbol{e}}_{i_2-1}]$, should satisfy the **merging condition**, i.e., $\bar{\boldsymbol{e}}_{i_2-k} = \boldsymbol{z}_{i_2-k} - \hat{\boldsymbol{z}}_{i_2-k}$ where $\boldsymbol{z}_{i_2-k}$ and $\hat{\boldsymbol{z}}_{i_2-k}$ belong to the same subset in the partition $\Omega(k)$ for all $k = 1, ..., \nu$.
3. No earlier ν elements satisfy the merging condition.

In MLSE, the merging condition requires $\bar{\boldsymbol{e}}_{i_2-k} = 0$ for $k = 1, ..., \nu$. However, this is not the case in RSSE. We call the error events $\bar{\boldsymbol{e}}(D) \in E$ whose last ν elements are not all zero the **early merged error events**, denoted by E^r. Clearly $E_1 \subseteq E^r$.

We now present a necessary and sufficient condition for an error event $\bar{\boldsymbol{e}}(D)$ to belong to E^r. We refer to this as the **early merging condition**. We first introduce some terminology.

For a partition Ω of the input constellation, the set of **intrasubset errors**, denoted by $\mathcal{E}_a(\Omega)$, is a collection of error symbols such that if there exist two input symbols $\boldsymbol{z}, \hat{\boldsymbol{z}}$ satisfying the condition that $\bar{\boldsymbol{e}} = \boldsymbol{z} - \hat{\boldsymbol{z}}$ and $\boldsymbol{z}, \hat{\boldsymbol{z}}$ belong to the same subset in Ω, then $\bar{\boldsymbol{e}} \in \mathcal{E}_a(\Omega)$. Similarly, the set of **intersubset errors**, denoted by $\mathcal{E}_b(\Omega)$, is a collection of error symbols such that if there exist two inputs $\boldsymbol{z}, \hat{\boldsymbol{z}}$ satisfying the condition that $\bar{\boldsymbol{e}} = \boldsymbol{z} - \hat{\boldsymbol{z}}$ and $\boldsymbol{z}, \hat{\boldsymbol{z}}$ belong to two different subsets in Ω, then $\bar{\boldsymbol{e}} \in \mathcal{E}_b(\Omega)$.

The following proposition gives the relationship between the intrasubset errors and intersubset errors for the set partition tree in Fig. 3.2. For convenience, the error symbols are indexed by the digits shown in Table 3.3.

Proposition 1. *For the proposed set partition tree in Fig. 3.2,* $\mathcal{E}_a(L_i) \cap \mathcal{E}_b(L_i) = \emptyset$ *for* $i = 1, 2, 3, 4$.

Proof. We prove the claim by enumeration.

1. If $\Omega = L_1$, all error symbols are intrasubset errors since there is only one subset.
2. If $\Omega = L_2$, $\mathcal{E}_a(L_2) = \{0, 1, 2, 3, 4\}$, $\mathcal{E}_b(L_2) = \{5, 6, 7, 8\}$.
3. If $\Omega = L_3$, $\mathcal{E}_a(L_3) = \{0, 1, 2\}$, $\mathcal{E}_b(L_3) = \{3, 4, 5, 6, 7, 8\}$.
4. If $\Omega = L_4$, all non-zero error symbols are intersubset errors, so $\mathcal{E}_a(L_4) = \{0\}$. ■

Proposition 2. *(Early merging condition)*
An error event $\bar{\boldsymbol{e}}(D) \in E^r$ *if and only if the last* ν *elements are not all zero symbols, and satisfy* $\bar{\boldsymbol{e}}_{i_2-k} \in \mathcal{E}_a(\Omega(k))$ *for all* $k = 1, ..., \nu$, *and no previous* ν*-tuple satisfies the condition.*

Proof. Given $\bar{\boldsymbol{e}}(D) \in E^r$, it is straightforward from the definition of "merging condition" that the last ν elements must be intrasubset error symbols in the corresponding patition $\Omega(k)$. On the other hand, if $\bar{\boldsymbol{e}}_{i_2-k} \in \mathcal{E}_a(\Omega(k))$ for all $k = 1, ..., \nu$, by Proposition 1, the sequences that produce $\bar{\boldsymbol{e}}(D)$ must satisfy that $\boldsymbol{z}_{i_2-k}$ and $\hat{\boldsymbol{z}}_{i_2-k}$ belong to the same subset in $\Omega(k)$. Therefore the decoding paths are merged at i_2, and $\bar{\boldsymbol{z}}(D) \in E^r$. ■

Remark 1. *Notice that Proposition 1 is also true for the QAM Ungerboeck set partition tree. So Proposition 2 also applies to the original RSSE formulation.*

Remark 2. *The single track error events are not affected by the RSSE algorithm if* $J_i > 1$ *for all* $i = 1, ..., \nu$.

Assume $\bar{\mathbf{e}}(D) \in E^r$ and starts from k_1. The distance parameter of $\bar{\mathbf{e}}(D)$ is given by

$$\begin{aligned} d_{\mathrm{r}}^2(\bar{\mathbf{e}}(D)) = & \frac{(1+\epsilon)^2}{2} \sum_{i=i_1}^{i_2} (\sum_{k=0}^{\nu} h_k \bar{e}_{i-k}^1)^2 \\ & + \frac{(1-\epsilon)^2}{2} \sum_{i=i_1}^{i_2} (\sum_{k=0}^{\nu} h_k \bar{e}_{i-k}^2)^2. \end{aligned} \tag{3.25}$$

The distance parameter measured by (3.25) is always smaller than or equal to that measured by (3.21) [14]. The possible reduction represents the price paid for using the reduced-state trellis. An example is given to illustrate the difference.

Example 3. *Consider the PR2 channel. Assume that* $\epsilon = 0.1$. *Assume a single error* $\bar{\mathbf{e}}_i = [4, 0]^\top$ *happens at time* i. *In ML detection, the paths remerge at time* $i + 2$, *and the distance parameter contributed by* $\bar{\mathbf{e}}_i$ *is* $\frac{(1+0.1)^2}{2} \cdot (4^2 + 8^2 + 4^2) = 58.08$. *However, if RSSE*$[4, 3]$ *is used, the paths will be early merged at time* $i + 1$, *since* $\bar{\mathbf{e}}_i \in \mathcal{E}_a(\Omega(2))$. *Therefore the distance parameter of this error event is reduced to* $\frac{(1+0.1)^2}{2} \cdot (4^2 + 8^2) = 48.4$ *in RSSE*$[4, 3]$.

Let

$$d_{\min}^2(E^r) = \min_{\bar{\mathbf{e}}(D) \in E^r} d_{\mathrm{r}}^2(\bar{\mathbf{e}}(D)). \tag{3.26}$$

The early merged error events $\bar{\mathbf{e}}^*(D)$ that achieve (3.26) are referred to as the dominant RSSE error events. To obtain good performance, it is essential that $d_{\min}^2(E^r) \geq d_{\min}^2$.

3.5.3 Error state diagram

An error state diagram can be employed to search for the minimum distance and enumerate the dominant error events. Consider a labeled directed graph $G = [V, E]$. The vertex set V is the collection of all possible error states $[\bar{\mathbf{e}}_{i-1}, \cdots, \bar{\mathbf{e}}_{i-\nu}]$, so $|V| = 9^\nu$. A state that satisfies the

Table 3.4: The dominant RSSE error events for channel $[1, 1.6, 1.1, 0.4]$.

	$\epsilon = 0.1$	$\epsilon = 0.2$	$\epsilon = 0.3$	$\epsilon = 0.4$
	$d^2_{\min} = 9.1304$	$d^2_{\min} = 9.4016$	$d^2_{\min} = 8.8592$	$d^2_{\min} = 6.5088$
RSSE$[3,3,3]$ (27 st)	[1]/9.6800	[5 2 1 2]/10.7168	[5 2 1 2]/10.0028	[5 2 1 2]/9.5472
RSSE$[3,3,2]$ (18 st)	[1]/9.6800	[5 2 1 2]/10.7168	[3 4 0 0]/8.2320	[3 4 0 0]/6.0480
RSSE$[4,3,2]$ (24 st)	[3 4 0 0]/13.6080	[3 4 0 0]/10.7520	[3 4 0 0]/8.2320	[3 4 0 0]/6.0480
RSSE$[4,2,2]$ (16 st)	[3 4 0]/10.4328	[3 4 0]/8.2432	[3 4 0]/6.3112	[3 4 0]/4.6368
RSSE$[3,2,2]$ (12 st)	[1]/9.6800	[3 4 0]/8.2432	[3 4 0]/6.3112	[3 4 0]/4.6368
RSSE$[4,2,1]$ (8 st)	[5 6 0 0]/8.4840	[3 4 0]/8.2432 [5 6 0 0]/8.7360	[3 4 0]/6.3112	[3 4 0]/4.6368

Table 3.5: The dominant RSSE error events for PR2 channel.

	$\epsilon = 0.1$	$\epsilon = 0.2$	$\epsilon = 0.3$	$\epsilon = 0.4$
	$d^2_{\min} = 16.16$	$d^2_{\min} = 16.64$	$d^2_{\min} = 15.68$	$d^2_{\min} = 11.52$
RSSE$[4,3]$ (12 st)	[5 (2 1)$^{\infty}$ 0]/24.24 [5 (2 1)$^{\infty}$ 2 0]/24.24	[5 (2 1)$^{\infty}$ 0]/24.96 [5 (2 1)$^{\infty}$ 2 0]/24.96	[5 (2 1)$^{\infty}$ 0]/26.16 [5 (2 1)$^{\infty}$ 2 0]/26.16	[5 (2 1)$^{\infty}$ 0]/27.84 [5 (2 1)$^{\infty}$ 2 0]/27.84
RSSE$[4,2]$ (8 st)	[(3 4)$^{\infty}$ 0]/19.44 [(3 4)$^{\infty}$ 3 0]/19.44	[(3 4)$^{\infty}$ 0]/15.36 [(3 4)$^{\infty}$ 3 0]/15.36	[(3 4)$^{\infty}$ 0]/11.76 [(3 4)$^{\infty}$ 3 0]/11.76	[(3 4)$^{\infty}$ 0]/8.64 [(3 4)$^{\infty}$ 3 0]/8.64
RSSE$[3,3]$ (9 st)	[1]/9.68 [5 2 1]/14.56 [(5 6)$^{\infty}$ 0 2]/16.16 [(5 6)$^{\infty}$ 5 0 1]/16.16 [(5 6)$^{\infty}$ 2 1]/16.16 [(5 6)$^{\infty}$ 5 1 2]/16.16	[1]/11.52 [5 2 1]/13.44 [(5 6)$^{\infty}$ 0 2]/16.64 [(5 6)$^{\infty}$ 5 0 1]/16.64 [(5 6)$^{\infty}$ 2 1]/16.64 [(5 6)$^{\infty}$ 5 1 2]/16.64	[1]/13.52 [5 2 1]/12.64	[5 2 1]/12.16

Table 3.6: The dominant RSSE error events for EPR4 channel.

	$\epsilon = 0.1$	$\epsilon = 0.2$	$\epsilon = 0.3$	$\epsilon = 0.4$
	$d^2_{\min} = 16.16$	$d^2_{\min} = 16.64$	$d^2_{\min} = 15.68$	$d^2_{\min} = 11.52$
RSSE$[4,3,3]$ (36 st)	[(5 0)$^{\infty}$ 1 0]/16.16	[(5 0)$^{\infty}$ 1 0]/16.64	[(5 0)$^{\infty}$ 1 0]/17.44	[(5 0)$^{\infty}$ 1 0]/18.56
RSSE$[4,3,2]$ (24 st)	[(5 0)$^{\infty}$ 1 0]/16.16	[(3 0)$^{\infty}$ 0]/15.36 [(3 4)$^{\infty}$ 3 0 0]/15.36 [3 4 (3 4)$^{\infty}$ 0 0]/15.36 [(5 0)$^{\infty}$ 1 0]/16.64	[(3 0)$^{\infty}$ 0]/11.76 [(3 4)$^{\infty}$ 3 0 0]/11.76 [3 4 (3 4)$^{\infty}$ 0 0]/11.76	[(3 0)$^{\infty}$ 0]/8.64 [(3 4)$^{\infty}$ 3 0 0]/8.64 [3 4 (3 4)$^{\infty}$ 0 0]/8.64
RSSE$[3,3,3]$ (27 st)	[1]/9.68 [5 6 1 2 1]/14.56 [5 2 1 2]/16.16 [(5 0)$^{\infty}$ 0 1]/16.16 [(5 0)$^{\infty}$ 1 0]/16.16 [(5 0)$^{\infty}$ 1 2]/16.16 [5 6 (5 6)$^{\infty}$ 0 0 2]/16.16 [5 (6 5)$^{\infty}$ 0 0 1]/16.16 [5 6 (5 6)$^{\infty}$ 1 2 1]/16.16 [5 (6 5)$^{\infty}$ 2 1 2]/16.16	[1]/11.52 [5 6 1 2 1]/13.44 [5 2 1 2]/16.64 [(5 0)$^{\infty}$ 0 1]/16.64 [(5 0)$^{\infty}$ 1 0]/16.64 [(5 0)$^{\infty}$ 1 2]/16.64 [5 6 (5 6)$^{\infty}$ 0 0 2]/16.64 [5 (6 5)$^{\infty}$ 0 0 1]/16.64 [5 6 (5 6)$^{\infty}$ 1 2 1]/16.64 [5 (6 5)$^{\infty}$ 2 1 2]/16.64	[1]/13.52 [5 6 1 2 1]/12.64	[5 6 1 2 1]/12.16

merging condition is called a **merging state**. For ML detection, the all-zero state is the only merging state, while for RSSE, additional early merging states are those which satisfy the early merging condition. If $\mathcal{T}$ denotes the set of merging states, then $|\mathcal{T}| = \prod_{k=1}^{\nu} |\mathcal{E}_a(\Omega(k))|$, which depends on the trellis configuration. An edge $(u, v) \in E$ starts from initial state $u = [\bar{\boldsymbol{e}}_{i-1}, \cdots, \bar{\boldsymbol{e}}_{i-\nu}]$ and ends in terminal state $v = [\bar{\boldsymbol{e}}_i, \cdots, \bar{\boldsymbol{e}}_{i-\nu+1}]$, with input/output label $\bar{\boldsymbol{e}}_i/\mathcal{L}_{\text{out}}$. Here

$$\mathcal{L}_{\text{out}} = \frac{(1+\epsilon)^2}{2}(\sum_{k=0}^{\nu} h_k \bar{e}^1_{i-k})^2 + \frac{(1-\epsilon)^2}{2}(\sum_{k=0}^{\nu} h_k \bar{e}^2_{i-k})^2. \tag{3.27}$$

Notice that all the merging states except the all-zero state are sink nodes, which have no outgoing edges. A path starting from the all-zero state and terminating at the merging state defines a closed error event, and the sum of the output labels of all edges in the path gives the distance parameter of this error event. A closed error event that ends at a non-zero merging state is an early merged error event. As proposed in [1], a depth-first algorithm can be used to find all the error events that lead to a distance parameter smaller than a given threshold.

We are interested in the dominant RSSE error events, i.e., the error events that end at non-zero merging states and produce the distance $d^2_{\text{min}}(E^r)$. Table 3.4 summarizes the dominant RSSE error events and their induced distances for several trellis configurations for the minimum phase channel $h(D) = 1+1.6D+1.1D^2+0.4D^3$. We simplify the table as follows: if $d^2_{\text{min}}(E^r) \geq d^2_{\text{min}}$, we only list the early merged error events that lead to $d^2_{\text{min}}(E^r)$; if $d^2_{\text{min}}(E^r) < d^2_{\text{min}}$, we list all the early merged error events whose distance parameters are smaller than or equal to d^2_{min}. The table is also simplified by considering the symmetry of the error events, i.e., $\pm(e^1(D), e^2(D))$ will produce the same distance parameter, and if the error events of track 1 and 2 are switched, the distance remains the same. So we group them together and only list the one whose first error symbol has a positive e^1_i component. As shown in Table 3.4, the early merged error events in the RSSE $[3, 3, 3]$ trellis always have distance parameter greater than d^2_{min}, under all ITI levels. Specifically, when $\epsilon = 0.1$, E_1 are the dominant RSSE error events. As ϵ increases, $d^2_{\text{min}}(E_1)$, which is proportional to Δ^2_1, also increases, and $[5, 2, 1, 2]$ becomes the dominant one. For the RSSE $[4, 3, 2]$ trellis, the error event $[3, 4, 0, 0]$ is dominant, and its distance parameter decreases as ϵ increases. In particular, for $\epsilon = 0.3$ and 0.4, its distance is strictly less than d^2_{min}, so it can be predicted that RSSE $[4, 3, 2]$ suffers greater performance loss compared to the ML detector at high ITI levels. One way to avoid this performance loss is to sacrifice complexity reduction and use RSSE$[4, 3, 3]$ which prevents the error event $[3, 4, 0, 0]$ from being early merged. RSSE $[4, 2, 2]$ has near-optimal performance at $\epsilon = 0.1$ and performs much worse when $\epsilon \geq 0.2$. A more aggressive configuration, RSSE$[4, 2, 1]$, cannot guarantee near-optimal performance since $d^2_{\text{min}}(E^r)$ is always smaller than d^2_{min}. Therefore, to retain near optimal performance as well as reduce complexity, we may use RSSE$[3, 2, 2]$ at $\epsilon = 0.1$, RSSE$[3, 3, 2]$ at $\epsilon = 0.2$, RSSE$[3, 3, 3]$ at $\epsilon = 0.3$ and 0.4.

For the PR2 and EPR4 channel, the error state diagrams contain zero cycles, leading to infinite recursive loops in the error event search. A zero cycle is a path that starts and ends at the same state, and accumulates zero path metric. The number of zero cycles depends on the reduced-state trellis configuration. In Examples 4 and 5, we summarize the zero cycles for PR2 and EPR4 channels. We follow the notations in [1] and let $(\boldsymbol{e}_1, \cdots, \boldsymbol{e}_k)^\infty$ represent an infinite periodic sequence with repeated pattern $\boldsymbol{e}_1, \cdots, \boldsymbol{e}_k$. Notice that a periodic sequence of the shifted pattern $(\boldsymbol{e}_i, \cdots, \boldsymbol{e}_k, \boldsymbol{e}_1, \cdots, \boldsymbol{e}_{i-1})^\infty$ is equivalent to $(\boldsymbol{e}_1, \cdots, \boldsymbol{e}_k)^\infty$.

Example4. *For PR2 channel, if the ML detector is used, the zero cycles are* 0^∞, $(1, 2)^\infty$, $(3, 4)^\infty$, $(5, 6)^\infty$, $(7, 8)^\infty$. *If RSSE*$[3, 3]$ *is used, both* $[2, 1]$ *and* $[1, 2]$ *becomes merging states, therefore* $(1, 2)^\infty$ *will not be a zero cycle, while other zero cycles still exist.*

Example5. *The zero cycles for the ML detector on EPR4 are* $(0)^\infty$,$\pm(0, 1)^\infty$, $\pm(0, 3)$, $\pm(0, 5)$, $\pm(0, 7)$, $\pm(1)^\infty$, $\pm(1, 2)^\infty$, $\pm(1, 3)^\infty$, $\pm(1, 4)^\infty$, $\pm(1, 5)^\infty$, $\pm(1, 6)^\infty$, $\pm(1, 7)^\infty$, $\pm(1, 8)^\infty$, $\pm(3)^\infty$, $\pm(3, 4)^\infty$, $\pm(3, 5)^\infty$, $\pm(3, 6)^\infty$, $\pm(3, 7)^\infty$, $\pm(3, 8)^\infty$, $\pm(5)^\infty$, $\pm(5, 6)^\infty$, $\pm(5, 7)^\infty$, $\pm(5, 8)^\infty$, $\pm(7)^\infty$, $\pm(7, 8)^\infty$. *Here* $-(\cdot)^\infty$ *represents taking the additive inverse of all symbols inside* $(\cdot)$.

Remark3. *The zero cycles do not intersect, so each state can only be visited by at most one zero cycle. We use* $\gamma(s)$ *to denote the zero cycle which starts and ends at state* u, *and* $\gamma(u, v)$ *to be* *the fragment of the zero cycle from state u to v. By an abuse of notation, we also use $\gamma(u, v)$ to represent the sequence of input labels on the fragment. The meaning will be clear according to the context.*

Let $\mathcal{Z}$ denote the collection of all the states visited by zero-cycles, and let $\mathcal{T}$ be the set of all merging states. A two-step algorithm introduced in [1] can be used to search for the dominant error events, with a slight modification that considers the additional early merging states in the RSSE trellis. The procedure is summarized below.

1. Given a threshold $\mathcal{D}$, apply the depth-first search algorithm to search for all the error fragments, whose path metric is no bigger than $\mathcal{D}$, and that start from some state $u \in \mathcal{Z}$ and end up at some state $v \in \mathcal{Z} \cup \mathcal{T}$ without having visited $\mathcal{Z} \cup \mathcal{T}$ in between. The path metric of such an error fragment is denoted as $d^2 \left(\bar{\boldsymbol{e}}(u, v)\right)$.
2. Construct a new graph F whose vertices are the states in $\mathcal{Z} \cup \mathcal{T}$. The edges in F are found as follows. If there is an error fragment $\bar{\boldsymbol{e}}(u, v)$ starting from state u and ending up at state v, then for each state $v' \in \gamma(v)$, there is an edge from state u to v'. The input label of the edge is $\bar{\boldsymbol{e}}(u, v) + \gamma(v, v')$, and the output label is $d^2 \left(\bar{\boldsymbol{e}}(u, v)\right)$, since the path metric from v to v' is zero. Parallel edges are allowed.
3. The same depth-first search on F can be used to search for and list all the closed error events whose distance parameters are less than $\mathcal{D}$.

Tables 3.5 and 3.6 list the dominant RSSE error events for several trellis configurations on the PR2 and EPR4 channels, respectively. They are constructed in the same manner as Table 3.4. The tables show a good match with the simulation results in Tables 3.2.

3.6 Asymmetric 2H2T System

The asymmetric 2H2T system is worth consideration because of its practical relevance. In this model, the ITI levels sensed by the two heads are different, i.e.,

$$\begin{bmatrix} r_i^1 \\ r_i^2 \end{bmatrix} = \begin{bmatrix} 1 & \epsilon - \Delta\epsilon \\ \epsilon + \Delta\epsilon & 1 \end{bmatrix} \begin{bmatrix} y_i^1 \\ y_i^2 \end{bmatrix} + \begin{bmatrix} \omega_i^1 \\ \omega_i^2 \end{bmatrix}. \tag{3.28}$$

Without loss of generality, we assume $0 \leq \Delta\epsilon \leq \epsilon$.

As in the discussion of the symmetric system, we analyze RSSE on the asymmetric channel by considering the transformed system. After the same coordinate transformation that was used before, the asymmetric 2H2T channel becomes

$$\begin{bmatrix} \bar{r}_i^1 \\ \bar{r}_i^2 \end{bmatrix} = \begin{bmatrix} 1 & \frac{\Delta\epsilon}{1+\epsilon} \\ \frac{\Delta\epsilon}{\epsilon-1} & 1 \end{bmatrix} \begin{bmatrix} \bar{y}_i^1 \\ \bar{y}_i^2 \end{bmatrix} + \begin{bmatrix} \bar{\omega}_i^1 \\ \bar{\omega}_i^2 \end{bmatrix}, \tag{3.29}$$

where $\bar{y}_i^1 = \sum_{k=0}^{\nu} h_k z_{i-k}^1$ and $\bar{y}_i^2 = \sum_{k=0}^{\nu} h_k z_{i-k}^2$, and $\bar{\boldsymbol{r}}_i$, $\bar{\boldsymbol{z}}_i$, and $\bar{\boldsymbol{\omega}}_i$ are obtained from equation (3.12), (3.13), and (3.14), respectively.

In the asymmetric system, the noiseless channel outputs become

$$f_i^1 = \bar{y}_i^1 + \frac{\Delta\epsilon}{1+\epsilon}\bar{y}_i^2, \qquad f_i^2 = \bar{y}_i^2 + \frac{\Delta\epsilon}{\epsilon-1}\bar{y}_i^1. \tag{3.30}$$

A joint trellis can be constructed by using the new output formulas, and then WSSJD is applicable. The same set partition tree shown in Fig. 3.2 is used to construct the subset trellis. We investigate change in performance by means of both simulation and error event analysis.

We first consider the case of parallel branches. Assume $J_1 > 1$. The effective squared distance between two parallel branches coming from the same state is

$$\begin{aligned}
d_{\mathrm{r}}^2(\bar{\boldsymbol{e}}_i \in E_1) &= \frac{(1+\epsilon)^2}{2}(h_0\bar{e}_i^1 + \frac{\Delta\epsilon}{1+\epsilon}h_0\bar{e}_i^2)^2 \\
&\quad + \frac{(1-\epsilon)^2}{2}(\frac{\Delta\epsilon}{\epsilon-1}h_0\bar{e}_i^1 + h_0\bar{e}_i^2)^2 \qquad (3.31)\\
&= \frac{h_0^2(1+\epsilon)^2}{2}[(\bar{e}_i^1)^2 + \frac{\Delta\epsilon^2}{(1+\epsilon)^2}(\bar{e}_i^2)^2] \\
&\quad + \frac{h_0^2(1-\epsilon)^2}{2}[\frac{\Delta\epsilon^2}{(\epsilon-1)^2}(\bar{e}_i^1)^2 + (\bar{e}_i^2)^2]. \qquad (3.32)
\end{aligned}$$

Here $\bar{\boldsymbol{e}}_i$ is defined as in (3.18). The second equality follows from the fact that when $J_1 = 2$ or $J_1 = 3$, $\bar{\boldsymbol{e}}_i$ always has a zero component, so $\bar{e}_i^1\bar{e}_i^2 = 0$.

$$\begin{aligned}
d_{\text{min,asy}}^2(E_1) &= \min_{\bar{\boldsymbol{e}}_i \in E_1} d_{\mathrm{r}}^2(\bar{\boldsymbol{e}}_i \in E_1) \\
&= \begin{cases} h_0^2\Delta_1^2 + 8\Delta\epsilon^2 h_0^2 & J_1 = 3 \\ h_0^2\Delta_2^2 + 8\Delta\epsilon^2 h_0^2 & J_1 = 2. \end{cases} \qquad (3.33)
\end{aligned}$$

Compared to the symmetric case, $d_{\text{min,asy}}^2(E_1)$ is increased both for $J_1 = 2$ and $J_1 = 3$.

For a longer error event $\bar{\boldsymbol{e}}(D)$, the induced squared distance is

$$\begin{aligned}
d^2(\bar{\boldsymbol{e}}(D)) &= \frac{(1+\epsilon)^2}{2}\|\bar{e}^1(D)h(D) + \frac{\Delta\epsilon}{1+\epsilon}\bar{e}^2(D)h(D)\|^2 \\
&\quad + \frac{(1-\epsilon)^2}{2}\|\frac{\Delta\epsilon}{\epsilon-1}\bar{e}^1(D)h(D) + \bar{e}^2(D)h(D)\|^2. \qquad (3.34)
\end{aligned}$$

Then the error state diagram and the error event search algorithm introduced in Section 3.5 can be applied to the asymmetric case, with the only modification being that the edge labels are calculated according to (3.34). Also notice that the zero cycles given in Example 4 and Example 5 remain the same in the asymmetric channel.

We search for $d_{\text{min}}^2(E^r)$ at two extreme values of ϵ and various offsets $\Delta\epsilon$ on the EPR4 channel. The results are listed in Tables 3.7. In each case, $\Delta\epsilon$ could take values from $\{0, 0.05, 0.1\}$. For comparison, we also give the minimum distance parameter of the ML detector, denoted as d_{min}^2, in each corresponding scenario. We find that $d_{\text{min}}^2(E^r)$ does not change much from the

Table 3.7: $d^2_{\text{min}}(E^r)$ for asymmetric 2H2T EPR4 channel under various $\Delta\epsilon$ where (a) $\epsilon = 0.1$, and (b) $\epsilon = 0.4$.

(a) $\epsilon = 0.1$

	$\Delta\epsilon = 0$	$\Delta\epsilon = 0.05$	$\Delta\epsilon = 0.1$
	$d^2_{\text{min}} = 16.16$	$d^2_{\text{min}} = 16.04$	$d^2_{\text{min}} = 16$
RSSE$[4,4,3]$	22.64	22.70	22.88
RSSE$[4,4,2]$	19.44	19.50	19.68
RSSE$[4,4,1]$	12.12	12.03	12.00
RSSE$[4,3,3]$	16.16	16.20	16.32
RSSE$[4,3,2]$	16.16	16.20	16.32
RSSE$[3,3,3]$	9.68	9.70	9.76

(b) $\epsilon = 0.4$

	$\Delta\epsilon = 0$	$\Delta\epsilon = 0.05$	$\Delta\epsilon = 0.1$
	$d^2_{\text{min}} = 11.52$	$d^2_{\text{min}} = 11.60$	$d^2_{\text{min}} = 11.84$
RSSE$[4,4,3]$	21.44	21.50	21.68
RSSE$[4,4,2]$	8.64	8.7	8.88
RSSE$[4,4,1]$	8.64	8.7	8.88
RSSE$[4,3,3]$	18.56	18.60	18.72
RSSE$[4,3,2]$	8.64	8.7	8.88
RSSE$[3,3,3]$	12.16	12.20	12.32

symmetric case ($\Delta\epsilon = 0$). In addition, some trellis configurations tend to have increased $d^2_{\text{min}}(E^r)$ under severe asymmetry, while some do not. We see that the performance of a configuration is closely related to the distance parameters of the length-1 error events, which provides an approach to design the set partition tree for other MHMT channel models. The conclusion is that the proposed RSSE algorithm is applicable to the asymmetric channel.

3.7 3H3T System

The ITI interference matrix of the 3H3T system is given by

$$A_3 = \begin{bmatrix} 1 & \epsilon & 0 \\ \epsilon & 1 & \epsilon \\ 0 & \epsilon & 1 \end{bmatrix}.$$

The WSSJD transformation decomposes the 3H3T system into 3 parallel channels. Recall

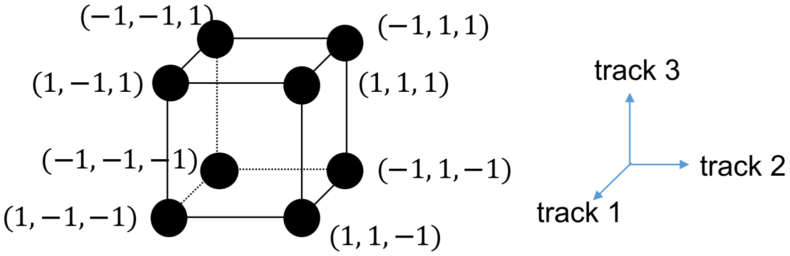


Figure 3.11: The input constellation of 3H3T system. The transposed vector beside each node represents the input symbol $\boldsymbol{x}_i = [x_i^1, x_i^2, x_i^3]^\top$. The corresponding dimensions are shown on the right.

that the eigen-decomposition of A_3 is $A_3 = V_3 \Lambda_3 V_3^\top$, where

$$\Lambda_3 = \begin{bmatrix} 1+\sqrt{2}\epsilon & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1-\sqrt{2}\epsilon \end{bmatrix}, \quad V_3 = \begin{bmatrix} \frac{1}{2} & \frac{\sqrt{2}}{2} & \frac{1}{2} \\ \frac{\sqrt{2}}{2} & 0 & -\frac{\sqrt{2}}{2} \\ \frac{1}{2} & -\frac{\sqrt{2}}{2} & \frac{1}{2} \end{bmatrix}.$$

The decomposed system is described by

$$\bar{\boldsymbol{r}}_i = \bar{\boldsymbol{y}}_i + \bar{\boldsymbol{\omega}}_i, \tag{3.35}$$

where

$$\bar{\boldsymbol{r}}_i = \Lambda_3^{-1} V_3^\top \boldsymbol{r}_i, \quad \bar{\boldsymbol{\omega}}_i = \Lambda_3^{-1} V_3^\top \boldsymbol{\omega}_i \tag{3.36}$$

are the transformed channel outputs and noises. The components of vector $\bar{\boldsymbol{y}}_i$ are given by $\bar{y}_i^j = \sum_{k=0}^{\nu} h_k z_{i-k}^j$, for $j = 1, ..., 3$, where $\boldsymbol{z}_i$ is the transformed channel input vector $\boldsymbol{z}_i = V_3^\top \boldsymbol{x}_i$. Since V_3 is independent of ϵ, the joint trellis constructed according to the combination of $\boldsymbol{z}_i$ is deterministic. Let $\boldsymbol{e}(D) = [e^1(D), e^2(D), e^3(D)]^\top$ be an error event of the system, where $e^j(D)$ is the error event on track j, $e^j(D) = x^j(D) - \hat{x}^j(D)$. An error symbol at time slot i is denoted as $\boldsymbol{e}_i = \boldsymbol{x}_i - \hat{\boldsymbol{x}}_i$. Then for the transformed 3H3T system,

$$\bar{\boldsymbol{e}}_i = V_3^\top \boldsymbol{e}_i = V_3^\top \bar{\boldsymbol{x}}_i - V_3^\top \hat{\boldsymbol{x}}_i, \tag{3.37}$$

and the distance associated with error event $\boldsymbol{e}(D)$ is

$$d^2\left(\boldsymbol{e}(D)\right) = \sum_{j=1}^{3} \lambda_j^2 \|\bar{e}^j(D)h(D)\|^2, \tag{3.38}$$

where $\lambda_1 = 1 + \sqrt{2}\epsilon$, $\lambda_2 = 1$ and $\lambda_3 = 1 - \sqrt{2}\epsilon$ are the eigenvalues on the diagonal of Λ_3.

To construct the reduced-state trellis, we first need to evaluate the distance between input symbols. Recall that the ESPD is proportional to the distance associated with length-1 error events, with the scaling factor h_0^2. Therefore, for symbols $\boldsymbol{x}_i$ and $\hat{\boldsymbol{x}}_i$ with difference $\boldsymbol{e}_i = \boldsymbol{x}_i - \hat{\boldsymbol{x}}_i$, their ESPD is calculated from

$$d(\boldsymbol{e}_i) = \sum_{j=1}^{3} (\lambda_j \bar{e}_i^j)^2. \tag{3.39}$$

Table 3.8 lists the ESPDs for the symbol pair differences. The table is simplified by symmetry considerations, i.e. $-\boldsymbol{e}_i$ produces the same distance as $\boldsymbol{e}_i$. The distances are functions of ϵ. They display different monotonicity behavior over the range $\epsilon \in [0, 0.5]$. For instance, the error symbol 6 has the same distance for all the values of ϵ. The distances of error symbols with index 4 and 7 decrease as ϵ increases, while for other error symbols the distance functions are increasing functions. Therefore, at different ITI levels, the dominant ESPD is different, which should be taken into account when designing the set partition tree.

In Fig. 3.12 we propose two set partition trees optimized for low or high ITI levels. For the low ITI case, the single track error symbols, corresponding to the error symbols 1 and 2 in Table 3.8, have smaller ESPDs. They are first removed from the level 1 to level 2 partitions in the type-1 construction, shown in Fig. 3.12(a). To further increase the intrasubset ESPD, the error symbols 4 and 5 are also removed in the level 3 partition, and error symbol 6 is avoided on level 4. Following a similar design rule, the type-2 set partition tree in Fig. 3.12(b) is constructed to handle the case of high ITI. It is necessary to first remove error symbols 4 and 7 from level 1 to level 2 since their induced distance is much smaller than others. Then the single track error symbols and the error symbol 6 are also avoided on level 3 and level 4, respectively. Notice that

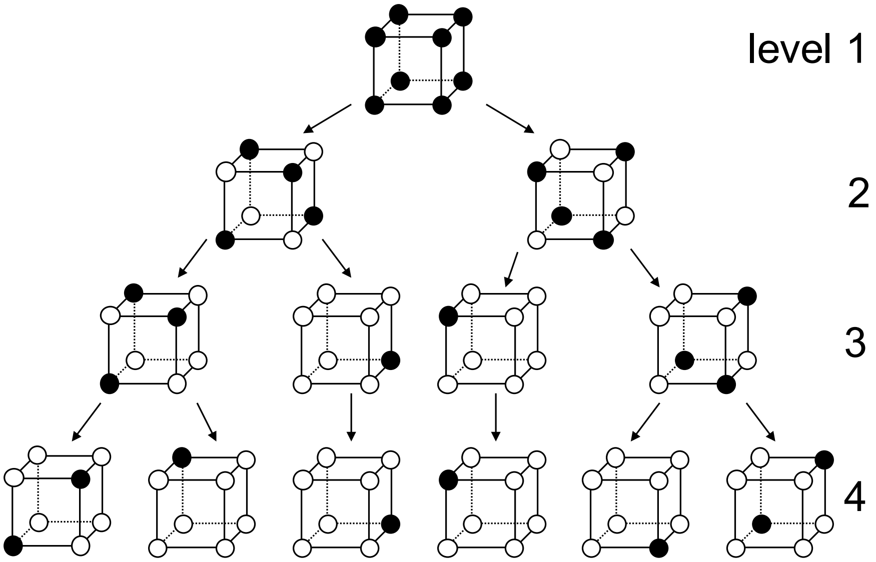
Table 3.8: ESPDs of 3H3T system. $\min d^2$ is the minimum value of $d^2(\boldsymbol{e}_i)$ achieved at ϵ^*.

index	$\boldsymbol{e}_i$	$d^2(\boldsymbol{e}_i)$	$\min d^2$	ϵ^*
1	$[2,0,0]$ $[0,0,2]$	$4+4\epsilon^2$	4	0
2	$[0,2,0]$	$4+8\epsilon^2$	4	0
3	$[2,2,0]$ $[0,2,2]$	$8+16\epsilon+12\epsilon^2$	8	0
4	$[2,-2,0]$ $[0,-2,2]$	$8-16\epsilon+12\epsilon^2$	3	0.5
5	$[2,0,2]$	$8+16\epsilon^2$	8	0
6	$[2,0,-2]$	8	8	$[0,0.5]$
7	$[2,-2,2]$	$12-32\epsilon+24\epsilon^2$	2	0.5
8	$[2,-2,-2]$ $[-2,-2,2]$	$12+8\epsilon^2$	12	0
9	$[2,2,2]$	$12+32\epsilon+24\epsilon^2$	12	0

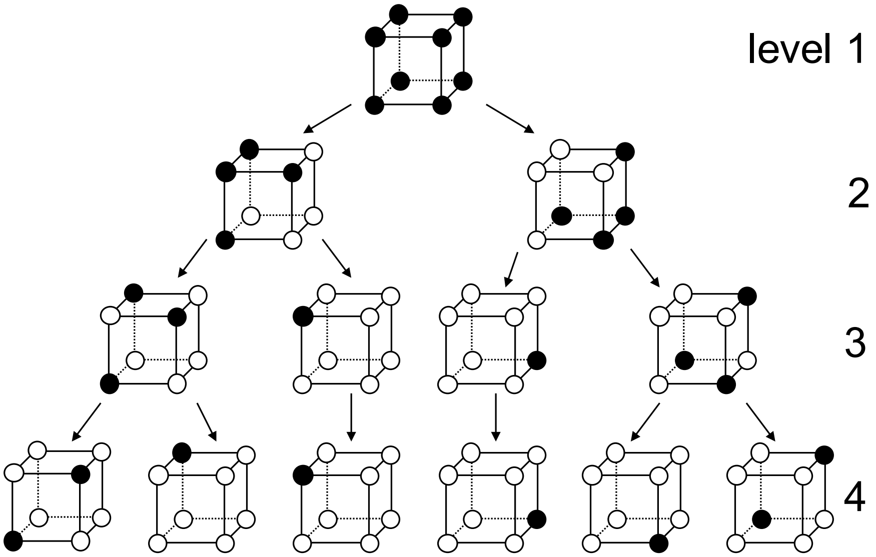
the type-1 and type-2 set partition trees differ only on level 2. Therefore, the subset trellis with $J_k \neq 2$ for all $k = 1, ..., \nu$ will yield the same performance, no matter which set partition tree is used.

In Figs. 3.13 and Figs. 3.14 we plot the simulation results for RSSE on the EPR4 channel and a minimum phase channel, respectively. Two extreme cases are considered, corresponding to a relatively low ITI level, $\epsilon = 0.1$, and a high ITI level, $\epsilon = 0.4$. We construct several subset trellises based on the type-1 set partition tree for $\epsilon = 0.1$ and the type-2 tree for $\epsilon = 0.4$. It can be observed from Fig. 3.13(a) that the type-1 RSSE $[8, 8, 2]$ configuration and the type- 1 RSSE $[8, 6, 2]$ configuration have near-ML performance. For comparison, we also plot the performance curve for the type-2 RSSE$[8, 6, 2]$ trellis, which is a subset trellis constructed using the type-2 set partition tree. This trellis suffers from significant performance loss although it has the same configuration as type-1 RSSE $[8, 6, 2]$. In Fig. 3.13(b), we see that type-2 RSSE $[8, 6, 4]$ can essentially achieve ML performance; the results for type-2 RSSE$[8, 6, 2]$ and type-1 RSSE$[8, 6, 2]$ are also plotted for comparison purposes.

For the minimum phase channel, whose performance results are shown in Fig. 3.14,

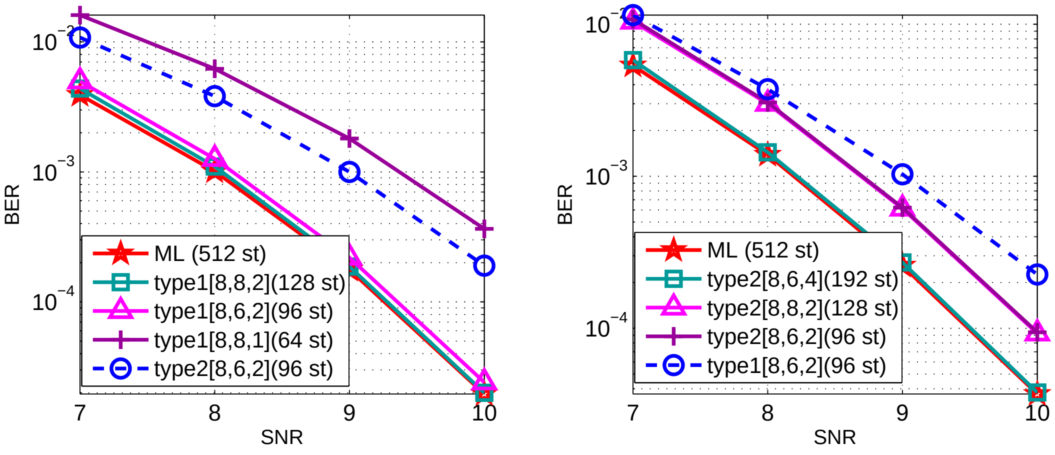


(a) type 1. better performance at low ITI



(b) type 2. better performance at high ITI

Figure 3.12: Set partition trees designed for 3H3T system. Both of the trees have 5 level partitions. To save space, the level 5, where each symbol itself is a subset, is not shown on the pictures.



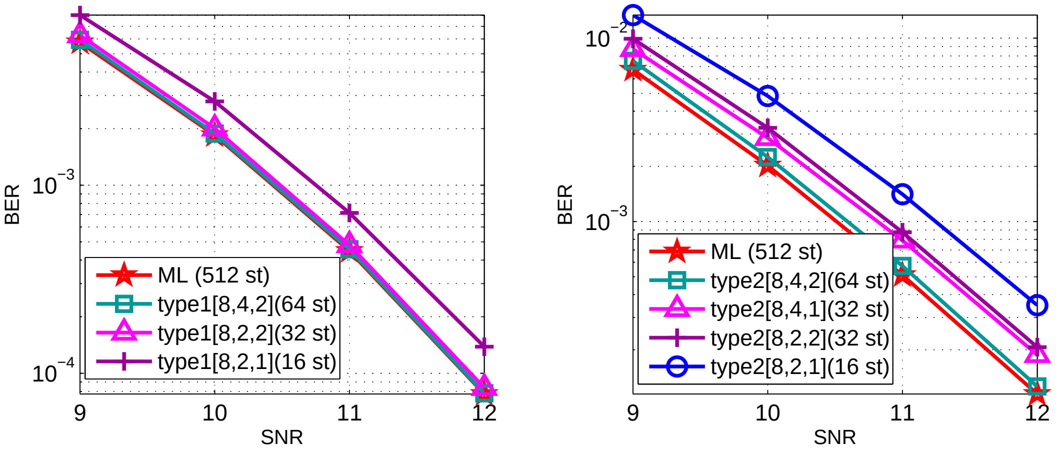
(a) $\epsilon = 0.1$ (b) $\epsilon = 0.4$

Figure 3.13: Simulation results for 3H3T system with EPR4 channel polynomial. The prefix "type1" and "type2" indicate if the subset trellis is based on the type 1 or type 2 set partition tree, respectively.

the required computational complexity is further reduced. From Fig. 3.14(a), we see that that the type-1 RSSE $[8, 2, 2]$ trellis with 32 states has performance nearly equal to that of the ML detector., which requires 512 states. In Fig. 3.14(b), we see that the type-1 RSSE $[8, 4, 2]$ trellis with 64 states provides essentially ML performance.

3.8 Conclusion

Due to its capability of combating ITI, MHMT detection is expected to play an important role in next generation magnetic recording. The conventional ML detector, however, suffers from high computational complexity. In this work we address this problem by applying RSSE techniques with properly designed set partition trees. In particular, we define an alternative distance measure on the input constellation, based on which we propose a three-level set partition tree for the 2H2T model. The BER comparison shows that RSSE can achieve near optimal performance while significantly reducing the number of trellis states. Error event analysis is used to explain the performance variations observed for different trellises under various conditions. We also investigate the performance of RSSE on an asymmetric 2H2T system because of the practical



(a) $\epsilon = 0.1$

(b) $\epsilon = 0.4$

Figure 3.14: Simulation results for 3H3T system with minimum phase channel $h(D) = 1 + 1.6D + 1.1D^2 + 0.4D^3$.

relevance of the model. For the 3H3T model, since the effective distances between input symbols show different monotonicity behavior as ϵ changes, we specifically design two set partition trees, where one is good for low ITI and the other is suitable for high ITI. Our work shows that the set partition tree plays a key role in applying RSSE to these channels. If the set partition tree can be properly designed, then the RSSE algorithm has the potential to be effectively applied to more general MHMT channels. This is a direction for future research.

Acknowledgements

This chapter contains material from the papers: Bing Fan, Hemant K. Thapar and Paul H. Siegel, “Multihead multitrack detection with reduced-state sequence estimation in shingled magnetic recording,” published in *IEEE Trans. Magn.*, vol. 51, no. 11, pp. 1-4, Nov. 2015, and Bing Fan, Hemant K. Thapar and Paul H. Siegel, “Multihead multitrack detection for next generation magnetic recording, part II: complexity reduction - algorithms and performance analysis,” published in *IEEE Trans. Commun.*, vol. 65, no. 4, pp. 1649-1661, April 2017. The dissertation author was the primary investigator and author of these papers, and co-authors have approved the use of the material for this dissertation.

Chapter 4

Generalized Weighted Sum Subtract Joint Detector

In this chapter we generalize WSSJD to MHMT systems whose interference matrices satisfy *the WSSJD property*. The implementation details are given for a symmetric 3H2T system, as an example. Other interference matrices where WSSJD is applicable are also presented. We also propose a new set partitioning tree for the 3H2T system with high ITI, and compare its performance with the set partitioning tree proposed in Chapter 3.

4.1 Notations

A length-L data sequence is represented by a polynomial $x(D) = \sum_{k=0}^{L} x_k D^k$. The inner product of two data sequences of length L is calculated by

$$\langle x(D), y(D) \rangle = \sum_{k=0}^{L} x_k y_k, \tag{4.1}$$

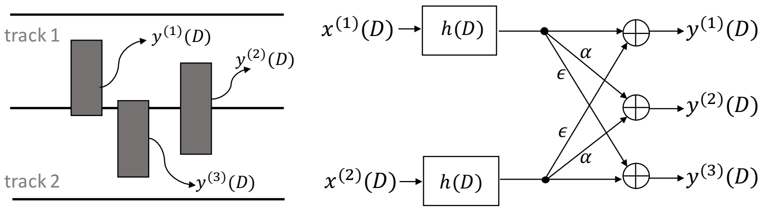


Figure 4.1: Three head/two track model with ISI and ITI.

and the norm of a data sequence is

$$\|x(D)\| = \sqrt{\langle x(D), x(D)\rangle} = \sqrt{\sum_{k=0}^{L} x_k^2}. \tag{4.2}$$

We use a capitalized bold symbol to represent a column vector of data sequences, e.g., $\boldsymbol{X}(D) = [x^{(1)}(D), \cdots, x^{(n)}(D)]^\top$. The coefficients of $\boldsymbol{X}(D)$ at time k form a column vector $\boldsymbol{x}_k = [x_k^{(1)}, \cdots, x_k^{(n)}]^\top$. The multiplication of $\boldsymbol{X}(D)$ with a data sequence $h(D)$ is equivalent to element-wise convolutions, i.e., $\boldsymbol{X}(D)h(D) = [x^{(1)}(D)h(D), \cdots, x^{(n)}(D)h(D)]^\top$.

Matrices are denoted by boldface symbols, with subscripts indicating the matrix size. For instance, $\mathbf{A}_{m,n}$ is an $m \times n$ matrix. We use $\mathbf{A}_n$ for short when $m = n$. A matrix element is referred by a lower-case symbol, e.g., a_{ij} on row i and column j in $\mathbf{A}_{m,n}$. The $n \times n$ identity matrix is represented by $\mathbf{I}_n$. The transpose of $\mathbf{A}$ is denoted by $\mathbf{A}^\top$. The i-th power of $\mathbf{A}$ is written as $\mathbf{A}^i$, and $\mathbf{A}^0 = \mathbf{I}$.

4.2 Three Head/Two Track Channel Model

We consider a 3H2T system with intersymbol interference (ISI) along the down-track direction and ITI in the cross-track direction, as shown in Fig. 4.1. The channel inputs, $\boldsymbol{X}(D) = [x^{(1)}(D), x^{(2)}(D)]^\top$, are data sequences stored on track 1 and track 2. We assume that both $x_k^{(1)}$ and $x_k^{(2)}$ are i.i.d and randomly chosen from $\{+1, -1\}$ with equal probability. We also assume there is no phase offset between $x^{(1)}(D)$ and $x^{(2)}(D)$.

Three heads are used to read back data simultaneously. Head 1 and head 3 are symmetrically placed around the boundary between two tracks, with an offset towards track 1 and 2, respectively. With this placement, head 1 senses a majority signal from track 1 (the target track) and a small interference from track 2 (the side track), while the case is reversed for head 3. Head 2 is placed symmetrically over the boundary, and senses equal level signals from two tracks. We use this head ordering since the resulting interference matrix shows a desired symmetric property. Let $\boldsymbol{R}(D) = [r^{(1)}(D), r^{(2)}(D), r^{(3)}(D)]^\top$ be the vector of discretized read back signals from head 1, 2, and 3, respectively. The channel model can be compactly written as

$$\boldsymbol{R}(D) = \mathbf{A}_{3,2}\boldsymbol{X}(D)h(D) + \boldsymbol{\Omega}(D), \tag{4.3}$$

where the electronic noise at the three heads, $\boldsymbol{\Omega}(D) = [\omega^{(1)}(D), \omega^{(2)}(D), \omega^{(3)}(D)]^\top$, is i.i.d and follows the Gaussian distribution, $\omega_k^{(i)} \sim \mathcal{N}(0, \sigma^2)$, $i = 1, 2, 3$. In this model, we assume that the data tracks are ideal partial response channels, with the same channel polynomial $h(D) = \sum_0^\nu h_k D^k$. The interference matrix $\mathbf{A}_{3,2}$ characterizes the ITI sensed at the three heads. Based on the head alignment, we assume $\mathbf{A}_{3,2}$ has the form

$$\mathbf{A}_{3,2} = \begin{bmatrix} 1 & \epsilon \\ \alpha & \alpha \\ \epsilon & 1 \end{bmatrix}, \tag{4.4}$$

where ϵ and α are variables whose values may change for various reasons. For example, the values of ϵ and α are expected to be very different at inner tracks and outer tracks, since the track pitch changes. Moreover, in some cases the accurate values of these variables may be unknown at the receiver that a parameter estimation process is needed. Due to these uncertainties in real drives, we use variables ϵ and α to represent the ITI levels. In our work ϵ is restricted to the range of $[0, 0.5)$, and $\alpha > 0$. A matrix whose elements do not contain variables is called **a deterministic matrix**.

At the receiver, the ML estimates of the input sequences $\boldsymbol{X}(D)$ are obtained by passing the outputs $\boldsymbol{R}(D)$ through a 2D joint Viterbi detector. Let $\boldsymbol{X}(D)$ and $\hat{\boldsymbol{X}}(D)$ denote the correct inputs and the estimated inputs, respectively. Their difference $\boldsymbol{E}(D) = \boldsymbol{X}(D) - \hat{\boldsymbol{X}}(D)$, if not zero, is called an error event. It is well known that the error event probability of a Viterbi detector can be approximated by $K \cdot Q(d_{\min}/2\sigma)$, where K is a constant related to the frequency of the dominant error events, σ^2 is the noise variance, and the Q-function is the tail probability of the standard Gaussian distribution. The parameter $d_{\min}$ is the minimum distance taken over all possible error events. For the 3H2T channel given by (4.3) and (4.4), we find

$$d^2_{\min,\mathbf{A}_{3,2}} = \begin{cases} (1+\alpha^2+\epsilon^2)d_0^2, & \text{if } \alpha^2 > 1-4\epsilon+\epsilon^2 \\ 2(1-\epsilon)^2 d_0^2, & \text{otherwise} \end{cases} \tag{4.5}$$

where d_0^2 is the squared minimum distance of the ISI channel $h(D)$. The derivation of (4.5) is presented in Appendix. The case of $\alpha = 0$ is equivalent to the 2H2T system with interference matrix

$$\mathbf{A}_2 = \begin{bmatrix} 1 & \epsilon \\ \epsilon & 1 \end{bmatrix}. \tag{4.6}$$

The incorporation of ITI parameters presents a barrier to the realization of an ML detector. To implement the 2D joint Viterbi detector, a trellis that simultaneously tracks $x^{(1)}(D)$ and $x^{(2)}(D)$ is constructed and stored in memory. Notice that the output labels of trellis branches, given by $\mathcal{L}_{\text{out}} = \boldsymbol{Y}_k$, where

$$\boldsymbol{Y}(D) = \mathbf{A}_{3,2}\boldsymbol{X}(D)h(D), \tag{4.7}$$

require the knowledge of α and ϵ. However, these two parameters are generally unknown to the receiver, and subject to change. Storing trellises for different parameters would consume a large memory, while recomputing output labels during the detection process could incur additional delay. Using static estimates could avoid these issues, but results in performance loss due to sub-optimality.

4.3 WSSJD for 3H2T System

We generalize WSSJD to the 3H2T system, as an alternative way to implement the ML detector. We show that by means of channel decomposition, the 3H2T system can be transformed to two parallel subchannels, whose joint trellis is free from ITI parameters. Then, a gain loop structure could be deployed to adaptively track the ITI parameters. Complexity reduction techniques will also be discussed. For convenience, the conventional ML detector directly designed for system (4.3) is referred to as "the ML detector", however, we emphasize that WSSJD also gives the ML solutions.

4.3.1 Channel Decomposition

The interference matrix $\mathbf{A}_{3,2}$ can be factored as

$$\mathbf{A}_{3,2} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{V}^{\top}, \tag{4.8}$$

where

$$\mathbf{V} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \tag{4.9}$$

is a 2×2 constant matrix. The 3×2 matrix $\mathbf{U}$ has the form

$$\mathbf{U} = [\lambda_1^{-1}\boldsymbol{u}_1, \lambda_2^{-1}\boldsymbol{u}_2, \boldsymbol{u}_3], \tag{4.10}$$

where

$$\boldsymbol{u}_1 = \left[\frac{1+\epsilon}{2}, \alpha, \frac{1+\epsilon}{2}\right]^{\top}, \quad \boldsymbol{u}_2 = \left[\frac{1-\epsilon}{2}, 0, \frac{\epsilon-1}{2}\right]^{\top}, \tag{4.11}$$

and $\lambda_1 = \|\boldsymbol{u}_1\|$, $\lambda_2 = \|\boldsymbol{u}_2\|$ are the normalization factors. The third column vector $\boldsymbol{u}_3$ has the unit length, and is orthogonal to both $\boldsymbol{u}_1$ and $\boldsymbol{u}_2$. Therefore, $\mathbf{U}$ is a unitary matrix, i.e.,

$\mathbf{U}^\top\mathbf{U} = \mathbf{U}\mathbf{U}^\top = \mathbf{I}_3$. Matrix $\mathbf{\Lambda}$ is a 3×2 diagonal matrix whose diagonal elements are λ_1 and λ_2,

$$\mathbf{\Lambda} = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \\ 0 & 0 \end{bmatrix}. \tag{4.12}$$

Notice that the factorization (4.8) is equivalent to the singular value decomposition (SVD) of $\mathbf{A}_{3,2}$, up to a scaling factor. The pseudoinverse of $\mathbf{\Lambda}$ is denoted as

$$\mathbf{\Lambda}^\dagger = \begin{bmatrix} \lambda_1^{-1} & 0 & 0 \\ 0 & \lambda_2^{-1} & 0 \end{bmatrix}. \tag{4.13}$$

To decompose the channel, we substitute (4.8) for the matrix $\mathbf{A}_{3,2}$,

$$\boldsymbol{R}(D) = \mathbf{U}\mathbf{\Lambda}\mathbf{V}^\top\boldsymbol{X}(D)h(D) + \mathbf{\Omega}(D), \tag{4.14}$$

Since $\mathbf{U}$ is unitary and $\mathbf{\Lambda}^\dagger\mathbf{\Lambda} = \mathbf{I}_2$, we can multiply both sides by $\mathbf{\Lambda}^\dagger\mathbf{U}^\top$ to obtain

$$\mathbf{\Lambda}^\dagger\mathbf{U}^\top\boldsymbol{R}(D) = \mathbf{V}^\top\boldsymbol{X}(D)h(D) + \mathbf{\Lambda}^\dagger\mathbf{U}^\top\mathbf{\Omega}(D). \tag{4.15}$$

Equation (4.15) is *the transformed 3H2T channel*. Define

$$\boldsymbol{Z}(D) = \mathbf{V}^\top\boldsymbol{X}(D) = [x^{(1)}(D) + x^{(2)}(D), x^{(1)}(D) - x^{(2)}(D)]^\top = [z^{(1)}(D), z^{(2)}(D)]^\top \tag{4.16}$$

to be the new channel inputs. Notice that $z_k^{(i)} \in \{-2, 0, +2\}$ is ternary. The new channel outputs are

$$\bar{\boldsymbol{R}}(D) = \mathbf{\Lambda}^\dagger\mathbf{U}^\top\boldsymbol{R}(D) = [\lambda_1^{-2}\boldsymbol{u}_1^\top\boldsymbol{R}(D),\ \lambda_2^{-2}\boldsymbol{u}_2^\top\boldsymbol{R}(D)]^\top = [\bar{r}^{(1)}(D), \bar{r}^{(2)}(D)]^\top. \tag{4.17}$$

The transformed noise components

$$\bar{\mathbf{\Omega}}(D) = \mathbf{\Lambda}^{\dagger}\mathbf{U}^{\top}\mathbf{\Omega}(D) \tag{4.18}$$

satisfy a zero-mean multivariate Gaussian distribution, with covariance matrix

$$E[\bar{\boldsymbol{\omega}}_k \bar{\boldsymbol{\omega}}_k^{\top}] = E[\mathbf{\Lambda}^{\dagger}\mathbf{U}^{\top}\boldsymbol{\omega}_k \boldsymbol{\omega}_k^{\top}\mathbf{U}(\mathbf{\Lambda}^{\dagger})^{\top}] = \begin{bmatrix} \sigma_1^2 & 0 \\ 0 & \sigma_2^2 \end{bmatrix}, \tag{4.19}$$

where $\sigma_1^2 = \sigma^2\lambda_1^{-2}$ and $\sigma_2^2 = \sigma^2\lambda_2^{-2}$.

4.3.2 WSSJD

After channel decomposition, the resulting system (4.15) consists of two parallel subchannels,

$$\bar{r}^{(1)}(D) = z^{(1)}(D)h(D) + \bar{\omega}^{(1)}(D), \tag{4.20}$$

$$\bar{r}^{(2)}(D) = z^{(2)}(D)h(D) + \bar{\omega}^{(2)}(D). \tag{4.21}$$

We call (4.20) the *sum channel*, and (4.21) the *subtract channel*, since $z^{(1)}(D)$ and $z^{(2)}(D)$ are basically the sum and the difference of the original channel inputs $x^{(1)}(D)$ and $x^{(2)}(D)$.

WSSJD searches for the joint ML solution to (4.20) and (4.21),

$$\begin{aligned} \boldsymbol{Z}^*(D) &= \arg\max_{\boldsymbol{Z}(D)} \log \Pr(\bar{\boldsymbol{R}}(D)|\boldsymbol{Z}(D)) && (4.22) \\ &= \arg\max_{\boldsymbol{Z}(D)} \log \Pr(\bar{r}^{(1)}(D)|z^{(1)}(D)) + \log \Pr(\bar{r}^{(2)}(D)|z^{(2)}(D)) && (4.23) \\ &= \arg\min \lambda_1^2 \|\bar{r}^{(1)}(D) - z^{(1)}(D)h(D)\|^2 + \lambda_2^2 \|\bar{r}^{(2)}(D) - z^{(2)}(D)h(D)\|^2. && (4.24) \end{aligned}$$

Here, (4.23) comes from the fact that the noises on the two subchannels are independent. The scale factors in (4.24), λ_1^2 and λ_2^2, are the *channel weights* due to the effect of unequal noise powers in two subchannels.

The WSSJD trellis constructed to decode $\boldsymbol{Z}(D)$ has the same number of states as the ML detector. In contrast to the ML detector whose trellis labels vary with ITI, the WSSJD trellis is deterministic. Namely, the trellis does not require knowledge of α and ϵ, remaining the same even though α and ϵ may change. This property would be preferable in hardware realization. Another difference from the ML detector is that, during detection, the branch metrics must be scaled by λ_1^2 and λ_2^2 to retain optimality. This additional step is easy to realize in practice.

Assume $\bar{\mathbf{E}}(D) = [\bar{e}^{(1)}(D), \bar{e}^{(2)}(D)]$ are the error events on the sum and subtract channels, respectively. The distance function is calculated by

$$d^2_{\text{WSSJD}}(\bar{\mathbf{E}}(D)) = \lambda_1^2 \|\bar{e}^{(1)}(D)h(D)\|^2 + \lambda_2^2 \|\bar{e}^{(2)}(D)h(D)\|^2. \tag{4.25}$$

The squared minimum distance

$$d^2_{\min} = \min_{\bar{\mathbf{E}}(D)} d^2_{\text{WSSJD}}(\bar{\mathbf{E}}(D)) \tag{4.26}$$

has the same closed form as (4.5).

4.3.3 Complexity Reduction

Reduced-state sequence estimation (RSSE), proposed in [14], allows one to make a tradeoff between performance and complexity in trellis-based detection and decoding. Instead of relying on the complete ML trellis, RSSE uses *subset trellis*, in which the *subset states* are disjoint groups of the original ML states. Suppose $J(k)$, $k = 1, ..., \nu$, are set partitionings of the input symbols. A subset state is represented by

$$\boldsymbol{s}_i = [a_{i-1}(1), a_{i-2}(2), ..., a_{i-\nu}(\nu)], \tag{4.27}$$

where $a_{i-k}(k)$ is a subset index in $J(k)$. The *configuration* of the resulting subset trellis is a vector $[|J_1|, |J_2|, ..., |J_\nu|]$, and the number of subset states is $\prod_{k=1}^{\nu} |J_k|$.

To obtain a valid subset trellis, the sequence of set partitionings $\{J_k\}_{k=1}^{\nu}$ is restricted to the condition that J_k is a further partition of the subsets in J_{k+1} for all k. A *set partitioning tree* can be used to assist the set partitioning selection process. In Figs. 4.3, we present two examples of the set partitioning tree. The set partitionings $\{J_k\}_{k=1}^{\nu}$ are chosen from four levels of the tree, with the restriction that J_{k+1} should not be a lower level than J_k.

A general design rule for a good set partitioning tree is to maximize the minimum intra-subset distance at each level. For the transformed 3H2T system, Since the two subchannels have different noise powers, we use *effective symbol pair distance* (ESPD), instead of Euclidean distance, to evaluate distances between input symbols. The ESPD between two symbols $\boldsymbol{z}$ and $\hat{\boldsymbol{z}}$ is defined by

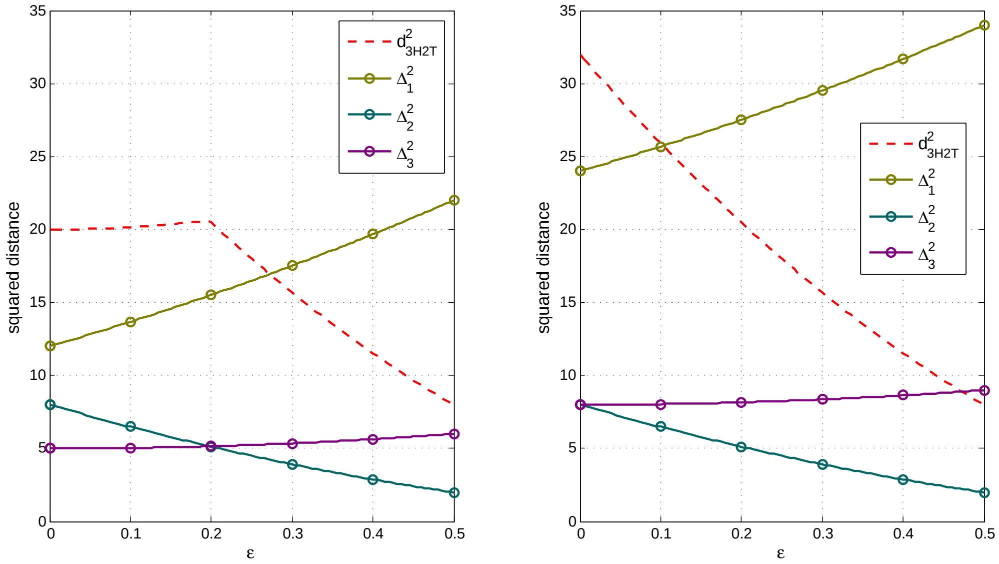
$$d^2_{\text{ESPD}}(\boldsymbol{z}, \hat{\boldsymbol{z}}) = \lambda_1^2 \cdot (z^{(1)} - \hat{z}^{(1)})^2 + \lambda_2^2 \cdot (z^{(2)} - \hat{z}^{(2)})^2. \tag{4.28}$$

Table 4.1 summarizes ESPDs for all pairs of input symbols. Compared to the ESPDs in 2H2T system (Table I in [18]), the additional head in 3H2T only affects Δ_1^2 and Δ_3^2. As shown in Fig. 4.2(a), for $\alpha = 0.5$, the curves of Δ_2^2 and Δ_3^2 cross near $\epsilon = 0.2$. For $\alpha = 1$ in Fig. 4.2(b), $\Delta_3^2 > \Delta_2^2$ for all $\epsilon > 0$. We also plot the squared minimum distance of the corresponding 3H2T system (with $h(D) = 1 + D - D^2 - D^3$) for comparison.

Table 4.1: ESPDs between input symbols of decomposed 3H2T system.

$(\boldsymbol{z}, \hat{\boldsymbol{z}})$	$d^2_{\text{ESPD}}(\boldsymbol{z}, \hat{\boldsymbol{z}})$
$([+2,0],[-2,0])$	$\Delta_1^2 = 8(1+\epsilon)^2 + 16\alpha^2$
$([0,+2],[0,-2])$	$\Delta_2^2 = 8(1-\epsilon)^2$
$([+2,0],[0,+2])$ $([+2,0],[0,-2])$ $([-2,0],[0,+2])$ $([-2,0],[0,-2])$	$\Delta_3^2 = 4(1+\epsilon^2) + 4\alpha^2$

We want to construct a set partitioning tree where the minimum intra-subset ESPD increases from the top level to the bottom. Since the ordering of Δ_1^2, Δ_2^2 and Δ_3^2 is not fixed, we propose two possible set partitioning trees that can be used in different ITI environments. The constructions are shown in Figs. 4.3. Tree-A (Fig. 4.3(a)) was originally proposed in [18] for the 2H2T channel, which is equivalent to the 3H2T system with $\alpha = 0$. Under the assumption that



(a) $\alpha = 0.5$ (b) $\alpha = 1$

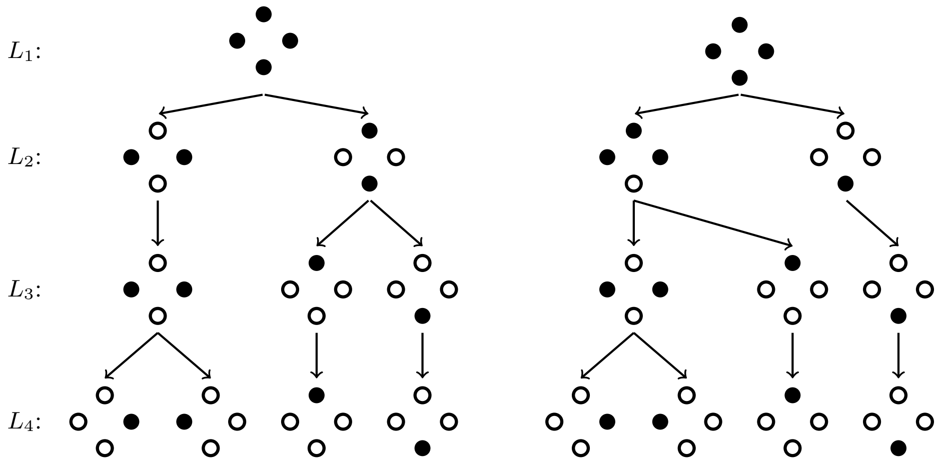
Figure 4.2: ESPDs for 3H2T system with (a) $\alpha = 0.5$, and (b) $\alpha = 1$.

Table 4.2: Error Event Analysis for different subset trellis configurations in Fig. 4.4(a).

	$\min d_{\mathrm{r}}^2(\bar{e}^{(1)}, \bar{e}^{(2)})$	$\bar{e}^{*(1)}, \bar{e}^{*(2)}$
ML	15.68	$[0,0,0,0], [4,-4,0,0]$
RSSE-A$[4,2]$	11.76	$[0,0,0], [4,-4,0]$
RSSE-B$[4,2]$	15.68	$[0,0,0,0], [4,-4,0,0]$
RSSE-A$[3,2]$	11.76	$[0,0,0], [4,-4,0]$
RSSE-B$[3,2]$	12.28	$[-2,4], [-2,0]$

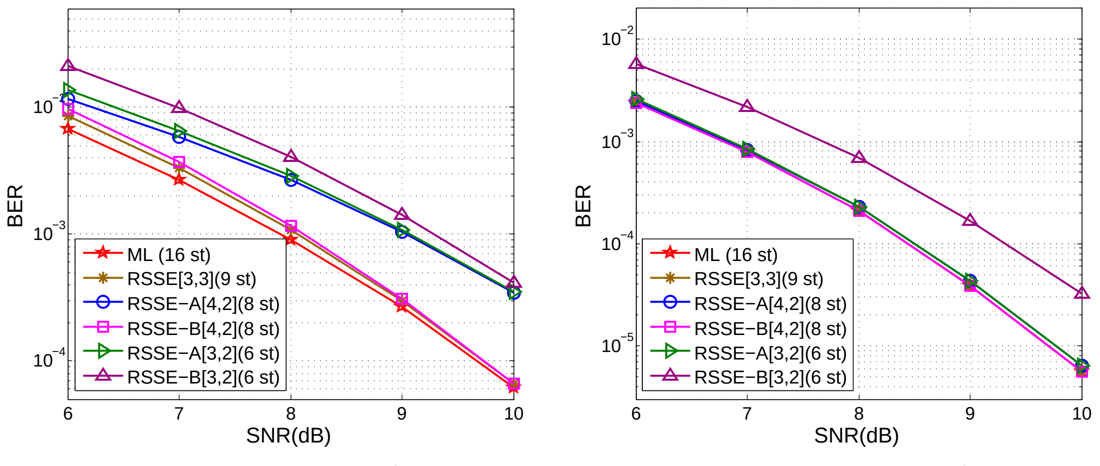
$\Delta_2^2 \geq \Delta_3^2$, the minimum intra-subset ESPDs from the top to the bottom are Δ_3^2, Δ_2^2, Δ_1^2 and ∞ in a sequential order. Tree-B (Fig. 4.3(b)) only differs from tree-A at level L_2. It is designed for the case when $\Delta_2^2 < \Delta_3^2$, so that the minimum ESPDs become Δ_2^2 followed by Δ_3^2, Δ_1^2 and ∞, from the top to the bottom. The performance of RSSE based on tree-A has been simulated and analyzed for the 2H2T system [18]. In this paper, we aim to show the performance improvement of using tree-B rather than tree-A in a 3H2T system where α is large.

We construct subset trellises based on tree-A and tree-B, and simulate RSSE for the 3H2T system where $\alpha = 1$ and $\epsilon = 0.3$. In Fig. 4.2(b) we see that $\Delta_3^2 > \Delta_2^2$ for all $\epsilon > 0$, so we expect the bit error rate (BER) performance of the trellises based on tree-B to be lower than that of the



(a) tree-A (b) tree-B

Figure 4.3: Two set partitioning trees designed for 3H2T system. Each tree has four levels, $\{L_i\}_{i=1}^4$. Four input symbols form a diamond-shape constellation.



(a) $h(D) = 1 + 2D + D^2$ (b) $h(D) = 4 + 7D + D^2$

Figure 4.4: Performance comparison of RSSE trellises with different configurations. The 3H2T systems are set to $\alpha = 1$, $\epsilon = 0.3$, but with different channel targets.

trellises constructed from tree-A. The simulation results on channels with two different length-3 channel polynomials are plotted in Figs. 4.4. The PR2 channel with $h(D) = 1 + 2D + D^2$ is linear phase, while the channel with target $h(D) = 4 + 7D + D^2$ is a minimum phase channel which is also considered in [49]. The set partitioning tree used to construct a subset trellis is indicated by "A" or "B" in the legends. For example, the term RSSE-A$[4, 2]$ represents the subset trellis constructed from tree-A with configuration $[4, 2]$. Since tree-A and tree-B have the same 3-subset partitioning (level L_3), they result in the same subset trellis with $[3, 3]$. We represent it by RSSE$[3, 3]$ for short. The signal-to-noise ratio (SNR) is defined by

$$\text{SNR(dB)} = 10 \log \frac{\|h(D)\|^2}{2\sigma^2}. \tag{4.29}$$

It is shown in Fig. 4.4(a) that RSSE$[3, 3]$ with 9 states and RSSE-B$[4, 2]$ with 8 states perform very close to the ML detector which uses 16 states, especially when SNR is high. In contrast to RSSE-B$[4, 2]$, RSSE-A$[4, 2]$, also with 8 states, suffers from large performance loss. RSSE-A$[3, 2]$ has a similar performance as RSSE-A$[4, 2]$. An interesting observation is that RSSE-B$[3, 2]$ is even worse than RSSE-A$[3, 2]$, which is different from our expectation.

We explain the BER performances through error event analysis. Due to the use of subset states, certain decoding paths will merge earlier in the subset trellis than in the ML trellis. These "early merged" paths can result in *early merged error events* [18] that possibly decrease the minimum distance of the system. In Table 4.2 we present the minimum distances for different trellises, with the same system setup as in Fig. 4.4(a). For each trellis configuration, we also give an example of the error events which achieve the minimum distance. Given $\bar{e}^{(1)}$ and $\bar{e}^{(2)}$, which are error events in the vector form, the minimum squared distance of using subset trellis is calculated by [18]

$$d_{\mathrm{r}}^2(\bar{e}^{(1)}, \bar{e}^{(2)}) = \lambda_1^2 \sum_{i=0}^{l-1} \left(\sum_{k=0}^{\nu} h_k \bar{e}_{i-k}^{(1)} \right)^2 + \lambda_2^2 \sum_{i=0}^{l-1} \left(\sum_{k=0}^{\nu} h_k \bar{e}_{i-k}^{(2)} \right)^2, \tag{4.30}$$

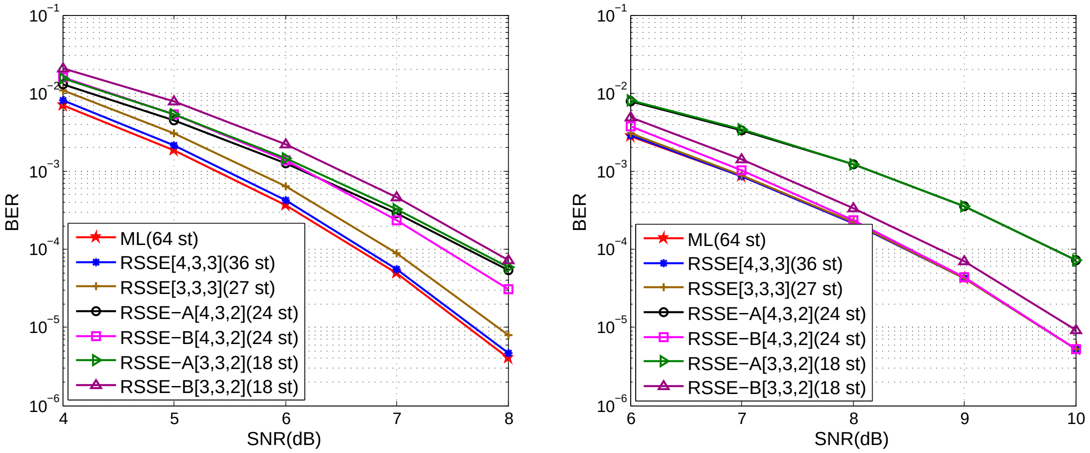
where l is the length of the error pattern.

It is shown in Table 4.2 that RSSE-B$[4,2]$ does not produce additional early merged error events whose distance is smaller than the minimum distance of the ML detector. In contrast, the minimum distances of RSSE-A$[4,2]$ and RSSE-A$[3,2]$ are smaller than that of the ML detector. Therefore, their BER performances are close and subject to be worse than the ML detector. Different from RSSE-B$[4,2]$, the performance of RSSE-B$[3,2]$ is dominated by early merged error events. Although it has a slightly larger $\min d_{\rm r}^2$ than RSSE-A$[3,2]$, the dominant early merged error events for RSSE-B$[3,2]$ appear more frequently so that its BER performance is worse than that of RSSE-A$[3,2]$.

Compared to the PR2 channel, RSSE algorithm generally performs better on channel $h(D)=4+7D+D^2$, as indicated in Fig. 4.4(b). It is shown that RSSE$[3,3]$, RSSE-A$[4,2]$, RSSE-B$[4,2]$, and RSSE-A$[3,2]$ essentially achieve the same performance as the ML detector. RSSE-B$[3,2]$, however, has roughly 0.8dB loss at BER= 10^{-4}. Our error event analysis shows that RSSE-B$[4,2]$ is dominated by the same error events as in the PR2 channel, and has a much smaller minimum distance than that of the ML detector.

We also present the performance of RSSE when $h(D)=1+D-D^2-D^3$ in Figs. 4.5. We consider the cases when ITI is low ($\epsilon=0.1$) and high ($\epsilon=0.3$). In Fig. 4.5(a) where $\epsilon=0.1$, RSSE$[4,3,3]$ with 36 states could nearly achieve the same performance as the ML detector. The performance loss of RSSE$[3,3,3]$ with 27 states is within 0.2dB at BER=10^{-5}. We observe that the performances of subset trellises constructed from tree-A and tree-B are very close. It is reasonable since when ITI is low, the difference between Δ_2^2 and Δ_3^2 is small. In Fig. 4.5(b) where $\epsilon=0.3$, the BER curves of the ML detector, RSSE$[4,3,3]$, RSSE$[3,3,3]$, and RSSE-B$[4,3,2]$ are essentially overlapped. RSSE-B$[3,3,2]$ has approximately 0.2dB performance loss at BER=10^{-5}. RSSE-A$[4,3,2]$ and RSSE-A$[3,3,2]$ have the same performance. We can see that for the case of high ITI, the trellises constructed from tree-B generally achieve better performance than those constructed from tree-A.

In [18] we have presented how to use error event diagram to search for the dominant error events for a given RSSE trellis on 2H2T channel. This technique can be easily extended to the 3H2T system. Once we know the dominant error events that mostly degrade the RSSE



(a) $\epsilon = 0.1$ (b) $\epsilon = 0.3$

Figure 4.5: Performance comparison of RSSE trellises on 3H2T systems with $\alpha = 1$, and $h(D) = 1 + D - D^2 - D^3$.

performance, it would be possible to design proper constrained codes to avoid such error events [4].

4.3.4 ITI Estimation

In general, the receiver may not have the information about α or ϵ. We show that gain loops can be incorporated in WSSJD to adaptively estimate the ITI parameters. Once we know the α and ϵ estimates, they are fed into WSSJD to evaluate the weighted branch metrics. First notice that in the transformed system, the subtract channel (4.21) has the same form as the one we obtained in the 2H2T system [17],

$$\frac{1}{1-\epsilon}(r^{(1)}(D) - r^{(3)}(D)) = z^{(2)}(D)h(D) + \bar{\omega}^{(2)}(D). \tag{4.31}$$

The parameter $1/(1-\epsilon)$ is a normalization factor, which could be estimated by a gain loop. Let $g_k^{(2)}$ be the estimate of $1/(1-\epsilon)$ at time k. In [17] we have presented a gain loop for tracking $g_k^{(2)}$, so the details are omitted here. We are more interested in the sum channel, which is different from [17].

For the sum channel, we again model it as a gain factor estimation problem,

$$\frac{1}{(1+\epsilon)^2+2\alpha^2}\left[(1+\epsilon)[r^{(1)}(D)+r^{(3)}(D)]+2\alpha r^{(2)}(D)\right]=z^{(1)}(D)h(D)+\bar{\omega}^{(1)}(D). \tag{4.32}$$

Let $g_k^{(1)}$ be the estimate of $\frac{1}{(1+\epsilon)^2+2\alpha^2}$ at time k, and let $f_k^{(1)}$ and $f_k^{(2)}$ be estimators of $(1+\epsilon)$ and 2α, respectively. Based on the Least Mean Squares (LMS) adaptive algorithm, we estimate $g_k^{(1)}$ by the following steps:

$$\bar{r}_k^{(1)} = g_{k-1}^{(1)}\,[f_{k-1}^{(1)}(r_k^{(1)}+r_k^{(3)})+f_{k-1}^{(2)}r_k^{(2)}], \tag{4.33}$$

$$e_{k-\delta}^{(1)} = \bar{y}_{k-\delta}^{(1)}\;-\;\bar{r}_{k-\delta}^{(1)}, \tag{4.34}$$

$$g_k^{(1)} = g_{k-1}^{(1)}+\beta\;\bar{y}_{k-\delta}^{(1)}\,e_{k-\delta}^{(1)}. \tag{4.35}$$

Here, parameter δ represents a small time delay, and $\bar{y}_{k-\delta}^{(1)}$ is the instantaneous estimate of the noiseless output of the sum channel. The step-size parameter β is used to control the convergence speed. The estimates of $f_k^{(1)}$ and $f_k^{(2)}$ are updated once we obtain the value of $g_k^{(1)}$. In some applications, the value of α is fixed and known. For this case, $f_k^{(2)}$ is set to 2α, and $f_k^{(1)}$ is updated by

$$f_k^{(1)} = \sqrt{\frac{1}{g_k^{(1)}}-2\alpha^2}. \tag{4.36}$$

If α is unknown, we can use $g_k^{(2)}$ from the subtract channel, and set

$$f_k^{(1)} = 2-\frac{1}{g_k^{(2)}},\quad f_k^{(2)} = \sqrt{2\left(\frac{1}{g_k^{(1)}}-\left(f_k^{(1)}\right)^2\right)}. \tag{4.37}$$

The updated $f_k^{(1)}$, $f_k^{(2)}$ and $g_k^{(1)}$ are used in the next iteration to calculate (4.33)-(4.35).

Recall that the path metric should be properly weighted in WSSJD. We replace λ_1^2, λ_2^2 by their estimators $(g_k^{(1)})^{-1}$, $(g_k^{(2)})^{-2}$, and the branch metric of the trellis edge from state $\boldsymbol{s}_{k-1}$ to $\boldsymbol{s}_k$

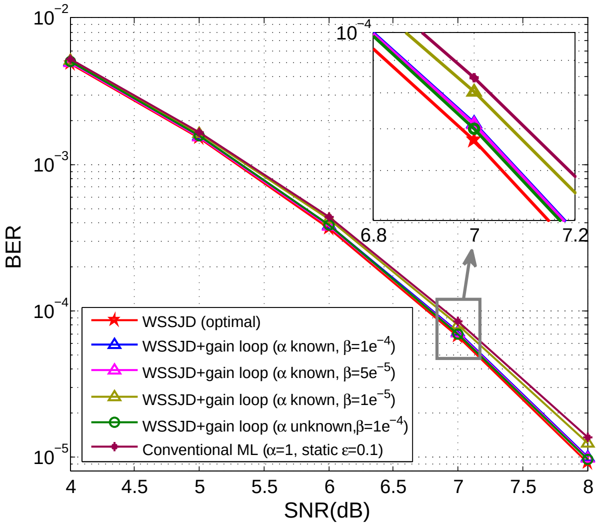


Figure 4.6: Performance comparison of WSSJD with gain loops on 3H2T channel. $h(D) = 4 + 7D + D^2$.

is calculated by

$$m(\boldsymbol{s}_{k-1}, \boldsymbol{s}_k) = \left(g_{k-1}^{(1)}\right)^{-1} \left[\bar{r}_k^{(1)} - \sum_{i=0}^{\nu} h_i z_{k-i}^{(1)}\right]^2 + \left(g_{k-1}^{(2)}\right)^{-2} \left[\bar{r}_k^{(2)} - \sum_{i=0}^{\nu} h_i z_{k-i}^{(2)}\right]^2. \quad (4.38)$$

Remark 4. *We estimate the gain factor instead of directly estimating the unknown coefficients of* $r^{(1)}(D)$, $r^{(2)}(D)$ *and* $r^{(3)}(D)$ *because the latter optimization problem has many global minima.*

In Fig. 4.6 we show some illustrative simulation results. The 3H2T system has channel target $h(D) = 4 + 7D + D^2$. Parameter α has a fixed value, $\alpha = 1$, and ϵ is assumed to be time-varying,

$$\epsilon(k) = \epsilon_0 + 0.1 \sin(4\pi(k/L)). \quad (4.39)$$

In our simulation, ϵ_0 is set to 0.1, and the sector length $L = 4096$. Hence, ϵ sinusoidly varies within the range $[0, 0.2]$. The optimal performance is obtained by implementing WSSJD with known α and $\epsilon(k)$. We then simulate the WSSJD algorithm incorporated with gain loop estimation.

The initial values of $g_0^{(1)}$, $g_0^{(2)}$ and $f_0^{(1)}$,$f_0^{(2)}$ are obtained by training the algorithm with one sample sector. For the case when α is known at the receiver, we simulate the gain loop structure with different values of step size β. We observe that both $\beta = 5 \times 10^{-5}$ and $\beta = 1 \times 10^{-4}$ could bring the performance of WSSJD close to the optimal one. WSSJD with gain loops when both α and ϵ are unknown is also simulated. It is surprising to see that lacking knowledge of α does not essentially degrade the performance. For comparison, we also show simulation results for a conventional ML detector with static $\alpha = 1$ and $\epsilon = 0.1$. The reason why we use static value is because of the ITI dependency of the ML trellis. We can see that WSSJD with gain loop estimates outperforms the static conventional ML detector.

4.4 The Generalized WSSJD

In this section we generalize the WSSJD algorithm to an extended MHMT family of channels. We show that, if the interference matrix satisfies a specific property, the transformed channel can be detected on a deterministic trellis whose branch labels do not depend on ITI.

4.4.1 WSSJD Property

Consider a general mHnT system given by

$$\boldsymbol{R}(D) = \mathbf{A}_{m,n}\boldsymbol{X}(D)h(D) + \boldsymbol{\Omega}(D), \tag{4.40}$$

where $\boldsymbol{X}(D)$ consists of input sequences stored on n tracks, $\boldsymbol{R}(D)$ corresponds to the readback sequences from m heads, and $\boldsymbol{\Omega}(D)$ is a length-m vector of Gaussian noise sequences. We assume that the entries of $\mathbf{A}_{m,n}$ contain unknown variables, whose values are affected by physical properties. For convenience, we call these variables the "ITI parameters". The matrices whose entries are independent of ITI parameters are said to be deterministic.

Suppose the SVD of $\mathbf{A}_{m,n}$ has the form of

$$\mathbf{A}_{m,n} = \begin{bmatrix} \mathbf{U}_{(1)} & \mathbf{U}_{(2)} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{\Lambda}_p & \mathcal{O} \\ \mathcal{O} & \mathcal{O} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{V}_{(1)}^{\top} \\ \mathbf{V}_{(2)}^{\top} \end{bmatrix}, \tag{4.41}$$

where $\mathbf{U}_{(1)}$ is $m \times p$, $\mathbf{U}_{(2)}$ is $m \times (m-p)$, $\mathbf{V}_{(1)}$ is $n \times p$, and $\mathbf{V}_{(2)}$ is $n \times (n-p)$. For full rank $\mathbf{A}_{m,n}$, $p = \min(m,n)$; otherwise, $p < \min(m,n)$. The submatrix $\mathbf{\Lambda}_p = \text{diag}(\lambda_1, \cdots, \lambda_p)$ is a diagonal matrix of singular values of $\mathbf{A}_{m,n}$.

Let $\mathbf{\Lambda}_p^{-1} = \text{diag}(\lambda_1^{-1}, \cdots, \lambda_p^{-1})$ be the inverse of $\mathbf{\Lambda}_p$. Substituting $\mathbf{A}_{m,n}$ by (4.41) and reorganizing (4.40), we can transform the original mHnT channel into

$$\mathbf{\Lambda}_p^{-1}\mathbf{U}_{(1)}^{\top}\mathbf{R}(D) = \mathbf{V}_{(1)}^{\top}\mathbf{X}(D)h(D) + \mathbf{\Lambda}_p^{-1}\mathbf{U}_{(1)}^{\top}\mathbf{\Omega}(D). \tag{4.42}$$

Define $\bar{\mathbf{R}}(D) = \mathbf{\Lambda}_p^{-1}\mathbf{U}_{(1)}^{\top}\mathbf{R}(D)$ as the new outputs, $\mathbf{Z}(D) = \mathbf{V}_{(1)}^{\top}\mathbf{X}(D)$ as the new inputs, and $\bar{\mathbf{\Omega}}(D) = \mathbf{\Lambda}_p^{-1}\mathbf{U}_{(1)}^{\top}\mathbf{\Omega}(D)$ as the new noise components. The transformed channel model is described by

$$\bar{\mathbf{R}}(D) = \mathbf{Z}(D) + \bar{\mathbf{\Omega}}(D). \tag{4.43}$$

It can be verified that the noise components in the new system are independent,

$$E[\bar{\boldsymbol{\omega}}_k \bar{\boldsymbol{\omega}}_k^{\top}] = E[\mathbf{\Lambda}_p^{-1}\mathbf{U}_{(1)}^{\top}\boldsymbol{\omega}_k\boldsymbol{\omega}_k^{\top}\mathbf{U}_{(1)}\mathbf{\Lambda}_p^{-1}] = \sigma^2\mathbf{\Lambda}_p^{-2}. \tag{4.44}$$

The transformed channel consists of p parallel subchannels, each of which is mathematically modeled as

$$\lambda_i^{-1}\left(\sum_{j=1}^{m} u_{ij} r^{(j)}(D)\right) = z^{(i)}(D)h(D) + \bar{\omega}^{(i)}(D), \tag{4.45}$$

for $i = 1, ..., p$. The ML solution of (4.43) is the vector of input sequences that minimize the *weighted* path metric,

$$\boldsymbol{Z}^*(D) = \arg \min_{\boldsymbol{Z}(D)} \sum_{i=1}^{p} \lambda_i^2 \|\bar{r}^{(i)}(D) - z^{(i)}(D)h(D)\|^2. \tag{4.46}$$

A WSSJD trellis is constructed to track $\boldsymbol{Z}(D)$. From (4.42), we see that $\boldsymbol{Z}(D)$ is obtained by applying a linear transform $V_{(1)}^{\top}$ to the original input $\boldsymbol{X}(D)$. Therefore, if $\mathbf{V}_{(1)}$ is a deterministic matrix, then the resulting WSSJD trellis will be independent of the ITI parameters.

Definition 1. *An MHMT interference matrix* $\mathbf{A}_{m,n}$ *satisfies* ***the WSSJD property*** *if its SVD has a deterministic* $\mathbf{V}_{(1)}$ *matrix.*

The WSSJD trellis is preferable in practice since it has fixed branch labels. It is possible that, on the real HDD, the ITI parameters vary from track to track and disk to disk. If the interference matrix satisfies the WSSJD property, then only one WSSJD trellis is needed to store in the memory. It helps to save memory space compared to maintaining separate trellises for each interference matrix.

For the case when ITI parameters are unknown, we could incorporate gain loops to adaptively train the detector. For instance, assume the entries in $\mathbf{A}_{m,n}$ are functions of an unknown parameter ϵ. Mathematically, the i-th subchannel is given by

$$\lambda_i^{-1}(\epsilon) \cdot \left(\sum_j u_{ij}(\epsilon) r^j(D) \right) = z^i(D)h(D) + \bar{\omega}_i. \tag{4.47}$$

Let $g^{(i)}$ and $\hat{u}_{ij}$ be the estimator of $\lambda_i(\epsilon)^{-1}$ and $u_{ij}(\epsilon)$, respectively. We estimate them by the following steps:

1. Treat $g^{(i)}$ as the gain factor, and estimate it by a gain loop.
2. Solve for $\hat{\epsilon}$ from $g^{(i)}$ by assuming $g^{(i)} = \lambda_i^{-1}(\hat{\epsilon})$.
3. Update the estimates $\hat{u}_{ij} = u_{ij}(\hat{\epsilon})$ using the new estimate $\hat{\epsilon}$.
4. Weight the branch metric of the i-th subchannel by $(g^{(i)})^{-2}$.

5. Repeat step 1)-4) using new estimates g^i and $\hat{u}_{ij}$.

4.4.2 WSSJD Channels

Proposition 3 and Proposition 4 give sufficient conditions for an interference matrix to satisfy the WSSJD property.

Proposition3. *Let* $\mathbf{T}_n$ *be an arbitrary* $n \times n$ *symmetric deterministic matrix. If an* $n \times n$ *matrix* $\mathbf{A}_n$ *can be written as*

$$\mathbf{A}_n = \sum_{i=0}^{N} \gamma_i \mathbf{T}_n^i, \tag{4.48}$$

where γ_i*,* $i = 0, ..., N$*, are variables, and* N *is an arbitrary nonnegative integer, then* $\mathbf{A}_n$ *satisfies the WSSJD property.*

Proof. The symmetric matrix $\mathbf{T}_n$ can be factorized as $\mathbf{T}_n = \mathbf{V}_n \mathbf{\Sigma}_n \mathbf{V}_n^\top$, where $\mathbf{V}_n$ is a unitary matrix, and $\mathbf{\Sigma}_n$ is diagonal. Since $\mathbf{V}_n^\top \mathbf{V}_n = \mathbf{I}_n$, we have $\mathbf{T}_n^i = \mathbf{V}_n \mathbf{\Sigma}_n^i \mathbf{V}_n^\top$. Therefore, (4.48) becomes

$$\mathbf{A}_n = \sum_{i=0}^{N} \gamma_i \mathbf{V}_n \mathbf{\Sigma}_n^i \mathbf{V}_n^\top = \mathbf{V}_n \left(\sum_{i=0}^{N} \gamma_i \mathbf{\Sigma}_n^i \right) \mathbf{V}_n^\top. \tag{4.49}$$

Since $\mathbf{V}_n$ is deterministic, $\mathbf{A}_n$ satisfies the WSSJD property. ■

Proposition4. *Let* $\mathbf{T}_n$ *be an arbitrary* $n \times n$ *symmetric deterministic matrix. If an* $m \times n$ *matrix* $\mathbf{A}_{m,n}$ *satisfies*

$$\mathbf{A}_{m,n}^\top \mathbf{A}_{m,n} = \sum_{i=0}^{N} \gamma_i \mathbf{T}_n^i, \tag{4.50}$$

where γ_i*,* $i = 0, ..., N$*, are variables, and* N *is an arbitrary nonnegative integer, then* $\mathbf{A}_{m,n}$ *satisfies the WSSJD property.*

Proof. Suppose the SVD of $\mathbf{A}_{m,n}$ is $\mathbf{A}_{m,n} = \mathbf{U}_m \mathbf{\Sigma} \mathbf{V}_n^\top$. Then,

$$\mathbf{A}_{m,n}^\top \mathbf{A}_{m,n} = \mathbf{V}_n \left(\mathbf{\Sigma}^\top \mathbf{\Sigma} \right) \mathbf{V}_n^\top. \tag{4.51}$$

As shown in the proof of Proposition 3, $\mathbf{V}_n$ is the deterministic unitary matrix in the eigen-decomposition of $\mathbf{T}_n$. Hence, $\mathbf{A}_{m,n}$ satisfies the WSSJD property. ■

We construct several matrices that satisfy the WSSJD property, based on Proposition 3 and Proposition 4. In the following examples, we assume $\mathbf{T}_n$ to be the $n \times n$ symmetric matrix of the form

$$\mathbf{T}_n = \begin{bmatrix} 0 & 1 & & & \mathcal{O} \\ 1 & 0 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & 0 & 1 \\ \mathcal{O} & & & 1 & 0 \end{bmatrix}. \tag{4.52}$$

On real HDDs, the dominant ITI often comes from the adjacent tracks, so (4.52) is a reasonable assumption for $\mathbf{T}_n$ to capture this phenomenon. The eigen-decomposition of $\mathbf{T}_n$ has been well studied [35]. Example 6-10 are constructed based on $\mathbf{T}_n$. The ITI parameters are presented by Greek letters.

Example 6. *An interference matrix of the form*

$$\mathbf{A}_n = \begin{bmatrix} \alpha & \epsilon & & & \mathcal{O} \\ \epsilon & \alpha & \epsilon & & \\ & \ddots & \ddots & \ddots & \\ & & \epsilon & \alpha & \epsilon \\ \mathcal{O} & & & \epsilon & \alpha \end{bmatrix}, \tag{4.53}$$

satisfies WSSJD property. It is easy to see that $\mathbf{A}_n = \alpha \mathbf{I}_n + \epsilon \mathbf{T}_n$.

Example7. *Consider the case when $n = 3$ and $N = 2$. Since*

$$\mathbf{T}_3^2 = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 2 & 0 \\ 1 & 0 & 1 \end{bmatrix}, \tag{4.54}$$

by Proposition 3 we know that $\mathbf{A}_3$ of the form

$$\mathbf{A}_3 = \alpha\mathbf{I}_3 + \beta\mathbf{T}_3 + \gamma\mathbf{T}_3^2 = \begin{bmatrix} \alpha+\gamma & \beta & \gamma \\ \beta & \alpha+2\gamma & \beta \\ \gamma & \beta & \alpha+\gamma \end{bmatrix} \tag{4.55}$$

satisfies the WSSJD property. We do not need to consider the case when $N \geq 3$ since

$$\mathbf{T}_3^i = \begin{cases} 2^{\lfloor i/2 \rfloor}\mathbf{T}_3 & i \text{ is odd} \\ 2^{i/2-1}\mathbf{T}_3^2 & i \text{ is even.} \end{cases} \tag{4.56}$$

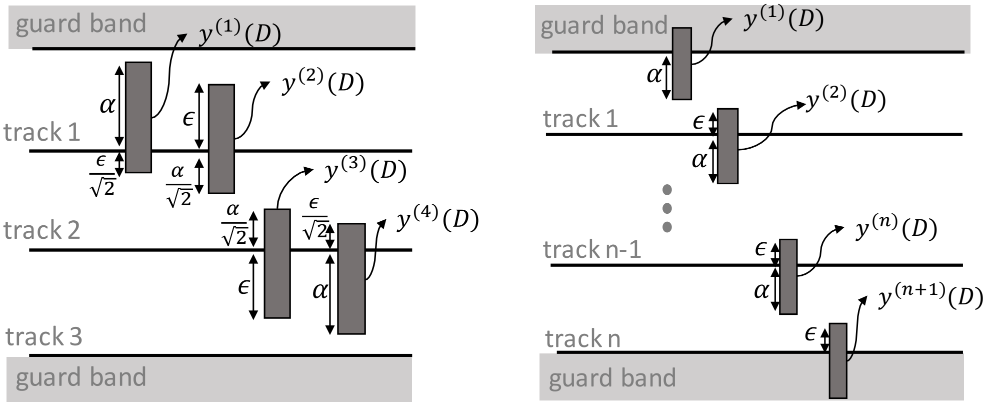
Example8. *It is obvious to see that $\mathbf{A}_{3,2}$ given in (4.4) satisfies*

$$\mathbf{A}_{3,2}^\top \mathbf{A}_{3,2} = (1 + \alpha^2 + \epsilon^2)\mathbf{I}_2 + (\alpha^2 + 2\epsilon)\mathbf{T}_2. \tag{4.57}$$

Example9. *The 4×3 matrix $\mathbf{A}_{4,3}$ given by*

$$\mathbf{A}_{4,3} = \begin{bmatrix} \alpha & \epsilon/\sqrt{2} & 0 \\ \epsilon & \alpha/\sqrt{2} & 0 \\ 0 & \alpha/\sqrt{2} & \epsilon \\ 0 & \epsilon/\sqrt{2} & \alpha \end{bmatrix}, \tag{4.58}$$

satisfies $\mathbf{A}_{4,3}^\top \mathbf{A}_{4,3} = (\alpha^2 + \epsilon^2)\mathbf{I}_3 + (\sqrt{2}\alpha\epsilon)\mathbf{T}_3$.



(a) Example 9

(b) Example 10

Figure 4.7: Head Alignments corresponding to the interference matrices in (a) Example 9, and (b) Example 10.

Example 10. *Suppose* $\mathbf{A}_{n+1,n}$ *is of the form*

$$\mathbf{A}_{n+1,n} = \begin{bmatrix} \alpha & & & \mathcal{O} \\ \epsilon & \alpha & & \\ & \ddots & \ddots & \\ & & \epsilon & \alpha \\ \mathcal{O} & & & \epsilon \end{bmatrix}. \tag{4.59}$$

Since $\mathbf{A}_{n+1,n}^{\top}\mathbf{A}_{n+1,n} = (\alpha^2+\epsilon^2)\mathbf{I}_n + (\alpha\epsilon)\mathbf{T}_n$, $\mathbf{A}_{n+1,n}$ satisfies the WSSJD property.

Fig.4.7(a) and Fig.4.7(b) show possible physical head layouts for Example 9 and Example 10, respectively. The ITI parameters are labeled on the figures to indicate the amount of ITI sensed by each head. Guard bands are added to prevent cross-talk between different track groups.

4.4.3 Discussion on Feasible Interference Matrices

In [49] the authors investigate the optimal placement of three heads that achieves the lowest BER. They show that, when ITI is high, the optimal placement is to put two heads symmetrically over two tracks, and place the third head at a similar off-track as either one of the other two. As mentioned in [49], this asymmetric placement could be derived from the distance analysis of the simplified model. We assume the same interference matrix as in [49],

$$\mathbf{B}_{3,2} = \begin{bmatrix} \alpha & 1-\alpha \\ 1-\alpha & \alpha \\ \beta & 1-\beta \end{bmatrix}, \tag{4.60}$$

where $\alpha \in [0.5, 1]$, and $\beta \in [0.5, 1]$. It indicates that a large α leads to small ITI in the first two heads. Assume the ISI channel is $h(D)$. The squared distance of a given error event $\boldsymbol{E}(D)$ is

$$\begin{aligned} d^2(\boldsymbol{E}(D)) &= \|\mathbf{B}_{3,2}\boldsymbol{E}(D)h(D)\|^2 \\ &= (\alpha^2 + (1-\alpha)^2 + \beta^2)\|e^{(1)}(D)h(D)\|^2 \\ &\quad + (\alpha^2 + (1-\alpha)^2 + (1-\beta)^2)\|e^{(2)}(D)h(D)\|^2 \\ &\quad + [4\alpha(1-\alpha) + 2\beta(1-\beta)]\left\langle e^{(1)}(D)h(D), e^{(2)}(D)h(D)\right\rangle. \end{aligned} \tag{4.61}$$

The minimum squared distance considering all single track error events is

$$d_{\mathrm{s}}^2 = \left[2(\alpha - \frac{1}{2})^2 + (\beta - 1)^2 + \frac{1}{2}\right] d_0^2, \tag{4.62}$$

which can be achieved by setting $e^{(1)}(D) = 0$ and $e^{(2)}(D)$ to be the non-zero error event that leads to d_0^2 on $h(D)$.

For double track error events, the minimum squared distance becomes

$$d_{\mathrm{d}}^2 = \left[8(\alpha - \frac{1}{2})^2 + 4(\beta - \frac{1}{2})^2\right] d_0^2. \tag{4.63}$$

To achieve (4.63), choose $e^{(1)}(D)$ to be the dominant error event on channel $h(D)$, and set $e^{(2)}(D) = -e^{(1)}(D)$.

The overall minimum squared distance

$$d_{\min,\mathbf{B}_{3,2}}^2 = \min d^2(\boldsymbol{E}(D)) = \min\{d_{\mathrm{s}}^2, d_{\mathrm{d}}^2\}. \tag{4.64}$$

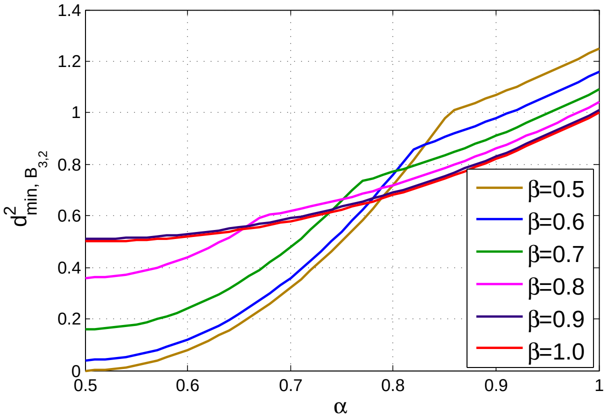


Figure 4.8: Minimum Squared Distance of 3H2T system with $\mathbf{B}_{3,2}$. Assume $d_0 = 1$.

Consider the effect of varying α and β in the range of $[0.5, 1]$. Increasing α will increase both d_{s}^2 and d_{d}^2. Increasing β will decrease d_{s}^2, but increase d_{d}^2. The difference between d_{d}^2 and d_{s}^2 is

$$d_{\mathrm{d}}^2 - d_{\mathrm{s}}^2 = 6(\alpha - \frac{1}{2})^2 + 3(\beta - \frac{1}{3})^2 - \frac{5}{6}. \tag{4.65}$$

When α is close to 1, i.e. the ITI is small, it is easy to see $d_{\mathrm{d}}^2 > d_{\mathrm{s}}^2$, so single track error events dominate. To make d_{min}^2 larger, a small value of β is preferred since it leads to larger d_{s}^2. In contrast, decreasing α to 0.5, i.e. large ITI, the double track error events become more severe. Therefore, a large β could help to improve the overall performance. In Fig. 4.8 we plot the minimum distance found at different values of β. Our analysis agrees with the simulation results in [49]. It is shown in Fig. 13 of [49] that the optimal placement of the third head is to set $\beta = 0.5$ for small ITI, and $\beta = 1$ for large ITI. In addition, the 3H2T system generally performs better at small ITI.

We examine the head placements that are amenable to apply WSSJD. When $\beta = 0.5$, the matrix $\mathbf{B}_{3,2}$ is equivalent to $\mathbf{A}_{3,2}$ in (4.4) up to a scaling and row permutation. When $\beta = 1$, the matrix

$$\mathbf{B}_{3,2} = \begin{bmatrix} \alpha & 1-\alpha \\ 1-\alpha & \alpha \\ 1 & 0 \end{bmatrix} \tag{4.66}$$

does not satisfy the WSSJD property. However, it can be partitioned into a 2×2 submatrix that has the desired property, and a 1×2 deterministic matrix. To apply WSSJD, only the 2×2 submatrix needs to be transformed. In [49] the authors also considered an interference matrix of the form

$$\mathbf{B}_{3,2} = \begin{bmatrix} \alpha & 1-\alpha \\ 1-\alpha & \alpha \\ \alpha & 1-\alpha \end{bmatrix}. \tag{4.67}$$

To apply WSSJD to this channel, the readback signals from head 1 and head 3 are averaged. Then the resulting channel becomes a symmetric 2H2T system that has been studied in [17].

4.5 Conclusion

The WSSJD algorithm is an alternative way to implement the ML detection for the MHMT channel. It is preferable in real drives since the branch labels in a WSSJD trellis are independent of the ITI parameters. In this work, we extend WSSJD to generalized MHMT channels whose interference matrices satisfy the WSSJD property. Gain loop structures are incorporated into WSSJD to adaptively estimate the unknown ITI parameters. We mainly focus on the performance of WSSJD on 3H2T systems, since they are potential candidates to implement in the new generation TDMR drives. Simulation results show that WSSJD with gain loop estimates could achieve near optimal performance in a time-varying ITI environment. The head alignment for the 3H2T system assumed in [49] is theoretically analyzed from the minimum distance perspective. We show that, WSSJD could be applied to many feasible interference matrices derived in [49], with a simple modification.

4.6 Appendix

Minimum Distance of 3H2T system We consider the channel model given by (4.3) and (4.4). Let $\boldsymbol{E}(D)$ be an error event. The squared minimum distance is calculated by

$$\begin{aligned} d^2(\boldsymbol{E}(D)) &= \|\mathbf{A}_{3,2}\boldsymbol{E}(D)h(D)\|^2 \\ &= \|e^{(1)}(D)h(D) + \epsilon \cdot e^{(2)}(D)h(D)\|^2 + \alpha^2\|e^{(1)}(D)h(D) + e^{(2)}(D)h(D)\|^2 \\ &\quad + \|\epsilon \cdot e^{(1)}(D)h(D) + e^{(2)}(D)h(D)\|^2 \\ &= (1+\alpha^2+\epsilon^2)\left(\|e^{(1)}(D)h(D)\|^2 + \|e^{(2)}(D)h(D)\|^2\right) \\ &\quad + (4\epsilon + 2\alpha^2)\left\langle e^{(1)}(D)h(D), e^{(2)}(D)h(D)\right\rangle \end{aligned} \tag{4.68}$$

We find $\min d^2(\boldsymbol{E}(D))$ by considering single track error events and double track error events separately.

1. Single track error events. Since the system is symmetric, it is sufficient to consider the case when $e^{(2)}(D) = 0$. Under this assumption, equation (4.68) reduces to

   $$d^2(e^{(1)}(D)) = (1+\alpha^2+\epsilon^2)\|e^{(1)}(D)h(D)\|^2 \geq (1+\alpha^2+\epsilon^2)d_0^2, \tag{4.69}$$

   where d_0 is the minimum distance measured on single track ISI channel with $h(D)$.

2. Double track error events. By Cauchy-Schwarz inequality, we have

   $$\begin{aligned} &\left\langle e^{(1)}(D)h(D), e^{(2)}(D)h(D)\right\rangle \\ &\geq -\left|\left\langle e^{(1)}(D)h(D), e^{(2)}(D)h(D)\right\rangle\right| \\ &\geq -\|e^{(1)}(D)h(D)\| \cdot \|e^{(2)}(D)h(D)\| \\ &\geq -\frac{1}{2}\left(\|e^{(1)}(D)h(D)\|^2 + \|e^{(2)}(D)h(D)\|^2\right). \end{aligned} \tag{4.70}$$

Using (4.70), we can get a lower bound of (4.68),

$$\begin{aligned} d^2(e^{(1)}(D), e^{(2)}(D)) &\geq (\epsilon^2 - 2\epsilon + 1)\left(\|e^{(1)}(D)h(D)\|^2 + \|e^{(2)}(D)h(D)\|^2\right) \\ &\geq 2(1-\epsilon)^2 d_0^2 \end{aligned} \tag{4.71}$$

To obtain $d_{\min}^2 = 2(1-\epsilon)^2 d_0^2$, choose $e^{(1)}(D)$ to be the error event that lead to d_0^2 on channel $h(D)$, and set $e^{(2)}(D) = -e^{(1)}(D)$.

Compare (4.69) and (4.71), we conclude that

$$d_{\min,3H2T}^2 = \begin{cases} (1 + \alpha^2 + \epsilon^2) d_0^2 & \text{if } \alpha^2 > 1 - 4\epsilon + \epsilon^2 \\ 2(1-\epsilon)^2 d_0^2 & \text{otherwise.} \end{cases} \tag{4.72}$$

Acknowledgements

This chapter is in part a reprint of the material in the paper: Bing Fan, Paul H. Siegel and Hemant K. Thapar, “Generalized weighted sum subtract joint detection,” which is being prepared for submission to *IEEE Transactions on Magnetics*. The dissertation author was the primary investigator and author of these papers, and co-authors have approved the use of the material for this dissertation.

Chapter 5

Enhancing the Average Lifetime of Flash Memory by Short q-Ary WOM Codes

Non-binary write-once memory (WOM) codes can be used to enhance the lifetime of multi-level flash memories by constraining unidirectional changes of cell levels. Most earlier works on WOM codes focus on their worst-case performance. In this work, we study the average number of writes of WOM codes, assuming on each write iteration the input alphabet is fixed. We model the WOM writing process as a Markov chain problem, and present techniques to evaluate the number of successful writes in average. Several code constructions are compared. A greedy algorithm is given to search for the optimal updating function of a given WOM labeling.

5.1 Introduction

Write-once memory (WOM) codes have been extensively studied as a coding technique for data storage with one-time programming properties, e.g., punch cards, optical disks, and flash memories. Flash memories are now the most widely used non-volatile memories due to their fast read/write speed, low power consumption, and better data integrity. However, the limited lifetime and the high cost per bit are two main properties that are less appealing than hard-disk drives. One of the most prominent features of flash memories is the asymmetry in programming and erasing. Namely, the level of flash memory cells can be easily increased by injecting electrons to their floating gates; however, to decrease the level of a single cell, the whole block of cells ($\sim 10^6$ cells) has to be erased and reprogrammed. The block erasure operation is not only time-consuming, but degrades the performance of flash memory cells as well. A typical 4-level flash cell can tolerate approximately 10^3 to 10^4 erasures before it no longer could be used due to the heavy programming noise at the end of its lifetime.

Considerable progress has been made to construct binary and non-binary WOM codes since the pioneering work in [39] and [55]. Different constructions based on algebraic codes [25], [59], and with error correction capability [60] were studied. For multi-level representation, WOM codes based on lattices were studied in [28], [7] and [8]. Some of the above codes have very good rate (close to capacity in [55]) and low encoding and decoding complexity.

Note that in all works mentioned above, it is assumed that the WOM codes should have t writes (i.e. information updating cycles) for the *worse case*. In this paper, we study an alternate criterion of WOM codes, that is, the *average number of writes* before an erasure is required. This criterion is reasonable because the worst case does not represent a typical sequence of information stored. If we assume the information message follows a distribution over a finite set Σ, then the average number of writes would be positively related, in particular linearly, to the expected lifetime of flash memory cells. Some initial results have been presented in [12] on the average-case performance of floating codes. In this paper, we focus on developing tools to evaluate the average number of writes of different WOM codes. The paper is organized as follows. In Section 5.2 we mathematically model the WOM codes that will be used in this work. In Section 5.3 we formally define the average number of writes of a given WOM code. Then, we focus on lattice-based WOM codes, and develop recursive formulas to calculate their average number of writes. In Section 5.4 we propose a greedy algorithm to search for the optimal updating function for a given WOM labeling. The “optimality” is in the sense of maximizing the number of successful writes in average-case. We also prove the optimality of the tile-labeling construction [8] when the alphabet size is 3. We conclude the paper in Section 5.5.

5.2 Preliminary

We assume that each memory cell can support q charge levels. Let $[q] \triangleq \{0, 1, \cdots, q-1\}$. The *state* of an n-cell memory unit is a vector of cell levels, denoted by $\boldsymbol{x} = (x_1, ..., x_n) \in [q]^n$. A state $\boldsymbol{y}$ is *accessible* from state $\boldsymbol{x}$ if $\boldsymbol{y} \succcurlyeq \boldsymbol{x}$, where "$\succcurlyeq$" indicates $y_i \geq x_i, \forall i = 1, ..., n$.

In the q-ary WOM model, messages can be stored on n cells under the restriction that the cell levels can only be increased. The use of WOM code allows us to rewrite the same memory cells for multiple times without violating the restriction. In this work, we assume that the set of information messages to be stored in each write is fixed, and denoted as Σ.

The q-ary WOM code is defined by specifying a pair of *decoding* and *updating* functions. The decoding function, $\Psi : [q]^n \rightarrow \Sigma$, maps each state of an n-cell memory to a message in Σ. Notice that we do not need to know the write number in the decoding process since Σ remains the same for each write iteration. The updating function $\Gamma : [q]^n \times \Sigma \rightarrow [q]^n$ generates the new cell levels given the current state and the message to be stored. For a successful write, the updating function finds an accessible state that carries the desired message, i.e. $\boldsymbol{y} = \Gamma(\boldsymbol{x}, m) \succcurlyeq \boldsymbol{x}$. If such a state does not exist, an erasure is required. For simplicity, we denote this process as $\Gamma(\boldsymbol{x}, m) = D$, where D is an imaginary state that represents an erasure. The decoding and the updating functions need to be consistent, i.e. $\Psi(\Gamma(\boldsymbol{x}, m)) = m$, for all successful writes. A t-write WOM code can guarantees t writes in the worst scenario.

5.3 Average Number of Writes of WOM Codes

In some works, a t-write WOM code is constructed by designing disjoint subsets of states for each write iteration. Therefore, the expected performance of such codes is the same as the worst case. In this paper, since we assume Σ is fixed for every write, the WOM code can reuse its current state, if the incoming message is the same as its current label. By this assumption, a t-write WOM code is possible to accommodate more number of writes.

Suppose that $\mathcal{M} = \{M_i\}_{i=1}^{\infty}$ is a random process, where M_i is drawn i.i.d. from Σ following a probability mess function p. For a WOM code(Ψ, Γ), define the random time, $T : [q]^n \times \mathcal{M} \to \mathbb{Z}$, to be the number of successful writes starting from a state $\boldsymbol{x} \in [q]^n$ for a given message sequence $\boldsymbol{m} \in \mathcal{M}$.

Definition 2. *The average number of writes from a state $\boldsymbol{x}$, denoted as $\mathbb{W}(\boldsymbol{x})$, is the expectation of the number of writes before an erasure is required. Mathematically,*

$$\mathbb{W}(\boldsymbol{x}) = \sum_{\boldsymbol{m} \in \mathcal{M}} Pr(\boldsymbol{m}) T(\boldsymbol{x}, \boldsymbol{m}), \tag{5.1}$$

The average number of writes of a WOM code is $\mathbb{W} = \mathbb{W}(\mathbf{0})$, where $\mathbf{0}$ is the all-zero state.

The WOM code writing process can be modeled as a Markov chain. Let $V = [q]^n \cup D$ be the set of possible states. The set of edges E is defined by

$$E = \{(\boldsymbol{x}, \boldsymbol{y}) \in V^2 : \exists m \in \Sigma \text{ s.t. } \Gamma(\boldsymbol{x}, m) = \boldsymbol{y}\}. \tag{5.2}$$

Given a message sequence $\boldsymbol{m} \in \mathcal{M}$, the states visited at each write form a random process $\{X_i\}_{i=0}^{\infty}, X_i \in V$. The transition probability associated with an edge $(\boldsymbol{x}, \boldsymbol{y}) \in E$ is defined by

$$\begin{aligned} \text{Pr}(\boldsymbol{x}, \boldsymbol{y}) &= \text{Pr}(X_{i+1} = \boldsymbol{y} | X_i = \boldsymbol{x}) && (5.3) \\ &= \begin{cases} p(\Psi(\boldsymbol{y})) & \text{if } \boldsymbol{y} \neq D \\ \sum\limits_{m : \Gamma(\boldsymbol{x}, m) = D} p(m) & \text{if } \boldsymbol{y} = D. \end{cases} && (5.4) \end{aligned}$$

Then, the random time T is the number of successful writes when $\{X_i\}$ first hits D, i.e.

$$T = \min\{i : X_i = D\} - 1. \tag{5.5}$$

Since T is a non-negative random variable, we have

$$\text{E}[T] = \sum_{i=0}^{\infty} i\text{Pr}(T = i) = \sum_{i=0}^{\infty} \text{Pr}(T > i) = \sum_{i=1}^{\infty} \text{Pr}(T \geq i). \tag{5.6}$$

Let $\mathbf{Q}$ denote the $q^n \times q^n$ transition matrix indexed by the states in $V \backslash \{D\}$ so that $\mathbf{Q}_{xy} = \text{Pr}(\boldsymbol{x}, \boldsymbol{y})$ for all states $\boldsymbol{x}, \boldsymbol{y} \neq D$. Theorem 1 shows how to calculate $\mathbb{W}(\boldsymbol{x})$ by using $\mathbf{Q}$.

Theorem 1. *The average number of writes starting from state $\boldsymbol{x}$ can be calculated by*

$$\mathbb{W}(\boldsymbol{x}) = \left[(\mathbf{I} - \mathbf{Q})^{-1}\mathbf{Q} \cdot \mathbf{1}\right]_{\boldsymbol{x}}, \tag{5.7}$$

where $\mathbf{1}$ is the all-one vector of length q^2, $[\boldsymbol{v}]_{\boldsymbol{x}}$ represents the entry in vector $\boldsymbol{v}$ indexed by $\boldsymbol{x}$.

Proof. Let $\text{Pr}_x(\cdot)$ and $\text{E}_x[\cdot]$ denote the probability and expectation given the initial state $X_0 = \boldsymbol{x}$, respectively. By definition of T,

$$\begin{aligned} \text{Pr}_x(T \geq i) &= \text{Pr}(X_0 \neq D, ..., X_i \neq D | X_0 = \boldsymbol{x}) \\ &= [\mathbf{Q}^i \mathbf{1}]_{\boldsymbol{x}}. \end{aligned} \tag{5.8}$$

According to (5.6), we have

$$\mathbb{W}(\boldsymbol{x}) = \text{E}_x(T) = \sum_{i=1}^{\infty} [\mathbf{Q}^i \mathbf{1}]_x = [(\mathbf{I} - \mathbf{Q})^{-1}\mathbf{Q} \cdot \mathbf{1}]_{\boldsymbol{x}} \tag{5.9}$$

■

For WOM codes with large amount of states, the calculation of $(\mathbf{I} - \mathbf{Q})^{-1}$ is computationally forbidden. We show that for some highly structured lattice WOM code, a recursive formula can be used to calculate the number of average write for each state.

Lemma 1. *The average number of writes satisfies*

$$\mathbb{W}(\boldsymbol{x}) = \sum_{m \in \Sigma : \Gamma(\boldsymbol{x}, m) \neq D} Pr(m)[1 + \mathbb{W}(\,\Gamma(\boldsymbol{x}, m)\,)]. \tag{5.10}$$

Proof. Suppose $X_0 = \boldsymbol{x}$. By first step analysis, we have

$$\mathrm{E}_x[T] = \sum_{\boldsymbol{y} \in V : \mathbf{Q}_{xy} > 0} \mathbf{Q}_{xy} \mathrm{E}[T | X_0 = \boldsymbol{x}, X_1 = \boldsymbol{y}]. \tag{5.11}$$

If the incoming message leads to an erasure, then

$$\mathrm{E}[T | X_0 = \boldsymbol{x}, X_1 = D] = 0$$

by definition. For $X_1 \neq D$,

$$\mathrm{E}[T | X_0 = \boldsymbol{x}, X_1 = \boldsymbol{y}] = 1 + \mathrm{E}[T | X_0 = \boldsymbol{y}].$$

The Lemma is obtained by considering these two cases. ■

Remark 5. *In [?], the authors introduce the write-constrained memory (WCM) model, where the transitions between states are associated with a costs. We find that Lemma 1 can be easily extended to WCM to evaluate the average cost, by changing* $1 + \mathbb{W}(\boldsymbol{x})$ *to* $c + \mathbb{C}(\boldsymbol{x})$*, where* c *and* $\mathbb{C}$ *represent the transition cost and accumulated average-cost, respectively.*

We then give two WOM codes to show how to use the recursive formula to obtain the closed form of $\mathbb{W}$. In these examples, we assume $p(m)$ is uniformly distributed. The simplest case is when $n = 1$. Using only one cell, each message has to result in a different charge increments ranging from 0 to $|\Sigma| - 1$, to guarantee $\lfloor (q-1)/(|\Sigma| - 1) \rfloor$ writes in the worst case.

Theorem 2. *Let* $a_i = \mathbb{W}(q - 1 - i)$ *and* $k = |\Sigma|$*. For* $i \leq k - 1$*,*

$$a_i = \sum_{j=1}^{i+1} \frac{\binom{i+1}{j}}{(k-1)^j}, \tag{5.12}$$

and for $i \geq k$*,* $a_i = 1 + \sum_{j=i-k+1}^{i} a_j$*.*

Proof. By Lemma 1, we have

$$a_i = \sum_{j=0}^{\min\{k-1,i\}} \frac{1}{k}(1 + a_{i-j}) \tag{5.13}$$

We use the mathematical induction to prove the case when $i \leq k-1$. It is obvious that $a_0 = \frac{1}{k}(1+a_0) \Rightarrow a_0 = \frac{1}{k-1}$. Assume (5.12) is true for states up to i. Then for $i+1$,

$$a_{i+1} = \sum_{j=0}^{i+1} \frac{1}{k}(1 + a_{i+1-j}) = a_i + \frac{1}{k}(a_{i+1} + 1). \tag{5.14}$$

Rearrange the equation,

$$\begin{aligned} a_{i+1} &= \frac{k}{k-1}a_i + \frac{1}{k-1} \\ &= \sum_{j=1}^{i+1} \frac{\binom{i+1}{j}}{(k-1)^j} + \sum_{j=2}^{i+2} \frac{\binom{i+1}{j-1}}{(k-1)^j} + \frac{1}{k-1} \\ &= \sum_{j=1}^{i+1} \frac{\binom{i+1}{j} + \binom{i+1}{j-1}}{(k-1)^j} + \frac{1}{(k-1)^{i+2}} \\ &= \sum_{j=1}^{i+2} \frac{\binom{i+2}{j}}{(k-1)^j}. \end{aligned}$$

Therefore (5.12) is also correct for $i+1$. ■

Tile-labeling is proposed in [8] to construct t-write WOM codes on 2 cells. We will show that such a construction method leads to a recursive structure that generating function can be developed to calculate $\mathbb{W}$. Consider the case when $n = 2$ and $\Sigma = \{1, 2, 3\}$. Fig. 5.1 shows an example of tile-labeling when $q = 4$. The decoding function is obtained by covering the $q \times q$ grid with the L-shape tile whose size is 3. The centers of the tile copies, which are the cubes labeled by 1, form a lattice whose generator matrix is

$$G = \begin{pmatrix} 1 & 1 \\ 2 & -1 \end{pmatrix} \tag{5.15}$$

Since $|\det(G)| = 3 = |\Sigma|$, the tile copies could disjointly fill the whole $q \times q$ grid. The

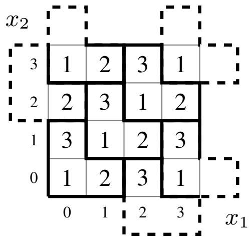


Figure 5.1: Example of tile labeling when $q = 4$ and $\Sigma = \{1, 2, 3\}$.

updating function $\Gamma((x_1, x_2), m)$ chooses the accessible state $\boldsymbol{x}'$ such that $\Phi(\boldsymbol{x}') = m$ and $\max(x'_1 - x_1, x'_2 - x_2)$ is minimized. It is easy to see that under such a rule, the updating region for each state is regularized: For the states on the right and upper boundaries (i.e. $x_1 = q - 1$ or $x_2 = q - 1$, respectively), only one cell is allowed to update, and the increment is bounded by 2; For other states (i.e. $x_1, x_2 < q - 1$), the updating region is an L-shape tile centered at the current state. The cell level increment, therefore, is bounded by 1. It can be proved that this construction can guarantee $\lfloor 3(q-1)/2 \rfloor$ writes.

Theorem3. *Let* $a_{i,j} = \mathbb{W}(q - i - 1, q - j - 1)$ *for* $0 \le i, j \le q - 1$. *Let* $b_i = a_{i,0} = a_{0,i}$. *Let* $A(x, y)$ *and* $B(x)$ *be the generating function of* $a_{i,j}$ *and* b_i*, respectively. Then*

$$A(x,y) = \frac{\frac{3xy}{(1-x)(1-y)} + (2-x)B(x) + (2-y)B(y) - 1}{2 - x - y}, \tag{5.16}$$

where

$$B(x) = \frac{x^2 + x + 1}{(1-x)^2(2+x)}. \tag{5.17}$$

Proof. By Lemma 1, we have the initial conditions: $b_0 = a_{0,0} = \frac{1}{2}$, $b_1 = a_{1,0} = a_{0,1} = \frac{5}{4}$. For $i > 1, j = 0$, we have the recursive formula

$$b_i = \frac{1}{2}(3 + b_{i-1} + b_{i-2}). \tag{5.18}$$

Let $B(x) = \sum_{n=0}^{\infty} b_n x^n$. Then

$$\begin{aligned} B(x) - b_1 x - b_0 = & \\ & \frac{1}{2}\left(3\sum_{n=2}^{\infty} x^n + x\left(B(x) - b_0\right) + x^2 B(x)\right), \end{aligned} \tag{5.19}$$

which leads to (5.17).

Similarly, for $i, j \geq 1$ we have

$$a_{i,j} = \frac{1}{2}\left(3 + a_{i-1,j} + a_{i,j-1}\right) \tag{5.20}$$

Let $A(x,y) = \sum_{m\geq 0}\sum_{n\geq 0} a_{m,n} x^m y^n$, we have

$$\begin{aligned} & A(x,y) - B(x) - B(y) + \frac{1}{2} = \\ & \frac{1}{2}\left(\frac{3xy}{(1-x)(1-y)} + x[A(x,y) - B(x)] + y[A(x,y) - B(y)]\right), \end{aligned} \tag{5.21}$$

which leads to (5.16). ■

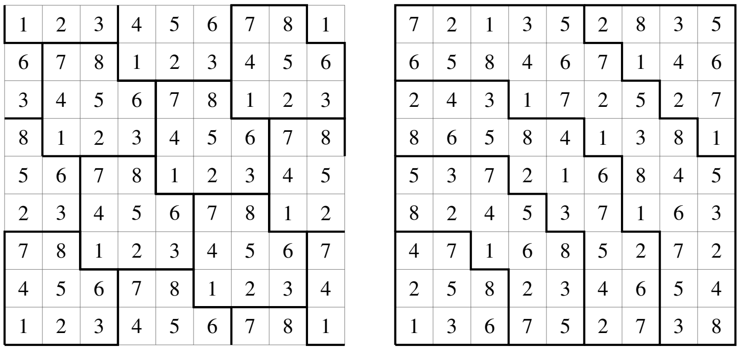
Remark 6. *Evaluating the generating functions we obtain*

$$b_i = i + 1 - \frac{2}{3} + \frac{(-1)^i}{3 \cdot 2^{i+1}}, \tag{5.22}$$

$$a_{i,i} = T_1 + 4T_2 - 2T_3 - T_4, \tag{5.23}$$

Table 5.1: $\mathbb{W}$ of $q \times q$ t-write WOM code with tile labeling.

q	4	5	6	7	8
t	4	6	7	9	10
$\mathbb{W}$	8.38	11.22	14.09	16.97	19.85



1	2	3	4	5	6	7	8	1
6	7	8	1	2	3	4	5	6
3	4	5	6	7	8	1	2	3
8	1	2	3	4	5	6	7	8
5	6	7	8	1	2	3	4	5
2	3	4	5	6	7	8	1	2
7	8	1	2	3	4	5	6	7
4	5	6	7	8	1	2	3	4
1	2	3	4	5	6	7	8	1

(a) $t = 4, \mathbb{W} = 8.17$

7	2	1	3	5	2	8	3	5
6	5	8	4	6	7	1	4	6
2	4	3	1	7	2	5	2	7
8	6	5	8	4	1	3	8	1
5	3	7	2	1	6	8	4	5
8	2	4	5	3	7	1	6	3
4	7	1	6	8	5	2	7	2
2	5	8	2	3	4	6	5	4
1	3	6	7	5	2	7	3	8

(b) $t = 5, \mathbb{W} = 7.88$

Figure 5.2: Two WOM codes for $q = 9$ and $|\Sigma| = 8$.

where

$$T_1 = \sum_{j=0}^{i-1} \sum_{l=i-j-1}^{2i-j-2} \frac{3}{2^{l+1}} \binom{l}{i-j-1}, \tag{5.24}$$

$$T_2 = \sum_{j=0}^{i} b_j \cdot \frac{1}{2^{2i-j+1}} \binom{2i-j}{i}, \tag{5.25}$$

$$T_3 = \sum_{j=0}^{i-1} b_j \cdot \frac{1}{2^{2i-j}} \binom{2i-j-1}{i}, \tag{5.26}$$

$$T_4 = \frac{1}{2^{2i+1}} \binom{2i}{i}. \tag{5.27}$$

Recall that for the $q \times q$ *WOM code,* $\mathbb{W} = a(q-1, q-1)$. *Table* **??** *lists the number of writes in the worst and average cases for different cell levels.*

Remark 7. *For the generalized two-dimensional tile-labeling with larger alphabet [8], the recursive formula also works, and can be naturally applied to calculate* $\mathbb{W}$.

5.4 Construct WOM Codes to Maximize $\mathbb{W}$

A WOM code designed to guarantee larger number of writes in the worst case may have a bad performance in average. For example, Figs. 5.2 compare the performances of two WOM codes in the worst-case and average-case. Construction 1 in Fig. 5.2(a) is a tile-labeling method that can guarantee $t = 4$ writes. The rewriting region is the shifted L-shape tile centered at the current state. Construction 2 in Fig. 5.2(b), proposed in [7] and [8], has the optimal worst-case performance. The updating region for each write iteration is indicated by the thick boundaries as shown in the figure. Under such updating rule, construction 2 can guarantee $t = 5$ writes. However, in average it can support $\mathbb{W} = 7.88$ writes, which is smaller than that of construction 1.

Generally it is very hard to find the optimal (Φ, Γ) pair that achieves the largest $\mathbb{W}$ for a given q and Σ. However, for a given decoding function Φ, the optimal updating function that leads to the largest $\mathbb{W}$ could be found by using greedy search. Algorithm 2 gives the pseudocode of greedy search when $n = 2$. The search starts from state $(q-1, q-1)$, and reversely scans each row. When it visits state (i, j), all accessible states are found and denoted as $\boldsymbol{A}$. The search order guarantees that the states in $\boldsymbol{A}$ have been visited before reaching to (i, j). If the incoming message equals to $m^* = \Phi(i, j)$, no cell level change is needed. For other messages $m \in \Sigma\backslash\{m^*\}$, we find the states in $\boldsymbol{A}$ that are labeled m. To maximize $\mathbb{W}(i, j)$, we choose the state $(x_m, y_m) \in \boldsymbol{B}$ which has the largest $\mathbb{W}(x_m, y_m)$ and set $\Gamma((i, j), m) = (x_m, y_m)$. Finally, the average number of writes at the current state is updated as

$$\mathbb{W}(i, j) = \frac{\sum\limits_{m \in \Sigma^* \backslash \{m^*\}} p(m)(\mathbb{W}(x_m, y_m) + 1) + p(m^*)}{1 - p(m^*)} \tag{5.28}$$

where $\Sigma^* \subseteq \Sigma$ consists of messages that do not lead to an erasure. It is easy to generalize Algorithm 2 to the case $n > 2$.

Theorem 4. *For a given Φ, the updating function Γ found by Algorithm 2 maximizes $\mathbb{W}(\boldsymbol{x}), \forall \boldsymbol{x} \in [q]^n$.*

Proof. We prove the theorem by induction. First notice that $\mathbb{W}(q-1, q-1)$ is upper bounded

Algorithm 2 Greedy search for optimal updating function

```
Input: a WOM labeling $\Psi$
Output: the updating function $\Gamma$ that maximizes $\mathbb{W}$
$W = q \times q$ zero matrix
Begin:
for $i = q - 1$ to 1 do
    for $j = q - 1$ to 1 do
        $m^* = \Phi(i, j)$
        $\Gamma((i, j), m^*) = (i, j)$
        $w = p(m^*)$
        $\boldsymbol{A} = \{(x, y) : (x, y) \succcurlyeq (i, j)\}$
        for $m \in \Sigma \backslash m^*$ do
            $\boldsymbol{B} = \{(x, y) \in \boldsymbol{A} : \Phi(x, y) = m\}$
            if $\boldsymbol{B} \neq \emptyset$ then
                $(x_m, y_m) = \arg\max_{(x,y) \in \boldsymbol{B}} W(x, y)$
                $\Gamma((i, j), m) = (x_m, y_m)$
                $w = w + p(m)[W(x_m, y_m) + 1]$
            end if
        end for
        $W(i, j) = w/(1 - p(m^*))$
    end for
end for
end
```

by $p(m)/(1 - p(m))$, where $m = \Phi(q - 1, q - 1)$. Assume Algorithm 2 maximizes $\mathbb{W}(\boldsymbol{y})$ for all accessible states $\boldsymbol{y} \succcurlyeq \boldsymbol{x}$. Then, by choosing the largest $\mathbb{W}(\boldsymbol{x}_m)$ for each writable message m, $\mathbb{W}(\boldsymbol{x})$ is also maximized. To see this, assume there is another Γ' giving a larger $\mathbb{W}(\boldsymbol{x})$. Then there must exist an accessible state $\boldsymbol{x}'_m$ labeled by m and $\mathbb{W}(\boldsymbol{x}'_m) > \mathbb{W}(\boldsymbol{x}_m)$, but not chosen as the next state in Γ. However, it conflicts with the greedy setting. Therefore, the assumption is not true. Updating function Γ maximizes $\mathbb{W}(\boldsymbol{x})$ for all states $\boldsymbol{x}$. ■

Corollary 1. $\mathbb{W}(\boldsymbol{x}) \geq \mathbb{W}(\boldsymbol{y})$, *if* $\boldsymbol{x} \succcurlyeq \boldsymbol{y}$.

Proof. Elementary. Notice that accessible states of $\boldsymbol{x}$ is also accessible from $\boldsymbol{y}$. ■

Theorem 5. *Assume* $|\Sigma| = 3$, *and* $p(m)$ *is a uniform distribution. The tile-labeling based WOM construction given in Fig. 5.1 has the best average-case performance, i.e.,* $\mathbb{W}$ *is maximized.*

Proof. We aim to show that the (Φ, Γ) pair in tile-labeling results in the largest $\mathbb{W}$. By Corollary

1, it is easy to see that Γ is the greedy search result. We then prove that Φ achieves the largest $\mathbb{W}$ by induction. First, tile-labeling on 2×2 grid maximizes $\mathbb{W}(\boldsymbol{x})$ for every $\boldsymbol{x} \in [2]^2$. Assume the statement is also true for the $q_1 \times q_2$ grid. Consider the labeling on grid $(q_1 + 1) \times (q_2)$. We use tile-labeling to assign messages to the cell states $(i, j), 1 \leq i \leq q_1, 0 \leq j \leq (q_2 - 1)$. Then, for the state $\boldsymbol{x} = (0, q_2 - 1)$, to achieve the largest $\mathbb{W}(\boldsymbol{x})$, the optimal way is to assign transitions from $\boldsymbol{x}$ to states $(1, q_2 - 1)$ and $(2, q_2 - 1)$, since they have the largest two $\mathbb{W}$. The assignment is possible if these three states carry different messages, which is satisfied by the tile-labeling rule. Next, for the state $\boldsymbol{x} = (0, q_2 - 2)$, we aim to assign it a different message from those of $(1, q_2 - 2)$ and $(0, q_2 - 1)$, since these two states have the largest $\mathbb{W}$. These three states form an L-shape tile. The step is repeated until all the states $(0, j)$ are labeled. The resulting decoding function is the same as the tile-labeling function Φ. Similarly, we can extend the result to $q_1 \times (q_2 + 1)$. From the discussion above, we can see that tile-labeling maximizes the average-case performance for every state.

Remark 8. *Notice that if $p(m)$ is not uniform, tile-labeling WOM may not be the optimal for $|\Sigma| = 3$.*

5.5 Conclusion

In this paper, we mathematically define the average number of writes, which is a quantity positively related to the average-case performance of WOM codes. For highly structured tile-labeling WOM codes, we develop recursive formulas to calculate their average number of writes, and present the analytical results. Generally it is very difficult to construct WOM codes that achieve the best average-case performance. However, for a given WOM labeling, greedy search could be used to find the optimal updating function that leads to the largest average number of writes. We have proved that tile-labeling construction is optimal when the alphabet size is 3. For alphabet with larger size, it remains an open problem to find the optimal WOM codes in average-case performance.

Acknowledgements

This chapter is in part a reprint of the material in the paper: Bing Fan, Minghai Qin and Paul H. Siegel, “Enhancing the average lifetime of flash memory by lattice-based WOM codes,” which is being prepared for submission to *IEEE Communications Letters*. The dissertation author was the primary investigator and author of these papers, and co-authors have approved the use of the material for this dissertation.

Bibliography

[1] S. A. Altekar, M. Berggren, B. E. Moision, P. H. Siegel, J. K. Wolf, and J. K. W. Fellow, "Error-event characterization on partial-response channels," *IEEE Trans. Inform. Theory*, vol. 45, no. 1, pp. 241–247, Jan. 1998.

[2] A. Amer, J. Holliday, D. D. E. Long, E. L. Miller, J. Paris, and T. Schwarz, "Data management and layout for shingled magnetic recording," *IEEE Trans. Magn.*, vol. 47, no. 10, pp. 3691–3697, Oct. 2011.

[3] A. Amer, J. Holliday, D. D. E. Long, E. L. Miller, J. F. Paris, and T. Schwarz, "Data management and layout for shingled magnetic recording," *IEEE Transactions on Magnetics*, vol. 47, no. 10, pp. 3691–3697, Oct 2011.

[4] R. M. R. B. H. Marcus and P. H. Siegel, "An introduction to coding for constrained systems," Oct. 2001, Available: http://cmrr-star.ucsd.edu/psiegel/book_draft.

[5] A. Baier and G. Heinrich, "Performance of m-algorithm mlse equalizers in frequency-selective fading mobile radio channels," in *IEEE Int. Conf. on Commun. (ICC)*, Boston, MA, June 1989, pp. 281–285.

[6] L. C. Barbosa, "Simultaneous detection of readback signals from interfering magnetic recording tracks using array heads," *IEEE Trans. Magn.*, vol. 26, no. 5, pp. 2163–2165, Sep. 1990.

[7] A. Bhatia, M. Qin, A. R. Iyengar, B. M. Kurkoski, and P. H. Siegel, "Lattice-based WOM codes for multilevel flash memories," *IEEE J. Selected Areas in Comm.*, vol. 32, no. 5, pp. 933–945, May 2014.

[8] Y. Cassuto and E. Yaakobi, "Short q-ary fixed-rate wom codes for guaranteed rewrites and with hot/cold write differentiation," *IEEE Transactions on Information Theory*, vol. 60, no. 7, pp. 3942–3958, July 2014.

[9] Y. Chen, P. Njeim, T. Cheng, B. J. Belzer, and K. Sivakumar, "Iterative soft decision feedback zig-zag equalizer for 2D intersymbol interference channels," *IEEE Journal on Selected Areas in Communications*, vol. 28, no. 2, pp. 167–180, February 2010.

[10] Y. Chen and S. G. Srinivasa, "Joint self-iterating equalization and detection for two-dimensional intersymbol-interference channels," *IEEE Trans. Commun.*, vol. 61, no. 8, pp. 3219–3230, August 2013.

[11] T. Cheng, B. J. Belzer, and K. Sivakumar, "Row-column soft-decision feedback algorithm for two-dimensional intersymbol interference," *IEEE Signal Processing Letters*, vol. 14, no. 7, pp. 433–436, July 2007.

[12] F. Chierichetti, H. Finucane, Z. Liu, and M. Mitzenmacher, "Designing floating codes for expected performance," *IEEE Trans. Inform. Theory*, vol. 56, no. 3, pp. 968–978, March 2010.

[13] M. R. Elidrissi, K. S. Chan, and Z. Yuan, "A study of smr/tdmr with a double/triple reader head array and conventional read channels," *IEEE Transactions on Magnetics*, vol. 50, no. 3, pp. 24–30, March 2014.

[14] M. V. Eyuboglu and S. U. H. Qureshi, "Reduced-state sequence estimation with set partitioning and decision feedback," *IEEE Trans. Commun.*, vol. 36, no. 1, pp. 13–20, Jan. 1988.

[15] B. Fan, H. K. Thapar, and P. H. Siegel, "Multihead multitrack detection with reduced-state sequence estimation in shingled magnetic recording," in *2015 IEEE Magnetics Conference (INTERMAG)*, Beijing, China, May 2015, pp. 1–1.

[16] ——, "Multihead multitrack detection with reduced-state sequence estimation in shingled magnetic recording," *IEEE Trans. Magn.*, vol. 51, no. 11, pp. 1–4, Nov. 2015.

[17] ——, "Multihead multitrack detection for next generation magnetic recording, part I: weighted sum subtract joint detection with ITI estimation," *IEEE Trans. Commun.*, vol. 65, no. 4, pp. 1635–1648, April 2017.

[18] ——, "Multihead multitrack detection for next generation magnetic recording, part II: complexity reduction - algorithms and performance analysis," *IEEE Trans. Commun.*, vol. 65, no. 4, pp. 1649–1661, April 2017.

[19] G. D. Forney, "Maximum-likelihood sequence estimation of digital sequences in the presence of intersymbol interference," *IEEE Trans. Inf. Theory*, vol. 18, no. 3, pp. 363–378, May 1972.

[20] M. Fujii and N. Shinohara, "Multi-track iterative ITI canceller for shingled write recording," in *2010 Int. Symp. on Commun. and Inform. Tech. (ISCIT)*, Tokyo, Japan, Oct. 2010, pp. 1062–1067.

[21] D. Hall, J. H. Marcos, and J. D. Coker, "Data handling algorithms for autonomous shingled magnetic recording hdds," *IEEE Transactions on Magnetics*, vol. 48, no. 5, pp. 1777–1781, May 2012.

[22] E. Hwang, R. Negi, and B. V. K. V. Kumar, "Signal processing for near 10 $\mathrm{Tbit/in}^2$ density in two-dimensional magnetic recording (TDMR)," *IEEE Transactions on Magnetics*, vol. 46, no. 6, pp. 1813–1816, June 2010.

[23] S. Karakulak, P. H. Siegel, J. K. Wolf, and H. N. Bertram, "A new read channel model for patterned media storage," *IEEE Trans. Magn.*, vol. 44, no. 1, pp. 193–197, Jan. 2008.

[24] ——, “Joint-track equalization and detection for bit patterned media recording,” *IEEE Trans. Magn.*, vol. 46, no. 9, pp. 3639–3647, Sep. 2010.

[25] S. Kayser, E. Yaakobi, P. H. Siegel, A. Vardy, and J. K. Wolf, “Multiple-write WOM-codes,” in *Proc. 48-th Annual Allerton Conference on Communication, Control and Computing*, Monticello, IL, September 2010.

[26] S. M. Khatami and B. Vasić, “Generalized belief propagation detector for TDMR microcell model,” *IEEE Trans. Magn.*, vol. 49, no. 7, pp. 3699–3702, July 2013.

[27] N. Kumar, J. Bellorado, M. Marrow, and K. K. Chan, “Inter-track interference cancelation in presence of frequency offset for shingled magnetic recording,” in *2013 IEEE Int. Conf. on Commun. (ICC)*, Budapest, Jun. 2013, pp. 4342–4346.

[28] B. Kurkoski, “Lattice-based wom codebooks that allow two writes,” in *International Symposium on Information Theory and its Applications*, Honolulu, Hawaii, October 2012, pp. 101–105.

[29] E. Kurtas, J. G. Proakis, and M. Salehi, “Reduced complexity maximum likelihood sequence estimation for multitrack high-density magnetic recording channels,” *IEEE Trans. Magn.*, vol. 35, no. 4, pp. 2187–2193, Jul. 1999.

[30] X. Ma and L. Ping, “Iterative detection/decoding for two-track partial response channels,” *IEEE Commun. Lett.*, vol. 8, no. 7, pp. 464–466, Jul. 2004.

[31] M. Marrow and J. K. Wolf, “Iterative detection of 2-dimensional ISI channels,” in *IEEE Information Theory Workshop (ITW)*, March 2003, pp. 131–134.

[32] G. Mathew, E. Hwang, J. Park, G. Garfunkel, and D. Hu, “Capacity advantage of array-reader-based magnetic recording (ARMR) for next generation hard disk drives,” *IEEE Trans. Magn.*, vol. 50, no. 3, pp. 155–161, Mar. 2014.

[33] M. Mehrnoush, B. J. Belzer, K. Sivakumar, and R. Wood, “Turbo equalization for two dimensional magnetic recording using Voronoi model averaged statistics,” *IEEE J. Sel. Areas Commun.*, vol. 34, no. 9, pp. 2439–2449, Sept 2016.

[34] J. Moon and B. Brickner, “Maximum transition run codes for data storage systems,” *IEEE Transactions on Magnetics*, vol. 32, no. 5, pp. 3992–3994, Sep 1996.

[35] S. Noschese, L. Pasquini, and L. Reichel, “Tridiagonal Toeplitz matrices: properties and novel applications,” *Numer. Linear Algebra Appl.*, vol. 20, no. 2, pp. 302–326, Mar. 2013.

[36] P. W. Nutter, I. T. Ntokas, and B. K. Middleton, “An investigation of the effects of media characteristics on read channel performance for patterned media storage,” *IEEE Trans. Magn.*, vol. 41, no. 11, pp. 4327–4334, Nov. 2005.

[37] E. Ordentlich and R. M. Roth, “Two-dimensional maximum-likelihood sequence detection is NP hard,” *IEEE Trans. Inf. Theory*, vol. 57, no. 12, pp. 7661–7670, Dec 2011.

[38] J. Park, B. Lengsfield, R. Galbraith, R. Wood, and S. Fu, “Optimization of magnetic read widths in two-dimensional magnetic recording based on micromagnetic simulations,” *IEEE Trans. Magn.*, vol. 51, no. 11, pp. 1–4, Nov 2015.

[39] R. Rivest and A. Shamir, "How to reuse a write-once memory," *Inform. and Contr.*, vol. 55, no. 1-3, pp. 1–19, December 1982.

[40] B. G. Roh, S.-U. Lee, J. Moon, and Y. Chen, "Single-head/single-track detection in interfering tracks," *IEEE Trans. Magn.*, vol. 38, no. 4, pp. 1830–1838, Jul. 2002.

[41] E. B. Sadeghian and J. R. Barry, "Soft intertrack interference cancellation for two-dimensional magnetic recording," *IEEE Trans. Magn.*, vol. 51, no. 6, pp. 1–9, June 2015.

[42] Seagate. Seagate barracuda SATA 2.5" product manual. [Online]. Available: http://www.seagate.com/www-content/product-content/seagate-laptop-fam/barracuda_25/en-us/docs/100804767c.pdf

[43] W.-H. Sheen and G. L. Stuber, "Error probability for reduced-state sequence estimation," *IEEE J. Sel. Areas Commun.*, vol. 10, no. 3, pp. 571–578, Apr. 1992.

[44] Y. Shiroishi, K. Fukuda, I. Tagawa, H. Iwasaki, S. Takenoiri, H. Tanaka, H. Mutoh, and N. Yoshikawa, "Future options for HDD storage," *IEEE Trans. Magn.*, vol. 45, no. 10, pp. 3816–3822, Oct. 2009.

[45] G. D. Smith, *Numerical Solution of Partial Differential Equations: Finite Difference Methods*, 3rd ed. Oxford: Clarendon Press, 1986.

[46] E. Soljanin and C. N. Georghiades, "Multihead detection for multitrack recording channels," *IEEE Trans. Inf. Theory*, vol. 44, no. 7, pp. 2988–2997, Nov. 1998.

[47] ——, "On coding in multi-track, multi-head, disk recording systems," in *IEEE Global Telecommun. Conf. (GLOBECOM)*, vol. 4, Houston, TX, Nov. 1993, pp. 18–22.

[48] G. Ungerboeck, "Channel coding with multilevel/phase signals," *IEEE Trans. Inf. Theory*, vol. 28, no. 1, pp. 55–67, Jan. 1982.

[49] N. Varnica, R. Radhakrishnan, S. K. Chilappagari, M. Khatami, and M. berg, "Comparison of two-reader and three-reader 2-d magnetic recording systems," *IEEE Transactions on Magnetics*, vol. 52, no. 2, pp. 1–8, Feb 2016.

[50] B. Vasic and E. M. Kurtas, Eds., *Coding and Signal Processing for Magnetic Recording Systems*. CRC Press, Nov. 2004.

[51] M. P. Vea and J. M. F. Moura, "Magnetic recording channel model with intertrack interference," *IEEE Transactions on Magnetics*, vol. 27, no. 6, pp. 4834–4836, Nov 1991.

[52] P. A. Voois and J. M. Cioffi, "Multichannel signal processing for multiple-head digital magnetic recording," *IEEE Trans. Magn.*, vol. 30, no. 6, pp. 5100–5114, Nov 1994.

[53] ——, "Upper bounds on achievable storage density: a two-dimensional approach," *IEEE Trans. Magn.*, vol. 33, no. 1, pp. 844–854, Jan 1997.

[54] Y. Wang and B. V. K. V. Kumar, "Multi-track joint detection for shingled magnetic recording on bit patterned media with 2-d sectors," *IEEE Transactions on Magnetics*, vol. 52, no. 7, pp. 1–7, July 2016.

[55] J. K. Wolf, A. D. Wyner, J. Ziv, and J. Korner, “Coding for a write-once memory,” *AT&T Bell Labs. Tech. J.*, vol. 63, no. 6, pp. 1089–1112, 1984.

[56] R. Wood, M. Williams, A. Kavcic, and J. Miles, “The feasibility of magnetic recording at 10 terabits per square inch on conventional media,” *IEEE Trans. Magn.*, vol. 45, no. 2, pp. 917–923, Feb. 2009.

[57] Y. Wu, J. A. O’Sullivan, N. Singla, and R. S. Indeck, “Iterative detection and decoding for separable two-dimensional intersymbol interference,” *IEEE Trans. Magn.*, vol. 39, no. 4, pp. 2115–2120, July 2003.

[58] H. Xia, L. Lu, S. Jeong, L. Pan, and J. Xiao, “Signal processing and detection for array reader magnetic recording system,” *IEEE Trans. Magn.*, vol. 51, no. 11, pp. 1–4, Nov. 2015.

[59] E. Yaakobi, S. Kayser, P. H. Siegel, A. Vardy, and J. K. Wolf, “Efficient two-write WOM-codes,” in *Proc. IEEE Inform. Theory Workshop*, Dublin, Ireland, August 2010.

[60] E. Yaakobi, P. H. Siegel, A. Vardy, and J. K. Wolf, “Multiple error-correcting WOM-codes,” *IEEE Trans. Inform. Theory*, vol. 58, no. 4, pp. 2220–2230, April 2012.

[61] N. Zheng, K. S. Venkataraman, A. Kavcic, and T. Zhang, “A study of multitrack joint 2-D signal detection performance and implementation cost for shingled magnetic recording,” *IEEE Trans. Magn.*, vol. 50, no. 6, pp. 1–6, Jun. 2014.