UNIVERSITY OF OKLAHOMA

GRADUATE COLLEGE

TWO DIMENSIONAL SIGNAL PROCESSING FOR STORAGE CHANNELS

A DISSERTATION

SUBMITTED TO THE GRADUATE FACULTY

in partial fulfillment of the requirements for the

Degree of

DOCTOR OF PHILOSOPHY

By

CINNA SOLTANPUR
Norman, Oklahoma
2016

TWO DIMENSIONAL SIGNAL PROCESSING
FOR STORAGE CHANNELS

A DISSERTATION APPROVED FOR THE
SCHOOL OF ELECTRICAL AND COMPUTER ENGINEERING

BY

Dr. J.R. Cruz, Chair

Dr. Kevin A. Grasse

Dr. Tian-You Yu

Dr. Choon Y. Tang

Dr. Samuel Cheng

Acknowledgements

I would like to express my gratitude to my advisor Dr. J. R. Cruz for his encouragement during my years of research. I have been fortunate to benefit from his knowledge of magnetic recording, and his insight on technical writing. He has been always available for me with help and advice.

I would like to thank Drs. K. Grasse, T. Yu, C. Tang and S. Cheng for their years of serving and support on my doctoral committee.

This research was supported in part by the National Science Foundation, Grant CCF 121-8452.

I would also like to extend my appreciation to Dr. R. Todd for providing numerous runs of Voronoi model, Dr. R. C. Keele for his help in many occasions, and ECE department staff - L. Wilkins, J. E. Bailey and S. R. Gill - for their diligent help.

Table of Contents

List of Tables

List of Figures

Abstract

Over the past decade, storage channels have undergone a steady increase in capacity. With the prediction of achieving 10 Tb/in^2 areal density for magnetic recording channels in sight, the industry is pushing towards different technologies for storage channels. Heat-assisted magnetic recording, bit-patterned media, and two-dimensional magnetic recording (TDMR) are cited as viable alternative technologies to meet the increasing market demand. Among these technologies, the two-dimensional magnetic recording channel has the advantage of using conventional medium while relying on improvement from signal processing. Capacity approaching codes and detection methods tailored to the magnetic recording channels are the main signal processing tools used in magnetic recording. The promise is that two-dimensional signal processing will play a role in bringing about the theoretical predictions.

The main challenges in TDMR media are as follows: i) the small area allocated to each bit on the media, and the sophisticated read and write processes in shingled magnetic recording devices result in significant amount of noise, ii) the two-dimensional inter-symbol interference is intrinsic to the nature of shingled magnetic recording. Thus, a feasible two-dimensional communication system is needed to combat the errors that arise from aggressive read and write processes.

In this dissertation, we present some of the work done on signal processing aspect for storage channels. We discuss i) the nano-scale model of the storage channel, ii) noise characteristics and corresponding detection strategies, iii) two-dimensional signal processing targeted at shingled magnetic recording.

1 Preliminaries

1.1 Introduction

The ubiquitous presence of digital information and rapid growth of digital data generation have been a paradigm shift at the beginning of this millennium. The rate of data generation has been expanding at an astounding pace, and the trend won't stop anytime soon. The data generation rate has taken a faster pace than Moore's law; data production is doubling every two years [1]. New frontiers to use the accumulated data have been discovered in many disciplines: national security, medicine, agriculture, advertising, etc.

Storage units are at the core of data infrastructures. Magnetic recording is an available technology for data storage. Magnetic recording devices provide a cost effective, reliable, high speed, non-volatile - maintaining the stored data with no power - and high capacity solution to the data storage units. These attributes have made magnetic devices relevant for personal use as well as the primary choice for the big data industry.

Storage devices have gone through incessant and often revolutionary changes since the invention of the first commercial hard drives by IBM [2]. Today, there are alternative technologies available to store information, for example solid state drives. The research on other technologies such as magnetoresistive random-access memory

(MRAM), is under way too. However, the magnetic recording devices persist as the dominant storage devices to store digital information.

The capacity of hard-disc drives is determined by the areal density. Areal density is the number of information bits on a given surface area and is stated in bits per square inch units. The current achievable areal density is slightly above 1 Tb/in^2 [3]. The increase in areal density is the result of several factors: better head design such as tunnel magneto-resistance read sensors, scaling of geometry of mechanical parts, superior materials for higher performing media, and improvement on signal processing techniques.

Magnetic recording channels are a baseband communication system. The basic components of the baseband communication system are: encoder, equalizer, detector, and decoder. The role of each component will be discussed in this chapter. For magnetic recording channels, stream of data is transferred over the channel in time, as opposed to space (in channels such as Ethernet.)

Signal processing improvements have been essential to increase capacity. The advent of partial response equalizers and advanced error control coding have had a huge impact on the current areal density. The implementation of such techniques has been made possible due to advances in semiconductor technology. However, reaching higher areal densities faces new challenges. Currently, the most promising technologies to increase the areal density beyond the current limit of 1 Tb/in^2 are heat assisted magnetic recording (HAMR) [4], bit patterned magnetic recording (BPMR) [5], and shingled magnetic recording for continuous medium [6]. All the possible candidates more or less encounter the same problem, namely, they require two-dimensional (2-D) signal processing [7]. The interference is two- dimensional; not only down-track signals interfere with each other but also the side tracks contribute to interference. This inter-track interference (ITI) introduces a new di-

mension to the problem which has its own implication from the signal processing perspective.

1.2 High Density Data Storage

A magnetic recording device has three basic components: the media, the write and read heads, and the signal processing unit. Digital information is stored by magnetization on the disc in different directions. For perpendicular magnetic discs, the magnetization directions are upwards and downwards, oriented perpendicular to the surface of disc, as depicted in Fig. 1.1. The magnetization directions and therefore the data is retained on the surface of the recording disc and can be recovered later on.

The head which carries write and read sensor flies over the disc at a close distance. The write sensor consists of a coil which translates the applied current provided by the write circuit to magnetic field. The magnetic flux circles through the permeable core of the head and then the layers of soft and hard magnets of the disc and closes at the the other end of the head. The head flies at a strategic distance to the surface of the magnetic disc. The gap between the head and the magnetic material is small enough to allow a high intensity of magnetic flux. The magnetic field saturates the ferromagnetic layer of the underlying media to write a binary symbol. The read sensor senses the magnetization direction written on the disc. The magnetization direction changes the impedance of the read sensor which is measured by the sampling circuit. The phenomenon that describes the changes in the impedance of read sensor is known as the megnetoresistance effect.

We discuss in detail each component of a magnetic device in this section. Development of higher user density magnetic devices is affected by the following factors;

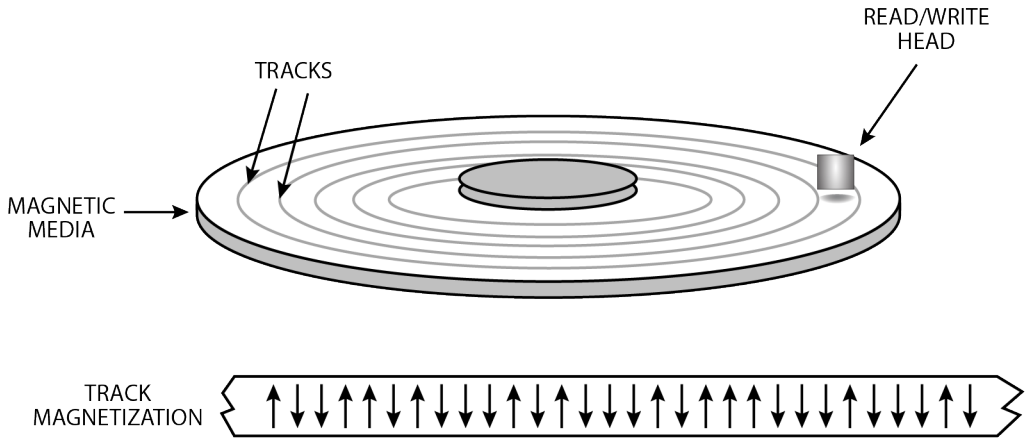


Figure 1.1: Components of magnetic recording device and magnetized cells in the media.

shrinking head dimensions, increasing sensitivity of read sensors, high resolution recording media, and advanced signal processing [8].

1.2.1 Media

Magnetic recording started a century ago and since then has gone through several breakthroughs. Hard drives remain the essential data storage device. The magnetic recording material used in the hard-disc drives is a thin film of ferromagnetic metal alloys. One of the properties of the quantum mechanical spin of electron is that it acts as a magnetic dipole. When a magnetic field is applied to a material, the magnetic dipoles are aligned to the direction of magnetic field. The magnetic moment is the result of microscopic current, change in the spin of electrons, and in smaller amounts the orbital angular moment of electrons around the nucleus. Magnetization is defined as a change in magnetic moment per unit volume.

The hysteresis curve shows the relation between magnetization M and the applied

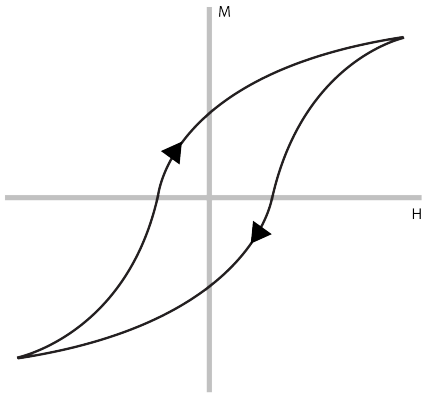


Figure 1.2: Magnetic hysteresis curve in ferromagnetic material.

magnetic field H (Fig. 1.2). For ferromagnetic materials, once the field is strong enough to reach the saturation level on the hysteresis curve, the direction of magnetization is preserved even after the magnetic field is removed. After magnetization, according to the hysteresis curve, extra energy is needed to demagnetize a ferromagnetic material, usually through an opposite magnetic field. We use the preserved magnetized directions to store information. On the other hand, the magnetic field is not the only source of energy that can change the magnetization. Thermal fluctuations also work against the magnetization direction and may switch the direction of magnetization.

The continuous medium is made of hard and soft layers on top of a substrate. The soft magnetic under-layer (SUL) is made of magnetically permeable material. This means it acts as a mirror and we can assume the image of the recording head or the read head is present in the SUL. The design factor in choosing the SUL are the thickness and magnetic moment requirement. The hard layer is made of a ferromagnetic alloy and has granular structure. Thin film poly-crystalline materials have been the primary choice for perpendicular devices. The thin film is deposited by sputtering in vacuum. The grains are made to form an isolated island on the

substrate which reduces the demagnetization effect.

High density media are perpendicular media. Perpendicular refers to the angle of magnetization direction with respect to the media surface [9]. The reason behind the use of perpendicular media is the demagnetization effect from the neighboring grains. The magnetized neighbors produce a magnetic field which could demagnetize the grain. Using the perpendicular media reduces the demagnetization effect. This is particularly important for higher density media. Another factor that play a role in higher densities is the thermal stability [6]. As discussed, the thermal fluctuation could change the magnetization. Unfortunately, thermal stability is a function of grain size and becomes an impending problem in practice for high areal densities .

1.2.2 Recording Process

Here, we give a sketch of what happens in the recording process. The aim of the recording process is to magnetize grains using the recording head. The recording head needs to produce a strong magnetic field to saturate the grains. The field also should be localized for narrow tracks and sharp transitions. In hard drives the recording head and read head are separated. The recording head is a single pole. The current in wire coiled around the pole produces a magnetic field. Therefore, the main pole and auxiliary pole tips form a magnetic field loop that goes through the SUL [10]. The recording pole is surrounded by shields. The role of shields is to improve the field gradient. The schematic is shown in Fig. 1.3.

For higher densities, the recording head needs to keep up with the size of grains; smaller recording heads are physically incapable of writing in downsized scale. The recording head size limits the magnetic field produced by the head. To overcome the issue the shingled recording process was introduced [6]. In this process the magnetic field is concentrated on a corner of the recording pole. After the information is

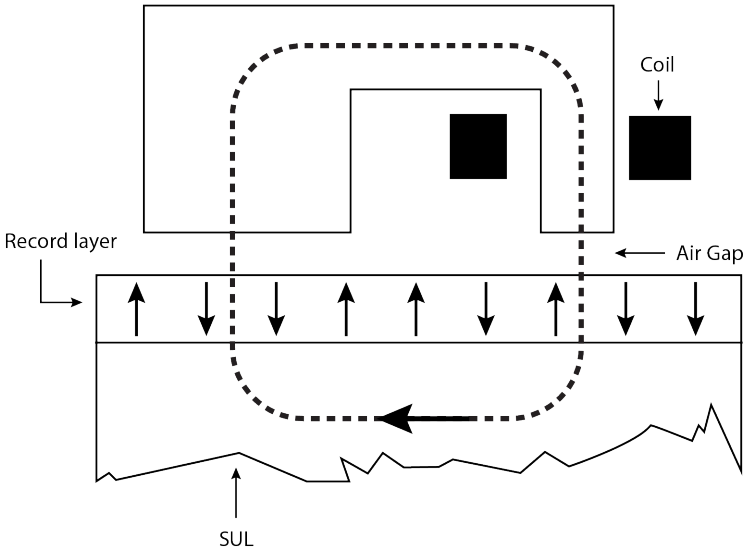


Figure 1.3: Write head for magnetic recording device [11].

written down on the track, the head moves to the next one. However, the written tracks partially overlap each other. The outcome of writing over a patch would look the same as shingles on the rooftop. The remnant of each track that has not been covered by the consequent tracks form bit-cells. A read head which is smaller in size, can read these bit-cells.

1.2.3 Read-back Process

Read sensor technology is based on the giant magnetoresistive (GMR) effect [12]. The GMR effect is the change in the resistance of connected layers of ferromagnetic and non-magnetic metals due to an external magnetic field. Current-perpendicular-to-plane (CPP)-GMR read heads are common read sensors in perpendicular recording. The CPP-GMRs have a small resistance and maintain a high magneto-resistance ratio. The small resistance reduces the thermal noise. These properties are appeal-

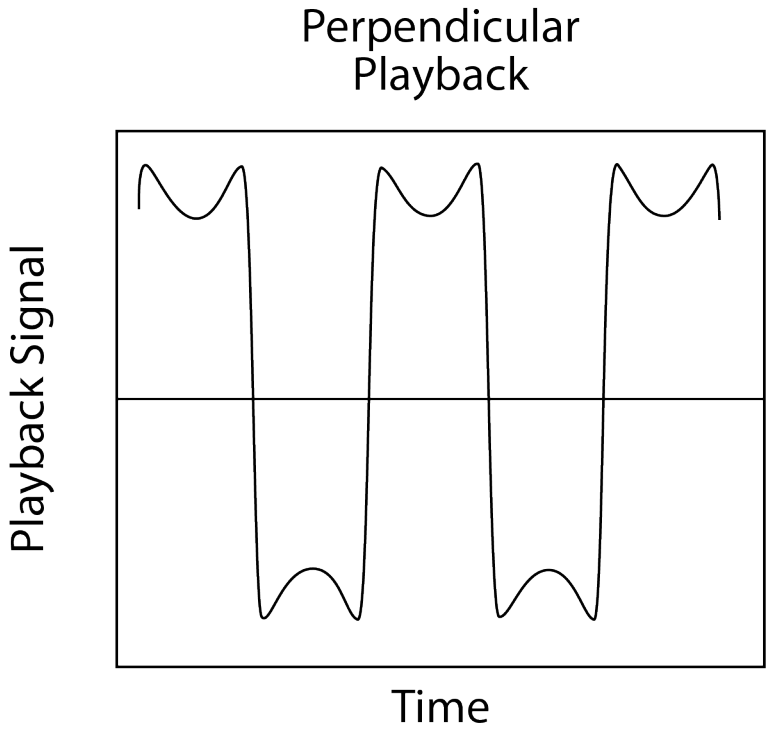


Figure 1.4: Sample of playback waveform for perpendicular recording [10].

ing for high areal density and have been the subject of many studies. A typical waveform at the output of a read sensor is shown in Fig. 1.4.

The GMR effect is related to electrons spins. According to the Mott model [13], in metals electrons mostly conserve their spin after scattering. Furthermore, in ferromagnetic metals the scattering coefficients are different for down and up spins. This is the basis for spin dependent resistance in ferromagnetic metals. We assume the scattering is stronger for electrons with spin opposite to the magnetization direction and weaker for parallel spins.

To explain the GMR effect, imagine electrons passing through layers of magnetized ferromagnetic metals each enveloped in layers of non-magnet metals. The interface between non-magnetic and ferromagnetic metal magnifies the spin depen-

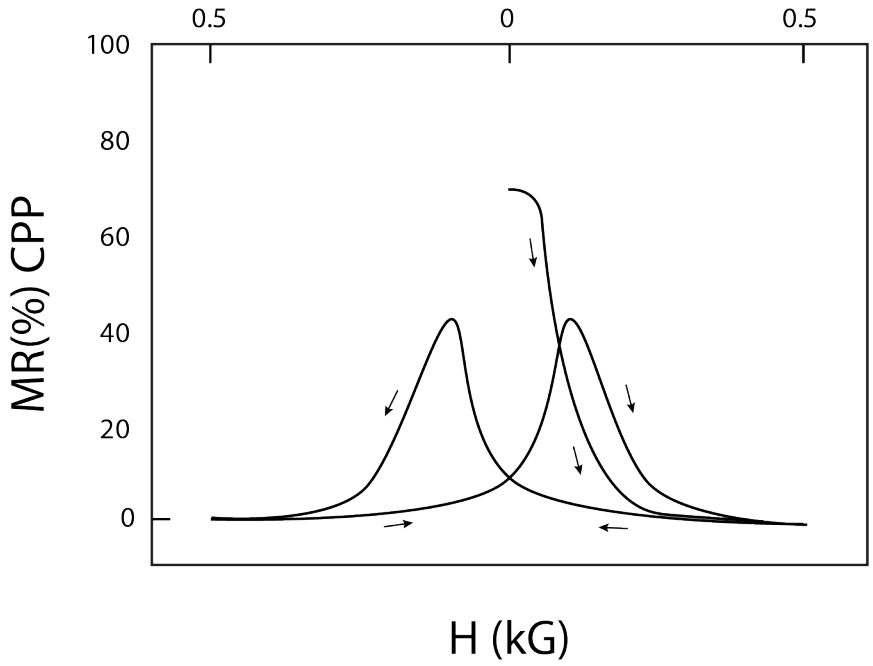


Figure 1.5: Giant magneto-resistive effect in read sensor of Nb/Co/Nb [14].

dency scattering. For two different magnetic fields in the opposite direction, the down and up spins experience the same amount of resistance as one of the layers lets them go through and the other one scatters them. On the other hand, for the magnetic fields in the aligned direction, either the up or down spin electrons are passing through the parallel magnetic field with ease and little scattering, and therefore the total resistance is much smaller. The magneto resistance ratio is simply the normalized difference between the two observed resistances for the aligned and opposite magnetization direction.

A variation of the read head sensor, the tunneling magneto-resistive (TMR) sensor, uses a thin insulator in between ferromagnetic metals. Electrons tunnel through the insulator barrier which is explained as a quantum mechanic phenomenon. In this case, the tunneling occurs with higher probability if the magnetic fields are aligned. This results in the magnetoresistive effect. The TMR usually shows a higher resis-

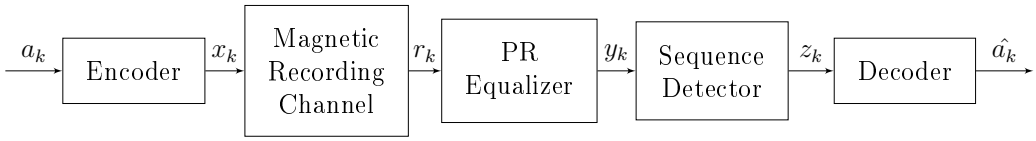


Figure 1.6: Components of baseband communication system: magnetic recording channel, partial response equalizer, channel detector, encoder and decoder.

tance and magneto-resistance ratio with respect to CPP-GMR. The TMR sensors are currently used in higher areal density disk drives.

1.3 Communication System Model

The components of a baseband communication system are explained in this section. The information is presented in binary format. The zeros and ones are turned into bipolar bits and go through an encoder. The encoder adds redundancy to the message to ensure that the message can be retrieved with no error at the other end of the communication link. The encoded message is written on the disc. Although the information is in binary format, the read out is an analog signal. The read out signals are sampled at baud rate. These samples are collected at the matched filter output to maximize the signal-to-noise ratio (SNR). The output goes through another filter called the equalizer. The role of the equalizer is to shorten the pulse response of the channel. The outputs of equalizer are then fed into the detector. The detector decides on what to make of the noisy outputs of the channel, and these decisions are handed to the decoder to decipher the original message. These components are considered the most basic elements of any communication systems. A brief explanation is given here as a reference for the other chapters.

1.3.1 Channel Characteristics

Although the read-back process was explained in Section 1.2.3, we would like to have the characteristics of the channel model in simple mathematical form. In this manner, the channel model can be used in simulations or to help our understanding of the channel. The more detailed version of noise characteristics in the read back channel is explained in Section 1.4.

A sequence of input bits $x_n \in \{-1, 1\}$, are written on the disc. However, the read sensor only responds to transitions of magnetization directions. To express the read out signal we define the transition sequence: the transition sequence b_n are the bipolar input bits that has gone through $1 - \mathbb{D}$ filter ($\mathbb{D}$ is the delay operator)

$$b_n = x_n - x_{n-1}. \tag{1.1}$$

At the output of read sensor after K consecutive read outs we can write the output as

$$r(t) = \sum_{k=1}^{K} b_k s(t - kT) + n(t). \tag{1.2}$$

The $n(t)$ represent the additive white Gaussian noise (AWGN). The function $s(t)$ is the step response of the channel that is to say the response of channel to a single isolated transition. An example of a step response for perpendicular recording is

$$s(t) = V \tanh\left(\frac{\ln 3}{T_{50}} t\right), \tag{1.3}$$

where T_{50} is the time that response reaches 75% of its maximum, and V is the magnitude of pulse response. To write the output signal in terms of input values we use the pulse response $h(t)$ defined as

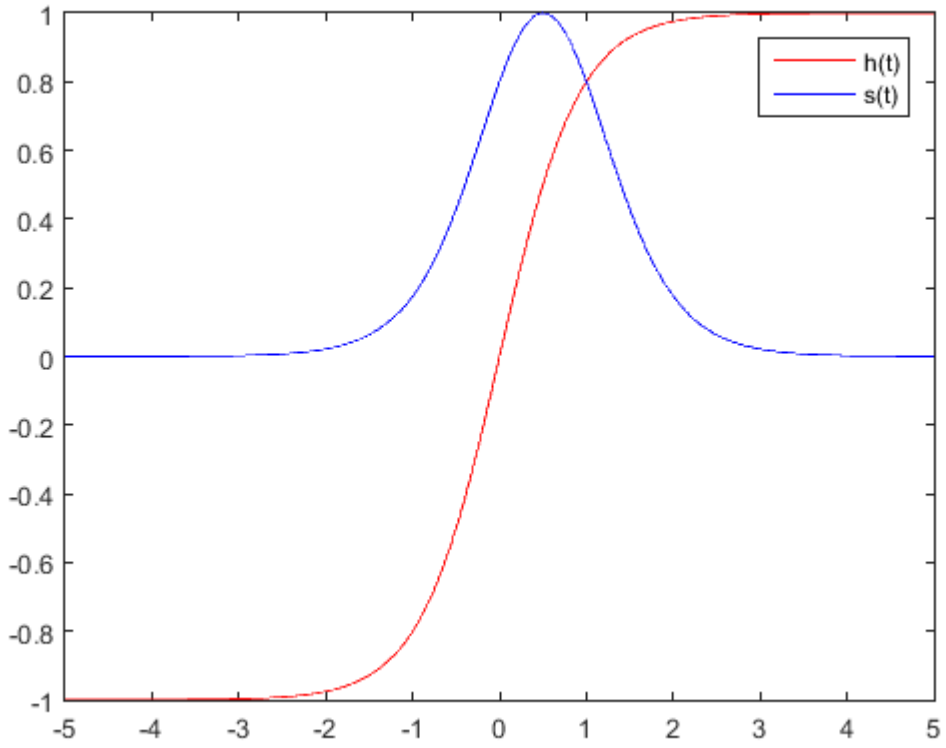


Figure 1.7: Pulse response and step response of perpendicular recording channel.

$$h(t) = s(t) - s(t - T), \tag{1.4}$$

which is the response of channel to two magnetic transitions in opposite directions. Then we are able to write the output function in terms of input values,

$$r(t) = \sum_{k=1}^{K} x_k h(t - kT) + n(t). \tag{1.5}$$

The pulse response and step response of a perpendicular magnetic recording is illustrated in Fig 1.7.

The output of the channel is further fed into a matched filter, defined as $h(-t)$. Then sampled at baud rate T. If the output of the matched filter is defined as $y(t)$,

the convolution of $r(t)$ and matched filter $r(t) \star h(-t)$, then $y(kT)$ are the collected samples, $\bar{y}_k$'s. These samples would go through a finite impulse response (FIR) filter known as the equalizer.

1.3.2 Equalizer and Partial Response

A very basic issue with magnetic recording channels is the presence of intersymbol interference (ISI). The moving read head scan over bit-cells and collects samples. The channel response has a long tail. The long tail increases the computational complexity of the detection algorithm prohibitively. To get rid of channel's long ISI, the common practice is to filter the output to either no ISI (full response equalizer) or reduced ISI (partial response equalizer).

The detector operates on the linear filtered response of the channel. The linear filter used is called the equalizer. The role of the equalizer is to shape the response of the channel and make it shorter. The combined output of the equalizer and channel is described by a polynomial and is called the target. The goal is to design the target as close as possible to the channel response. Also, the target - the short span polynomial describing the ISI in the channel - helps to find the parameters of the detection algorithm.

Examples of targets that have been used in the beginning era of magnetic recording channels are PR4 and EPR4. The polynomial describing a PR4 channel is $(1 - \mathbb{D}^2)$, and for EPR4 $(1 - \mathbb{D})(1 + \mathbb{D})^2$. The modern targets use polynomial with real coefficients, generalized partial response (GPR) [15].

Let the equalizer be a polynomial of order M and the target have $\xi + 1$ elements. The steps to derive an equalizer are as follows

- Choose the suitable size for equalizer and target $\mathbf{f}_1^M, \mathbf{g}_1^\xi$

- Define $e_k = \mathbf{f}[\overline{y}_{k+1}, ..., \overline{y}_{k+M}]^T - \mathbf{g}[x_{k-\xi}, ..., x_k]^T$
- Minimize the best linear estimator's mean squared error, $E[e_{i,j}^2|\mathbf{f}, \mathbf{g}]$ with respect to constraint $g_1 = 1$ to find $\mathbf{f}, \mathbf{g}$.

1.3.3 Detection

At the receiver the decision on received samples is made by minimizing the risk which in turn is an optimization problem. In the presence of ISI, each sample depends on more than one input and therefore the solution to the optimization problem is the sequence that comes closest to the transmitted message in our search space. In choosing a detector, the pertinent criterion besides performance is computational complexity. There are two efficient algorithms available: The Viterbi algorithm [16] and the Bahl-Cocke-Jelinek-Raviv (BCJR) algorithm [17]. These two algorithms under the AWGN assumption would find the optimum solution. They have a linear computational complexity with respect to number of transmissions and exponential with respect to the number of states on the trellis diagram.

Both BCJR and Viterbi algorithms use the trellis diagram to find the optimum solution in channels with memory. The trellis diagram is the manifestation of the underlying finite state machine through the passage of time. At each step, the system is at one of the possible states in finite state machine. For example, a finite state machine with two memory bits has four states (00), (01), (10), (11). The trellis structure is shown in Fig. 1.8. The current state represents the content of the memory. The new incoming bits advance the system forward in time into a new state. In this manner, the detector uses the diagram to traverse on all possible paths.

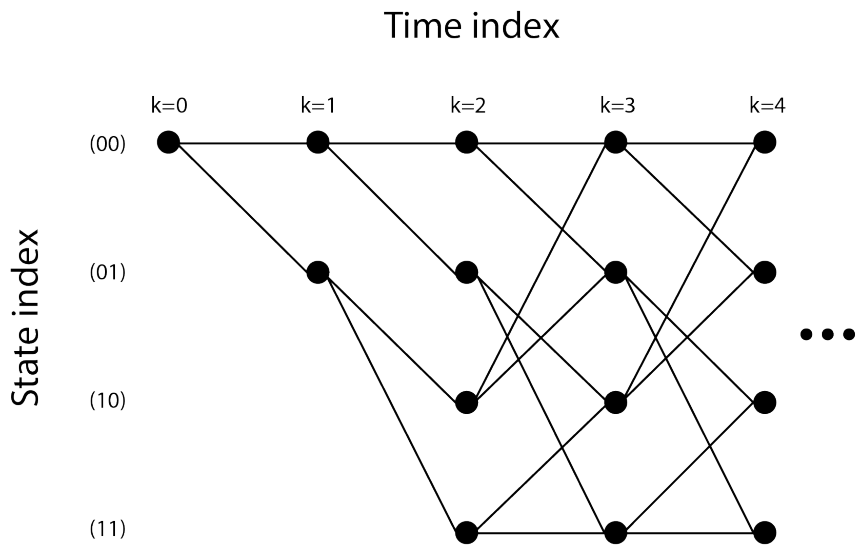


Figure 1.8: Structure of trellis for a channel with memory.

Maximum Likelihood Sequence Detection

The maximum likelihood sequence detection (MLSD) of input data given the received samples is stated as

$$\hat{\mathbf{x}}_{ML} = \arg\max_{\mathbf{x}} f_{Y|X}(\mathbf{y}|\mathbf{x}), \tag{1.6}$$

where the elements of $\mathbf{y}$ is the observation and $\mathbf{x}$ is the message which belongs to $\{-1, 1\}^{K_x}$. Suppose the message has K_x elements and $x_i = 0$ for $i < 0$. Assuming the causality the conditional distribution can be written as

$$f_{Y|X}(\mathbf{y}|\mathbf{x}) = \prod_{i=1}^{K_x} f_{Y|X}(y_i|\mathbf{y}_1^{i-1}, \mathbf{x}_1^i) = \prod_{i=1}^{K_x} f_{Y|X}(y_i|\mathbf{x}_{i-\xi}^i), \tag{1.7}$$

where y_i represent an element in $\mathbb{R}$ identified by $\mathbf{f}[\overline{y}_{i+1}, ..., \overline{y}_{i+M}]^T$. The second equality is by conditional independence. The conditional distribution term for AWGN channel can be written as

Algorithm 1.1 Viterbi Algorithm

- Initialize the state values
- Do the following steps for all received signals ($i = 1, ..., K_x$) on trellis diagram
  - Calculate the branch metrics as given in (2.15)
  - Add the state values at time i and outgoing branch metrics
  - Assign the smallest incoming branch values to the state values $i + 1$
- Starting from the last time, choose the smallest state values at each step as the survivor path
- Report back the survivor path's branch labels as the detected message

$$f_{Y|X}(y_i|\mathbf{x}_{i-\xi}^i) = (2\pi\sigma_e^2)^{-1/2} exp\left[-\frac{\left[y_i - \mathbf{g}[x_{i-\xi}, ...x_i]^T\right]^2}{2\sigma_e^2}\right], \tag{1.8}$$

where σ_e^2 is the variance of the white noise. The branch metrics on the trellis is defined by $-\ln f_{Y|X}(y_i|\mathbf{x}_{i-\xi}^i)$. The Viterbi algorithm use the log of conditional distribution to solve

$$\hat{\mathbf{x}}_{ML} = \arg \min_{\mathbf{x}} \sum_{i=1}^{K_x} \left[y_i - \mathbf{g}[x_{i-\xi}, ...x_i]^T\right]^2. \tag{1.9}$$

The answer to the optimization problem (2.15) is given through the Viterbi algorithm on the trellis diagram depicted in Fig. 1.8. On the trellis, the $\mathbf{x}_{i-\xi}^i$ defines the states at each time i and x_i indicate the branch labels. The number of states at each step is 2^ξ with 2 outgoing/incoming branches to each state. The Viterbi algorithm implementation is given in Algorithm 1.1.

The Viterbi algorithm is simply based on Bellman Principle of Optimality: "An optimal policy has the property that what ever the initial state and initial decision are the remaining decision must constitute an optimal policy with regard to the state resulting from the first decision."

The Maximum A-posteriori Detection

The alternative algorithm to solve the sequence detection is to use the maximum *a-posteriori* (MAP) formulation. The complexity of the algorithm is in the same order as the Viterbi algorithm. However, the BCJR algorithm is a symbol-by-symbol algorithm unlike the Viterbi algorithm. The decoded message at the output of the Viterbi algorithm is a sequence of bits. The symbol-by-symbol nature of BCJR has its own advantages: the estimated bits can be expressed in terms of each bit independent of the sequence. These outputs are in form of soft values meaning that its magnitude shows our confidence in the estimated bit. The exact mathematical term for the soft value is the log-likelihood ratio (LLR) and defined as

$$L_k \triangleq \log \frac{p(x_k = 1|\mathbf{y})}{p(x_k = -1|\mathbf{y})}. \tag{1.10}$$

It is easy to turn the soft values into the decoded message; we simply look at the signs of LLR's for each bit. Before getting into an explanation of the algorithm we give the MAP formulation as

$$\hat{\mathbf{x}}_{MAP} = \arg\max_{\mathbf{x}} f(\mathbf{x}, \mathbf{y}). \tag{1.11}$$

The estimation can be further simplified under Markovian property of input values and conditional independence as

$$f_{Y,X}(\mathbf{y}, \mathbf{x}) = f_{Y|X}(\mathbf{y}|\mathbf{x}) f_X(\mathbf{x}) = \prod_{i=1}^{K_x} f_X(x_i|x_{i-1}) \prod_{i=1}^{K_x} f_{Y|X}(y_i|\mathbf{x}_{i-\xi}^{i}), \tag{1.12}$$

The conditional distribution is the same as given in (1.8).

On the trellis diagram, s is the current state, and s' indicates the previous state. The numerator sums over all the transitions that pin down $a_k = 1$, and the denominator

for $a_k = -1$. On the trellis diagram L_k can be written as

$$L_k = \log \frac{\sum_{a_{k=1}} p(s_{k-1} = s', s_k = s, \mathbf{y}) p(\mathbf{y})}{\sum_{a_{k=-1}} p(s_{k-1} = s', s_k = s, \mathbf{y}) p(\mathbf{y})} + \log \frac{p(a_k = 1)}{p(a_k = -1)}, \tag{1.13}$$

where the first term is called extrinsic information and the second term is *a priori* LLR. The joint probability $p(s_{k-1} = s', s_k = s, \mathbf{y})$ is then expanded into three independent probabilities

$$p(s_{k-1}, s_k, \mathbf{y}) = \overbrace{p(s', \mathbf{y}_1^k)}^{\alpha_{k-1}(s')} \overbrace{p(y_k, s|s')}^{\gamma_k(s',s)} \overbrace{p(\mathbf{y}_{k+1}^K|s)}^{\beta_k(s)}, \tag{1.14}$$

To write the equation in a format useful for the BCJR algorithm we need to rewrite γ_k as follows

$$\gamma_k = p(s|s')p(y_k|s', s). \tag{1.15}$$

On a given branch it is feasible to calculate the probability $p(y_k|s', s)$ for the branch label. For binary inputs, the conditional probability $p(s|s')$ depends on the a_k being -1 or 1. Once available, in iterative detection, the value for $p(s|s')$ can be found through *a priori* LLR's. In absence of such information we assume *a priori* LLRs are zero, i.e. $p(s|s') = 1$. The description of BCJR algorithm is given in Algorithm 1.2.

State of the Art Detection in Magnetic Recording

The current magnetic recording systems use a GPR channel using partial response equalization. The number of states on the trellis diagram is determined by the ISI length. For current densities, the BCJR algorithm is a feasible algorithm. The soft values at the output of the detector are fed into a decoder. Some systems use extra

Algorithm 1.2 BCJR Algorithm

- Initialize the initial state values
- Do the following steps for all received signals ($i = 1, ..., K_x$) on the trellis diagram
  - Calculate the branch metrics $\gamma_k(s, s')$ as given in (2.15)
  - Recursively calculate $\alpha_k = \sum_{s' \in S} \alpha_{k-1}(s')\gamma_k(s', s)$ for $k = 1, ..., K$
  - Also find $\beta_{k-1}(s') = \sum_{s' \in S} \beta_k(s')\gamma_k(s, s')$ for all states $k = K, .., 1$
  - Assign the log likelihood ratio L_k for each time k on the trellis
- Detect message $\hat{a}_k$ by setting a_k equals to 1 or 0 according to the sign of L_k.

iterations of detection and decoding provided that error control coding updates the *a-priori* information on each bit which results in performance gain of the overall system. The described turbo equalized system is described in [18].

We assumed, in implementing the detector so far, that the noise at the output of the equalizer is white. However, the white assumption is not true in the physical channel. The improvement over performance of the available detectors in the magnetic recording channel comes from noise correlation. The assumption is that the noise is self correlated and also correlated to the input signal. The pattern dependent noise predictive (PDNP) Viterbi algorithm was first introduced in [19]. The noise is assumed to be an auto-regressive process and signal dependent. The characteristics of the noise are detailed in Section 1.4. Essential to the optimum solution is that the noise has the following description

$$n_i = b_1(\mathbf{x}^i_{i-\xi})n_{i-1} + \cdots + b_\nu(\mathbf{x}^i_{i-\xi})n_{i-\nu} + \sigma(\mathbf{x}^i_{i-\xi})w_i, \tag{1.16}$$

where $\mathbf{b} = [b_1(\mathbf{x}^i_{i-\xi}), ..., b_\nu(\mathbf{x}^i_{i-\xi})]$ represents the coefficients of auto-regressive filter. The noise is modeled as Gauss-Markov process of length ν. For brevity, let $s(\mathbf{x}^i_{i-\xi}) = \mathbf{g}[x_{i-\xi}, ...x_i]^T$. Then the output of channel is expressed as

$$y_i = s(\mathbf{x}_{i-\xi}^i) + n_i. \tag{1.17}$$

We skip the full derivation of the algorithm as it is given in [20]. The branch metrics for MLSD is updated as follows

$$\hat{\mathbf{x}}_{ML} = \arg\min_{\mathbf{x}} \sum_{i=1}^{K} \log \sigma^2(\mathbf{x}_{i-\xi}^i) + \frac{\left([-\mathbf{b}^T, 1]\left(\mathbf{y}_{i-\nu}^i - \mathbf{s}(\mathbf{x}_{i-\xi-\nu}^i)\right)\right)^2}{\sigma^2(\mathbf{x}_{i-\xi}^i)}. \tag{1.18}$$

The log-likelihood given in (1.18) defines the branch metrics on the extended states trellis. The Viterbi algorithm as described in Algorithm 1.1 with the new branch metrics finds the optimum solution to the auto-regressive correlated noise problem.

1.3.4 Error Correcting Coding

Error control coding consists of encoding and decoding methods which try to achieve reliable and efficient data transmission through an unreliable channel with the aim to reach rates close to the Shannon's limit. This limit shows the ultimate amount of information that can be transferred through a communication channel while the integrity of information remains intact. The limit has been known for long since the original paper was published in 1949 [21], but there were no algorithms available to get close to the capacity until the 1990's.

Since the rediscovery of low-density parity-check (LDPC) codes [22] by MacKay [23] as a class of capacity approaching codes, they have found different applications in various digital standards and technologies. Over the past decade, a great deal of research has been made on design, construction, encoding, decoding, performance analysis, generalization and applications of these codes.

LDPC codes are linear block codes that are constructed by designing a sparse

parity-check matrix and achieve good error performance close to the Shannon limit [24]. Various methods for constructing LDPC codes have been proposed with a great deal of research into their design and construction.

A regular binary LDPC code C is given by the null space of a sparse parity-check matrix H over $GF(2)$ for which each column has weight γ and each row has weight ρ, where γ and ρ are small compared to the code length [25]. Such an LDPC code is said to be (γ, ρ)-regular. If H is an array of sparse circulants of the same size over $GF(2)$, then the null space of H gives a quasi-cyclic (QC)-LDPC code. If H consists of a single sparse circulant or a column of sparse circulants, then the null space of H gives a cyclic LDPC code. Detailed explanation of the LDPC constructions listed above has been omitted to avoid repetition of the content of the respective papers from which the construction are based.

There are various methods for decoding LDPC codes. The sum-product algorithm (SPA) decoding gives the best error performance out of all methods and is practically implementable providing near-optimal performance [26].

The SPA algorithm determines the a posterior probability of each message symbol as a function of the parity-check equations and the channel characteristics which are conveniently described using a Tanner graph [27]. A Tanner graph is a bipartite graph consisting of factor nodes and symbol nodes. On the Tanner graph, the constraints under which the codewords are constructed defines the factors. Each symbol node d_j send to each of its children factor nodes h_i an estimate of the probability that the parity-check node is in state x, based on the information provided by the other children nodes of that symbol node. On the other hand, each factor node h_i send to each of its parents symbol nodes d_j an estimate of the probability that the parity check equation i related to the factor node h_i is satisfied, if the symbol or parent node is in state x, as illustrated in Figure 1.9. This is an iterative process

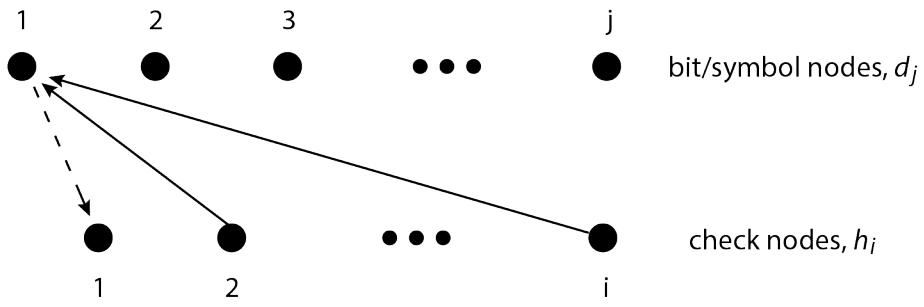


Figure 1.9: A Tanner graph linking symbol and factor nodes.

of interchanging information between the two types of nodes on the Tanner graph. The iterative process finishes after successfully calculating the syndrome condition over the estimated decoded vector, or halted when it reaches a given predetermined number of iterations. Under certain circumstances the convergence to exact solution is guaranteed [28].

There are several other capacity approaching error control coding methods available today; polar codes [29], repeat and accumulate codes [30], and turbo codes [31]. These codes are designed for memory-less channels. The research on a more comprehensive construction that covers a channel with memory is still on going.

1.4 Noise Characteristics

An accurate description of characteristics of the noise can have a considerable impact on the performance of the communication system. The idea is to model the sophisticated underlying physics in a simple mathematical form to capture the complex behavior of the system. On one hand simplicity is required to assure that other components of the communication systems is adapted to the new model. On the other hand, the model should be descriptive of physical phenomena that happen in

the physical channel. In this section, we explain the known useful models and give a brief description.

1.4.1 Markov and non-Markov noise

From the perspective of detection the independent noise characteristics is ideal. The computational complexity of such a detector reduces significantly. However, in practice, the independent assumption rarely matches the physical channel noise. One of the concepts that helps us define alternative characteristics is the Markov property [32]. Let $\{N_i\}$ be a sequence of random variables for $i = 1, ..., K$. We assume collected noise samples are correlated. Provided that the sequence has a Markov property, the joint density of the noise samples can be factored as follows

$$f_{\mathcal{N}}(\mathbf{n}) = \prod_{i=1}^{K} f_{\mathcal{N}_i}(n_i|\mathbf{n}_{i-p}^{i-1}), \tag{1.19}$$

with Markovian noise of order p. Knowing the value of p previous samples tells us about the characteristics of current sample as much as if we had the complete sequence.

We give two sequences generated by MA and AR filter as an example. Let w_i's being the independent random variables and the relation between discrete random variables N_i's be expressed as

$$n_i = a_1 n_{i-1} + \cdots + a_\nu n_{i-\nu} + w_i, \tag{1.20}$$

where $\mathbf{a} = [a_1, ..., a_\nu]$ is the coefficient of AR process with finite degree. Given that w_i is AWGN noise with mean μ and variance σ_w^2, we can write

$$f_{\mathcal{N}}(n_i|\mathbf{n}_{i-p}^{i-1}) = \frac{1}{\sqrt{2\pi\sigma_w^2}} \exp\left(-\frac{(n_i - \mu - \mathbf{a}^T\mathbf{n}_{i-p}^{i})^2}{2\sigma_w^2}\right). \tag{1.21}$$

The sum of random variables that defines n_i are jointly Gaussian. Therefore, the sequence is described by Gauss-Markov process.

We define the sequence as MA with following description

$$n_i = a_1 w_{i-1} + \cdots + a_\nu w_{i-\nu} + w_i. \tag{1.22}$$

The sequence $\{N_i\}$ does not admit a Markovian property. By Bayes' law we have

$$f_{\mathcal{N}}(\mathbf{n}) = \prod_{i=1}^{K} f_{\mathcal{N}_i}(n_i|\mathbf{n}_1^{i-1}). \tag{1.23}$$

Note that the indices on the condition parameters starts from the beginning of the sequence this time.

1.4.2 Pattern Dependency

In the previous section we assumed a rather linear model to define the correlation of noise terms. The linear model was described by constant coefficients. A more elaborate model may assume these coefficients could change with time or according to another parameter available in the system. In magnetic recording the third parameter is the input signal. In other words, the received noise characteristics differ based on the pattern of the written bits.

Unlike the MA and AR correlated noise, the covariance of the noise is not constant anymore. Since the pattern of written data changes, the correlation is also modified over time. This means that the noise covariance matrix is dependent on time and hence the process is non-stationary.

Let us give an example by defining the noise term $n_i \sim \mathcal{N}(0, \sigma(x_i))$ as follows

$$\mathbb{E}[n_i^2] = \sigma^2(x_i), \tag{1.24}$$

and for signal dependent MA model in (1.22) we have

$$n_i(x_{i-\nu}^i) = a_1(x_{i-1})w_{i-1} + \cdots + a_\nu(x_{i-\nu})w_{i-\nu} + a_0(x_i)w_i. \tag{1.25}$$

Then, for $w_i \sim \mathcal{N}(0, 1)$ the elements of covariance matrix are given by

$$\mathbb{E}[n_i n_j] = \sum_{k=i-\nu, l=j-\nu}^{i,j} a_{i-l}(x_l)a_{j-k}(x_k)\delta[l-k]. \tag{1.26}$$

1.4.3 Jitter Noise

To model the timing errors or grain boundary effects we use the jitter noise model. The timing error stems from asynchronous head positioning. The guide bits written on the media try to station the head sensor on the exact location on its attempt to read the information. Despite the efforts the exact positioning is not always possible. On the other hand, the bit cell boundaries do not always match the grain boundaries. The grains on amorphous media have fuzzy boundaries which result in an extra disturbance experienced at read out.

Here, we give the currently used jitter noise model. The transition sequence goes through the channel with the same step response $s(t)$. However, the jitter moves the step response from its origin. The jitter values are random variables with Gaussian distribution with zero mean and variance σ_j^2. The received signal in (1.2) is updated as follows

$$y(t) = \sum_{k=1}^{K} b_k s(t - kT + j_k) + n(t). \tag{1.27}$$

The jitter values are changing the response of channel. The j_k terms are nonlinear terms in the pulse response. The conventional way of dealing with jitter is to use the Taylor series as an approximation of the step response. The Taylor expansion of $s(t)$ is

$$s(t + j) = s(t) + js'(t) + \frac{j}{2}^2 s(t) + \cdots \tag{1.28}$$

provided that the jitter values are small, the first two terms gives a good enough approximation

$$s(t + j) \simeq s(t) + js'(t). \tag{1.29}$$

Let us name the $s(t+j)$ approximation the "first order jitter" model. Replacing the first order jitter in (1.27) we get

$$y(t) = \sum_{k=1}^{K} b_k s(t - kT) + \sum_{k=1}^{K} b_k j_k s'(t - kT) + n(t). \tag{1.30}$$

The first term corresponds to signal, second term is the jitter noise, and $n(t)$ represent the electronic noise and magnetic noise available in magnetic recording channel.

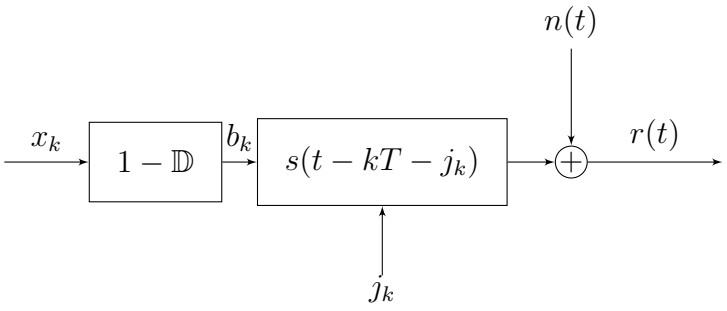


Figure 1.10: Magnetic recording channel in the presence of jitter.

2 Vector Noise Predictive Viterbi Algorithm

2.1 Introduction

State-of-the-art sequence detectors for hard-disc drives are capable of dealing with a pattern dependent Markovian noise model. This model is the *de facto* standard in the industry and proved useful in improving the performance of the system. We have given the description of the model in Chapter 1. Specifically, the pattern dependent model is advantageous over the simple Markov noise model as characteristics of noise depends on the written bits.

We briefly discuss the channel and noise model for magnetic recording. We show that the noise model exhibits regressive characteristics. The noise model is important for derivation of the sequence detector. If the correlated noise is Gauss-Markov, the optimal MLSD detector has been derived in [20]. In contrast, we present a variation of the noise predictive Viterbi algorithm noise predictive algorithm that is tailored to the linear regressive noise model. The vector noise predictive (VNP) Viterbi algorithm introduced here is the generalization of the noise predictive algorithm in [33] with extended states on the trellis diagram. The vector noise predictive (VNP) Viterbi algorithm uses a modified block LDL decomposition to factorize the

noise covariance matrix and hence obtain the likelihood metrics. We further extend the algorithm for regressive noise where the correlation with the input signal is modeled by the signal dependent regression coefficients.

2.2 Channel Model

First, let us establish some notation. We use uppercase letters for matrices. Vectors are lower case bold, and underline designates that the vector or matrix is formed of building sub-blocks. Given vectors $\mathbf{x}_i, ..., \mathbf{x}_{i+k}$, $\underline{\mathrm{x}}_i^{i+k}$ is defined as a vector consisting of all elements of $\mathbf{x}_i$ up to $\mathbf{x}_{i+k}$. We let $X_0, ..., X_q$ and $Y_1, ..., Y_q$ be matrices of the same size, and the block Toeplitz matrix $\underline{\mathrm{T}}$, by enumeration of its elements $\underline{\mathrm{T}} = [X_0, ...X_q; Y_1, ..., Y_q]$, represents

$$\underline{\mathrm{T}} = \begin{bmatrix} X_0 & X_1 & X_2 & & X_q \\ Y_1 & X_0 & X_1 & & \\ Y_2 & Y_1 & X_0 & & \\ \vdots & & & \ddots & \\ Y_q & & & & X_0 \end{bmatrix}. \tag{2.1}$$

For a magnetic recording channel, given the step response $s(t)$ as discussed in Chapter 1, the received signal can be stated as

$$y(t) = \sum_{k=1}^{K} (x_k - x_{k-1}) s(t - kT + j_k) + n(t). \tag{2.2}$$

where x_k are binary input values, j_k are jitter values with Gaussian distribution with zero mean and variance σ_j^2 and $n(t)$ is the additive noise term. The first order jitter model can be described as

$$y(t) \backsimeq \sum_{k=1}^{K}(x_k - x_{k-1})s(t - kT) + \sum_{k=1}^{K}(x_k - x_{k-1})j_k s'(t - kT) + n(t). \tag{2.3}$$

This approximation is accurate for small values of jitter noise variance with respect to pulse's width T_{50}. Therefore (2.2) can be written as

$$\bar{n}(t) = \sum_{k=1}^{K}(x_k - x_{k-1})j_k s'(t - kT) + n(t). \tag{2.4}$$

where we assumed $\bar{n}(t)$ contains the additive and jitter elements from (2.3). The boundary values are $x_0, x_K = -1$. We sample the signal at half the transmission period, i.e., $t = T/2 + mT$, and after rearranging we have

$$y(T/2+mT) = \sum_{l\geq 0}(x_{m-l} - x_{m-l-1} - x_{m+l+1} + x_{m+l})\left(s\left(T/2 + lT\right)\right) + \bar{n}(T/2+mT). \tag{2.5}$$

For hyperbolic tangent, $1 - s(t)$ decays exponentially. Indeed, if the following inequality holds

$$\frac{1 - s\left(3T/2\right)}{1 - s\left(T/2\right)} \ll 1 \tag{2.6}$$

the following three terms (i.e., for $l = 0$) sufficiently express the discrete response of the channel [34]

$$y_m = y(T/2 + mT) = -x_{m-1}s(T/2) + 2x_m s(T/2) - x_{m+1}s(T/2). \tag{2.7}$$

For $T_{50} = 0.5$, the left-hand side of (2.6) is less than 0.015. Going over the same steps for the first-order jitter model, we get the following for noise term

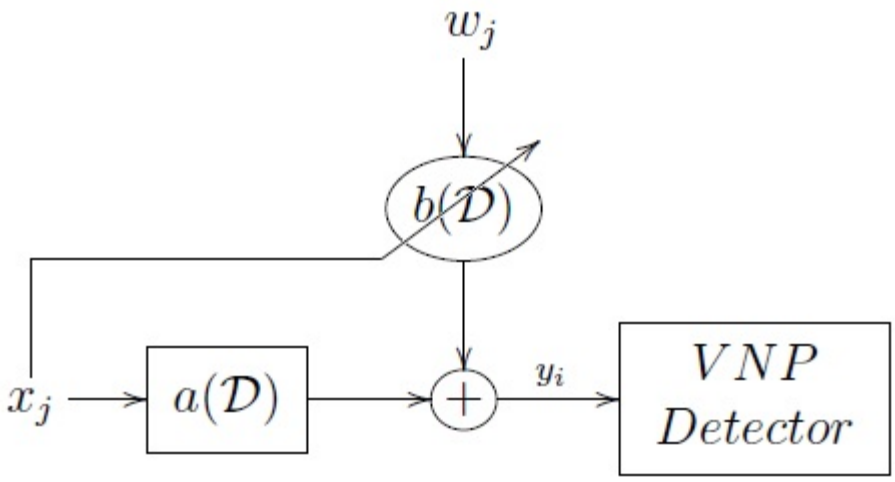


Figure 2.1: Signal dependent regressive noise where the proposed detection algorithm is effective.

$$\bar{n}(T/2+mT) = (x_{m+1}-x_m)j_{k+1}s'(T/2)+(x_m-x_{m-1})j_k s'(T/2)+n(T/2+mT). \quad (2.8)$$

The model introduced in (2.7) and (2.8) helps us to develop the detector. The channel has an infinite impulse response (IIR). Since the complexity of the Viterbi algorithm depends on the ISI length, we curtailed the ISI length in (2.7). However, the noise at the output is colored. The derived noise model is regressive and does not admit a Gauss-Markov form. Furthermore, this noise model cannot be approximated by an autoregressive model as its parameters depend on the signal. The model is depicted in Fig. 2.1, where $\mathcal{D}$ is the delay operator and additive zero mean white Gaussian noise, w_j, goes through a signal dependent filter $b(\mathcal{D})$ to represent the colored noise. The output of the channel is fed into the sequence detector. The derivation of the VNP detector is given next.

2.3 Problem Statement

Let the output of a partial-response channel be expressed as

$$y_i = \sum_{j=0}^{r} a_j x_{r+i-j} + n_i, \tag{2.9}$$

where the noise n_i is Gaussian (colored) with zero mean, a_j's are the known discrete channel response of ISI length r, and x_i belongs to alphabet set $\mathcal{X} = \{-1, 1\}$. Let us rewrite (2.9) as

$$\mathbf{y}_k = \begin{bmatrix} a_r \cdots a_0 & & \\ & \ddots & \\ & & a_r \cdots a_0 \end{bmatrix} \begin{bmatrix} x_{(k-1)m} \\ \vdots \\ x_{km+r-1} \end{bmatrix} + \begin{bmatrix} n_{(k-1)m} \\ \vdots \\ n_{km-1} \end{bmatrix}, \tag{2.10}$$

where $\mathbf{y}_k \triangleq [y_{(k-1)m}, ..., y_{km-1}]^T = A\mathbf{x}_k + \mathbf{n}_k$. The matrix A is of size $m \times (m + r)$ and column vector $\mathbf{x}_k$ has $m + r$ elements. Note the vector $\mathbf{x}_k$ shares m elements with $\mathbf{x}_{k+1}$ and so forth. The vectors $\mathbf{y}_k, \mathbf{n}_k, A\mathbf{x}_k$ are each a column vector of size m and the counter is updated as $k = 1, ..., p$. The indices start from 1 for vectors $\mathbf{y}_k, \mathbf{n}_k, \mathbf{x}_k$ so that the notation is consistent.

The problem is to find an efficient algorithm for the MLSD expressed as

$$\underline{\hat{x}}_{ML} = \arg \max_{\mathcal{X}^{pm+r}} f(\underline{y}|\underline{x}), \tag{2.11}$$

where $f(\underline{y}|\underline{x})$ is the conditional distribution of all blocks of received signal $\underline{y} = [\mathbf{y}_1^T, ..., \mathbf{y}_p^T]^T$, which has length pm, given the input values $\underline{x} = [x_0, ..., x_{pm+r-1}]^T$. The set $\mathcal{X}^k$ indicates the k-th Cartesian product of $\mathcal{X}$.

2.4 Vector Noise Predictive Viterbi Algorithm

2.4.1 Colored Noise

We derive the VNP Viterbi algorithm for a multivariate moving average noise model defined as

$$\mathbf{n}_k = \Theta_q \mathbf{w}_{k-q} + ... + \Theta_0 \mathbf{w}_k, \tag{2.12}$$

for some q, where Θ_i's are known $m \times m$ matrices and the $\mathbf{w}_i$'s are $m \times 1$ white Gaussian noise vectors. The noise is a discrete time stationary process and its covariance matrix has a block Toeplitz structure [35]. We let $\underline{\mathrm{T}}$, the covariance matrix of Gaussian noise, be expressed in the following block Toeplitz form $\underline{\mathrm{T}} = \left[B_1, ..., B_p; B_2^T, ..., B_p^T\right]$, where the B_i's are $m \times m$ matrices. The covariance matrix is defined as the expectation of noise terms in (2.12) and matrices B_i's are defined as $B_i = E[\mathbf{n}_k \mathbf{n}_{k-i+1}]$ and directly calculated from Θ_i's. Since the covariance matrix is a Toeplitz matrix, its inverse can be written as [36]

$$\underline{\mathrm{T}}^{-1} = \underline{\mathrm{L}}_1^T \underline{\mathrm{D}}^{-1} \underline{\mathrm{L}}_1 - \underline{\mathrm{L}}_2^T \underline{\mathrm{D}}^{-1} \underline{\mathrm{L}}_2, \tag{2.13}$$

with block diagonal $\underline{\mathrm{D}} = diag\left[D_p, ..., D_p\right]$ and upper block triangular matrices $\underline{\mathrm{L}}_1^T = [I, R_2, ..., R_p; \mathbf{0}, ..., \mathbf{0}]$, and $\underline{\mathrm{L}}_2^T = [\mathbf{0}, JR_p, ..., JR_2; \mathbf{0}, ..., \mathbf{0}]$, where $R_2, ..., R_p, D_p$ are calculated offline directly from the B_i's, and matrix J is the anti-identity matrix. This factorization is a variation of the celebrated Gohberg-Semencul decomposition [37]. The derivation is given in Appendix A.

Without loss of generality, let the monic polynomial $\Theta(z) = I + \Theta_0^{-1}\Theta_1 z^{-1} + ... + \Theta_0^{-1}\Theta_q z^{-q}$ be the description of a multivariate moving average filter where z is a complex value. As long as the zeros of $\det \Theta(z)$ lie strictly inside the unit circle, the whitening filter matrices R_i's norms decay exponentially with respect to i [38].

Recall that the R_i's are in reverse order in the definition of $\underline{\mathrm{L}}_2$. Under the aforementioned circumstances, one can show that $\underline{\mathrm{T}}^{-1}$ and $\underline{\mathrm{L}}_1^T \left[\underline{\mathrm{D}}^{-1}\right] \underline{\mathrm{L}}_1$ are asymptotically equivalent. Proof is given in Appendix B. Hence, we use the $\underline{\mathrm{L}}_1^T \left[\underline{\mathrm{D}}^{-1}\right] \underline{\mathrm{L}}_1$ as an approximation of the inverse of the covariance matrix. The rapid decrease in norm of the R_i's guarantees that the approximation has a negligible effect on the detection outcome. Similar approximations for the likelihood function are used in [35].

In order to derive the likelihood function, let us use the approximation of the inverse of the covariance matrix to express the conditional distribution in (3.4) as

$$f(\underline{\mathbf{y}}|\underline{\mathbf{x}}) \simeq (2\pi)^{-(pm+r)/2} \det(\underline{\mathrm{T}})^{-1/2} \exp\left\{-\frac{1}{2}\left(\underline{\mathbf{y}} - \underline{\mathrm{A}}\underline{\mathbf{x}}\right)^T \left[\underline{\mathrm{L}}_1^T \underline{\mathrm{D}}^{-1} \underline{\mathrm{L}}_1\right] \left(\underline{\mathbf{y}} - \underline{\mathrm{A}}\underline{\mathbf{x}}\right)\right\}. \tag{2.14}$$

The logarithm is a strict monotonic function, which allows us to replace the objective function in (3.4) with the minimization of the following

$$-2\ln f(\underline{\mathbf{y}}|\underline{\mathbf{x}}) \simeq (pm+r)\ln(2\pi) + \ln\det(T) + \left(\underline{\mathrm{y}} - \underline{\mathrm{A}}\underline{\mathbf{x}}\right)^T \underline{\mathrm{L}}_1^T \left[\underline{\mathrm{D}}^{-1}\right] \underline{\mathrm{L}}_1 \left(\underline{\mathrm{y}} - \underline{\mathrm{A}}\underline{\mathbf{x}}\right). \tag{2.15}$$

In our calculation of the branch metrics, we skip the first term since the constant term does not change the outcome of the minimization. The determinant in (2.15) does not depend on $\underline{\mathbf{x}}$ and therefore the second term is a constant too. We write the branch metrics as follows: Given $\underline{\mathrm{L}}_1^T \left[\underline{\mathrm{D}}^{-1}\right] \underline{\mathrm{L}}_1$, we simply multiply the noise vector $[\underline{\mathrm{y}} - \underline{\mathrm{A}}\underline{\mathbf{x}}]^T$ by the upper triangular matrix $\underline{\mathrm{L}}_1^T$ in (2.15). Since the inverse of matrix $\underline{\mathrm{D}}$ is block diagonal, the sum of the branch metrics is given by

$$\sum_{k=1}^{p} \left((\mathbf{y}_k - A\mathbf{x}_k) + \hat{\mathbf{n}}_k \right)^T D_p^{-1} \left((\mathbf{y}_k - A\mathbf{x}_k) + \hat{\mathbf{n}}_k \right), \tag{2.16}$$

and the predicted noise term $\hat{\mathbf{n}}_k$ in the branch metric at time k is

$$\hat{\mathbf{n}}_k = \sum_{j=1}^{k-1} R_{j+1}^T \left(\mathbf{y}_{k-j} - A\mathbf{x}_{k-j} \right). \tag{2.17}$$

This formulation dictates the following changes to the trellis diagram implementation: i) there are 2^m branches leaving/entering each state, and the number of states remains 2^r (for $r \geq m$); ii) there is a possibility of some branches overlapping. The prediction term $\hat{\mathbf{n}}_k$ is estimated by the tentative decisions on the trellis diagram (see [33] and [39] for details.) The VNP Viterbi algorithm is sub-optimal since the decisions are made based on a finite number of states. In practice, we approximate the sum in (2.17) by its first few terms: the iterator j starts from $\max\{k-l, 1\}$ for a fixed number l, which is the number of predictor taps.

2.4.2 Pattern Dependent Colored Noise

We would like to extend our results to signal dependent noise. Unfortunately, the problem does not conform to a symmetric Toeplitz structure and therefore we are unable to use the Gohberg-Semencul decomposition. In this section, we present a practical way of incorporating the noise predictor in the Viterbi algorithm for linear regressive noise with signal dependent coefficients. To do so, we present an alternative decomposition, which allows us to write the branch metrics. The data dependency can have different formulations. We let the noise model $\mathbf{n}_k$ in (2.10) be defined by its elements as

$$\mathbf{n}_k = \Theta_q(\underline{\mathrm{x}}_{k-q}^k)\mathbf{w}_{k-q} + ... + \Theta_0(\underline{\mathrm{x}}_{k-q}^k)\mathbf{w}_k, \tag{2.18}$$

where each matrix Θ_i depends on $\underline{\mathrm{x}}_{k-q}^{k}$ and $\mathbf{w}_i$ are white Gaussian noise vectors of size $m \times 1$.

First, we give a factorization of the inverse of the signal dependent covariance matrix. Based on this factorization, we derive a signal dependent VNP Viterbi algorithm. Let $\underline{\mathrm{T}}$ be a block matrix with its LDU decomposition [36]

$$\underline{\mathrm{L}} = \begin{bmatrix} L_{1,1} & & & & \\ L_{2,2} & L_{1,2} & & & \\ \vdots & L_{2,3} & \ddots & & \\ L_{q,q} & \vdots & \ddots & & \\ & L_{q,q+1} & & & \\ & & & & L_{1,p} \end{bmatrix}, \tag{2.19}$$

$$\underline{\mathrm{U}} = \begin{bmatrix} U_{1,1} & U_{2,2} & \cdots U_{q,q} & & \\ & U_{1,2} & & \ddots & U_{q,p} \\ & & U_{1,3} & \ddots & \vdots \\ & & & \ddots & U_{2,p} \\ & & & & U_{1,p} \end{bmatrix}, \tag{2.20}$$

and $\underline{\mathrm{D}} = diag\left[\tilde{D}_1, ..., \tilde{D}_p\right]$. For $m \times m$ matrices $L_{i,j}$ and $U_{i,j}$, the first subscript i is an indicator of the distance from the matrix diagonal and second subscript is the row. Given that all leading principal minors $\tilde{D}_i, i = 1, ..., p$ are non-singular, we introduce a factorization for the signal dependent covariance matrix of a regressive model. Based on this factorization, the result in Section 2.4.1 is generalized for signal dependent noise. We write the block UDL decomposition of the inverse as in the following lemma.

Lemma 2.1. *A banded block matrix $\underline{T}$ with its UDL decomposition is invertible if*

the following equations are solvable

$$\sum_{k=1}^{i} R_k^{j+k-i} U_{i-k+1,j} = I\delta[i-1], i = 1, ..., j, j = 1, ..., p, \tag{2.21}$$

$$\sum_{k=1}^{i} L_{i-k+1,j} W_k^{j+k-i} = I\delta[i-1], i = 1, ..., j, j = 1, ..., p, \tag{2.22}$$

and $\tilde{D}_i$ *is invertible for all* $j = 1, ..., p$. *Furthermore, the inverse can be written as*

$$\underline{T}^{-1} = \begin{bmatrix} R_1^1 & R_2^2 & R_3^3 & \cdots & R_p^p \\ & R_1^2 & R_2^3 & & \\ & & R_1^3 & \ddots & \vdots \\ & & & \ddots & R_2^p \\ & & & & R_1^p \end{bmatrix} \underline{D}^{-1} \begin{bmatrix} W_1^1 & & & & \\ W_2^2 & W_1^2 & & & \\ & \ddots & \ddots & & \\ \vdots & & & W_1^{p-1} & \\ W_p^p & \cdots & & W_2^p & W_1^p \end{bmatrix}. \tag{2.23}$$

For $m \times m$ matrices R_i^j and W_i^j, the first subscript i is an indicator of the distance from the matrix diagonal and superscript is the row. The decomposition is simply a variation of the Cholesky decomposition. Proof is given in Appendix C.

We derive the branch metrics for $q = 1$, generalization to other values is straightforward. Let the linear regression covariance matrix of the pattern dependent noise be expressed as $\underline{\mathrm{C}}(\underline{\mathrm{x}}) = \underline{\mathrm{L}}(\underline{\mathrm{x}})\underline{\mathrm{D}}(\underline{\mathrm{x}})\underline{\mathrm{U}}(\underline{\mathrm{x}})$ where $\underline{\mathrm{D}}(\underline{\mathrm{x}}) = diag[C_0(\mathbf{x}_1), ..., C_0(\mathbf{x}_p)]$ and

$$\underline{\mathrm{L}}(\underline{\mathrm{x}}) = \begin{bmatrix} I & & & \\ L_{2,2}(\mathbf{x}_1, \mathbf{x}_2) & I & & \\ & & \ddots & \\ & & L_{2,p}(\mathbf{x}_{p-1}, \mathbf{x}_p) & I \end{bmatrix}, \tag{2.24}$$

$$\underline{\mathrm{U}}(\underline{\mathrm{x}}) = \begin{bmatrix} I & U_{2,2}(\mathbf{x}_1, \mathbf{x}_2) & & \\ & I & & \\ & & \ddots & U_{2,p}(\mathbf{x}_{p-1}, \mathbf{x}_p) \\ & & & I \end{bmatrix}, \tag{2.25}$$

where we assume $U_{1,i} = L_{1,i} = I$. To emphasize that the covariance sub-matrices depend on the input signal, we denote each as a function of vectors $\mathbf{x}_k$. Each element of the the covariance matrix of noise can be written as $E[\mathbf{n}_k \mathbf{n}_k^T] = \sum_{i=0}^{1} \Theta_i(\underline{\mathrm{x}}_{k-1}^k)\Theta_i(\underline{\mathrm{x}}_{k-1}^k)^T$ and $E[\mathbf{n}_k \mathbf{n}_{k-1}^T] = \Theta_1(\underline{\mathrm{x}}_{k-1}^k)\Theta_0(\underline{\mathrm{x}}_{k-2}^{k-1})^T$. The noise model $\mathbf{n}_k$ in (2.18) has an LDU decomposition in the form of (2.24) and (2.25). This can be seen as a block tridiagonal decomposition of the covariance matrix [40]. The matter is best explained by an example, as given below.

Once the noise covariance matrix is in LDU form, we solve (2.21) and (3.14) to find R, W's. We recall that the set of equations in (2.21) is in reduced echelon form. Therefore, from (2.21), for $i = 1$, we have $R_1^j = I$ for all $j = 1, ..., p$. We use the matrices R_1^j's to find R_2^j's. The dependency of R_2^j's on $\underline{\mathrm{x}}_{j-1}^j$ becomes apparent by setting $i = 2$, for $j = 2, ..., p$, i.e., $R_1^{j-1} U_{2,j}(\underline{\mathrm{x}}_{j-1}^j) + R_2^j = \mathbf{0}$. Iterating over the remaining equations we find out that R_k^j is dependent on $\underline{\mathrm{x}}_{j-k+1}^j$. The same result holds for W_j^k.

The sum of branch metrics in the VNP Viterbi algorithm (2.16) can be updated as

$$\sum_{k=1}^{p} \Lambda(\mathbf{y}_k, \underline{\mathrm{x}}_1^k) = \sum_{k=1}^{p} \Big[\ln \det C_0(\mathbf{x}_k) + \left((\mathbf{y}_k - \mathbf{A}_k \mathbf{x}_k) + \hat{\mathbf{n}}_\mathbf{k}\right)^T C_0(\mathbf{x}_k)^{-1} \left((\mathbf{y}_k - \mathbf{A}_k \mathbf{x}_k) + \tilde{\mathbf{n}}_\mathbf{k}\right)\Big], \tag{2.26}$$

since the determinants of $\underline{\mathrm{U}}$ and $\underline{\mathrm{L}}$ are both 1. The predicted noise becomes

$$\hat{\mathbf{n}}_k = \sum_{j=\max\{k-l,1\}}^{k-1} R_{j+1}^k (\underline{\mathrm{x}}_{k-j}^k)^T \left(\mathbf{y}_{k-j} - A\mathbf{x}_{k-j}\right), \tag{2.27}$$

$$\tilde{\mathbf{n}}_k = \sum_{j=\max\{k-l,1\}}^{k-1} W_{j+1}^k (\underline{\mathrm{x}}_{k-j}^k) \left(\mathbf{y}_{k-j} - A\mathbf{x}_{k-j}\right). \tag{2.28}$$

The predictor coefficients depend on the input data, and therefore the states are further extended to account for different choices of R and W. These matrices, which carry data dependent prediction parameters, are calculated offline.

Example 2.1. Assuming $\sigma_w^2 = 1$ and a (1-D) partial response channel, let the linear regressive model for noise be expressed as

$$n_i = b_0(x_{i-1}^i)w_i + b_1(x_{i-1}^i)w_{i-1}. \tag{2.29}$$

where $b_i(x_{i-1}^i)$'s are non-linear functions of input values. Then elements of the covariance matrix are $E[n_i n_i] = b_0^2(x_{i-1}^i) + b_1^2(x_{i-1}^i)$, $E[n_i n_{i-1}] = b_0(x_{i-2}^{i-1})b_1(x_{i-1}^i)$. In matrix block form we can write the Θ_i's as

$$\Theta_0 = \begin{bmatrix} b_0(x_{i-1}^i) & \\ b_1(x_i^{i+1}) & b_0(x_i^{i+1}) \end{bmatrix}, \Theta_1 = \begin{bmatrix} b_1(x_{i-1}^i) \\ \\ \end{bmatrix}. \tag{2.30}$$

Since the linear regression is of order two we pick $m = 2$. In the matrix form if we define matrix $\underline{\mathrm{L}}'(\underline{\mathrm{x}})$ as

$$\underline{\mathrm{L}}'(\underline{\mathrm{x}}) = \begin{bmatrix} b_0(x_0^1) & & & \\ b_1(x_1^2) & b_0(x_1^2) & & \\ & b_1(x_2^3) & \ddots & \\ & & b_1(x_{n-1}^n) & b_0(x_{n-1}^n) \end{bmatrix}, \tag{2.31}$$

the covariance matrix is

$$\underline{\mathrm{C}}(\underline{\mathrm{x}}) = E[\underline{\mathrm{n}}\underline{\mathrm{n}}^T] = \underline{\mathrm{L}}'(\underline{\mathrm{x}})E[\underline{\mathrm{w}}\underline{\mathrm{w}}^T]\underline{\mathrm{L}}'(\underline{\mathrm{x}})^T. \tag{2.32}$$

The first equality follows from the definition of covariance matrix and the second equality comes from the linearity of expectation.

In a block form, to get the $\underline{\mathrm{L}}(\underline{\mathrm{x}})$ in LDU decomposition, we simply multiply the $\underline{\mathrm{L}}'(\underline{\mathrm{x}})$ with the the inverse of $diag[B_0(\mathbf{x}_1), ..., B_0(\mathbf{x}_p)]$, where $B_0(\mathbf{x}_i)$ is defined as

$$B_0(\mathbf{x}_i) = \begin{bmatrix} b_0(x_{i-1}^i) & \\ b_1(x_i^{i+1}) & b_0(x_i^{i+1}) \end{bmatrix}. \tag{2.33}$$

Therefore, we can rewrite the LDU decomposition as

$$\underline{\mathrm{L}}(\underline{\mathrm{x}}) = \begin{bmatrix} I & & & & \\ L_{2,2}(\mathbf{x}_1, \mathbf{x}_2) & I & & & \\ & L_{2,3}(\mathbf{x}_2, \mathbf{x}_3) & I & & \\ & & \ddots & \ddots & \end{bmatrix}, \tag{2.34}$$

in which $L_{2,i}(\mathbf{x}_{i-1}, \mathbf{x}_i)$'s are as follows

$$L_{2,i}(\mathbf{x}_{i-1}, \mathbf{x}_i) = B_0^{-1}(\mathbf{x}_{i-1}) \begin{bmatrix} 0 & b_1(x_{i-1}^i) \\ 0 & 0 \end{bmatrix}, \tag{2.35}$$

and $C_0(\mathbf{x}_i) = B_0(\mathbf{x}_i)B_0(\mathbf{x}_i)^T$ and so on. We derive the R, W 's by solving (2.21), (3.14).

2.5 Simulations and Discussion

We evaluate the performance of the proposed algorithms with an additive Gaussian noise with known correlation.

First, we use a channel with ISI memory $r = 2$, with $\mathbf{a} = [1, 0.5, 0.2]$, and MA noise model with coefficients $\mathbf{b} = [1, 0.6, 0.4]$. The noise is of order $m = 3$. Let $\|\cdot\|$ indicate the norm, then the SNR is defined as the signal power, i.e., $\|\mathbf{a}\|^2$, over the noise variance $\sigma_n^2 = \sigma_w^2 \|\mathbf{b}\|^2$. We change the value of the noise variance, σ_w^2, to achieve different SNR's. We show the bit-error rate (BER) performance for different detectors in Fig. 2.2. The curve marked as Euclidean Viterbi represents the conventional Viterbi algorithm that computes the Euclidean distance of the received noise. This configuration does not consider any compensation for the noise correlation. We also show the noise predictive maximum likelihood (NPML) [33] performance. The trellis has four states for both algorithms, but the VNP algorithm requires $2^5 \times 14$ multiplications at each step while NPML only needs 2^4. Although the NPML algorithm uses the predicted noise to improve the performance, the error propagation degrades the result. The analysis of the phenomenon is given in [41]. The VNP algorithm label denotes the algorithm derived in (2.16) with l prediction taps. If we let $l = 0$, the algorithm reduces to Altekar and Wolf's [39]. For $l = 2$, we did not observe a significant improvement over the one tap predictor. At a BER of 10^{-3}, the proposed detector shows a 1 dB improvement over Altekar and Wolf's algorithm which is itself 1.25 dB away from the Viterbi detector. The matched filter lower bound for the probability of error is given by [42]

$$P_e \geq Q\left(\frac{d_{min}\,\|\mathbf{b}\|}{\sigma_n}\right), \tag{2.36}$$

where the function $Q(u)$ is the probability of a normal random variable being greater than u. The d_{min} is the adjusted minimum distance for the channel response filtered through the whitening filter.

To test the pattern dependent VNP algorithm, we used a channel with $r =$

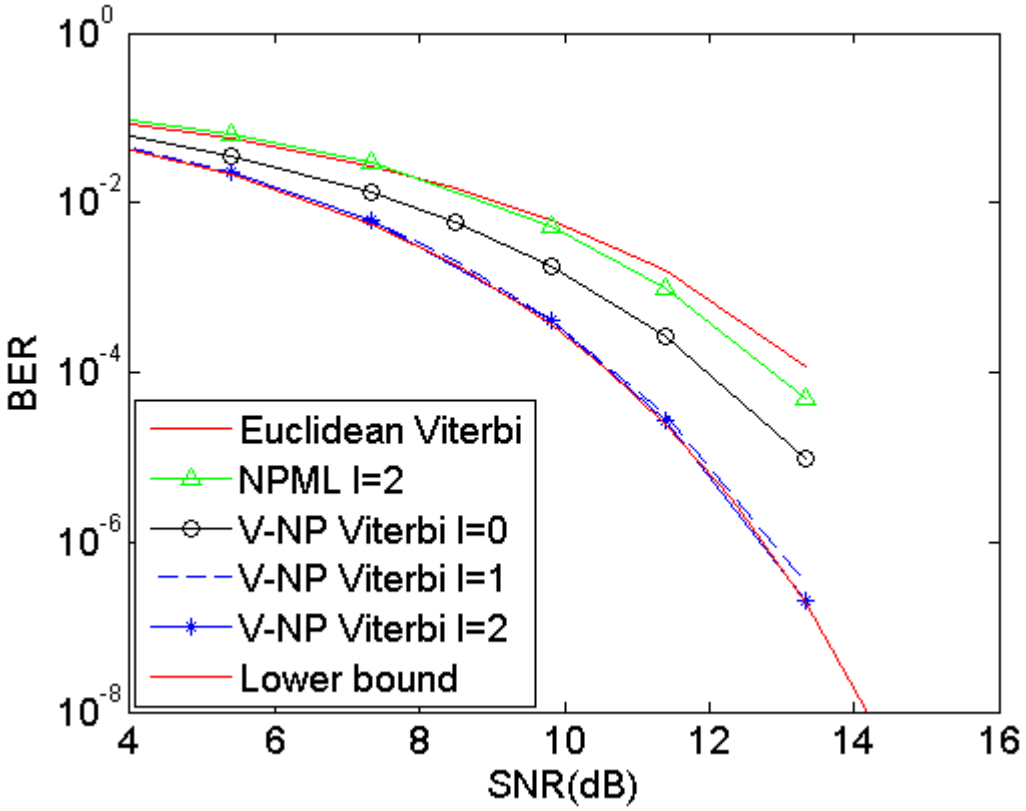


Figure 2.2: Noise predictive Viterbi algorithms for an ISI channel with MA noise.

1, namely, $\mathbf{a} = [1, -1]$, and the parameters that are given in Table 2.1. Noise power changes over time depending on which data has been transmitted: $\sigma_n^2(\mathbf{x}) = \sigma_w^2 \|\mathbf{b}(\mathbf{x})\|^2$ for each row in Table 2.1. Since we assume the input bits are equiprobable, the SNR is the signal power over the average of these noise powers, σ_n^2. The known correlation coefficients generate the prediction block matrices R and W which have been computed off-line. In all noise predictive Viterbi algorithms, the predictor was restricted to one tap, $l = 1$. Since the regressive noise is of order two, $m = 2$, we build the trellis with 8 states at each step with 4 branches leaving/entering each state. The branch metrics are as given in (2.26) with one tap predictor, $l = 1$. The detector uses 64 additions and 32×5 vector multiplications at a time. The performance of the detector is compared to a pattern dependent

noise predictive maximum likelihood (PDNP-ML) algorithm described in [20]. The PDNP-ML algorithm has the same number of states on the trellis diagram and 32 multiplication at each stage. The noise parameter given here does not match a Markovian description, but, for the purpose of comparison, we use the output of the channel to estimate the parameters of a pattern dependent Markov noise model of order 3. In our simulations the VNP algorithm performed as well as or better than the PDNP-ML algorithm depending on the noise characteristics. For the channel given in Table 2.1 we show the results in Fig. 2.3. At 10^{-3}, the proposed detector shows almost 1 dB improvement over the Viterbi algorithm and 0.5 dB improvement over the PDNP-ML algorithm. The poor performance of the PDNP-ML algorithm is mostly rooted in the Markovian assumption: the mismatch between the actual noise and the Markov model degrades its performance.

A genie aided detector that knows *a-priori* the two possible candidates for the transmitted signal provides a lower bound for our detector. Assume one candidate is the transmitted signal $\underline{\mathrm{x}}$ and the other candidate has a single bit error. For each transmitted signal $\underline{\mathrm{x}}$, we determine the minimum distance. That is to find the errors that are most probable. The derivation of d_{min} is explained in [43],[20]. We scale d_{min} by $\|\mathbf{b}(\mathbf{x})\|$ for which the erroneous bit has occurred. Once we find d_{min} for each transmitted signal then the probability of error is bounded by [44]

$$P_e \geq \sum_{\underline{\mathrm{x}}} P(\underline{\mathrm{x}}) Q\left(\frac{d_{min}(\underline{\mathbf{x}})\,\|\mathbf{b}(\mathbf{x})\|}{\sigma_n}\right). \tag{2.37}$$

We have kept the length of vector $\underline{\mathrm{x}}$ small enough to be able to run the simulations. The comparison shows our algorithm is less effective when larger number of errors are present at lower SNRs. A more elaborate error analysis, although possible, is beyond the scope of this dissertation.

Table 2.1: Partial response channel with pattern dependent regressive Gaussian correlated noise with known coefficients **b**.

x	**y**	**b(x)**
00	0.0	[-0.5,1.2]
01	2.0	[0.5,1.0]
10	-2.0	[0.2,1.0]
11	0.0	[-0.5,1.2]

Appendices

A. Derivation of (2.13)

We designate a column or row of matrices in uppercase bold. The $m \times m$ matrices $R_2, ..., R_p$ and $D_1, ..., D_p$ can be retrieved from $B_1, ..., B_p$ through a recursive Trench-like algorithm [44]. Let the anti-identity matrices J and $\underline{\mathrm{J}}_k$, i.e., matrices with ones on the diagonal connecting opposite vertices of leading diagonal, be of size $m \times m$ and $(k-1)m \times (k-1)m$, respectively. The block column matrix $\mathbf{\Gamma}_k = col[B_2, ..., B_k]_{km \times m}$ consists of the input elements up to step k. We want to show that the last column $\mathbf{R}_k = col[R_2, ..., R_k]$ updates the next level $\mathbf{R}_{k+1}$ through

$$\begin{bmatrix} R_2 \\ \vdots \\ R_k \end{bmatrix} = \begin{bmatrix} \mathbf{R}_k + \underline{\mathrm{J}}_k \mathbf{R}_k diag\left[D_k^{-1}(\mathbf{R}_k^T \mathbf{\Gamma}_k + B_{k+1})\right] \\ J \times D_k^{-1}(\mathbf{R}_k^T \mathbf{\Gamma}_k + B_{k+1}) \end{bmatrix}, \tag{2.38}$$

and the diagonal elements at step k are $D_k = B_1 + [R_2^T, ..., R_k^T]\underline{\mathrm{J}}_k \mathbf{\Gamma}_k$. We derive (2.38) by following the steps in [45]: suppose the LDL decomposition of $\underline{\mathrm{T}}_{k+1} = \left[B_1, ..., B_k; B_2^T, ..., B_k^T\right]$ is written as

$$\underline{\mathrm{T}}_{k+1} = \begin{bmatrix} I & \\ -\mathbf{R}_k^T & I \end{bmatrix} \begin{bmatrix} \underline{\mathrm{T}}_k & \\ & D_k \end{bmatrix} \begin{bmatrix} I & -\mathbf{R}_k \\ & I \end{bmatrix}, \tag{2.39}$$

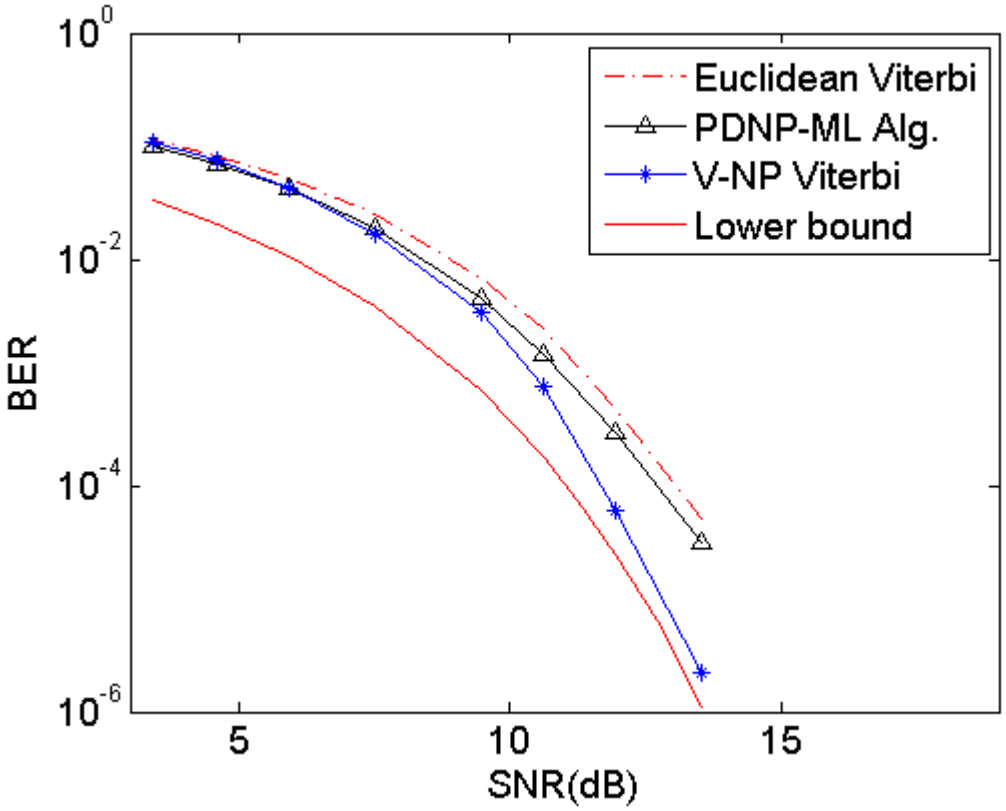


Figure 2.3: Pattern dependent Viterbi algorithms for an ISI channel with signal dependent regressive noise.

which reveals the following decomposition for the inverse of $\underline{\mathrm{T}}_{k+1}$ as

$$\begin{bmatrix} I & \mathbf{R}_k \\ & I \end{bmatrix} \begin{bmatrix} \underline{\mathrm{T}}_k^{-1} & \\ & D_k^{-1} \end{bmatrix} \begin{bmatrix} I & \\ \mathbf{R}_k^T & I \end{bmatrix} = \underline{\mathrm{T}}_{k+1}^{-1}, \tag{2.40}$$

since the product of (5.7) and (5.8) is the identity.

Solving the equality in (5.7) for D_k and $\mathbf{\Gamma}_k$, we have $-\underline{\mathrm{T}}_k\mathbf{R}_k = \underline{\mathrm{J}}_k\mathbf{\Gamma}_k$, and $D_k = B_1 + \mathbf{R}_k^T\underline{\mathrm{J}}_k\mathbf{\Gamma}_k$. Let us rewrite the equation for the next iteration $k+1$; $\mathbf{R}_{k+1} = -\underline{\mathrm{T}}_{k+1}^{-1}\underline{\mathrm{J}}_k\mathbf{\Gamma}_{k+1} = -\underline{\mathrm{J}}_k\underline{\mathrm{T}}_{k+1}^{-1}\mathbf{\Gamma}_{k+1}$ where we used the per-symmetric property for the second equality [45]. By substituting $\underline{\mathrm{T}}_{k+1}^{-1}$ with the left-hand side of (5.8) we get (2.38).

B. Proof of asymptotic equivalence

The Frobenius norm is denoted as $\|.\|_F$, and $|A|$ indicates $\frac{1}{\sqrt{n}}\|A\|_F$ for matrix A of size $n \times n$. To show that the two matrices are asymptotically equivalent we need to show that $\lim_{p\to\infty} \left|\underline{\mathrm{L}}_2\underline{\mathrm{D}}^{-1}\underline{\mathrm{L}}_2\right| = 0$.

We know that we can bound the norm by the product of the norms of each matrix [46]

$$\left|\underline{\mathrm{L}}_2^T\underline{\mathrm{D}}\underline{\mathrm{L}}_2\right| \leq \left\|\underline{\mathrm{L}}_2^T\right\|_2 |\underline{\mathrm{D}}\underline{\mathrm{L}}_2| \leq \|\underline{\mathrm{L}}_2\|_2 \|\underline{\mathrm{D}}\|_2 |\underline{\mathrm{L}}_2| . \tag{2.41}$$

The norm is bounded below and therefore we only need to show that $\lim_{p\to\infty} |\underline{\mathrm{L}}_2| = 0$. In addition, the Frobenius norm of a block matrix can be stated in terms of the norm of each block

$$\|\underline{\mathrm{L}}_2\|_F^2 = \sum_{i,j=1}^{p} \left\|L_2^{i,j}\right\|_F^2 = \left(\sum_{i=1}^{p-1} i \left\|R_{i+1}\right\|_F^2\right), \tag{2.42}$$

$$|\underline{\mathrm{L}}_2|^2 = \frac{1}{pm+r}\sum_{i,j=1}^{pm+r} a_{i,j}^2 = \frac{1}{pm+r}\sum_{i=1}^{p-1} i \left\|R_{i+1}\right\|_F^2 . \tag{2.43}$$

As expressed in [38] if the zeros of $\det\Theta(z)$ are strictly inside the unit circle, the R_i's decay exponentially and therefore

$$\lim_{i\to\infty} i \left\|R_{i+1}\right\|_F^2 = 0. \tag{2.44}$$

Then, by the Cesaro mean convergence theorem [47], we have

$$\lim_{p\to\infty} |\underline{\mathrm{L}}_2|^2 = 0. \tag{2.45}$$

C. Proof of Lemma 2.1

Assume (14) is solvable, we want to show $Ker(\underline{\mathrm{U}}) = \{\varphi | \underline{\mathrm{U}}\varphi = \mathbf{0}\}$ contains only the zero element. Suppose $\underline{\mathrm{U}}\varphi = 0$. From (14), for $j = k$ we have

$$[..., R_1^k, ..., R_{p-k+1}^p] \begin{bmatrix} U_{1,1} & \cdots U_{l,l} & & \\ & U_{1,2} & \ddots & U_{l,p} \\ & & \ddots & \vdots \\ & & & U_{1,p} \end{bmatrix} = \begin{bmatrix} & \\ & \\ I & \\ & \\ & \end{bmatrix}^T, \tag{2.46}$$

where the identity matrix in (2.46) sits on the k-th block row. In addition, we know that

$$\mathbf{0} = [0, .., R_1^k, R_2^{k+1} ..., R_{p-k+1}^p]\underline{\mathrm{U}}\varphi = \begin{bmatrix} & \\ & \\ I & \\ & \\ & \end{bmatrix}^T \varphi. \tag{2.47}$$

Since (2.47) is true for all $k = 1, ..., p$, therefore $\varphi = \mathbf{0}$. Similarly, given (15) is solvable, we can show that $\underline{\mathrm{L}}$ is invertible. Elements of $\underline{\mathrm{D}}$ are each invertible and this proves that matrix $\underline{\mathrm{T}}$ is invertible. From (2.47) and the fact that $\underline{\mathrm{U}}$ is upper triangular we deduce $\underline{\mathrm{U}}^{-1}$ is the juxtaposition of all matrices $[0, .., R_1^k, R_2^{k+1} \cdots, R_{p-k+1}^p]$, for $k = 1, ..., p$. Equation (16) is the result of multiplying the inverse of each matrix in the opposite direction.

3 Sequence Detection in the Presence of Jitter Nuisance Parameter

3.1 Introduction

This chapter presents a sequence detector in the presence of a random nuisance parameter with a continuous distribution. The nuisance parameter that caused by deviation or displacement of signal pulses known as jitter affects the performance of the communication system. Conventionally, the effect of jitter is considered an extra source of noise in the system. In the previous chapter, we discussed the sequence detectors that handle colored noise. Here, we opt for a different approach for jitter contaminated channel. Due to jitter, the channel response is random in nature. In this context several papers have observed a benefit of oversampling empirically. In magnetic recording, Victora *et al.* [48, 49] found that twice sampling will result in better detection and decoding performance. Using a first order Taylor approximation of pulse response, Pighi *et al.* [50] proposed a linear predictive algorithm which requires a bank of filters to sample the received signal. Our approach in this chapter does not rely on approximation of channel response. The formulation given enable

us to work out a sequence detector that handle the random jitter noise present at the channel response directly. It is worthy to note the jitter is a common issue in many other applications, such as optical recording [51], Ethernet [52], wireless communication [53].

Our first contribution is the derivation of a Viterbi like detector for sequence detection in presence of random jitter. We propose the layout of over-sampling communication system, and derive a detection algorithm based on the Gauss-Markov assumption. We also derive a genie aided bound for the detector. Our second contribution is to study our detector in the context of faster-than-Nyquist signaling [54]. We found that the nuisance parameter is closely related to the concept of faster-than-Nyquist signaling. In the presence of jitter nuisance parameter the amount of ISI changes for each symbol despite the fixed transmission rate. The faster-than-Nyquist signaling achieve only 20% percent increase in transmission rate and in many applications that is not enough to justify the excess complexity. In contrast, in physical channels where noise is described by a nuisance parameter the gain in performance can be significant. Our aim here is to show the benefit of exploiting the nuisance parameter in boosting the performance of the system.

In this chapter, it is first shown that observations twice the number of transmitted symbols are the minimum number of points required for detection. To collect these samples we need sampling kernels. We have discussed different sampling kernels. These sampling kernels are particularly important in the sense that the subspace spanned by sampling kernels confines the search region which contains the solution. Our take on nuisance parameter estimation techniques is explained briefly for Gaussian shaped pulses. The derivation of the nonlinear optimum detector is highly complex and computationally prohibitive. Therefore, a Viterbi-like sequence detector is proposed. Finally, the BER bounds, decision boundaries and minimum

distance are discussed and the simulation results for the detector are presented.

3.2 Formulation of the Problem

We used the following notation throughout the chapter. We use uppercase letters for random variables and lower case for their realization. Matrices are shown in underlined bold uppercase letters and vectors are designated either in bold lowercase letters, e.g., $\mathbf{x}$, or through enumeration of their elements, $\mathbf{x}_1^K = [x_1, x_2, ...x_K]$ where x_i is the i-th element of vector $\mathbf{x}$. Given a map A, we show the restriction of a map to a subset S of its original domain by $A|_S$. For brevity, we later use $p_k(t)$ as a $k-$translation of $p(t)$, i.e., $p(t - kT)$. Symbols $\|.\|_2, \|.\|_{\mathcal{H}}$ represent the Euclidean and L^2 norm. The norm induced by matrix $\mathbf{C}$ is indicated by$\|\mathbf{x}\|_{\mathbf{C}} = \sqrt{\mathbf{x}^T\mathbf{C}\mathbf{x}}$, where $\mathbf{x}^T$ is the transpose of vector $\mathbf{x}$.

Consider the signal at the receiver after fixed finite K successive transmissions as being given by

$$y(t) = \sum_{k=1}^{K} x_k p(t - kT + j_k) + n(t), \tag{3.1}$$

where $p(t)$ is the pulse response of the channel where each binary information input $X_k \in \{-1, 1\}$ has a Bernoulli distribution. We assume the additive noise term is AWGN with auto-correlation function $\frac{N_0}{2}\delta(t)$. Random variables J_k are the displacement of the channel response with respect to the sampler and are restricted to half the sampling interval, $J_k \in (-\frac{T}{2}, \frac{T}{2})$.

A single pulse $p(t + j_k)$ along with a sequence of modulated signals is depicted in Fig. 3.1. The solid line shows the dislocated Gaussian-shaped pulse and dash lines are Gaussian-shaped pulses without jitter. The dashed lines also represent the position of three matched filters sampling the incoming signals. The jitter nuisance

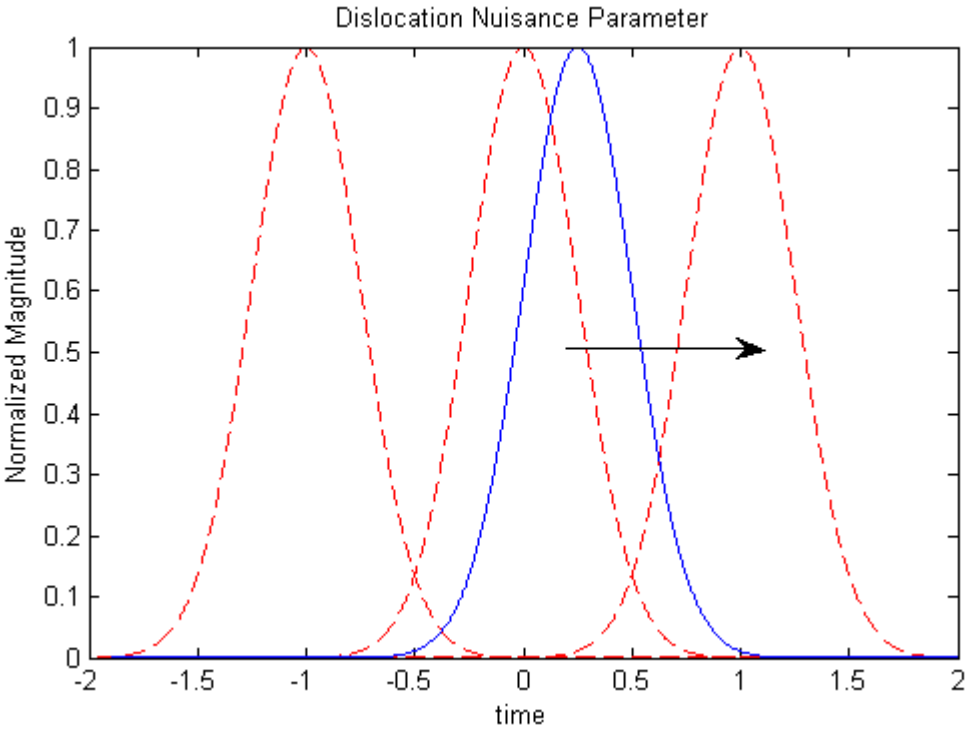


Figure 3.1: Sequence of modulated signals (dashed line) and a randomly dislocated response (solid line) for Gaussian-shaped pulses.

parameter shifts the channel response from the matched filter centered at the origin. In this paper, we only consider the jitter nuisance parameter on the abscissa axis, which is the dislocation of the channel response from its origin.

Our problem is to find the maximum likelihood (ML) estimate of vector $\mathbf{x}_1^K$ given the observation $y(t)$. In the absence of jitter noise, (3.1) can be fully expressed in a subspace of a Hilbert space with an orthonormal basis $\{\varphi_i(t)\}$, which can be generated through the Gram-Schmidt algorithm. But in the presence of jitter noise, the orthogonalization is not as efficient in the sense that there is no finite set of $\varphi_i(t)$'s that would span the whole signal subspace.

3.2.1 Sufficient Statistic

Let signal $s(t)$ be the received signal excluding the additive noise term. Suppose an invertible transform $\mathbb{T}$ exists such that $\mathbb{T}$ maps the signal into its representation as follows

$$\sum_{k=1}^{K} x_k p(t - kT + j_k) \overset{\mathbb{T}}{\mapsto} \sum_{i=1}^{N} \hat{x}_i(\mathbf{j}_1^K, \mathbf{x}_1^K) q(t - iT'), \tag{3.2}$$

for some $N \in \mathbb{N}$. Let us name $q(t)$ which belongs to Hilbert space a sampling kernel. Note that the sampling rate, i.e., $1/T'$, is not the same as the transmission rate. With this setup, the jitter noise will be implicit in $\hat{x}_i(\mathbf{j}_1^K, \mathbf{x}_1^K) \in \mathbb{R}$. The purpose of the transform is to confine the search region. This is done by expressing the signal in union of subspaces defined as follows [55];

Definition 3.1. Let $\mathcal{H}$ be a Hilbert space. The signal $s(t) \in \mathcal{H}$ is in a union of subspaces

$$\chi = \bigcup_{(j_1,...,j_K) \in \mathbb{R}^K} S_{(j_1,...,j_K)}, \tag{3.3}$$

if and only if there is some fixed $(j_1, ..., j_K) \in (-\frac{T}{2}, \frac{T}{2})^K$ such that $s(t) \in S_{(j_1,...,j_K)}$. The $S_{(j_1,...,j_K)}$'s are subspaces of $\mathcal{H}$ spanned by orthogonal functions $\delta_k(j_k)$'s.

Note in the definition each subspace has a finite basis, but the union is over infinite set. The choice of Dirac delta functions in our definition of union of subspaces is necessary since the index set for jitter elements cover the whole period $(-\frac{T}{2}, \frac{T}{2})$. Next, we express our signal in union of subspaces. Eq. (3.1) can be adapted to the definition as follows: channel is time invariant, and the $p_k(t)$'s are k-translations of the same pulse response $p(t)$ and therefore we may as well assume that the sampled points are given by (for $i = 1, ..., N$)

$$s_i = \sum_{k=1}^{K} x_k \langle p_k(t + j_k), q_i(t) \rangle = \sum_{k=1}^{K} x_k \left\langle \delta_k(t + j_k), \left(p(\tau) * \overline{q_i(-\tau)} \right)(t) \right\rangle, \qquad (3.4)$$

where $*$ stands for convolution of two functions, $\langle \cdot, \cdot \rangle$ is the inner product, and $\overline{q(t)}$ is the complex conjugate of $q(t)$. Eq. (3.4) shows that our signal $s(t)$ belongs to a union of subspaces. On the other hand, we achieve our representation in (3.2) by replacing $\hat{x}_i(\mathbf{j}_1^K, \mathbf{x}_1^K) = \sum_{k=1}^{K} x_k \langle p_k(t + j_k), q_i(t) \rangle$, where the coefficients $\hat{x}_i$'s absorb the effect of the jitters. Next step is to show transform $\mathbb{T}$ is invertible. Since our signal belongs to union of subspaces, we need to show that transform $\mathbb{T}$ is invertible in the union of subspaces. The following lemma considered here has been proved in [55].

Lemma 3.1. *Given a linear transformation $\mathbb{T} : \chi \to \mathbb{T}(\chi) \subseteq \mathbb{R}^N$, $\mathbb{T}$ is invertible for any $\tilde{s} \in \chi$ if and only if $\mathbb{T}|_{S_\gamma + S_\theta} : S_\gamma + S_\theta \to \mathbb{T}|_{S_\gamma + S_\theta}(S_\gamma + S_\theta)$ is invertible for any $\gamma = (j_1, ..j_K) \in (-\frac{T}{2}, \frac{T}{2})^K, \theta = (j_1', ..., j_K') \in (-\frac{T}{2}, \frac{T}{2})^K$ where $S_\gamma + S_\theta = \{\alpha \tilde{s}_1 + \beta \tilde{s}_2 | \tilde{s}_1 \in S_\gamma, \tilde{s}_2 \in S_\theta, \alpha, \beta \in \mathbb{R}\}$.*

Proof. Proof is given in Appendix A. □

This lemma simply says that if the transform is invertible for any $2K$ delta functions, then it is invertible for χ. The immediate corollary of the lemma is that if we let $\bar{q} = \{\bar{q}_n(t)\}_{n=1}^N$ be a set of sampling kernels and $\Phi_{\gamma,\theta} = \{\delta(t + j_k)\}_{k=1}^{2K}$ be a basis for $S_\gamma + S_\theta$, then $\bar{q}$ provides an invertible sampling operator for χ if and only if the Gram matrix $\underline{G}_{\gamma,\theta}$ defined as

$$\mathbf{G}_{\gamma,\theta} \triangleq \begin{bmatrix} \langle \delta(t+j_1), \bar{q}_1(t) \rangle & \langle \delta(t+j_2), \bar{q}_1(t) \rangle & \dots & \langle \delta(t+j_{2K}), \bar{q}_1(t) \rangle \\ \langle \delta(t+j_1), \bar{q}_2(t) \rangle & \langle \delta(t+j_2), \bar{q}_2(t) \rangle & \dots & \langle \delta(t+j_{2K}), \bar{q}_2(t) \rangle \\ \vdots & \vdots & \ddots & \vdots \\ \langle \delta(t+j_1), \bar{q}_N(t) \rangle & \langle \delta(t+j_2), \bar{q}_N(t) \rangle & \dots & \langle \delta(t+j_{2K}), \bar{q}_N(t) \rangle \end{bmatrix}_{N \times 2K}, \tag{3.5}$$

has full column rank for every $\gamma = (j_1, ..j_K) \in \mathbb{R}^K, \theta = (j_1^{'}, ..., j_K^{'}) \in \mathbb{R}^K$, because $\mathbf{G}_{\gamma,\theta}$ is obtained by a change of basis between two subspaces. Note that the smallest number for N so that the Gram matrix has full column rank is twice the number of transmissions, i.e., $N \geq 2K$. Therefore, we keep N as a fixed number greater than $2K$ for the rest of the paper. The easiest way to build such a matrix is to use the Vandermonde matrix with Dirac delta functions as our orthogonal basis $\{\delta(t+j_i)\}$ which has expressed in [56].

Example 3.1. Assume that $p(t) = \delta(t)$ in (3.1). Then the Gram matrix can be expressed as

$$\mathbf{G}_{\gamma} = \begin{bmatrix} q_1(j_1) & \cdots & q_N(j_1) \\ \vdots & \ddots & \vdots \\ q_1(j_{2K}) & \cdots & q_N(j_{2K}) \end{bmatrix}. \tag{3.6}$$

For $q_n(t) = t^{n-1}$, the matrix $\mathbf{G}_{\gamma}$ for $\gamma = (j_1, ...j_{2K})$, which is the Vandermonde matrix, is an invertible matrix if $det(\mathbf{G}_{\gamma}) \neq 0$. Matrix $\mathbf{G}_{\gamma}$ has full rank for all $j_1 \neq j_2 \cdots \neq j_{2K}$. Any system with some $q(t)$ that has nonzero determinant for the entire set $\{j_i\}_{i=1}^{2K}$ such that $j_1 \neq j_2 \cdots \neq j_{2K}$ is called a Chebychev system [57].

Remark 3.1. To show that the transform $\mathbb{T}$ in (3.2) is invertible we substitute $\bar{q}(t) = \left(p_k(\tau) * \overline{q_n(-\tau)}\right)(t)$ and write the Gram matrix as in (3.5).

Due to the additive noise term, our received signal $y(t)$ does not belong to the union of subspaces in general, but we can show that the projection of $y(t)$ onto our sampling subspace is what really matters to the estimation problem.

Proposition 3.1. *Suppose $y(t)$ defined as in (3.1) belongs to a Hilbert space. Given an invertible map $\mathbb{T}$ defined by a set of sampling kernels $\{q_n(t)\}_{n=1}^N$, the samples present a sufficient statistic for the ML estimation of $\{X_k\}_{k=1}^K$.*

Proof. Proof is given in Appendix B. □

Similarly, one can show that the set of $\{\langle y, q_i \rangle\}_{i=1}^N$ is also a sufficient statistic for the jitter noise estimation.

3.3 Sampling Strategies

The set of sufficient statistic is not unique. The next step in the design of our detector is to choose the appropriate sampling kernels. The sampling kernels are chosen according to an additional criterion to fulfill the requirement of the particular application. In the absence of jitter noise, the conventional detector, the criterion is to maximize the SNR, and the sampling kernel is matched to the pulse response of the channel. But in the presence of jitter we know from Section 3.2, that the minimum number of sampling kernels must be at least twice the number of transmitted symbols.

In this section we discuss different approaches to choose the sampling kernels. Since uniform sampling is desired, changing the sampling period to $T/2$ is one way of building the $2K$ sampling kernels. Using two different sampling kernels q and $\tilde{q}$, which could be realized through two filter banks with sampling period T for each, may achieve the maximum SNR as long as the conditions discussed in

the Section 3.2 are satisfied. However, if the detection strategy is based on the practical implementation of detector, the uniform sampling kernels may suit better the purpose of sequence detection.

3.3.1 Sampling Kernels Matched to Pulse Response

Let us define the SNR as

$$SNR = \frac{E_J\left[\left(\int p(t+J)h(-t)dt\right)^2\right]}{E_N\left[\left(\int N(t)h(-t)dt\right)^2\right]} = \frac{E_J\left[\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} p(v+J)p(u+J)h(-u)h(-v)dudv\right]}{\frac{N_0}{2}\int\int h(-u)h(-v)dvdu}. \quad (3.7)$$

We want to choose the function $h(t)$ that maximizes the expected received signal power over noise power. From the generalized Hölder's inequality we have

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} |E_J[p(v+J)p(u+J)]h(-u)h(-v)dudv| \leq \left(\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} |E_J[p(v+j)p(u+j)]|^2 dudv\right)^{1/2} \times \left(\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} |h(-u)h(-v)|^2 dudv\right)^{1/2}, \quad (3.8)$$

and equality holds [58] if and only if there exist $q, \tilde{q} \in \mathcal{H}$ such that for some $\lambda_1, \lambda_2 \in \mathbb{R}$,

a) $E_J[p(u+J)p(v+J)] = q(u)\tilde{q}(v)$,

b)$|q(u)| = \lambda_1|h(-u)|$ and $|\tilde{q}(v)| = \lambda_2|h(-v)|$.

An example is given next to elaborate on the matter. The nuisance parameter has

a discrete distribution. This case is interesting since the resulting sampling kernels matched to the channel response agree with the intuition.

Example 3.2. Let $p(t) = e^{-t^2}$ and J be a discrete random variable with equal probability of taking values in$\{+\epsilon, -\epsilon\}$. To find the matched filter, for (a) we have $E_J[p(t+J)p(\tau+J)] = \frac{1}{2}[p(t+\epsilon)p(\tau+\epsilon) + p(t-\epsilon)p(\tau-\epsilon)]$. Suppose $q, \tilde{q}$ exist such that $\frac{1}{2}[p(t+\epsilon)p(\tau+\epsilon) + p(t-\epsilon)p(\tau-\epsilon)] = q(t)\tilde{q}(\tau)$, then

$$q(t)\tilde{q}(\tau) = \frac{1}{2}[e^{-(t+\epsilon)^2}e^{-(\tau+\epsilon)^2} + e^{-(t-\epsilon)^2}e^{-(\tau-\epsilon)^2}]. \tag{3.9}$$

Since $p(t)$ is symmetric

$$q(t)\tilde{q}(\tau) = \frac{1}{2}[e^{-(t+\epsilon)^2}e^{-(\tau-\epsilon)^2} + e^{-(t+\epsilon)^2}e^{-(\tau-\epsilon)^2}] = e^{-(t+\epsilon)^2}e^{-(\tau-\epsilon)^2},$$

and therefore the matched filters are $e^{-(t+\epsilon)^2}$ and $e^{-(t-\epsilon)^2}$.
It should be noted that it is not always possible to analytically find the exact matched filters.

3.3.2 Uniform Sampling

Uniform sampling is defined by collection of finite samples at sample points with a uniform distribution, i.e., equally spaced in time or space. This approach is particularly appealing to practical applications. In our setup, the sampling is done by multiplying the received signal by sampling kernels as shown in Fig. 3.2. Then integrated to form the inner product expressed in (3.4) for each sample.

For uniform sampling the two sampling kernels $q, \tilde{q}$ are chosen from the same function, $\tilde{q}$ is offset by $T/2$. The two similar sampling kernels eliminate the need for extra filter by speeding up the sampling to twice the baud rate. This can be achieved by higher rate sampler while keeping the matched filter intact. Here and

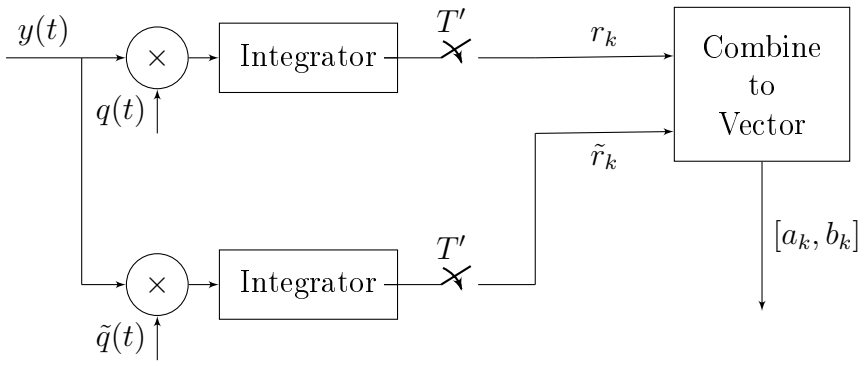


Figure 3.2: Over-sampling layout.

for the rest of the paper, by matched filter we mean the filter that is matched to the pulse response in absence of jitter.

Note that according to the requirement of Proposition 3.1 for the continuous signal at the receiver, the proposed setup is not the only possible scenario. For instance, the derivative of Gaussian function as second sampling kernel also satisfy the proposition. Using derivative of a Gaussian filter has already been used in [50]. But we have only considered uniform sampling in this paper.

3.4 Recovering Jitter Values

The pulse response and sampling kernels are required to have certain properties to enable us estimate the jitter noise in our framework. These properties have been discussed in previous sections. In this section, we further examine those for our pulse response of interest, the Gaussian shaped pulse response, but most of the material can be extended to any function that is a Chebychev system.

Let $p(t) = \mathcal{N}(t, \sigma_s^2) = (2\pi\sigma_s^2)^{-1/2} e^{-\frac{(t)^2}{2\sigma_s^2}}$ where $\mathcal{N}$ is the normal distribution. Choose the sampling kernel $q(t) = (2\pi\sigma_s^2)^{-1/2} e^{-\frac{t^2}{2\sigma_s^2}}$ matched to the pulse response. Solve the Fredholm integral equation of first kind

$$\int_{-\infty}^{\infty} \tilde{q}(t)\mathcal{N}(t-\tau,\sigma_s^2)dt = \tau\mathcal{N}(\tau,2\sigma_s^2), \tag{3.10}$$

to find the second sampling kernel. Solution to the above problem can be obtained by means of Weierstrass's integral equation [59]or an integral table as

$$\tilde{q}(t) = (2\pi\sigma_s^2)^{-1/2} t e^{-\frac{t^2}{2\sigma_s^2}}. \tag{3.11}$$

Therefore, the sampled data for each sampling kernel $q(t), \tilde{q}(t)$, for sampling points $n = T, ..., KT$, are expressed as

$$a_n = (4\pi\sigma_s^2)^{-1/2} \sum_{k=0}^{K} x_k e^{-\frac{(j_k-(n-k)T)^2}{4\sigma_s^2}},$$
$$b_n = (4\pi\sigma_s^2)^{-1/2} \sum_{k=0}^{K} x_k (j_k-(n-k)T) e^{-\frac{(j_k-(n-k)T)^2}{4\sigma_s^2}}. \tag{3.12}$$

Our aim is to show that the Gram matrix is invertible for any choice of jitter and further find an estimate of jitter values. Define $\tilde{a}_n = a_n e^{\frac{(nT)^2}{4\sigma_s^2}}$ and $\tilde{b}_n = b_n e^{\frac{(nT)^2}{4\sigma_s^2}}$ then

$$\tilde{a}_n = \sum_{k=1}^{K} x_k e^{-\frac{(j_k+kT)^2}{4\sigma_s^2}} e^{\frac{(j_k+kT)nT}{2\sigma_s^2}},$$
$$\tilde{b}_n - n\tilde{a}_n = \sum_{k=1}^{K} x_k (j_k+kT) e^{-\frac{(j_k+kT)^2}{4\sigma_s^2}} e^{\frac{(j_k+kT)nT}{2\sigma_s^2}}. \tag{3.13}$$

The Gram matrix as in (3.5), for the sampling kernels is given (3.14).

$$\begin{bmatrix} e^{-\frac{(j_1+T)^2}{4\sigma_s^2}} e^{\frac{(j_1+T)T}{2\sigma_s^2}} & (j_1'+T)e^{-\frac{(j_1'+T)^2}{4\sigma_s^2}} e^{\frac{(j_2+T)T}{2\sigma_s^2}} & \cdots & (j_K'+KT)e^{-\frac{(j_k'+KT)^2}{4\sigma_s^2}} e^{\frac{(j_k'+KT)T}{2\sigma_s^2}} \\ e^{-\frac{(j_1+T)^2}{4\sigma_s^2}} e^{\frac{(j_1+T)2T}{2\sigma_s^2}} & (j_1'+T)e^{-\frac{(j_1'+T)^2}{4\sigma_s^2}} e^{\frac{(j_k+T)2T}{2\sigma_s^2}} & \cdots & (j_K'+T)e^{-\frac{(j_k'+KT)^2}{4\sigma_s^2}} e^{\frac{(j_k'+KT)2T}{2\sigma_s^2}} \\ \vdots & & \ddots & \vdots \\ e^{-\frac{(j_1+T)^2}{4\sigma_s^2}} e^{\frac{(j_1+T)KT}{2\sigma_s^2}} & (j_1'+T)e^{-\frac{(j_1'+T)^2}{4\sigma_s^2}} e^{\frac{(j_k+T)KT}{2\sigma_s^2}} & \cdots & (j_K'+T)e^{-\frac{(j_k'+KT)^2}{4\sigma_s^2}} e^{\frac{(j_k'+KT)KT}{2\sigma_s^2}} \end{bmatrix}_{K\times 2K}. \tag{3.14}$$

The determinant of such a matrix is nonzero as long as $j_1+T \neq j_2+2T \neq ... \neq j_K+KT$ and $j_1'+T \neq j_2'+2T \neq ... \neq j_K'+KT$, or equivalent condition, $j_k, j_k' \in (-\frac{T}{2}, \frac{T}{2})$. We omit the proof as it is given in [57].

Next step is to find an explicit estimate of the jitter vector. To have an estimation of the jitter we need more than $4K$ samples. Let $a_n$and b_n be as defined in (3.12), but this time sampled at $n = \frac{T}{2}, T, \frac{3T}{2}, ...$ Build Hankel matrices H_0 and H_1 from $\tilde{a}_n$ and $\tilde{b}_n$'s sampled at $\frac{T}{2}$ as follows

$$\mathbf{H}_0 = \begin{bmatrix} \tilde{a}_1 & \tilde{a}_2 & \cdots & \tilde{a}_K \\ \tilde{a}_2 & \tilde{a}_3 & & \tilde{a}_{K+1} \\ \vdots & & \ddots & \\ \tilde{a}_K & \tilde{a}_{K+2} & \cdots & \tilde{a}_{2K-1} \end{bmatrix}_{K\times K}, \tag{3.15}$$

$$\mathbf{H}_1 = \begin{bmatrix} \tilde{b}_1 & \tilde{b}_2 & \cdots & \tilde{b}_K \\ \tilde{b}_2 & \tilde{b}_3 & & \tilde{b}_{K+1} \\ \vdots & & \ddots & \\ \tilde{b}_K & \tilde{b}_{K+2} & \cdots & \tilde{b}_{2K-1} \end{bmatrix}_{K\times K} \tag{3.16}$$

The Vandermonde decomposition of Hankel matrices can be written $H_i = VD_iV^T$, where V is the Vandermonde matrix of power functions of $e^{\frac{(j_k+T)nT}{2\sigma_s^2}}$,

$$\mathbf{V} = \begin{bmatrix} e^{\frac{(\tau_1+T)T}{4\sigma_s^2}} & e^{\frac{(\tau_2+2T)T}{4\sigma_s^2}} & \cdots & e^{\frac{(\tau_K-KT)T}{4\sigma_s^2}} \\ e^{\frac{(\tau_1+T)2T}{4\sigma_s^2}} & e^{\frac{(\tau_2+2T)2T}{4\sigma_s^2}} & & e^{\frac{(\tau_K-KT)2T}{4\sigma_s^2}} \\ \vdots & & \ddots & \vdots \\ e^{\frac{(\tau_1+T)KT}{4\sigma_s^2}} & e^{\frac{(\tau_2+2T)KT}{4\sigma_s^2}} & \cdots & e^{\frac{(\tau_K-KT)KT}{4\sigma_s^2}} \end{bmatrix}, \tag{3.17}$$

and $\mathbf{D}_0$ and $\mathbf{D}_1$ are the diagonal matrices filled with elements $e^{-\frac{(j_i+iT)^2}{4\sigma_s^2}}$ and $(j_i + iT)e^{-\frac{(j_i+iT)^2}{4\sigma_s^2}}$ for $i = 1, ..., K$, respectively. Since both $\mathbf{H}_0, \mathbf{H}_1$ are symmetric the generalized eigenvalues are real [36]. With this formulation, jitter values are the generalized eigenvalues of H_0, H_1,

$$\mathbf{H}_0\mathbf{v}_i = j_i\mathbf{H_1}\mathbf{v}_i, \tag{3.18}$$

where $\mathbf{v}_i$'s are the generalized eigenvectors.

Remark 3.2. Estimation of generalized eigenvalues of Hankel matrices is numerically unstable. The numerical solution becomes unstable since the Hankel matrices of higher size regardless of the values of the elements are ill conditioned. This is due to the nature of the Vandermonde matrices [11].

Remark 3.3. Since generalized eigenvalues of Hankel matrices $\mathbf{H}_0, \mathbf{H}_1$ is $\mathbf{j}$, our transform $\mathbb{T}$ is one-to-one and onto.

3.5 Detection Based on Gauss-Markov Assumption

In the presence of jitter, when signals are not T-orthogonal and transmitted symbols are interfering with each other, the collected samples are fed into a sequence detector. Derivation of an optimal sequence detector is complex in the presence of jitter. The computational complexity of such an optimal detector is also prohibitive. In

this section we use Gauss-Markov assumption for likelihood function to derive a suboptimal sequence detector for oversampled received signal. Let vectors $\boldsymbol{a}, \boldsymbol{b}$ be the matched filter output at T intervals starting from $0, 1/2T$. Since $\boldsymbol{a}, \boldsymbol{b}$ are two vectors which are a sufficient statistic for detection, then the ML detection of $\mathbf{x}$ is stated as

$$\hat{\mathbf{x}}_{ML} = \arg\max_{\boldsymbol{x}} f_{A,B|X}(\boldsymbol{a}, \boldsymbol{b}|\boldsymbol{x}), \tag{3.19}$$

where $f_{A,B|X}$ is the conditional distribution of received signals given the input values. The collected samples $\boldsymbol{a}, \boldsymbol{b}$ depend on jitter values and $f_{A,B|X}$ is only the marginal distribution. The marginal distribution $f_{A,B|X}$ does not have a closed form unless the distribution $f_{A,B,J|X}$ is jointly Gaussian. However, using the Gauss-Markov assumption, we can derive a sequence detection algorithm. This assumption results in a sub-optimal detector. The Viterbi algorithm for Gauss-Markov is thoroughly described in [20] and the BCJR equivalent is given in [60]. These algorithms apply to signal-dependent covariance matrices. Let us, first, briefly highlight the differences between our algorithm and the conventional sequence detection for Gauss-Markov noise. Instead of having an AR model to describe the noise characteristic, we resort to a multivariate AR model. In this case, the parameter estimation is done through multivariate Yule-Walker equations [61]. To derive the branch metrics we assume Markovianity of finite order ν. The full derivation of the algorithm is given below. Using the chain rule factorization and the Markovianity of order ν we could expand the the joint distribution as follows

$$f_{A,B|X}(\boldsymbol{a}, \boldsymbol{b}|\boldsymbol{x}) = \prod_{k=1}^{K} f_{A,B|X}(a_k, b_k|\mathbf{a}_{k-\nu}^{k-1}, \mathbf{b}_{k-\nu}^{k-1}, \mathbf{x}_{k-\nu-\xi}^{k}), \tag{3.20}$$

where ξ is the ISI length. Next, we use the Bayes rule to rewrite (3.20) as follows

$$f_{A,B|X}(\boldsymbol{a},\boldsymbol{b}|\boldsymbol{x}) = \prod \frac{f(\mathbf{a}_{k-\nu}^{k}, \mathbf{b}_{k-\nu}^{k}|\mathbf{x}_{k-\nu-\xi}^{k})}{f(\mathbf{a}_{k-\nu}^{k-1}, \mathbf{b}_{k-\nu}^{k-1}|\mathbf{x}_{k-\nu-\xi}^{k})}. \tag{3.21}$$

The log likelihood function can be expressed as sum of branch metrics as

$$\hat{\mathbf{x}}_{ML} = \arg\min_{\mathbf{x}} \sum_{k=1}^{K} \Lambda(\mathbf{a}_{k-\nu}^{k}, \mathbf{b}_{k-\nu}^{k}, \mathbf{x}_{k-\nu-\xi}^{k}), \tag{3.22}$$

where ML branch metrics are

$$\Lambda(\mathbf{a}_{k-\nu}^{k}, \mathbf{b}_{k-\nu}^{k}, \mathbf{x}_{k-\nu-\xi}^{k}) = -\log \frac{f(\mathbf{a}_{k-\nu}^{k}, \mathbf{b}_{k-\nu}^{k}|\mathbf{x}_{k-\nu-\xi}^{k})}{f(\mathbf{a}_{k-\nu}^{k-1}, \mathbf{b}_{k-\nu}^{k-1}|\mathbf{x}_{k-\nu-\xi}^{k})}. \tag{3.23}$$

The ML estimate of input sequence can be found through (3.22). However, the marginal distribution given in the numerator and denominator of (3.23) is quite complicated to calculate in the presence of the nuisance parameter. At this point, we use the first and second order statistics of the process to approximate the marginal distribution with a normal distribution given as

$$f_{A,B|X}(\mathbf{a}_{k-\nu}^{k}, \mathbf{b}_{k-\nu}^{k}|\mathbf{x}_{k-\nu-\xi}^{k}) \sim \mathcal{N}\left(\left[\mathbf{s}_{k-\nu}^{k}(\mathbf{x}_{k-\nu-\xi}^{k}), \tilde{\mathbf{s}}_{k-\nu}^{k}(\mathbf{x}_{k-\nu-\xi}^{k})\right], \mathbf{C}(\mathbf{x}_{k-\nu-\xi}^{k})\right). \tag{3.24}$$

The denominator also is a marginal distribution and has a normal distribution with mean $\left[\mathbf{s}_{k-\nu}^{k-1}(\mathbf{x}_{k-\nu-\xi}^{k-1}), \tilde{\mathbf{s}}_{k-\nu}^{k-1}(\mathbf{x}_{k-\nu-\xi}^{k-1})\right]$, and upper $2\xi \times 2\xi$ principal minor of $\mathbf{C}(\mathbf{x}_{k-\nu-\xi}^{k})$, named $\overline{\mathbf{C}}(\mathbf{x}_{k-\nu-\xi}^{k-1})$, as the covariance. For brevity, we drop the signal dependency whenever its clear from the context. For instance, we abbreviate matrices $\mathbf{C}(\mathbf{x}_{k-\nu-\xi}^{k})$, $\overline{\mathbf{C}}(\mathbf{x}_{k-\nu-\xi}^{k-1})$ as $\mathbf{C}$ and $\overline{\mathbf{C}}$, respectively. Substituting the normal distribution in (3.23) and canceling the common terms, we are left with the following expression for sum of branch metrics

$$\Lambda(\mathbf{a}_{k-\nu}^{k}, \mathbf{b}_{k-\nu}^{k-1}, \mathbf{x}_{k-\nu-\xi}^{k-\nu}) = \log \frac{\det \mathbf{C}}{\det \overline{\mathbf{C}}} + \\ + \left\| \begin{bmatrix} a_{k-\nu} - s_{k-\nu}(\mathbf{x}_{k-\nu-\xi}^{k-\nu}) \\ b_{k-\nu} - \tilde{s}_{k-\nu}(\mathbf{x}_{k-\nu-\xi}^{k-\nu}) \\ \vdots \\ a_k - s_k(\mathbf{x}_{k-\xi}^{k}) \\ b_k - \tilde{s}_k(\mathbf{x}_{k-\xi}^{k}) \end{bmatrix} \right\|_{\mathbf{C}}^2 - \left\| \begin{bmatrix} a_{k-\nu} - s_{k-\nu}(\mathbf{x}_{k-\nu-\xi}^{k-\nu}) \\ b_{k-\nu} - \tilde{s}_{k-\nu}(\mathbf{x}_{k-\nu-\xi}^{k-\nu}) \\ \vdots \\ a_{k-1} - s_{k-1}(\mathbf{x}_{k-\xi}^{k}) \\ b_{k-1} - \tilde{s}_{k-1}(\mathbf{x}_{k-\xi}^{k}) \end{bmatrix} \right\|_{\overline{\mathbf{C}}}^2 . \quad (3.25)$$

For the full description the reader should refer to [20]. The branch metrics given in (3.25) can be further simplified. To that extent, we invoke the matrix inversion lemma [20]

$$\mathbf{C}^{-1} = \begin{bmatrix} \overline{\mathbf{C}} & \mathbf{U}(\mathbf{x}_{k-\nu-\xi}^{k}) \\ \mathbf{U}^T(\mathbf{x}_{k-\nu-\xi}^{k}) & \mathbf{V}(\mathbf{x}_{k-\nu-\xi}^{k}) \end{bmatrix}^{-1} = \\ \begin{bmatrix} \overline{\mathbf{C}}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} + \mathbf{W}(\mathbf{x}_{k-\nu-\xi}^{k}) \mathbf{\Gamma}^{-1}(\mathbf{x}_{k-\nu-\xi}^{k}) \mathbf{W}(\mathbf{x}_{k-\nu-\xi}^{k})^T, \quad (3.26)$$

where $\mathbf{\Gamma}(\mathbf{x}_{k-\nu-\xi}^{k}) = \mathbf{V}(\mathbf{x}_{k-\nu-\xi}^{k}) - \mathbf{U}(\mathbf{x}_{k-\nu-\xi}^{k})^T \overline{\mathbf{C}}^{-1} \mathbf{U}(\mathbf{x}_{k-\nu-\xi}^{k})$, and $\mathbf{W}(\mathbf{x}_{k-\nu-\xi}^{k})$ is defined as

$$\mathbf{W}(\mathbf{x}_{k-\nu-\xi}^{k}) = \begin{bmatrix} -\overline{\mathbf{C}}^{-1}\mathbf{U} \\ \mathbf{I}_{2\times 2} \end{bmatrix} = \begin{bmatrix} -\mathbf{B}_{\nu}^{T}(\mathbf{x}_{k-\nu-\xi}^{k}) \\ \vdots \\ -\mathbf{B}_{1}^{T}(\mathbf{x}_{k-\nu-\xi}^{k}) \\ \mathbf{I}_{2\times 2} \end{bmatrix}. \quad (3.27)$$

The $\mathbf{B}_i(\mathbf{x}_{k-\nu-\xi}^k)$'s and $\mathbf{\Gamma}(\mathbf{x}_{k-\nu-\xi}^k)$ are 2×2 matrices and define a multivariate FIR filter [61]. They are determined by the solution to multivariate Yule-Walker equations [61]. On the other hand, from definition of $\mathbf{\Gamma}(\mathbf{x}_{k-\nu-\xi}^k)$ it can be shown that

$$\frac{\det \mathbf{C}\left(\mathbf{x}_{k-\nu-\xi}^k\right)}{\det \overline{\mathbf{C}}(\mathbf{x}_{k-\nu-\xi}^k)} = \det(\mathbf{\Gamma}(\mathbf{x}_{k-\nu-\xi}^k)). \tag{3.28}$$

Thus, the branch metrics are given by

$$\Lambda(\mathbf{a}_{k-\nu}^k, \mathbf{b}_{k-\nu}^k, \mathbf{x}_{k-\nu-\xi}^k) = \ln(\det(\mathbf{\Gamma}(\mathbf{x}_{k-\nu-\xi}^k))) + \left\| \begin{bmatrix} a_k - s_k(\mathbf{x}_{k-\xi}^k) - u_k \\ b_k - \tilde{s}_k(\mathbf{x}_{k-\xi}^k) - \tilde{u}_k \end{bmatrix} \right\|^2_{\Gamma^{-1}(\mathbf{x}_{k-\nu-\xi}^k)}, \tag{3.29}$$

where the predicted noise terms $u_k, \tilde{u}_k$ are defined as

$$\begin{bmatrix} u_k(\mathbf{x}_{k-\nu-\xi}^{k-1}) \\ \tilde{u}_k(\mathbf{x}_{k-\nu-\xi}^{k-1}) \end{bmatrix} = \sum_{i=1}^{\nu} \mathbf{B}_i(\mathbf{x}_{k-\nu-\xi}^k) \begin{bmatrix} a_{k-i} - s_{k-i}(\mathbf{x}_{k-\xi-i}^{k-i}) \\ b_{k-i} - \tilde{s}_{k-i}(\mathbf{x}_{k-\xi-i}^{k-i}) \end{bmatrix}. \tag{3.30}$$

The prediction filter matrices $\mathbf{B}_i$'s are calculated offline, see Appendix C. To implement the Viterbi algorithm, the states on the trellis diagram at each step are extended to include all possible symbols $\mathbf{x}_{k-\nu-\xi}^k$. Then, the branch metrics are calculated for each branch considering the two incoming values a_k, b_k at a time. At the end, the path that minimizes the log likelihood over all different $\mathbf{x}$'s is chosen as the output of the detector. Note that all along we assumed s and $\tilde{s}$ depends on the same input signals. The generalization to the case where they are dependent on different number of x's is straightforward.

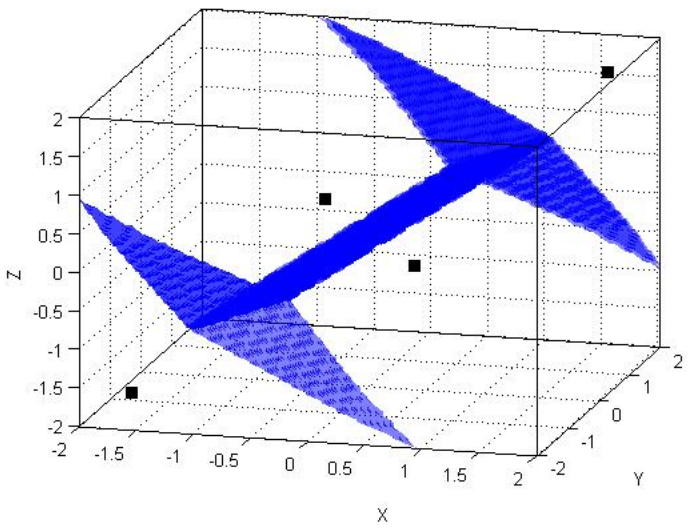


Figure 3.3: Decision boundaries for $\mathbf{x} \in \{[1,1],[-1,1],[-1,1],[-1,-1]\}$ where $\mathbf{s}(\mathbf{x}), \tilde{s}(\mathbf{x})$ are designated with square dots. The axes indicate the three samples' values. The decisions are made based on the region to which the received vector belongs. The ratio of sampling period to standard deviation of pulse response is $1/\sqrt{2}$.

3.5.1 How Good Is the Gauss Markov Assumption?

In this section, we are interested in determining the validity of the Gauss-Markov assumption. Decision boundaries for single shot transmission provide us with visualization for binary hypothesis testing. In the presence of jitter, we put the Gauss Markov assumption to the test by drawing three dimensional decision boundaries for two input values. We sketch the decision boundaries for binary transmission of vector $[x_1, x_2]$. We assume the channel is Gaussian-shaped and samples are collected through a matched filter at times $0, 1/2T, T$. The optimum decision is given by solving the following

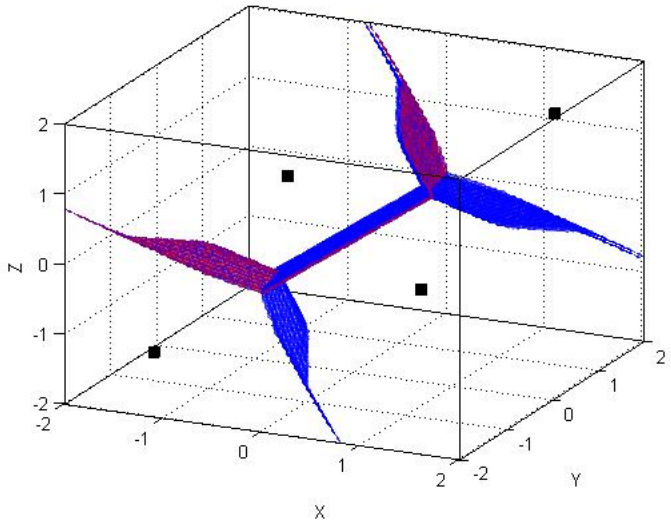


Figure 3.4: Decision boundaries for $\mathbf{x} \in \{[1,1],[-1,1],[-1,1],[-1,-1]\}$. The axes indicate the three samples' values. The top left region wrapped by the red surface indicates decision $\mathbf{x} = [-1,1]$. The ratio of sampling period to standard deviation of pulse response is $1/\sqrt{2}$.

$$\arg\max_{x_1,x_2} f(y_1,y_2,y_3|x_1,x_2) = \arg\max_{x_1,x_2} \int f(y_1,y_2,y_3|j_1,j_2 x_1,x_2) f(j_1,j_2) dj_1 dj_2. \quad (3.31)$$

The integral on the right hand side of (3.31) can be calculated numerically. The sampled points (y_1, y_2, y_3) contain three values that can be shown on a three dimensional plot. The decision boundaries are designated through three surfaces. As depicted in Fig. 3.3, the surfaces are hyperplanes in three dimensions. These boundaries are equivalent to the decision boundaries in the case of a Gauss-Markov distribution [62]. However, when α increases, the decision boundaries get more complicated. As the corresponding points to $[1,-1]$ and $[-1,1]$ get farther away from each other, the hyperplane caves in and the boundary surfaces change shape (See Fig. 3.4.)

3.6 Discussion

This section discusses lower bounds for performance of sequence detection in the presence of jitter. The first idea is to derive a genie-aided bound where the jitter values are revealed to the detector. Another approach is to find the lower bound for probability of error through minimum distance.

3.6.1 Genie Aided Detector

Since $\boldsymbol{a}, \boldsymbol{b}$ are two vectors which are a sufficient statistic for detection then there exists an ML detection of $\mathbf{x}$ stated as

$$\hat{\mathbf{x}}_{ML} = \arg\max_{\boldsymbol{x}} f_{A,B|X}(\boldsymbol{a}, \boldsymbol{b}|\boldsymbol{x}). \tag{3.32}$$

There have been reports of detectors that deal with discrete distributed jitter noise in the literature [63, 64]. Basically, if the distribution of the jitter noise is discrete, a joint estimation of jitter and information can be performed on a single trellis as in [63]. But in the case of continuous jitter random variables, (3.32) can be expressed in an integral form using Bayes' rule

$$\hat{\mathbf{x}}_{ML} = \arg\max_{\boldsymbol{x}} \int_{\mathcal{J}} f_{\mathbf{A},\mathbf{B}|\mathbf{X},\mathbf{J}}(\boldsymbol{a}, \boldsymbol{b}|\boldsymbol{x}, \boldsymbol{j}) f_{\mathbf{J}}(\mathbf{j}) d(\boldsymbol{j}). \tag{3.33}$$

Our approach is based on approximating the marginal distribution in (5.7). Assuming jitter displacements are identically independently Gaussian distributed with zero mean, we can write

$$\hat{\mathbf{x}}_{ML} = arg\max_{\mathbf{x}}(2\pi\sigma_j)^{-K}det(\mathbf{R})^{-1/2}$$
$$\int exp\left(-\frac{1}{2}\left[\left\|\boldsymbol{a}-\mathbf{s}_1^K, \boldsymbol{b}-\mathbf{s}_{K+1}^{2K}\right\|^2_{\mathbf{R}^{-1}} + \frac{1}{\sigma_j^2}\left\|\mathbf{j}\right\|^2\right]\right)d\mathbf{j}, \quad (3.34)$$

where $\underline{\mathbf{R}}$ is the covariance matrix for correlated noise in received samples, and $\mathbf{s} = [\langle s(t), q_1(t)\rangle\ , \langle s(t), q_2(t)\rangle\ , ..., \langle s(t), q_{2K}(t)\rangle]$.

Let $Q(\mathbf{x}_1^K, \mathbf{j}_1^K) = [\boldsymbol{a}-\mathbf{s}_1^K, \boldsymbol{b}-\mathbf{s}_{K+1}^{2K}]\mathbf{R}^{-1}[\boldsymbol{a}-\mathbf{s}_1^K, \boldsymbol{b}-\mathbf{s}_{K+1}^{2K}]^T$ and p be a smooth function. Then, we define $\mathbf{g}_{\partial P_k}^T$ and $\mathbf{g}_{\partial^2 P_k}$to be the sum of k, and $K+k$-th column of Gram matrix for the first and second partial derivative of the channel response with respect to j_k respectively, e.g., the i-th element of$\mathbf{g}_{\partial P_k}$ is given by$\int \frac{\partial p(t-kT+j_k)}{\partial j_k}[q_i(t)]dt$. If our transform $\mathbb{T}$ is onto, we can choose jitter vector $\mathbf{j}^*$ ideally such that $a_i - s_i = 0,\ b_i - s_{K+i} = 0$ for all $1 \le i \le K$. Then, writing the Taylor series expansion around the extrema of exponents, i.e., $\mathbf{j}^*$, first partial derivatives of $Q(x_1^K, j_1^K)$ with respect to j_k's are zero,

$$\left.\frac{\partial Q(\mathbf{x}_1^K, \mathbf{j}_1^K)}{\partial j_k}\right|_{j_k^*} = \left(-2[x_k\mathbf{g}_{\partial P_k}]\mathbf{R}^{-1}[\boldsymbol{a}-\mathbf{s}_1^K, \boldsymbol{b}-\mathbf{s}_{K+1}^{2K}]^T\right)\big|_{j_k^*} = 0.$$

Therefore, by the saddle-point approximation [65], the approximate marginal distribution is given by

$$\int f_{A,B|J,X}(\boldsymbol{a},\boldsymbol{b}|\boldsymbol{x},\mathbf{j})f_J(\mathbf{j})d\mathbf{j} \simeq$$
$$(2\pi\sigma_j)^{-K}det(\mathbf{R})^{-1/2}exp\left(-Q(\mathbf{x}_1^K, \mathbf{j}_1^{*K})\right)$$
$$\int exp\left(-\frac{1}{2\sigma_j^2}\left\{(\mathbf{j}-\mathbf{j}^*)(\sigma_j^2\mathbf{B}^{-1})(\mathbf{j}-\mathbf{j}^*)^T + \mathbf{j}\mathbf{j}^T\right\}\right)d\mathbf{j}. \quad (3.35)$$

where matrix $\underline{\mathbf{B}}$ is defined by its elements $\underline{\mathbf{B}}_{k,m} = \frac{\partial^2 Q(\mathbf{x}_1^K, \mathbf{j}_1^K)}{\partial j_k \partial j_m}$, through

$$\begin{aligned}\left.\frac{\partial^2 Q(\mathbf{x}_1^K, \mathbf{j}_1^K)}{\partial j_k^2}\right|_{j_k^*} &= \left(-2[x_k \mathbf{g}_{\partial P_k}]\mathbf{R}^{-1}[x_k \mathbf{g}_{\partial P_k}\right)\big|_{j_k^*} \\ &+ \left(-2[x_k \mathbf{g}_{\partial^2 P_k}]\mathbf{R}^{-1}[\boldsymbol{a} - \mathbf{s}_1^K + \boldsymbol{b} - \mathbf{s}_{K+1}^{2K}]^T]^T\right)\big|_{j_k^*} \\ &= -2[\mathbf{g}_{\partial P_k}]\mathbf{R}^{-1}[\mathbf{g}_{\partial P_k}]^T\big|_{j_k^*}, \quad (3.36)\end{aligned}$$

$$\left.\frac{\partial^2 Q(\mathbf{x}_1^K, \mathbf{j}_1^K)}{\partial j_k \partial j_m}\right|_{j_k^*, j_m^*} = 2x_k x_m \left[\mathbf{g}_{\partial P_k}\right] \mathbf{R}^{-1} \left[\mathbf{g}_{\partial P_m}\right]\big|_{j_k^*, j_m^*}. \quad (3.37)$$

Finally, we can put the integrand into the quadratic form of a normal distribution by replacing the mean value of the exponential term $\mathbf{j}^*$, by $\mathbf{j}^*(\sigma_j^2\mathbf{B}^{-1}(\mathbf{I}+\sigma_j^2\mathbf{B}^{-1})^{-1})$. Hence (5.8) reduces to

$$\begin{aligned}(2\pi\sigma_j)^{-K}(2\pi)^{K/2} det(\sigma_j^2\mathbf{B}^{-1} + \mathbf{I}) exp\left(-\frac{1}{2}Q(\mathbf{x}_1^K, \mathbf{j}_1^{*K})\right) \\ exp\left(-\frac{\sigma_j^2}{2}(\mathbf{j}^*)(\sigma_j^2\mathbf{B} + \mathbf{B}^2)^{-1}(\mathbf{j}^*)^T\right). \quad (3.38)\end{aligned}$$

We know by definition, $\underline{\mathbf{B}}$ is a symmetric matrix where diagonal elements of $\underline{\mathbf{B}}$, and also any powers of $\underline{\mathbf{B}}$, do not depend on x_k's, for $k = 1, ..., K$. Since the determinant of a matrix can be expressed in terms of traces of powers of that matrix [66, Thm. 1], the determinant of matrix $\underline{\mathbf{B}}$ does not depend on the $\mathbf{x}_k$'s . Nor does the determinant of $det(\sigma_j^2\mathbf{B}^{-1} + \mathbf{I})$. The the covariance matrix for the second exponential in (3.38) cane be approximated as $\sigma_j^2(\sigma_j^2\mathbf{B} + \mathbf{B}^2)^{-1} \simeq \sigma_j^2\mathbf{B}^{-2} + \sigma_j^4\mathbf{B}^{-2} + \mathcal{O}(\sigma_j^6)$, using matrix inversion lemma[67]. For small values of jitter variance this term may be neglected. In this manner, given the optimum values of jitter noise,

the ML values are dependent on $Q(x_1^K, j_1^{*K})$ which is expressed as

$$\hat{\mathbf{x}}_{ML} = \arg\max_{\mathbf{x}} det(\underline{\mathbf{R}})^{-1/2} exp\left\{-\frac{1}{2}\left\|[\boldsymbol{a} - \mathbf{s}^{*K}_{1}, \boldsymbol{b} - \mathbf{s}^{*2K}_{K+1}]\right\|^2_{\underline{\mathbf{R}}^{-1}}\right\},$$

where $\mathbf{s}^*$ is the vector $\mathbf{s}$ evaluated at $\mathbf{j}^*$. The jitter values can be recovered for Gaussian pulses as explained in Section 3.4. No prior knowledge of input values are needed for such a recovery.

3.7 Simulation Results

In this section, we give a setup of communication system, using the sampler, detector, and etc. together to exploit the benefit of such a system in presence of jitter. This is the motivation for this section.

In the absence of jitter, the detection layout depends on the transmission rate. Symbol-by-symbol detection is optimum for a band-limited channel with T-orthogonal pulse response, if the data transmission rate is $1/T$ [68]. By T-orthogonal pulses we mean that the inner product of two distinct integer T-shifted signals is zero. An example of such pulses is $\text{sinc}(t/T)$. Mazo [54] showed that sinc pulses can be transmitted at a rate faster than $1/T$ without loss in performance. The scheme is called faster-than-Nyquist (FTN) signaling. For FTN signaling, the presence of ISI in channel requires a detector of some sort [69].

With jitter dislocations, we cannot claim that the transmission is ISI free at any transmission rate. However, the FTN serves as a lower bound for the jitter contaminated channel and we also observe that for jitter with Gaussian distribution, transmission rate has an important role in the performance of the system. In the remainder of the section, we discuss uniform sampled receivers for Gaussian and sinc-shaped pulses and give different setups for the equalizer and the receiver.

3.7.1 Transmission Pulses

In many applications it is common to model the physical channel as Gaussian-shaped. In this section we consider a Gaussian model of the following form

$$y(t) = \sum_{k=1}^{K} x_k \exp\left(-\frac{(t - k\tau T + j_k)^2}{2\sigma_s^2}\right) + n(t). \tag{3.39}$$

In (3.39), σ_s^2 is the the variance of the signal, and $1/\tau T$ indicates the signaling rate. The jitter value, j_k, is a random variable with a truncated Gaussian distribution that captures the effect of inaccuracy in timing or positioning. We also define a parameter α as the normalized transmission rate over Gaussian signal variance $\alpha = \sqrt{2}\frac{\tau T}{\sigma_s}$. The normalized standard deviation of jitter is σ_j/σ_s. AWGN noise of -30dB is added to the signal. The signal $y(t)$ is sampled using a matched filter at twice the baud rate.

For ISI free transmission over a bandlimited AWGN channel, sinc-shaped functions are a common choice. In this section, we study the sinc pulses transmitted at a higher rate than the Nyquist Criterion where random jitter is also present. Our sinc-function transmission model is given by

$$y(t) = \sum_{k=1}^{K} x_k \mathbf{sinc}\left(t - \tau k T + j_k\right) + n(t). \tag{3.40}$$

For $T = 1$, if we indicate the signaling rate as $1/\tau$, then for faster than the Nyquist signaling transmission we have $\tau < 1$. The other parameters are kept the same as in the model for Gaussian pulses. To calculate the SNR, we estimate the noise power at the output due to the jitter. Then, the SNR is defined as signal power over the total output noise power for different variances of random jitter.

The asymptotic performance of the detector is determined by its minimum distance. For transmission above the Mazo limit, the performance is hindered by the loss in d_{min}. The numerical calculation of the minimum distance is given in [69].

For sinc functions the Mazo limit is $\tau = 0.8$, and for Gaussian pulses our simulation results show that the Mazo limit corresponds to $\alpha = 0.85$.

The main source of noise is assumed to be induced by jitter displacements. Given that the displacements have a truncated Gaussian distribution, to calculate the SNR, we estimate the noise power at the output due to the random jitter. Then, the SNR is defined as signal power over the total output noise power for different variances of random jitter

$$SNR = 10 \log \frac{\int |p(t)|^2 \, dt}{M_0 + N_0}, \tag{3.41}$$

where $M_0 = E_J \left[\int |p(t+j)|^2 - |p(t)|^2 \, dt \right]$ and N_0 is single sided power spectral density due to the AWGN noise.

3.7.2 Minimum Distance

Consider the original transmitted signal given in (3.1). The ML detector for such a transmission with ISI provides the minimum probability of error. Although, a computationally feasible implementation of ML detector is not possible in case of jitter, the probability of error of such a system is bounded below by the minimum distance. The minimum distance depend on the pulse response and therefore is a function of jitter values, however, as a measure of performance we derive the d_{min} for jitter free system. The distance between transmission of two different input vectors $\mathbf{x}^*, \hat{\mathbf{x}}$ is given by

$$d^2(\mathbf{x}^*, \hat{\mathbf{x}}) = \int_{-\infty}^{\infty} [\mathbf{s}_{\mathbf{x}^*}(t) - \mathbf{s}_{\hat{\mathbf{x}}}(t)]^2 dt, \tag{3.42}$$

where $\mathbf{s}(t)$ is the signal in (3.1) minus the AWGN noise. The system is linear with respect to input values and therefore can be expressed in terms of error sequence

$\mathbf{e} = \mathbf{x}^* - \hat{\mathbf{x}}$, where $e_k \in \{0, \pm 2\}$, as follows

$$d^2(\mathbf{e}) = \sum_l \sum_k e_k e_l [p(t - kT) * \overline{p(-t + lT)}]. \tag{3.43}$$

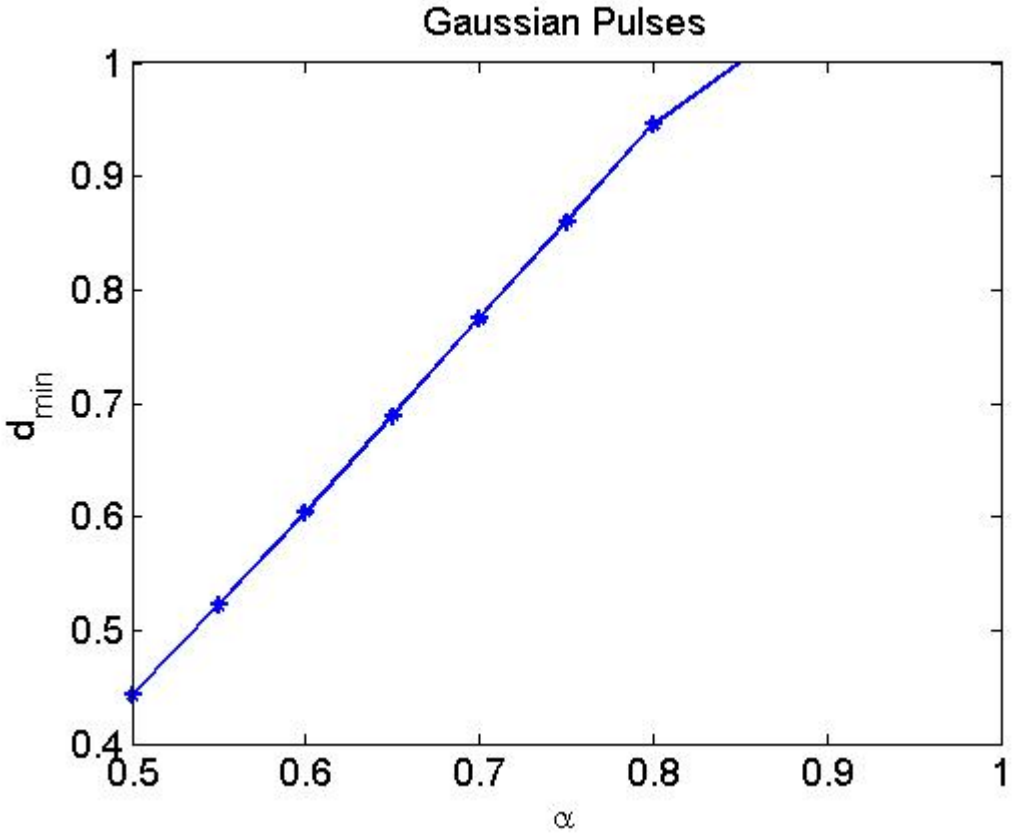
The expression in (3.43) can be calculated numerically for different error sequences. Over all possible sequences, there exists an error sequence that minimizes the distance. The normalized minimum distance d_{min} is defined as [54]

$$d^2_{min} = \min_{\mathbf{e} \neq \mathbf{0}} \frac{d^2(\mathbf{e})}{4 \int |s(t)|^2 \, d(t)}. \tag{3.44}$$

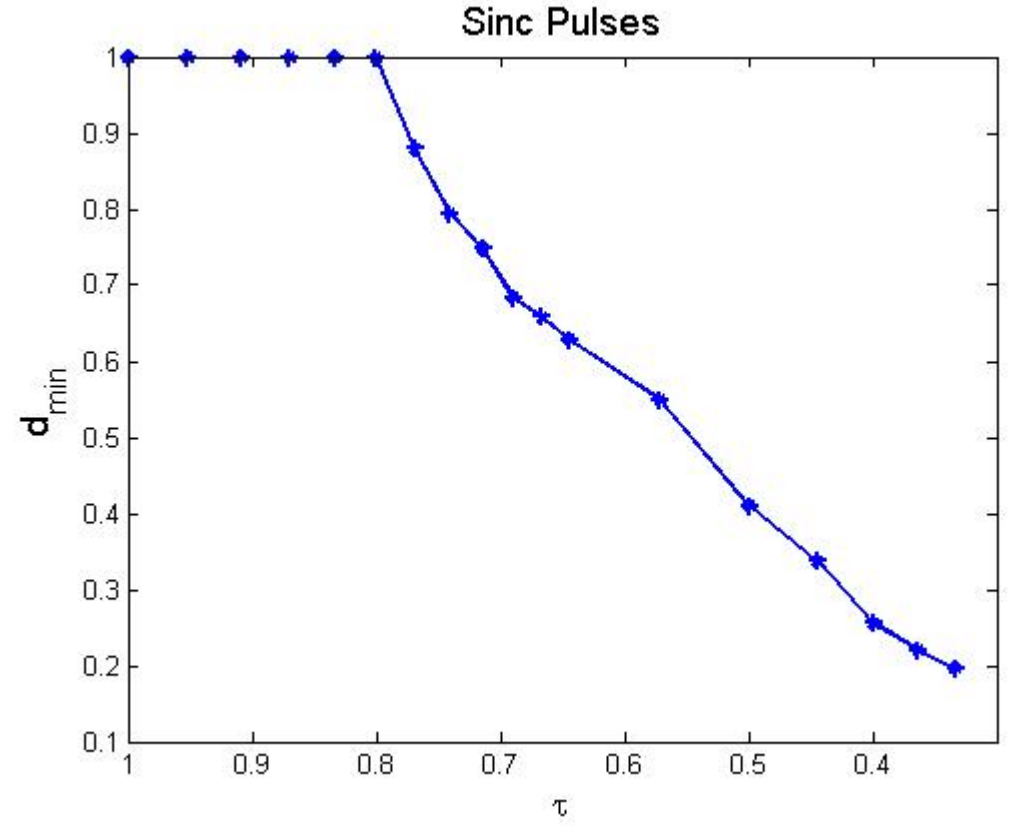
The parameter d_{min} is an indicator of the optimum detector's performance. The results for sinc and Gaussian pulses are shown in Fig. 3.5. The error events that cause loss in minimum distance are long blocks of alternating error sequences as reported in several papers [69, 70, 54].

3.7.3 Truncated Equalizer

The Gaussian pulse has an infinite support, however at both ends the pulse decays exponentially fast. We truncate the pulse to values greater than 10^{-3}. We set the variance of AWGN noise to zero. In this manner, there are two sources of noise in the channel: i) Due to jitter, the observed signal will differ from the nominal signal; ii) We assume that the residual ISI resulted from the truncation behaves as additive Gaussian noise. Since there is no additive noise, the output of the matched filter can be used directly on the detection algorithm. Note that in the case of AWGN noise, a modified version of the Viterbi algorithm can still be used for the truncated equalizer [71]. Fig. 3.6 shows the performance of the Viterbi algorithm on the output of jitter translated Gaussian shaped signal once it goes through a matched filter. We also truncate the tail of Gaussian pulses and use the oversampled signals for the



(a)



(b)

Figure 3.5: Minimum Euclidean distance for different transmission rates.

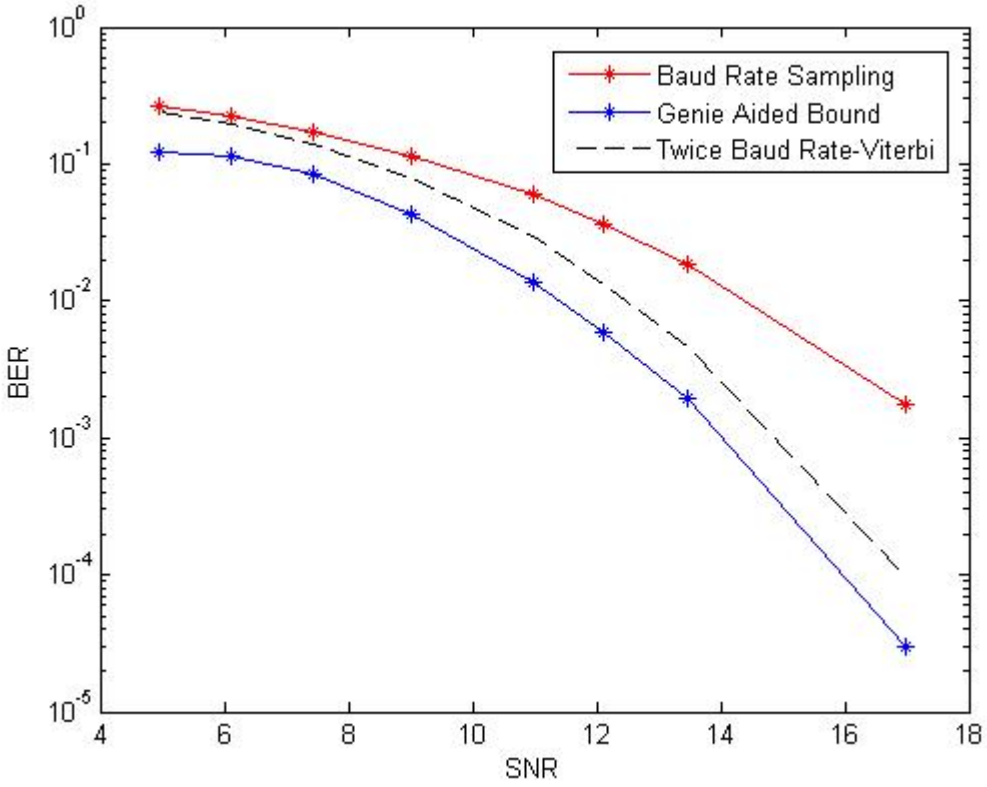


Figure 3.6: Comparison of Viterbi algorithm for baud rate and twice baud rate sampling versus the genie aided bound.

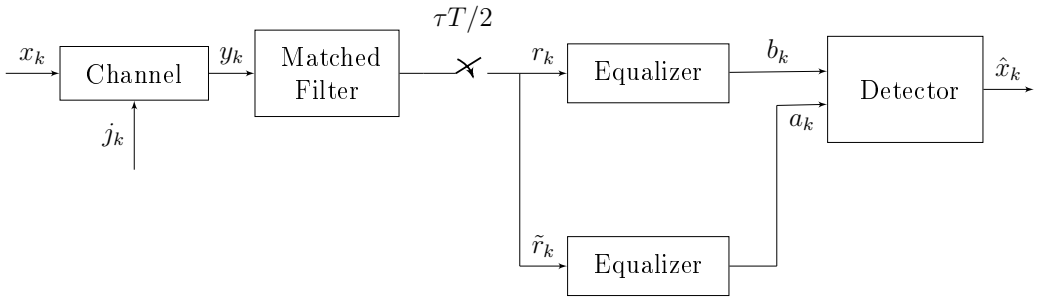


Figure 3.7: Communication system used for twice the baud rate sampling.

calculation of branch metrics to compare with the baud rate sampling. The branch metrics for twice the baud rate sampling are calculated simply as the squared sum of each metric$\|[a_i - s_i, b_i - \tilde{s}_i]\|^2$at step i. In another experiment, we truncate the the same signal that was used to calculate the performance of the Viterbi algorithm but this time the jitter values are revealed to the detector. The details of the genie-aided detector was explained in Section 3.5. The difference between the two Viterbi curves shows that the minimum distance has changed. For higher variance of jitter the Viterbi results are farther away from the genie bound.

3.7.4 Partial Response Equalization

In order to evaluate the performance of our sampling strategy we need a practical detection algorithm with feasible computational complexity. We use a modified version of the PDNP algorithm [60] adapted for the oversampled channel. For simulations, AWGN noise of -30dB is added to the signal. We have purposefully chosen a small value for AWGN noise to examine the effect of jitter. To use the detector, we need a finite support channel response. It is common to use an equalizer to curtail the tail of a infinite support response. In our simulations, two equalizers are used to find the targets of length 2, one offset by $T/2$. The noise at the output of the equalizers is used to find the parameters of the multivariate AR model. The

number of predictive taps in the detector is 1. After the equalizers and targets are fitted to the channel response, we used the target as the description of the channel to define the branch metrics. Since the target is finite, we are able to utilize the PDNP detector. The diagram of the transmission system is depicted in Fig. 3.7.

Fig. 3.8 shows the simulation results for different transmission rates. The σ_s is an indicator of the bandwidth required for transmission over the channel. For a fixed bandwidth, namely fixing σ_s, the lower α simply means faster transmission rate. As we expect, higher transmission rate results in loss of performance for the conventional detector. Interestingly, the extra samples taken from the received signal compensate for the loss. The conventional PDNP detector with baud rate sampling uses only one of the equalizers depicted in Fig. 3.8. The proposed sequence detector proves resilient to lower values of α. As explained in Section 3.6, lower α makes boundary decisions more complicated and therefore result in a sub-optimal detector. We conjecture that the relatively small loss in the performance for higher transmission rates is due to the sub-optimality of our detector. The gain in the performance is significant compared to the baud-rate sampling.

Fig. 3.9 shows the performance of the communication system for faster-than-Nyquist signaling. In this setup, we used the symbol-by-symbol detection for orthogonal Nyquist signals in the presence of random jitter. The performance reveals the optimum bit-error rate for different variances of the jitter random variable. The jitter accounts for most of the noise in the system as the AWGN noise is limited to -30dB in the channel. The matched filter output is sampled at twice the transmission rate, then samples are fed into the modified PDNP algorithm. The length of the target is 2, which limits the computational complexity. The number of predictor taps is kept constant as 1. As shown in the figure, the bit-error rate for the Mazo limit remains close to optimum. However, the performance degrades for data

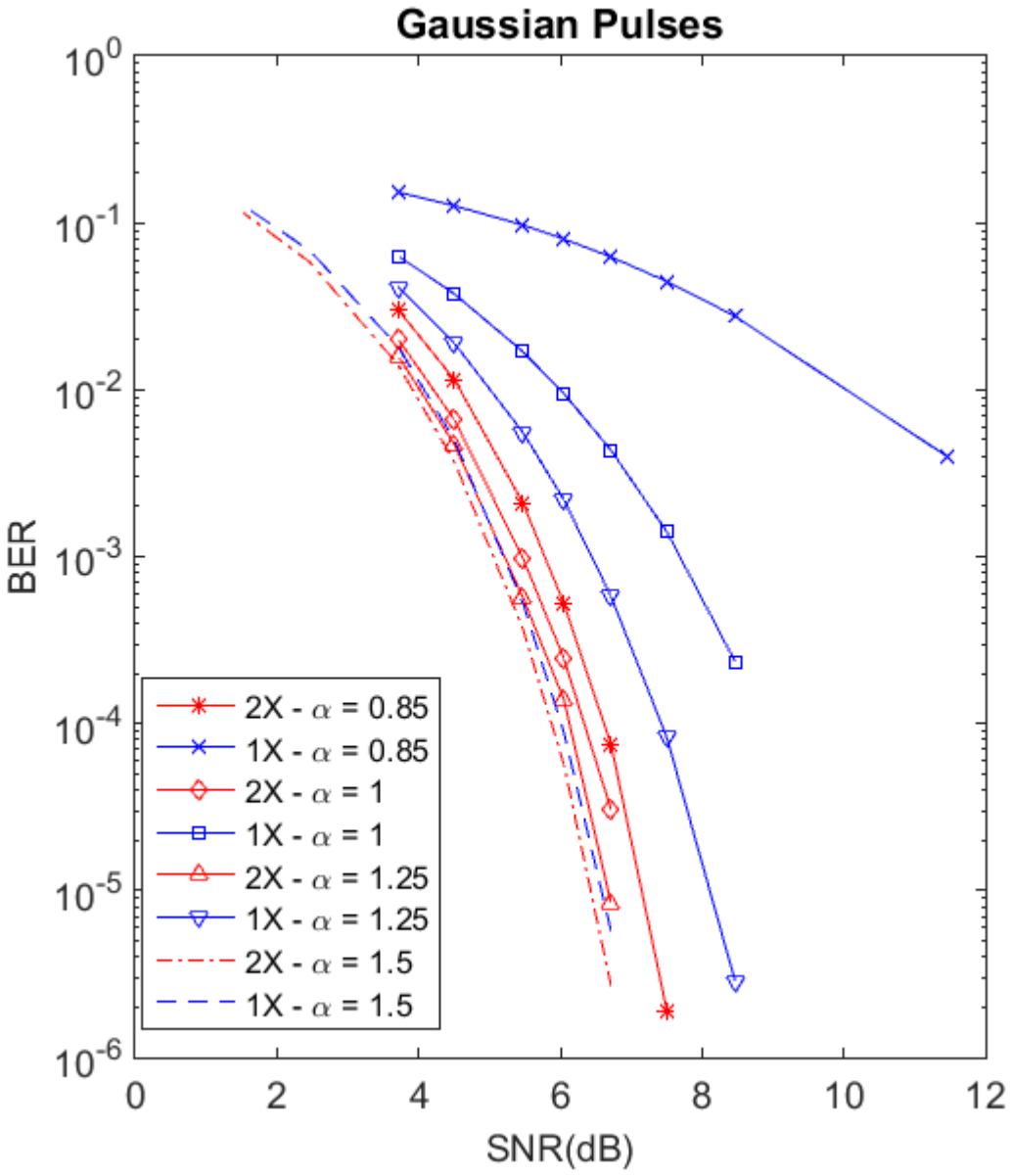


Figure 3.8: Comparison of pattern dependent noise predictive algorithm for baud rate (labeled as 1X) and twice baud rate sampling (labeled as 2X).

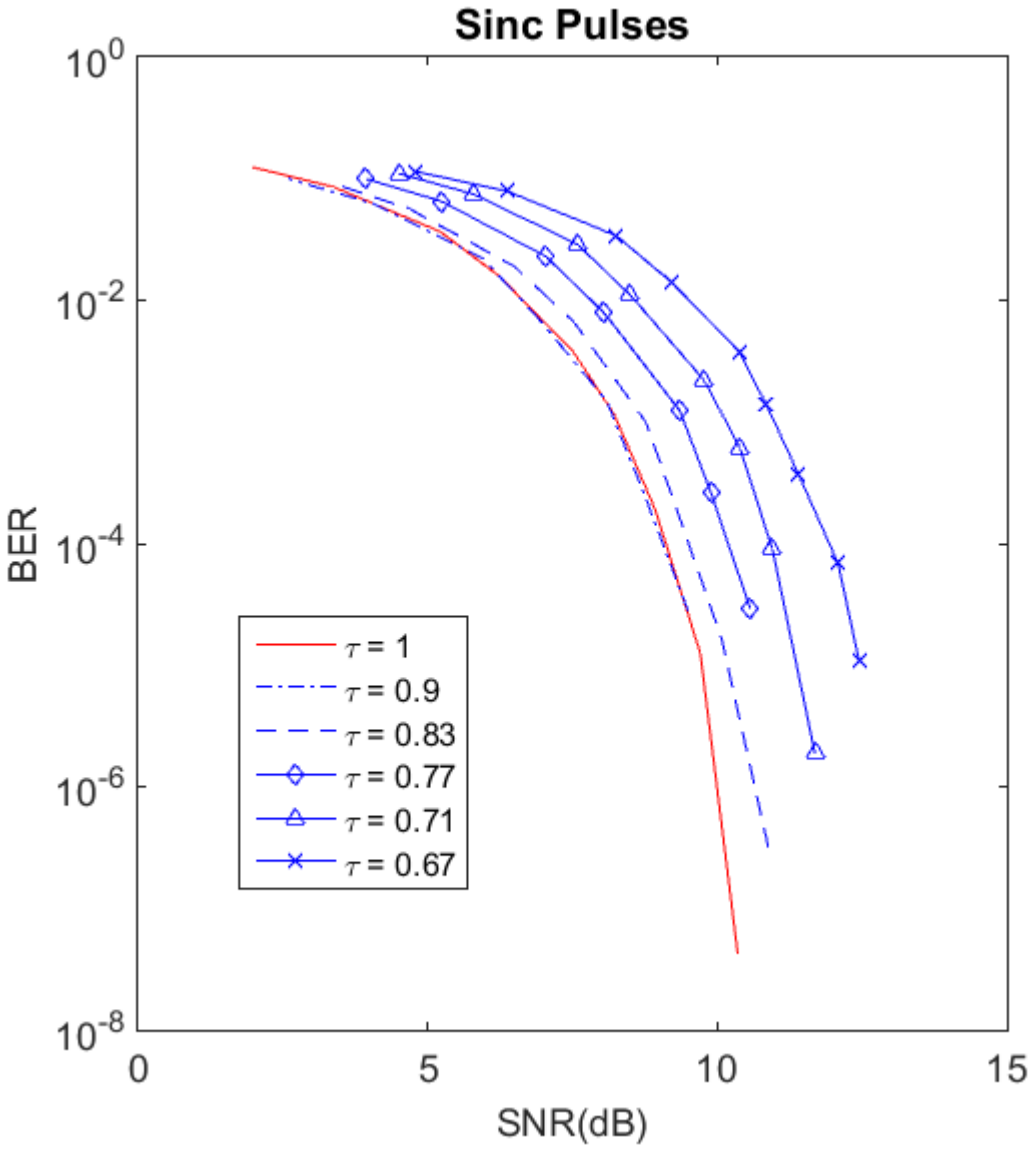


Figure 3.9: Comparison of performance of proposed communication system at higher rate to an ISI free transmission (fater-than-Nyquist signaling).

transmitted at higher rates. This can be associated with the sub-optimal equalizer and detector, as well as loss in d_{min}. We observed the same gap as in the Gaussian case between the baud-rate sampling and the proposed algorithm.

Appendices

A. Proof of Lemma 4.1

Proof. Suppose $\mathbb{T}$ is invertible. For any $s_1, s_2 \in \chi$ by definition there exists γ_0, θ_0 such that $s_1 \in S_{\gamma_0}$and $s_2 \in S_{\theta_0}$. Since $S_\gamma + S_\theta$ is a subspace then $\mathbb{T}|_{S_\gamma + S_\theta}$ is also linear and we only need to show its kernel is zero, i.e., $ker(\mathbb{T}|_{S_\gamma + S_\theta}) = 0$. From the hypothesis we know that $s_1 = s_2$ if and only if $\mathbb{T}s_1 - \mathbb{T}s_2 = 0$. But $s_1 - s_2 \in S_{\gamma_0} + S_{\theta_0}$ and hence $\mathbb{T}|_{S_\gamma \cup S_\theta}(s_1 - s_2) = \mathbb{T}|_{S_\gamma \cup S_\theta} s_1 - \mathbb{T}|_{S_\gamma \cup S_\theta} s_2 = \mathbb{T}s_1 - \mathbb{T}s_2 = 0$.

Conversely, we want to show $\mathbb{T}s_1 = \mathbb{T}s_2$ if and only if $s_1 = s_2$. By hypothesis $\mathbb{T}$ is linear. If $\mathbb{T}(s_1 - s_2) = 0$ then $s_1 - s_2$ belongs to some $S_{\gamma_1} + S_{\theta_1}$. But we know the restriction of $\mathbb{T}$ to such a subspace is invertible. Therefore $\mathbb{T}|_{S_{\gamma_1} + S_{\theta_1}} s_1 = \mathbb{T}|_{S_{\gamma_1} + S_{\theta_1}} s_2$ which implies $s_1 = s_2$. □

B. Proof of Proposition 4.1

Proof. If we sample the received signal with $\{q_1(t), ..., q_N(t)\}$ to get the vector $\mathbf{r} = [r_1, ..., r_N]$, then, the MLSD strategy can be stated as

$$\hat{\mathbf{x}}_{\mathbf{ML}} = \mathbf{arg} \max_{\mathbf{x}} f_{\mathbf{R}|\mathbf{X}}(\mathbf{r}|\mathbf{x}), \tag{3.45}$$

where $f_{\mathbf{R}|\mathbf{X}}(\mathbf{r}|\mathbf{x})$ is the conditional distribution of sampled vector $\mathbf{r}$ given the information vector $\mathbf{x}$. Since $\mathbb{T}$ is invertible, due to invariance-to-data-transformation property [66], the likelihood is unchanged - with respect to the likelihood of receiving

$y(t)$ given $\mathbf{x}$. The likelihood function can be expanded for different values of jitter noise through the Bayes rule

$$f_{\mathbf{R}|\mathbf{X}}(\mathbf{r}|\mathbf{x}) = \int_{\mathbb{J}} f_{\mathbf{R}|\mathbf{X},\mathbf{J}}(\mathbf{r}|\mathbf{x},\mathbf{j}) f_{\mathbf{J}|\mathbf{X}}(\mathbf{j}|\mathbf{x}) d\mathbf{j}, \tag{3.46}$$

where we assumed the J_k's are independent of X_k's for any $k, i = 1, ..., K$.

There exists a set of orthonormal basis $\{\psi_i(t)\}_{i=1}^M$ that we choose, for some $M \in \mathbb{N}$, to express sampling kernels $\{q_1(t), ..., q_N(t)\}$ so that the representation of signal $s(t)$ and channel output can be projected onto the subspace of sampling kernels by the vector $\mathbf{s} = [\langle s(t), \psi_1(t)\rangle, \langle s(t), \psi_2(t)\rangle, ..., \langle s(t), \psi_M(t)\rangle]$ and $\mathbf{y} = [\langle y(t), \psi_1(t)\rangle, \langle y(t), \psi_2(t)\rangle, ..., \langle y(t), \psi_M(t)\rangle]$, respectively. We write the conditional distribution in terms of vector $\mathbf{y}$, noting that the $\mathbf{y}$ is a linear transformation of $\mathbf{r}$, $\mathbf{y} = \Psi\mathbf{r}$, and the distributions are translated into each other by a Jacobian matrix. Since the noise present in the channel is AWGN, $f_{\mathbf{Y}|\mathbf{X},\mathbf{J}}(\mathbf{y}|\mathbf{x},\mathbf{j})$ is a jointly normal distribution with mean $\mathbf{s}$ and we have

$$f_{\mathbf{Y}|\mathbf{X},\mathbf{J}}(\mathbf{y}|\mathbf{x},\mathbf{j}) = (\pi N_0)^{-M/2} exp\left(-\frac{1}{N_0}\sum_{i=1}^{M}(y_i - s_i)^2\right). \tag{3.47}$$

We expand the quadratic term and by definition of $\mathbf{s}$ we have

$$f_{\mathbf{Y}|\mathbf{X},\mathbf{J}}(\mathbf{y}|\mathbf{x},\mathbf{j}) = (\pi N_0)^{-M/2} exp\left(-\frac{1}{N_0}\left[\sum_{i=1}^{M}(y_i^2 + s_i^2) - 2\left(\int s(t)\sum_{i=1}^{M} y_i\psi_i(t)dt\right)\right]\right). \tag{3.48}$$

Next, we let $s(t) = \sum_{l=1}^{N} \hat{x}_l(\mathbf{j}_1^K, \mathbf{x}_1^K) q_l(t)$ be the representation of our signal in the subspace spanned by $\{\psi_i\}_{i=1}^M$, then we have

$$f_{\mathbf{Y}|\mathbf{X},\mathbf{J}}(\mathbf{y}|\mathbf{x},\mathbf{j}) = (\pi N_0)^{-M/2} exp\left(-\frac{1}{N_0}\left[\sum_{i=1}^{N}(y_i^2+s_i^2) \right.\right. \\ \left.\left. -2\sum_{i=1}^{N}\hat{x}_i(\mathbf{j}_1^K,\mathbf{x}_1^K)\left(\int y(t)q_i(t)dt\right)\right]\right). \quad (3.49)$$

where we used orthogonality of the basis' elements. But by definition, a basis obtained through the Gram-Schmidt algorithm is a linear combination of the original functions q_i's for some weights $w_{i,k}$, that is $\psi_i(t) = \sum_{k=1}^{N} w_{i,k}q_k(t)$. Then (3.49) becomes a function of collected samples,

$$f_{\mathbf{Y}|\mathbf{X},\mathbf{J}}(\mathbf{y}|\mathbf{x},\mathbf{j}) = (\pi N_0)^{-M/2} exp\left(-\frac{1}{N_0}\left[\sum_{i=1}^{N}(y_i^2+s_i^2) \right.\right. \\ \left.\left. -2\sum_{i=1}^{M}\sum_{k=1}^{N} w_{i,k}\hat{x}_i(\mathbf{j}_1^K,\mathbf{x}_1^K)\left(\int y(t)q_k(t)dt\right)\right]\right). \quad (3.50)$$

The term $\sum_{i=1}^{M}(y_i^2)$ is an energy term and does not depend on $\{X_i\}_{i=1}^K$, and can be written in terms of sampled values. We deduct the likelihood $f_{\mathbf{R}|\mathbf{X}}(\mathbf{r}|\mathbf{x})$ is only a function of our sampled points $\int y(t)q_k(t)dt$ for $k = 1, ..., N$ and vector $\mathbf{x}$. Hence, by the factorization theorem [66], the samples are sufficient statistic for$\{X_i\}_{i=1}^K$. □

C. Multivariate Yule-Walker Equations

The difference between the proposed detection algorithm and the conventional sequence detection for Gauss-Markov noise is that instead of having an AR model to describe the noise characteristic, we have a multivariate AR model. In this case, the parameter estimation is done through multivariate Yule-Walker equations [61]. We fix the input bits $\mathbf{x}_{k-\nu-\xi}^k$ to estimate the parameters of $\mathbf{W}, \mathbf{\Gamma}$. Let the $\mathbf{R}_j^T$ be the covariance matrix of collected samples defined as

$$\mathbf{R}_j^T = E\left\{ \begin{bmatrix} a_k - s_k(\mathbf{x}_{k-\nu}^k) \\ b_k - \tilde{s}(\mathbf{x}_{k-\nu}^k) \end{bmatrix} \left[a_{k-j} - s(\mathbf{x}_{k-j-\nu}^{k-j}), b_{k-j} - \tilde{s}(\mathbf{x}_{k-j-\nu}^{k-j}) \right] \right\}, \tag{3.51}$$

One can recognize the multivariate Yule-Walker equations in estimation of parameters

$$\underbrace{\begin{bmatrix} \mathbf{R}_\nu \\ \\ \vdots \\ \\ \mathbf{R}_1 \end{bmatrix}}_{\mathbf{U}} = \underbrace{\begin{bmatrix} \mathbf{R}_0 & \mathbf{R}_1 & & \mathbf{R}_{\nu-1} \\ \mathbf{R}_1 & \mathbf{R}_0 & & \mathbf{R}_{\nu-2} \\ \vdots & & \ddots & \vdots \\ \mathbf{R}_{\nu-1} & & & \mathbf{R}_0 \end{bmatrix}}_{\bar{\mathbf{C}}} \underbrace{\begin{bmatrix} \mathbf{B}_\nu^T \\ \\ \vdots \\ \\ \mathbf{B}_1^T \end{bmatrix}}_{\mathbf{B}}. \tag{3.52}$$

where $\bar{\mathbf{C}}$ is a full rank matrix provided that R_0 is invertible. The estimate of parameters $\mathbf{B}$ are given through

$$\hat{\mathbf{B}} = \bar{\mathbf{C}}^{-1}\mathbf{U}. \tag{3.53}$$

And for Γ we have

$$\mathbf{\Gamma} = \mathbf{R}_0 - \mathbf{U}^T\bar{\mathbf{C}}^{-1}\mathbf{U}. \tag{3.54}$$

4 Storage Channel Model

4.1 Introduction

Steady increase in storage density in the past decade has driven magnetic recording to the limits of current technology. The state of the art hard-disk drives operate at 1 Tb/in^2. The underlying driver for higher densities has been the down scaling of the grain size. At the current limit, shrinking grain size is no longer viable due to thermal instability [6]. The storage industry is looking at alternative approaches to increase the recording density. To keep with the current market demands, hard-disk drives need to go through fundamental changes. There are several competing technologies available today. In BPMR, the magnetic area is etched into the media using lithography. The resulting islands can be magnetized by the writing head. HAMR uses energy to heat up grains through the writing process. This sophisticated process achieves high density that ensures thermal stability. Two-dimensional magnetic recording (TDMR) utilizes conventional media and data are stored on smaller regions. Theoretically, on conventional media each bit can be saved on one single grain, which means it can achieve capacity up to 10 Tb/in^2.

To discuss the challenges of data storage, a comprehensive channel model is needed. In current state-of-the-art magnetic recording the bit size is on the scale of the grain size, resulting in irregular bit boundaries. These irregularities are captured

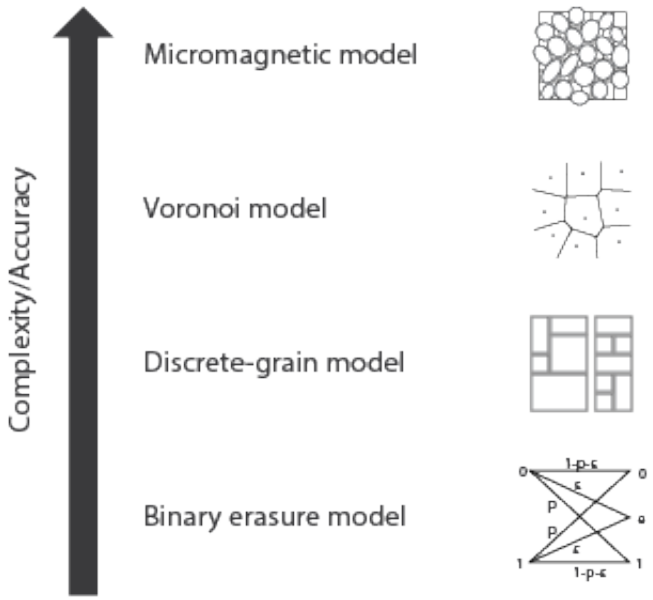


Figure 4.1: Various data storage channel models [73].

by the read head sensor as dominant source of noise. An accurate modeling of the phenomenon can be very effective in our understanding of the physical model and also in designing new signal processing methods. The available models range from mathematical models to micromagnetic modeling. The simple mathematical models are used to calculate the capacity of channel. The micromagnetic model follows physical rules that form the medium and represent a realistic model. This model is highly complex and may not be used for performance evaluation of signal processing algorithms. Fig. 4.1 shows models in different degrees of accuracy and complexity. In this chapter, we study the Voronoi model that has been developed and studied in [72]. The Voronoi provides a trade off between complexity and accuracy and will be used for evaluation of our detection and error control coding algorithms.

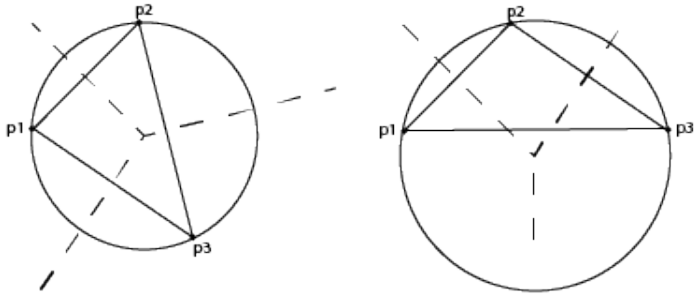


Figure 4.2: Three centers, Delaunay triangulation (solid lines), and Voronoi diagram (dashed line) using bisections [74].

4.2 Voronoi Model

Hard-disk drives use magnetic material to store data. The equally spaced tracks accommodate the data on the disk. The direction of magnetization indicates a binary sequence of zeros and ones on each track. The actual process in the physical channel is more complicated. The media is amorphous; nanometer grains are randomly distributed and shaped. An accurate model would express these characteristics and manifest similar properties of the magnetic media. We use the Voronoi model for the magnetic recording channel. The Voronoi model is an effective way of tessellation in many applications, including graphic processing units (GPU), finite element models, flexible mesh generation, etc. To explain the details of the Voronoi model, we first need to lay out the basics of the model.

Tessellation Given an open set $\Omega \subset \mathbb{R}^2$, the finite set$\{\mathcal{A}_i\}$ is called a tessellation of the open subset Ω if i) for all i's $\mathcal{A}_i \subset \Omega$. ii) $\mathcal{A}_i \cap \mathcal{A}_j = \emptyset$ for all $i \neq j$. iii) $\cup \bar{\mathcal{A}}_i = \bar{\Omega}$.

Voronoi Region Let S be the set of points on a Euclidean plane as centers. Voronoi region of $v \in S$, $\mathcal{A}_s$, is the set of all points on the plane that are closer to v

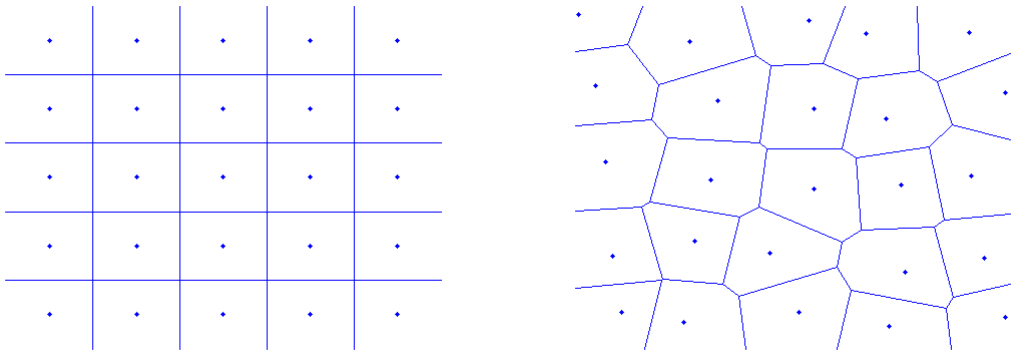
than any other points in S.

The set $\{\mathcal{A}_i\}$ is the Voronoi diagram of the disc and generating points $\nu \in S$ describe the grain centers.

One of closely related subjects to Voronoi diagram is Delaunay triangulation. We define the *empty circle* property for a circle that goes through three vertices and circumferencing a region that does not contain any other vertices inside. By Delaunay triangulation we mean a collection of edges that satisfy the empty circle property.

There are several algorithms available to construct the Voronoi diagram with different computational complexities. One easy way is the construction through Delaunay triangulation. The gradual construction is done by adding one new vertex at a time. Then, we search for any existing triangle that contains the new vertex. If the triangle existed, we split the triangle in three such that remaining edges satisfy the *empty circle* property. The complexity of such an algorithm is of order $O(n^2)$. Since the Delaunay triangulation problem is the dual of Voronoi diagram, a solution for one can translate to the other. We give an example for three points. The Voronoi diagram for three points is shown in Fig. 4.2. To draw the Voronoi diagram, we find the Delaunay triangulation of the dual graph, i.e., the three points triangle. The bisectors of each side collide at the center of a circle that passes through the three points, and represents the Voronoi diagram.

The Voronoi region each represents a grain which can be magnetized. These grains in the Voronoi diagram are indicated by their centers, i.e., grain nuclei. The number of grains on the media can be calculated by knowing the media density and average grain size. However, the initial uniformly generated set of grain nuclei does not produce the desired statistics. The following steps are taken to readjust the randomly generated center set: i) Replacing the too small grains; ii) Breaking up



(a) (b)

Figure 4.3: Voronoi regions for centers placed (a) equally distanced placed, (b) and randomly shifted version [75].

the too big grains by adding extra points; iii) Shrinking the grain size to account for non-magnetic boundaries of each grain.

After the grains are structured on the virtual media with the desired statistics through refined Voronoi diagram, the next step is to assign the bit cells. Each bit cell indicates the grains that belong to one bit of information, as depicted in Fig. 4.4. The associated grains for each bit are assigned randomly with respect to the center of grain by an indicator function - zero if outside the bit cell and one if inside the bit cell. The aspect ratio of the rectangle that encompasses the bit cell is known as bit aspect ratio (BAR).

TDMR is composed of two techniques: shingled writing and 2-D signal processing. Several papers in the literature [76, 72, 6] have emphasized the importance of the Voronoi model for TDMR signal processing. We note that although the models for emerging magnetic recording devices do not share the same physics, they all suffer from the same signal processing issue, namely the 2-D detection problem. The higher areal density inevitably brings the tracks closer together. Therefore, interference from side tracks, which corresponds to 2-D ISI, will complicate the

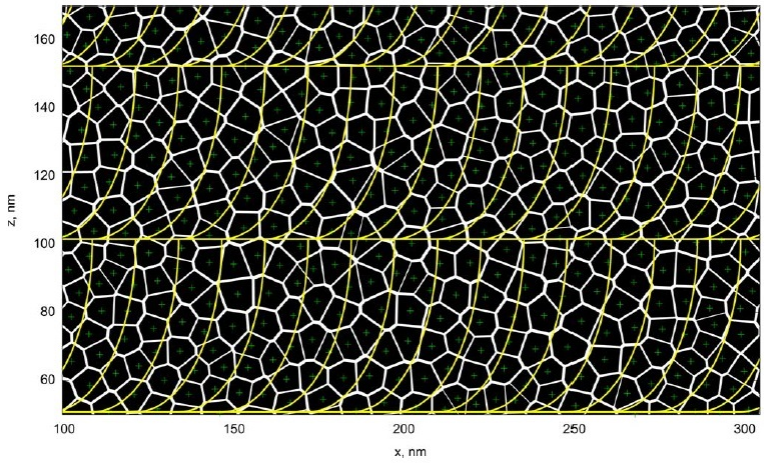


Figure 4.4: Simulated grains with their centers as green dots and shingled writing boundaries as yellow indicating each bit [72].

detection. In the next section, we explain the read and write processes which store and retrieve the data on the media.

4.2.1 Write Process (Shingled Magnetic Recording)

The write process consists of a moving head magnetizing the ferromagnetic material on the underlying substrate. On a magnetic disc a thin layer of ferromagnetic material (e.g., CoCr of 15 nm thickness) is deposited on top of SUL, (e.g., NiFe of 80 nm thickness) [77]. The ferromagnetic layer is grown by a sputtering process which results in random grain size and shapes. Fig. 4.5 shows a STEM (scanning transmission electron microscopy) image of the grains on the surface of magnetic recording substrate.

To write each bit, the magnetic field emanating from the write pole must be strong enough to flip the polarization of grains, penetrating through thickness of ferromagnetic medium. The magnetic permeable SUL acts as a magnetic image which allows a single pole head to record the data on the ferromagnetic layer. Smaller

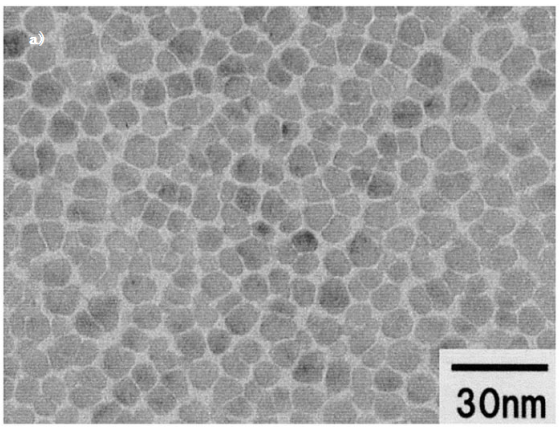


Figure 4.5: STEM view of grains in CoCrPt:Sio2 thin layer [78].

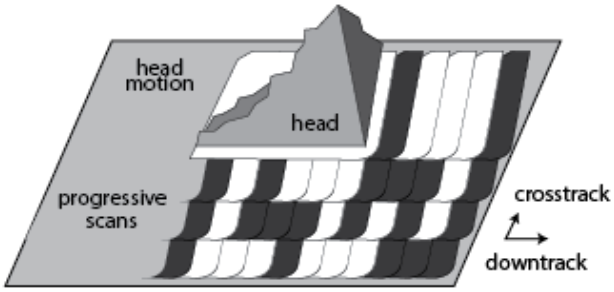


Figure 4.6: Writing process on a shingled magnetic recording disc [6].

size of grains will lead to thermal instability. The challenge is that at smaller scales the local thermal fluctuations may end up de-magnetizing the stored data. The shingled writing process manages to maintain the higher magnetic field by keeping the larger size of write head while preserving the thermal stability. Shingled writing empowers the higher magnetizing flux on the write corner by writing overlapped data on top of each other. In this configuration, the tracks are written sequentially towards the center. These write tracks overlap each other, so when the next track is written the narrow part that is left behind from the previous track represents the data. The process is shown in Fig (4.6).

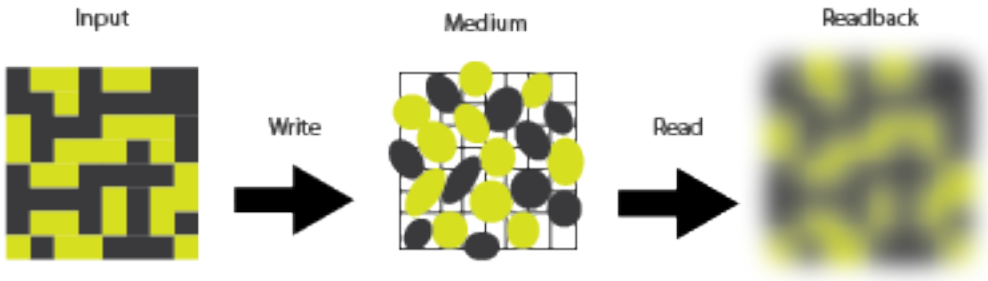


Figure 4.7: Illustration of write and read processes on the magnetic media [73].

4.2.2 Readback Modeling (Voronoi Model)

In this section, the focus is on the modeling of the nature of the readback process. Unlike the write head, the readback sensor can be made significantly small (e.g., 3nm×4nm [6]). The grains' development on the surface is a factor in our model in addition to readback sensor's characteristics.

The statistical model for grain distribution is generated using Voronoi regions; with known average grain area and its standard deviation, the random grains are distributed over a Euclidean plane. An example of such a simulated plane is illustrated in Fig. 4.4. Once a web of random grains is created, the probability of a given grain being magnetized to the value of bit s on track r (bits indicated by yellow boundaries in Fig. 4.4) is given by

$$p_m^{v_{r,s}}(x,y) = \mathbb{I}_{\{(x,y)\in\mathcal{A}_{v_{r,s}}} * G(x,y)\Big|_{v_{r,s}}, \tag{4.1}$$

where $\mathbb{I}$ is the indicator function for point (x,y) being in Voronoi region of point s and G is a 2-D Gaussian function centered at s.

The readback response model is captured by convolution of magnetization func-

tion $m(x, y)$ and head response $h(x, y)$. The $m(x, y)$ describes the magnetization value of the grain to which point x, y belongs. The head response is defined as

$$h(x,y) = \frac{1}{2\pi\sigma_x\sigma_y} exp\left[-\frac{1}{2}\left(\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}\right)\right] - \frac{1}{2\pi l_x l_y} K_0\left(\frac{x}{l_x^2} + \frac{y}{l_y^2} + \epsilon\right), \tag{4.2}$$

where K_0 is the Bessel function and l_x, l_y indicates the width of undershoot in head response (the small amount ϵ is added to avoid singularity of Bessel function). The parameter σ_x, σ_y are responsible for bulk of Gaussian shape of head response.

During the readback, the read head captures the magnetic influx from neighboring bit cells. These interferences come from downtrack bit cells as well as side tracks bit cells. The results of readback are depicted in Fig. 4.8 for different densities. The figures shows that the blurring effect of the readback sensor for higher densities is pronounced. This amounts to higher 2-D ISI from both directions, which makes the detection more complicated.

4.3 Other Technologies

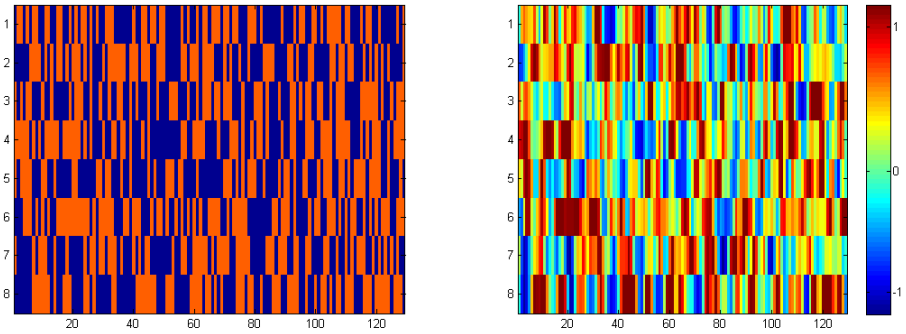
4.3.1 Bit Patterned Magnetic Recording

An alternative competing technology for magnetic storage devices is the BPMR. On the surface of the disc, the data is stored on separated islands, which are fabricated with the help of lithography. As a result, the BPMR does not suffer from transition noise due to irregular boundaries for higher densities. The thermal stability is less of a problem as the whole island switches magnetization as opposed to each grain in conventional medium. However, the traditional lithography does not provide adequate density for BPMR. For island sizes of less than 20nm electron beam

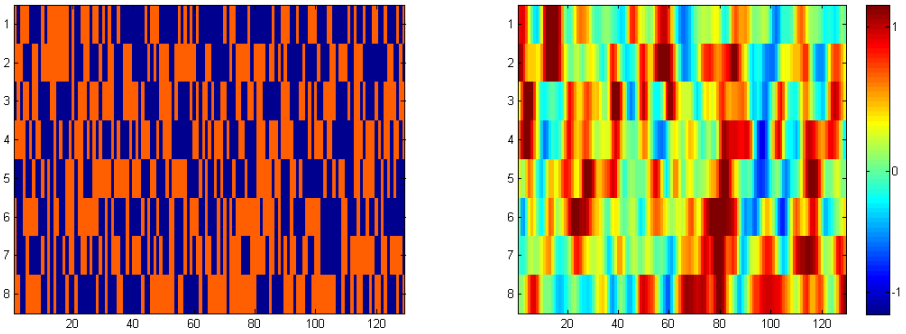
lithography is required. The currently available electron beam lithography does not yield enough accuracy. The variation in island position and size are common problem at higher densities. These variations will steer any small mis-synchronization or fluctuation to a writing error. At higher densities the ITI and ISI will be still present in BPMR due to the close proximity of islands.

4.3.2 Heat Assisted Magnetic Recording

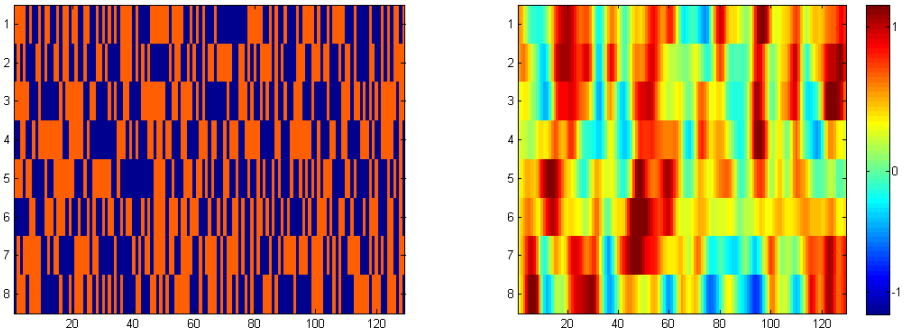
Another technology to help alleviate the concerns with thermal instability of medium is HAMR. In HAMR the grains are heated locally to facilitate the writing process. For a highly anisotropic magnetic material, the temperature is raised on the specific grains so that the magnetization switching happens at a lower magnetic field. After writing process, the grains cool down quickly and the data is stored on the medium. The heating process can be done by means of a guided laser beam and it can happen in the scale of a nanosecond. The remaining parts of the read and write heads will be the same as perpendicular magnetic recording. The challenges facing HAMR are the medium with thermal and magnetic stability for repeated heating process, and integration of optical and magnetic head resistant to high temperature.



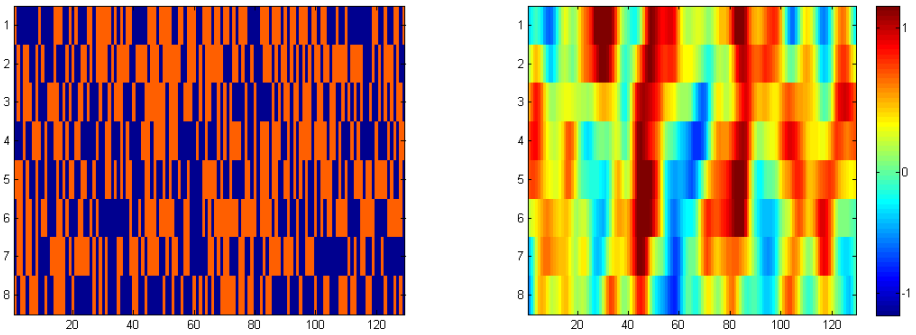
(a) Voronoi model normalized output and input data at 1Tb, BAR=3.



(b) Voronoi model normalized output and input data at 4Tb, BAR=3.



(c) Voronoi model normalized output and input data at 7Tb, BAR=3.



(d) Voronoi model normalized output and input data at 10Tb, BAR=3.

Figure 4.8: Readback sensor output of Voronoi diagram and input data for different densities.

5 Two-Dimensional Detection

5.1 Introduction

The MLSD for channels with ISI has an important role in digital communications [79]. Digital data is mapped to analog signals before transmission or storage. The received signal is converted back to digital with the help of a detector. The detector makes a great impact on the overall performance of the system. At the detector the decisions are made based on minimizing the risk and therefore reducing the probability of error. The objective is to find an MLSD for a channel with a 2-D pulse response. In short, the answer in polynomial time is not available to-date [56]. Nonetheless, the interest in 2-D detection encourages us to seek sub-optimal solutions with manageable complexity.

In this chapter, a 2-D communication system and its components are described. The challenges of a 2-D storage channel are discussed. A Voronoi model that closely approximates a magnetic recording channel is simulated. The random outputs of the model simulations are used to determine the equalizer's coefficients, target, and performance of a 2-D detector. Our contribution is to show the improvement on the performance by means of 2-D detection. For higher densities, the performance of the detector degrades. This is not only due to significant amount of noise, but also due to spatial band-limitedness of this magnetic recording channel; the violation of

the 2-D Nyquist ISI criterion leads to poor performance.

5.2 Two-Dimensional Channel Model

The following notation is used through out this chapter. We use uppercase letters for random variables and lower case for their realization. Matrices are shown in underlined bold uppercase letters and vectors are designated either in bold lowercase letters, e.g., $\mathbf{x}$, or through enumeration of their elements, $\mathbf{x}_1^K = [x_1, x_2, ...x_K]$ where x_i is the i-th element of vector $\mathbf{x}$. The $\langle \cdot, \cdot \rangle$ shows the inner product. The parentheses represent a function, e.g., $f(\cdot)$.

One of the major disturbance is ISI and ITI: The pulse responses of a band-limited channel overlap, meaning the received signal not only depends on the transmitted data at the current state of time but also on other transmitted data. How the interference occurs plays an important role in deciphering the original message. Suppose the stream of binary data $\overline{a}_{i,j} \in \{-1, 1\}$ are stored. Given a 2-D pulse response, $p(x, y) \in \mathcal{L}_2(\mathbb{R}^2)$, the interference from adjacent input data can be modeled as

$$\overline{y}(x, y) = \sum_{m=1}^{K_x} \sum_{n=1}^{K_y} \overline{a}_{r-m,s-n} p(x - nT, y - mT) + n(x, y), \tag{5.1}$$

where $n(x, y)$ is AWGN with auto-correlation function $\sigma_n^2 \delta(x, y)$. The received sample at location r, s is given by (for $p(x, y)$ having finite support)

$$\overline{y}_{r,s} = \sum_{j=-J}^{J} \sum_{i=-I}^{I} \overline{a}_{r-j,s-i} p(x - iT, y - jT) * h(x - sT, y - rT) + \overline{n}_{r,s}, \tag{5.2}$$

where $h(x, y) \in \mathcal{L}_2(\mathbb{R}^2)$ is the matched filter output and $*$ represent the 2-D convo-

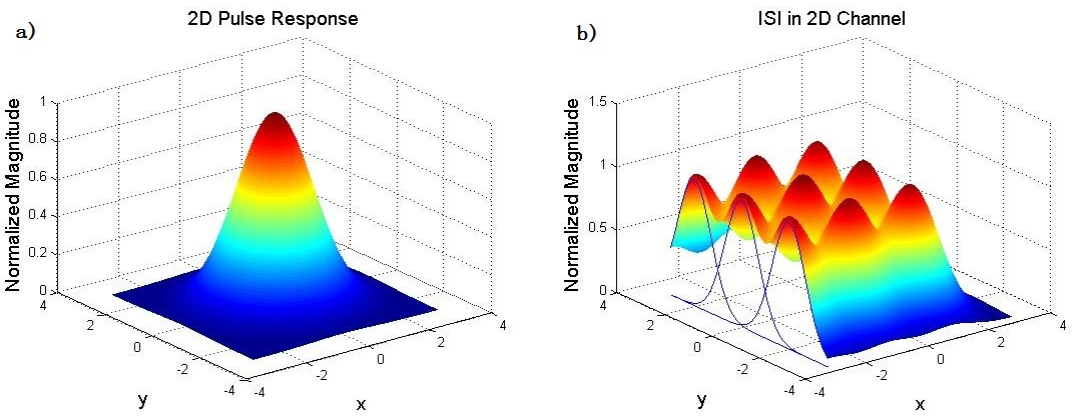


Figure 5.1: a) The normalized 2-D pulse response of a channel with respect to its parameters x, y. b) Overlapping 2-D pulse responses of a channel shows ISI in both directions.

lution operator evaluated at origin. The last terms $\bar{n}_{r,s}$ represent the AWGN with variance σ_n^2.

Alternatively, we derive the equivalent formulation in vector form by concatenating rows of input data and received samples

$$\mathbf{a} \triangleq [\overline{a}_{1,1}...\overline{a}_{K_y,1}, \overline{a}_{1,2}, ...\overline{a}_{K_y,2}, ...\overline{a}_{1,K_x}, ..., \overline{a}_{K_y,K_x}]^T, \tag{5.3}$$

$$\mathbf{y} \triangleq [\overline{y}_{1,1}...\overline{y}_{K_y,1}, \overline{y}_{1,2}, ...\overline{y}_{K_y,2}, ...\overline{y}_{1,K_x}, ..., \overline{y}_{K_y,K_x}]^T, \tag{5.4}$$

which result in a form similar to the one dimension (1-D) problem

$$\begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix} = \mathbf{H} \begin{bmatrix} a_1 \\ \vdots \\ a_N \end{bmatrix} + \begin{bmatrix} n_1 \\ \vdots \\ n_N \end{bmatrix}, \tag{5.5}$$

with matrix $\mathbf{H}$ consisting of elements $h_{i,j}$ defined as $p(x,y) * h(x - iT, y - jT)$,

$$\begin{bmatrix} h_{-J,-I}\ldots & \mathbf{0}_{1\times K_{y-2J}} & h_{-J,-I}\ldots h_{J,-I} & ,\mathbf{0}_{1\times K_{y-2J}} & h_{-J,I}\cdots h_{J,I} & \\ & h_{-J,-I}\ldots h_{J,-I} & \mathbf{0}_{1\times K_{y-2J}} & \cdots & & \\ & & & & \ddots & \\ & & h_{-J,-I}\ldots h_{J,-I} & ,\mathbf{0} & h_{-J,-I}\ldots h_{J,-I} & \cdots h_{J,I} \end{bmatrix}. \tag{5.6}$$

5.3 Two-Dimensional Equalizer

In practice, the pulse response of a channel has a long tail which in turn results in higher complexity for the detector. To combat the problem the channel response is shaped into the desired structure using an equalizer.

Let $I^{i,j}$, $J^{i,j}$ be the finite sequence of ordered pair that represent a window centered at i,j of size K and M respectively. The notation simplifies the equalizer to 1-D counterpart [80]. The steps to derive an equalizer are as follows

- Choose the suitable size for vectors $\mathbf{f}_1^M, \mathbf{g}_1^K$
- Define $e_{i,j} = \mathbf{f}[\overline{y}_{I_1^{i,j}}, ..., \overline{y}_{I_M^{i,j}}]^T - \mathbf{g}[a_{J_1^{i,j}}, ..., a_{J_K^{i,j}}]^T$
- Minimize the best linear estimator's mean squared error, $E[e_{i,j}^2|\mathbf{f},\mathbf{g}]$ with respect to constraint $g_{\lceil K/2\rceil} = 1$ to find $\mathbf{f},\mathbf{g}$.

The constraint is to prevent the trivial answer. The solution to the optimization problem is given by the Lagrange method,

$$\sigma_e^2 = E[e_{i,j}^2|\mathbf{f},\mathbf{g}]+\lambda(\mathbf{p}\mathbf{g}^T-1) = \mathbf{f}E[\tilde{\mathbf{y}}^T\tilde{\mathbf{y}}]\mathbf{f}^T+\mathbf{g}E[\tilde{\mathbf{a}}^T\tilde{\mathbf{a}}]\mathbf{g}^T-2\mathbf{f}E[\tilde{\mathbf{y}}^T\tilde{\mathbf{a}}]\mathbf{g}^T+-\lambda(\mathbf{p}\mathbf{g}^T-1), \tag{5.7}$$

where $\tilde{y} = [y_{I_1^{i,j}}, ..., y_{I_M^{i,j}}]$, $\tilde{\mathbf{a}} = [a_{J_1^{i,j}}, ..., a_{J_K^{i,j}}]$, and $\mathbf{p} = [0, ...0, 1, 0, ...0]$. To find the vector $\mathbf{f}, \mathbf{g}$ we set $\frac{\partial}{\partial \mathbf{f}} \sigma_e^2 = 0$,

$$(E[\tilde{\mathbf{y}}^T \tilde{\mathbf{y}}]^T + E[\tilde{\mathbf{y}}^T \tilde{\mathbf{y}}])\mathbf{f}^T - 2E[\tilde{\mathbf{y}}^T \tilde{\mathbf{a}}]\mathbf{g}^T = 0. \tag{5.8}$$

Replacing $\mathbf{f}$ from (5.8) into (5.7) and taking the derivative with respect to $\mathbf{g}$ we find

$$2E[\tilde{\mathbf{y}}^T \tilde{\mathbf{a}}]^T E[\tilde{\mathbf{y}}^T \tilde{\mathbf{y}}]^{-1} E[\tilde{\mathbf{y}}^T \tilde{\mathbf{a}}]\mathbf{g}^T + 2E[\tilde{\mathbf{a}}^T \tilde{\mathbf{a}}]\mathbf{g}^T - 4E[\tilde{\mathbf{y}}^T \tilde{\mathbf{a}}]^T E[\tilde{\mathbf{y}}^T \tilde{\mathbf{y}}]^{-1} E[\tilde{\mathbf{y}}^T \tilde{\mathbf{a}}]\mathbf{g}^T + \lambda \mathbf{p}^T = 0, \tag{5.9}$$

and further setting $\frac{\partial}{\partial \mathbf{g}} \sigma_e^2 = 0$ leaves us

$$\mathbf{g}^T = 0.5 \left(E[\tilde{\mathbf{a}}^T \tilde{\mathbf{a}}] - E[\tilde{\mathbf{y}}^T \tilde{\mathbf{a}}]^T E[\tilde{\mathbf{y}}^T \tilde{\mathbf{y}}]^{-1} E[\tilde{\mathbf{y}}^T \tilde{\mathbf{a}}] \right)^{-1} \lambda \mathbf{p}^T, \tag{5.10}$$

and therefore

$$\lambda = 2 / \left[\mathbf{p} \left(E[\tilde{\mathbf{a}}^T \tilde{\mathbf{a}}] - E[\tilde{\mathbf{y}}^T \tilde{\mathbf{a}}]^T E[\tilde{\mathbf{y}}^T \tilde{\mathbf{y}}]^{-1} E[\tilde{\mathbf{y}}^T \tilde{\mathbf{a}}] \right)^{-1} \mathbf{p}^T \right]. \tag{5.11}$$

Now, we can replace $h_{i,j}$ in (5.5) with the corresponding elements of the target, i.e., vector $\mathbf{g}$ and the received samples are the output of the equalizer. The equalized signal has the desired properties of the channel which is forced on the target by means of constraints and size.

5.3.1 Implementation

Table 5.1 shows the pattern dependency of the noise for $1.2Tb/in^2$ and BAR=4. Also we see the effect of head size on the noise. The head size of 15 nm shows better performance on the chosen areal density. The size of the head has two effects: i)

$\sigma_x/pattern$	0	1	2	3	4	5	6	7
1nm	0.29	0.24	0.52	0.50	0.51	0.52	0.21	0.30
2nm	0.29	0.24	0.49	0.48	0.48	0.49	0.22	0.30
5nm	0.25	0.21	0.33	0.34	0.34	0.33	0.20	0.25
10nm	0.20	0.17	0.25	0.25	0.25	0.25	0.17	0.20
15nm	0.18	0.15	0.25	0.26	0.26	0.25	0.15	0.18
18nm	0.19	0.15	0.28	0.29	0.29	0.28	0.15	0.19

Table 5.1: Variance of noise at the output of equalizer for different head sizes ($\sigma_x = \sigma_z$) $1.2Tb/in^2$, BAR=4.

the smaller read head size would make the response more sensitive to the fuzzy boundaries of grains; ii) read head size determines the amount of interference in the channel response.

5.4 Areal Density and Nyquist ISI Criterion

This section discusses the limits and benefits of 2-D signal processing for shingled magnetic recording devices. The minimum size for grains that ensures the thermal stability dictates the areal density of the magnetic medium. However, we would like to discuss the shapes of the bit cells for higher densities which affects the signal processing. The magnetization of these bit cells is captured through a read head sensor. The shape of the read head response is spatially band-limited. With shrinking size of bit cells, for higher areal density, the spatial bandlimited response of the channel results in poor performance of the signal processing unit. We aim to study the effect of such a matter on shingled magnetic recording devices.

The goal of currently available signal processing methods is to seek suboptimal solutions in a channel with 2-D pulse responses with manageable complexity. The interest in 2-D detection encourages us to seek the benefits of such a detector. Our methodology enables us to answer the following questions:

1. What is the best read head layout?

2. How the performance of the system is affected by shrinking the tracks closer together?

In this section, the focus is on the modeling of the readback process. Unlike the write-head, the readback sensor can be made significantly small (e.g., 3nm $\times$ 4nm [81]). The head response is defined as 2-D Gaussian function and a Bessel function which indicates the width of undershoot in head response. If we consider the undershoot negligible the received signal from reading a plane is given by

$$s(x,y) = \sum_{m=1}^{N}\sum_{k=1}^{K} a_{k,m} \exp\left[-\frac{1}{2}\left(\frac{(x-m/\alpha)^2}{\sigma_x^2} + \frac{(y-k/\beta)^2}{\sigma_y^2}\right)\right] + n(x,y), \qquad (5.12)$$

where $a_{k,m}$'s are chosen from a binary alphabet and $n(x,y)$ is the observed noise. The parameters σ_x, σ_y are responsible for the bulk of the Gaussian shape of the head response, and α, β are measures of distance between centers of the bit cells from one another in each direction. The higher values of α, β indicate shrinking in the x or y direction, respectively.

Using the simple model in (5.12) we are able to numerically calculate the normalized minimum distance over pulse energy for different head parameters [69]. The minimum distance gives us the lower bound for the probability of error. The result of our simulations with symmetric head size for different values of α, β is shown in Fig. 5.2. For the fixed head response, shrinking of bit cell size or squeezing the tracks closer together will lead to loss in performance of the detector. As depicted in our model (5.12), the degradation in performance occurs regardless of the noise characteristics. The result also shows that ITI does not necessarily degrade the performance of the system. Also, the number of erroneous bits for corresponding

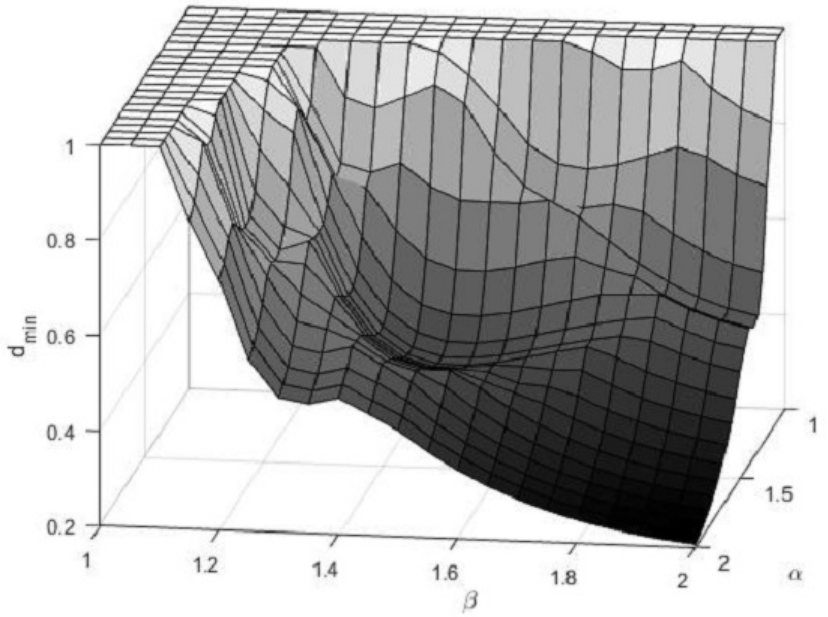


Figure 5.2: Normalized minimum distance for different parameters of the channel pulse response.

error events is given in Table 5.2. These errors are particularly interesting in the design of suboptimal 2-D detectors which process stripes of rows or columns at a time; the 3x3 mask for the 2-D detector would be able to observe the error events of size 9, however, for higher number of error bits a larger mask is needed which itself increases the complexity of the detector significantly. Next, we run the simulations on a Voronoi model of a shingled magnetic recording at different densities over $1Tb/in^2$ for different readback sensor head sizes. The results are shown in Table 5.3. The performance of a PDNP detector degrades for higher densities. This can be associated with loss in minimum distance as well as higher noise variance.

α/β	2	1.8	1.6	1.4	1.2	1.0
2.0	9	9	9	49	121	1
1.8	9	9	9	169	81	1
1.6	9	9	25	25	81	1
1.4	49	169	25	25	1	1
1.2	121	81	81	1	1	1
1.0	1	1	1	1	1	1

Table 5.2: Number of erroneous bits for minimum distance events.

Density (Tb/in2)	**BAR**	**Noise Variance**	**Bit-cell Width (nm)**	**Track Pitch (nm)**	**Read Head Size (nm2)**	**BER**
1.5	4	0.5719	10.37	41.48	2×2	0.229
1.5	4	0.3710	10.37	41.48	7×7	0.164
1.5	4	0.3442	10.37	41.48	12×12	0.170
1.5	4	0.4033	10.37	41.48	18×18	0.249
1.2	4	0.5061	11.59	46.37	2×2	0.2114
1.2	4	0.3409	11.59	46.37	5×5	0.1511
1.2	4	0.2688	11.59	46.37	10×10	0.1094
1.2	4	0.2817	11.59	46.37	15×15	0.1447
1	3	0.2076	14.66	43.99	6.48×6.48	0.061

Table 5.3: Simulation results for shingled Voronoi model.

5.5 Two-Dimensional Detector

In this section we discuss the performance of a 2-D detector on a Voronoi model of a shingled magnetic recording channel. Several suboptimal 2-D detectors have been proposed in the literature. We here briefly go over the currently available 2-D detectors. The ITI canceling equalizer can turn the 2-D problem into one dimension [82]. Wu *et al.* [83] imposed a constraint on the channel characteristics to derive a separable 2-D detector. The assumed separable channel has the property that the channel response can be written as a product of two vectors. With this configuration in place, it was shown that two 1-D detectors can be used on rows and columns; the first detector to handle the ITI is a non-binary detector concatenated with a second binary MAP detector. These detectors iteratively exchange soft values to achieve close to optimum performance. The exchange of soft values can be seen from a belief propagation (BP) perspective too. The BP algorithm gives an exact solution on a tree factor graph, but becomes suboptimal in a loopy factor graph which stands for our channel with memory [84, 85]. Lehmann [86, 87] designed a BP Kalman filter to estimate the signal with a considerable low complexity. However, the performance of such a detector is inferior to trellis based detectors. In [88], Khatami and Vasic used the idea of BP and suggested to bundle the graph nodes. These bundles based on the local constraints form an opinion and exchange information with other nodes by sending messages to achieve a consensus on the estimation of the transmitted signal.

We use a row and column detector. The basic idea is that two detectors sweep stripes of data for each column or row and soft values are exchanged at the output of each individual detector. In fact, each detector assumes the data is laid in a narrow long ribbon band where data on column/row is boxed into one symbol.

These symbols overlap as the detector moves forward to the next symbol. There are several variations of such a detector. Authors in [89] introduced a reduced state detector. Cheng *et al.* [90] added a soft decision feedback from adjacent side tracks and considered different sweeping combinations such as zig-zag iterations [91]. We stick to the basic row and column detector with mask size of 3×3. Our goal here is not to compare different 2-D detectors, but to assess the performance of a 2-D detector on a Voronoi model of a shingled magnetic recording channel. We are specifically interested in examining the effect of ITI at the output of the detector.

We run the Voronoi model to generate 20 batches of 8 tracks each containing 16384 bits. We choose 1.5 and 1.2 Tb/in^2 as the areal density for our simulations. The 1.5 Tb/in^2 has 6.67 grains in each bit cell on average and 1.2 Tb/in^2 has 8.33 grains. The head size is kept constant at $10 \times 10nm^2$ for $1.2Tb/in^2$ and $7 \times 7nm^2$ for 1.5 Tb/in^2. Our aim here is to investigate the effect of squeezing the tracks closer together. By squeezing the tracks we increase the ITI from side tracks. At the same time, the change in the layout of grains on the surface could change the noise variance at the output of equalizer. For BAR greater than 3 the ITI in the channel is very small. For a fixed density, the performance of the detector is determined by the minimum distance and SNR. Interestingly, for 1.2 Tb/in^2 we see that number of errors are halved for BAR=3 and BAR=1. The BAR=3 corresponds to less ITI and BAR=1 represents the same density but more interference with side-tracks. The SNR for both BARs is almost the same. The only explanation for such improvement is a minimum distance increase. The ITI from side-tracks helps to achieve a better performance. However, as we observed in Fig. 5.2, further increasing the ITI lead to poorer performance as seen for BAR=0.5. As we increase the density to 1.5 Tb/in^2, the probability of error increases. This is partially due to lower SNR at higher densities.

Density (Tb/in2)	**BAR**	**Noise Variance**	**SNR (dB)**	**Bit-cell Width (nm)**	**Track Pitch (nm)**	**BER**
1.5	0.5	0.4707	4.14	29.32	14.66	0.2077
1.5	1	0.3551	5.46	20.73	20.73	0.1522
1.5	4	0.3671	5.15	10.36	41.47	0.1821
1.2	0.5	0.3483	5.59	31.07	17.29	0. 1627
1.2	1	0.2187	7.22	23.18	23.18	0.0715
1.2	3	0.2432	7.16	13.38	40.16	0.1410
1.2	4	0.2650	6.77	11.59	46.37	0.1510

Table 5.4: Simulation results for shingled Voronoi model.

6 Conclusion

In this dissertation, advanced signal processing techniques for shingle magnetic recording are investigated, and novel detection methods are proposed. This chapter outlines the contributions of this dissertation.

Chapter 2 proposes a noise predictive detection algorithm for ISI channels with linear regressive noise. The MLSD algorithm for ISI channels with correlated noise has been known for Gauss-Markov noise and is widely used for magnetic recording channels. Our contribution is the derivation of a noise predictive algorithm that is tailored to the linear regressive noise model. A block factorization of the covariance matrix of the linear regressive Gaussian noise is used to derive the branch metrics. This algorithm is tested on a partial response channel and exhibits near optimal performance. A generalization of this algorithm to signal dependent linear regressive noise is also presented and its performance improvement over conventional algorithms with comparable complexity is shown using simulation results.

Chapter 3 proposes a layout of a communication system that deals with a nuisance parameter. This chapter presents a sequence detector in the presence of a random nuisance parameter with a continuous distribution. Conventionally, the nuisance parameter assumed to contribute to the observed noise at the receiver and therefore degrades the performance of the system. We show that with the right setup the disturbance from the nuisance parameter in the received signal can be

alleviated. Our first contribution is the derivation of a Viterbi like detector for sequence detection in the presence of random jitter with polynomial time computational complexity. We propose the layout of an over-sampling communication system including sampling, equalization, and detection; derive a detection algorithm based on the Gauss-Markov assumption. We also derive a genie-aided bound for the detector. Our second contribution is to study the performance of our detector with respect to minimum distance. The performance of the detector is a function of the minimum distance of the channel which is subject to changes due to the random parameter. Our aim is to show the benefit of exploiting the nuisance parameter in boosting the performance of the system, especially once the system performs at the boundaries of the Nyquist ISI criterion. The proposed system is tested for Gaussian and sinc-shaped pulses. The detector exhibits a performance gain over conventional detectors. These gains are magnified at higher transmission rates. The simulation results for Gaussian pulses and sinc pulses show that despite the sub-optimality of our detection, the bit-error probability remains relatively close to optimum.

In Chapters 4 and 5, we explore the challenges of signal processing for a storage channel. In Chapter 4, we present the micro-magnetic model of hard-disk drives and discuss the basics of the Voronoi model. We study the detailed parameters of the Voronoi model and their effects on the channel output. This model gives an accurate imitation of the storage channel and is used for evaluating the performance of detection methods. Chapter 5 is dedicated to the discussion of 2-D signal processing techniques on a storage channel. We give an overview of each component of a 2-D communication system, 2-D channel model, 2-D channel equalizers, and 2-D detection. Our contribution in this chapter is to show the effectiveness and limits of 2-D detection on a storage channel. We provide an analytical relation between the performance of a 2-D detector and minimum distance of the channel response. We

also illustrate the improved performance of a 2-D detector for the Voronoi model at an areal density that is beyond the current standard in the industry.

Bibliography

[1] G. Bell, J. Gray, and A. Szalay, "Petascale computational systems," *Comput.*, vol. 39, no. 1, pp. 110–112, Jan. 2006.

[2] C. M. Christensen, "The rigid disk drive industry: A history of commercial and technological turbulence," *Business History Review*, vol. 67, no. 04, pp. 531–588, Oct. 1993.

[3] R. Wood, "The feasibility of magnetic recording at 1 terabit per square inch," *IEEE Trans. Magn.*, vol. 36, no. 1, pp. 36–42, Jan. 2000.

[4] M. Kryder, E. Gage, T. McDaniel, W. Challener, R. Rottmayer, J. Ganping, Y. Hsia, and M. F. Erden, "Heat assisted magnetic recording," *Proc. of the IEEE*, vol. 96, no. 11, pp. 1810–1835, Nov. 2008.

[5] C. Ross, "Patterned magnetic recording media," *Annu. Rev. Mater. Res.*, vol. 31, no. 1, pp. 203–235, Nov. 2001.

[6] 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.

[7] E. Hwang, R. Negi, B. V. K. V. Kumar, and R. Wood, "Investigation of two-dimensional magnetic recording (TDMR) with position and timing uncertainty at 4 Tb/in^2," *IEEE Trans. Magn.*, vol. 47, no. 12, pp. 4775–4780, Dec. 2011.

[8] J. Heidmann and M. T. Alexander, *Handbook of Magnetic Materials (Volume 19) Chapter one: Magnetic Recording Head.* Amsterdam, Holland: Elsevier, 2011.

[9] M. Mallary, A. Torabi, and M. Benakli, "One terabit per square inch perpendicular recording conceptual design," *IEEE Trans. Magn.*, vol. 38, no. 4, pp. 1719–1724, Jul. 2002.

[10] D. Litvinov and S. Khizroev, "Perpendicular magnetic recording: Playback," *J. Appl. Phys.*, vol. 97, no. 7, pp. 071–101, Apr. 2005.

[11] E. E. Tyrtyshnikov, "How bad are Hankel matrices?" *Numerische Mathematik*, vol. 67, no. 2, pp. 261–269, Mar. 1994.

[12] E. Y. Tsymbal and I. Zutic, *Handbook of spin transport and magnetism.* Boca Raton, FL: CRC Press, 2012.

[13] N. F. Mott, "The basis of the electron theory of metals, with special reference to the transition metals," *Proc. Phys. Soc. A*, vol. 62, no. 7, pp. 416–422, Apr. 1949.

[14] J. Bass, "Giant magnetoresistance: Experiment," in *Handbook of Spin Transport and Magnetism*, Tsymbal and Zutic, Eds. Boca Raton, FL: CRC Press, 2012.

[15] P. Kovintavewat, I. Ozgunes, E. Kurtas, J. Barry, and S. McLaughlin, "Generalized partial-response targets for perpendicular recording with jitter noise," *IEEE Trans. Magn.*, vol. 38, no. 5, pp. 2340–2342, Sep. 2002.

[16] G. Forney, "The Viterbi algorithm," *Proc. of the IEEE*, vol. 61, no. 3, pp. 268–278, Mar. 1973.

[17] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. Inform. Theory*, vol. 20, no. 2, pp. 284–287, Mar. 1974.

[18] M. Tuchler, R. Koetter, and A. Singer, "Turbo equalization: principles and new results," *IEEE Trans. Commun.*, vol. 50, no. 5, pp. 754–767, May 2002.

[19] I. Lee and J. M. Cioffi, "Performance analysis of the modified maximum likelihood sequence detector in the presence of data-dependent noise," in *26th Asilomar Conf. on Signals, Systems and Computers*, Oct. 1992, pp. 961–964.

[20] A. Kavčić and J. M. Moura, "The Viterbi algorithm and Markov noise memory," *Trans. Inform. Theory*, vol. 46, no. 1, pp. 291–301, Jan. 2000.

[21] R. G. Gallager, *Information theory and reliable communication.* New York, NY: Wiley, 1968.

[22] R. Gallager, "Low-density parity-check codes," *IEEE Trans. Inform. Theory*, vol. 8, no. 1, pp. 21–28, Jan. 1962.

[23] R. MacKay, D.J.C.Neal, "Near shannon limit performance of low density parity check codes," *Electron. Lett.*, vol. 33, no. 6, p. 457, Aug. 1997.

[24] R. Richardson, T.J.Urbanke, "The capacity of low-density parity-check codes under message-passing decoding," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 599–618, Feb. 2001.

[25] D. J. Lin, ShuCostello, *Error control coding.* Englewood Cliffs, N.J.: Prentice-Hall, 1983.

[26] F. Kschischang, B. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 498–519, Feb. 2001.

[27] R. Tanner, "A recursive approach to low complexity codes," *IEEE Trans. Inform. Theory*, vol. 27, no. 5, pp. 533–547, Sep. 1981.

[28] H. J. Mooij, Joris M.Kappen, "Sufficient conditions for convergence of the sum product algorithm," *IEEE Trans. Inform. Theory*, vol. 53, no. 12, pp. 4422–4437, Dec. 2007.

[29] E. Arikan, "Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels," *IEEE Trans. Inform. Theory*, vol. 55, no. 7, pp. 3051–3073, Jul. 2009.

[30] A. Abbasfar, D. Divsalar, and K. Yao, "Accumulate-repeat-accumulate codes," *IEEE Trans. Commun.*, vol. 55, no. 4, pp. 692–702, Apr. 2007.

[31] A. Berrou, C.Glavieux, "Near optimum error correcting coding and decoding: turbo-codes," *IEEE Trans. Commun.*, vol. 44, no. 10, pp. 1261–1271, Oct. 1996.

[32] D. Grimmett, GeoffreyStirzaker, *Probability and random processes.* Oxford, UK: Oxford Univ. Press, 2001.

[33] J. Coker, E. Eleftheriou, R. Galbraith, and W. Hirt, "Noise-predictive maximum likelihood (NPML) detection," *IEEE Trans. Magn.*, vol. 34, no. 1, pp. 110–117, Jan. 1998.

[34] S. Gratrix, R. Jackson, T. Parnell, and O. Zaboronski, "Viterbi detector for non-Markov recording channels," *IEEE Trans. Magn.*, vol. 44, no. 1, pp. 198–206, Jan. 2008.

[35] G. Phadke, M. S.Kedem, "Computation of the exact likelihood function of multivariate moving average models," *Biometrika*, vol. 65, no. 3, p. 511, Dec. 1978.

[36] G. H. Golub and C. F. Van Loan, *Matrix computations.* Baltimore, MD: Johns Hopkins University Press, 1996.

[37] I. Gohberg and H. Georg, *Convolution Equations and Singular Integral Operators (Operator Theory: Advances and Applications 206).* Basel, Switzerland: Springer, 2010.

[38] G. T. Wilson, "The estimation of parameters in multivariate time series models," *J. of the Roy. Statistical Soc. Series B (Methodological)*, pp. 76–85, Aug. 1973.

[39] S. Altekar and J. Wolf, "Improvements in detectors based upon colored noise," *IEEE Trans. Magn.*, vol. 34, no. 1, pp. 94–97, Jan. 1998.

[40] G. Meurant, "A review on the inverse of symmetric tridiagonal and block tridiagonal matrices," *SIAM. J. Matrix Anal. & Appl.*, vol. 13, no. 3, pp. 707–728, Jul. 1992.

[41] Y. Kim and J. Moon, "Noise-predictive maximum-likelihood method combined with infinite impulse response equalization," *IEEE Trans. Magn.*, vol. 35, no. 6, pp. 4538–4543, Nov. 1999.

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

[43] ——, "Lower bounds on error probability in the presence of large intersymbol interference," *IEEE Trans. Commun.*, vol. 20, no. 1, pp. 76–77, Feb. 1972.

[44] W. F. Trench, "An algorithm for the inversion of finite Toeplitz matrices," *Journal of the Society for Industrial and Appl. Mathematics*, vol. 12, no. 3, pp. 515–522, Sep. 1964.

[45] M. K. Ng, *Iterative methods for Toeplitz systems.* Oxford, UK: Oxford Univ. Press, 2004.

[46] R. M. Gray, *Toeplitz and circulant matrices.* Boston, MA: Now Publisher, 2006.

[47] G. H. Hardy, *Divergent series.* New York, NY: Chelsea Pub. Co., 2000, vol. 334.

[48] Y. Wang, B. Yuan, K. Parhi, and R. Victora, "Two-dimensional magnetic recording using a rotated head array and ldpc code decoding," *IEEE Trans. Magn.*, vol. 50, no. 11, pp. 1–4, Nov. 2014.

[49] Y. Wang, M. F. Erden, and R. Victora, "Novel system design for readback at 10 terabits per square inch user areal density," *IEEE Magn. Lett.*, vol. 3, Dec. 2012.

[50] R. Pighi, R. Raheli, and U. Amadei, "Multidimensional signal processing and detection for storage systems with data-dependent transition noise," *IEEE Trans. Magn.*, vol. 42, no. 7, pp. 1905–1916, Jul. 2006.

[51] R. Wood, "Jitter vs. additive noise in magnetic recording: Effects on detection," *IEEE Trans. Magn.*, vol. 23, no. 5, pp. 2683–2685, Sep. 1987.

[52] A. Shami, X. Bai, C. M. Assi, and N. Ghani, "Jitter performance in Ethernet passive optical networks," *J. Lightw. Technol.*, vol. 23, no. 4, pp. 1745–1753, Apr. 2005.

[53] A. Demir, "Phase noise and timing jitter in oscillators with colored-noise sources," *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.*, vol. 49, no. 12, pp. 1782–1791, Dec. 2002.

[54] J. E. Mazo, "Faster-than-Nyquist signaling," *Bell System Techn. J.*, vol. 54, no. 8, pp. 1451–1462, Oct. 1975.

[55] Y. Lu and M. Do, "A theory for sampling signals from a union of subspaces," *IEEE Trans. Signal Process.*, vol. 56, no. 6, pp. 2334–2345, Jun. 2008.

[56] M. Vetterli, P. Marziliano, and T. Blu, "Sampling signals with finite rate of innovation," *IEEE Trans. Signal Process.*, vol. 50, no. 6, pp. 1417–1428, Jun. 2002.

[57] S. Karlin and W. J. Studden, *Tchebycheff systems: With applications in analysis and statistics.* New York, NY: Interscience Publishers, 1966.

[58] H. Finner, "A generalization of Holder's inequality and some probability inequalities," *The Ann. of Probability*, vol. 20, no. 4, pp. 1893–1901, Oct. 1992.

[59] A. Polyanin and A. V. DManzhirov, *Handbook of integral equations.* Boca Raton, FL: CRC Press, 1998.

[60] J. Moon and J. Park, "Pattern-dependent noise prediction in signal-dependent noise," *IEEE J. Select. Areas Commun.*, vol. 19, no. 4, pp. 730–743, Apr. 2001.

[61] P. Stoica, "Generalized Yule-Walker equations and testing the orders of multivariate time series," *Int. J. of Control*, vol. 37, no. 5, pp. 1159–1166, Sep. 1983.

[62] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern classification.* Hoboken, NJ: John Wiley & Sons, 2012.

[63] W. Zeng and A. Kavcic, "Optimal soft-output detector for channels with intersymbol interference and timing errors," *IEEE Trans. Magn.*, vol. 39, no. 5, pp. 2555–2557, Sep. 2003.

[64] J. Barry, A. Kavcic, S. McLaughlin, A. Nayak, and W. Zeng, "Iterative timing recovery," *IEEE Signal Process. Mag.*, vol. 21, no. 1, pp. 89–102, Jan. 2004.

[65] R. W. Butler, *Saddlepoint approximations with applications.* Cambridge, UK: Cambridge University Press, 2007.

[66] G. Casella and R. L. Berger, *Statistical inference*. New York, NY: Duxbury Pacific Grove, 2002.

[67] K. Abadir and J. R. Magnus, *Matrix algebra*. Cambridge, UK: Cambridge University Press, 2005.

[68] J. Proakis, *Digital communications*. New York, NY: McGraw-Hill, 1995.

[69] A. Liveris and C. Georghiades, "Exploiting faster-than-Nyquist signaling," *IEEE Trans. Commun.*, vol. 51, no. 9, pp. 1502–1511, Sep. 2003.

[70] D. Hajela, "On computing the minimum distance for faster than Nyquist signaling," *IEEE Trans. Inform. Theory*, vol. 36, no. 2, pp. 289–295, Mar. 1990.

[71] G. Ungerboeck, "Adaptive maximum-likelihood receiver for carrier-modulated data-transmission systems," *IEEE Trans. Commun.*, vol. 22, no. 5, pp. 624–636, May 1974.

[72] R. M. Todd, E. Jiang, R. L. Galbraith, J. R. Cruz, and R. W. Wood, "Two-dimensional voronoi-based model and detection for shingled magnetic recording," *IEEE Trans. Magn.*, vol. 48, no. 11, pp. 4594–4597, Nov. 2012.

[73] A. R. Krishnan, "Signal processing for two-dimensional magnetic recording," Ph.D. dissertation, Univ. of Arizona, Tucson, AZ, Nov. 2011.

[74] R. Green, P. J.Sibson, "Computing Dirichlet tessellations in the plane," *The Comput. J.*, vol. 21, no. 2, pp. 168–173, May 1978.

[75] A. Krishnan, R. Radhakrishnan, B. Vasic, A. Kavcic, W. Ryan, and F. Erden, "2-D magnetic recording: Read channel modeling and detection," *IEEE Trans. Magn.*, vol. 45, no. 10, pp. 3830–3836, Oct. 2009.

[76] 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.

[77] T. Oikawa, M. Nakamura, H. Uwazumi, T. Shimatsu, H. Muraoka, and Y. Nakamura, "Microstructure and magnetic properties of CoPtCr-SiO_2 perpendicular recording media," *IEEE Trans. Magn.*, vol. 38, no. 5, pp. 1976–1978, Sep. 2002.

[78] H. Uwazumi, K. Enomoto, Y. Sakai, S. Takenoiri, T. Oikawa, and S. Watanabe, "CoPtCr-SiO_2 granular media for high-density perpendicular recording," *IEEE Trans. Magn.*, vol. 39, no. 4, pp. 1914–1918, Jul. 2003.

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

[80] J. Moon and W. Zeng, "Equalization for maximum likelihood detectors," *IEEE Trans. Magn.*, vol. 31, no. 2, pp. 1083–1088, Mar. 1995.

[81] A. Weiss and B. Friedlander, "Efficient dynamic programming in presence of nuisance parameters," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 25, no. 2, pp. 277–280, Mar. 1989.

[82] W. Chang and J. R. Cruz, "Inter-track interference mitigation for bit-patterned magnetic recording," *IEEE Trans. Magn.*, vol. 46, no. 11, pp. 3899–3908, Nov. 2010.

[83] Y. Wu, J. O'Sullivan, S. N., and R. Indeck, "Iterative detection and decoding for separable two-dimensional intersymbol interference," *IEEE Trans. Magn.*, vol. 39, no. 4, pp. 2115–2120, Jul. 2003.

[84] G. Colavolpe, G.Germi, "On the application of factor graphs and the sum-product algorithm to ISI channels," *IEEE Trans. on Commun.*, vol. 53, no. 5, pp. 818–825, Apr. 2005.

[85] F. Kschischang, B. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 498–519, Feb. 2001.

[86] F. Lehmann, "Iterative mitigation of intercell interference in cellular networks based on Gaussian belief propagation," *IEEE Trans. Veh. Technol.*, vol. 61, no. 6, pp. 2544–2558, Jul. 2012.

[87] ——, "Joint binary image deconvolution and blur identification in the context of two-dimensional storage channels," *Signal Process.*, vol. 91, no. 10, pp. 2426–2431, May 2011.

[88] S. M. Khatami and B. Vasic, "Generalized belief propagation detector for TDMR microcell model," *IEEE Trans. Magn.*, vol. 49, no. 7, pp. 3699–3702, Jul. 2013.

[89] J. Zheng, X. Ma, Y. L. Guan, K. Cai, and K. S. Chan, "Low-complexity iterative row-column soft decision feedback algorithm for 2-D inter-symbol interference channel detection with Gaussian approximation," *IEEE Trans. Magn.*, vol. 49, no. 8, pp. 4768–4773, Aug. 2013.

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

[91] Y. Chen, P. Njeim, T. Cheng, B. Belzer, and K. Sivakumar, "Iterative soft decision feedback zig-zag equalizer for 2D intersymbol interference channels," *IEEE J. Select. Areas Commun.*, vol. 28, no. 2, pp. 167–180, Feb. 2010.