Seagate

# Annual Report

Barrett Brickner and Jaekyun Moon

September 26, 1995

*University of Minnesota*

CENTER FOR MICROMAGNETICS AND INFORMATION TECHNOLOGIES

# 1 Introduction

Various detection schemes for a linear channel corrupted with additive white Gaussian noise (AWGN) were studied. Except where noted, the simulations and analysis were performed using an actual magnetoresistive (MR) head response. The focus of the research was on the use of fixed-delay tree search with decision feedback (FDTS/DF) with $(0,k)$ run length limited (RLL) codes. The performance of FDTS/DF is compared to a partial response maximum likelihood (PRML) detector and the optimum detector, maximum likelihood sequence detection (MLSD). Alternate formulations of the FDTS detector, based on a signal space representation, are presented. Finally, a new coding technique, which can be used either to reduce detector complexity or improve margin, is discussed.

# 2 Channel Model

The channel model used for all the work presented is linear with additive white gaussian noise. A block diagram for the channel/detector is shown in Figure 1. The write current samples are $a_k$, $h_n$ is the sampled, filtered step response, $n_k$ are the AWGN samples, $c_n$ is the forward equalizer, and $b_n$ is the feedback filter. For a PRML detector, then $b_n$ block is not included. The value $\tau$ is the delay of the detector and the depth of a FDTS structure.

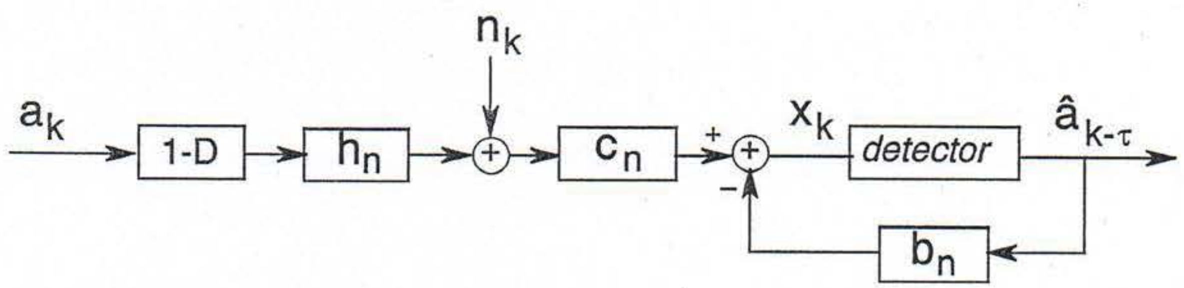


**Figure 1.** Channel model and detector block diagram

The sampled transition response is shown in Figure 2. The magnitude spectrum of the reponse, prior to processing by the anti-aliasing low pass filter, is shown in Figure 3. The equalizers were optimized at each SNR point before measuring the bit error rate (BER). The optimization criterion was minimum mean square error (MMSE). The effects of equalizer length were not considered. A sufficiently large number of taps (41 for the forward equalizer and 21 for the feedback filter) was used so that the filter lengths would not degrade the detector performance.

# 3 Detector Distance Properties

The performance of a detection scheme is dominated by the pair of symbols that are closest to each other in Euclidean distance. For an error to occur, the amount of noise that must be added is half of this distance. Because the noise amplitude is assumed to be Gaussian distributed, a small increase in distance can greatly reduce the probability of an error.

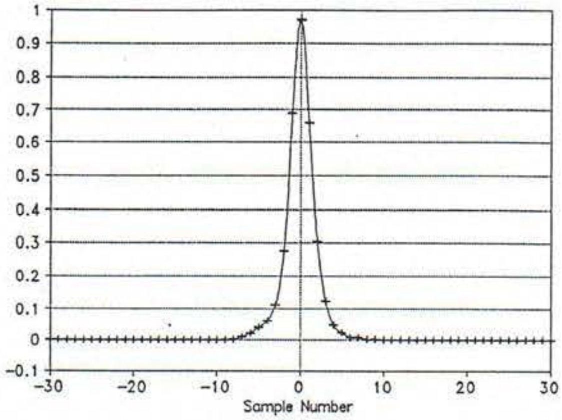


**Figure 2.** Sampled, filtered transition response

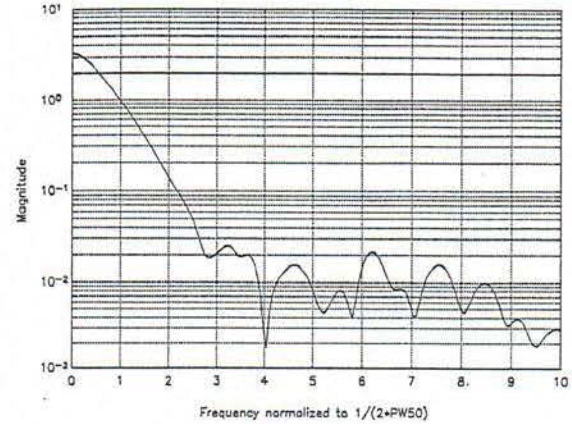


**Figure 3.** Spectrum of an unfiltered transition

## 3.1 Maximum Likelihood Sequence Detection

Maximum Likelihood Sequence Detection (MLSD) yields the optimum detector performance for a linear channel with AWGN. Although implementing an unconstrained MLSD in a commercial disk drive is not economically feasible, it is useful as an absolute performance bound against which other detection schemes may be compared. The distance between the two closest sequences is denoted by $d_{min}$. The performance of the MLSD detector is approximately

$$P_{MLSD}(e) \approx K \cdot Q\left[\frac{d_{\min}}{2\sigma}\right] \tag{1}$$

where Q(·) is the complementary distribution function for a Gaussian distribution, $\sigma$ is the square root of the noise power at the output of the forward equalizer, and $K$ as a constant independent of $d_{min}$.

The minimum distance can be bounded with the following [1]

$$\underset{\{e_k\}_0^L}{MIN} \sum_{k=0}^{L} \left[\sum_{j=0}^{J} b_j e_{k-j}\right]^2 \leq d_{\min}^2 \leq \underset{\{e_k\}_0^L}{MIN} \sum_{k=0}^{\infty} \left[\sum_{j=0}^{J} b_j e_{k-j}\right]^2 \tag{2}$$

where $\{b_n;\ n=1,\ldots,J\}$ are the feedback filter coefficients and $b_0=1$. The term $b_0=1$ is a result of training the equalizers for a DFE; i.e., for ideal operation, the samples at the input to a DFE are $\pm b_0$. The error sequence of length $L+1$, $\{e_k\}$ is taken to be all possible error sequences where $e_0 = \pm 2$, and all other $e_k$ are 0 or $\pm 2$. This assumes that the desired data sequence is taken from $a_k = \pm 1$ so that the differences are in $\{\pm 2, 0\}$. These upper and lower bounds represent non-increasing and non-decreasing functions, respectively, of the error length $L$. These bounds will converge as $L \rightarrow \infty$. The bounds, as a function of $L$, for the case where the user density is $D_u = 2.5$ are shown in Figure 4.

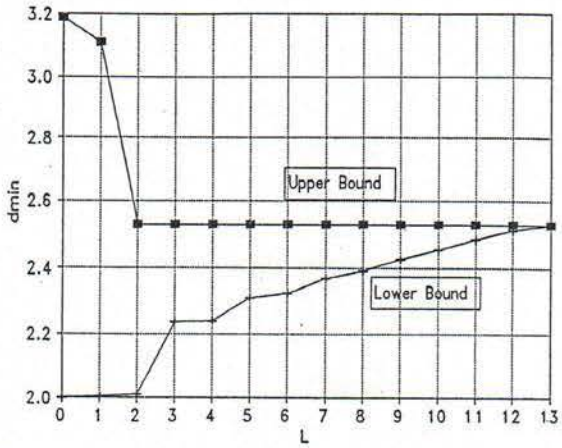


Figure 4. Bounding of $d_{min}$ for $D_u$ = 2.5

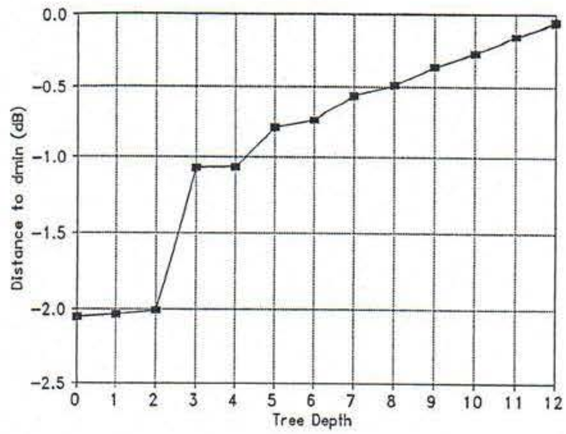


Figure 5. FDTS/DF Minimum Distance ($\beta_{min}$)

As shown, the upper bound rapidly converges to $d_{min}$ while the lower bound approaches more slowly. The rapid convergence of the upper bound is explained by noting that for $L \geq 2$, the minimum distance error sequence determined by the upper bound was $e = \pm(-2,+2,-2)$. Thus, when $L = 2$, the upper bound will be evaluated for this minimum distance event. Assuming that an event with a smaller distance does not exist beyond the range of L that was examined, there will be no decrease in the upper bound, because the minimum error event will already have been accounted for. It has been shown that at high densities, the minimum distance error sequence is of the form $e_{min} = \pm(-2,+2,-2)$, a result which agrees with Figure 4 [2].

## 3.2 Fixed Delay Tree Search

A measure of performance similar to $d_{min}$ exists for FDTS/DF. In the case of the tree search, the quantity of interest is the distance between the pair of conflicting symbols with the smallest Euclidean distance. This value, termed $\beta_{min}$ is determined by [3]

$$\beta^2_{\min} = \underset{\{e_k\}_0^\tau}{MIN} \sum_{m=0}^{\tau} \left[ \sum_{j=0}^{J} b_j e_{k-j} \right]^2 \tag{3}$$

Notice that this is the same as the lower bound in (2). Thus, the minimum distance in FDTS/DF is a non-decreasing function of the depth $\tau$. Note that because $\beta_{min}$ is the lower bound for $d_{min}$, the performance of FDTS/DF must converge to the MLSD bound. This distance (in dB) from $d_{min}$ is shown as a function of $\tau$ in Figure 5. The only question remaining is how large the tree depth should be. This criterion is still dominated by the need to balance complexity against performance, but the distance properties provide some assistance in choosing a depth. For example, there is less incentive to increase the complexity from depth $\tau$=1 to $\tau$=2, but increasing from $\tau$=2 to $\tau$=3 results in a noticeable improvement for this particular channel.

# 4 Coding to Improve Signal Margin

The bit error rate (BER) performance of optimal and suboptimal sequence estimators in an additive white Gaussian noise (AWGN) channel is dominated by the Euclidean distance between the two closest, conflicting sequences. For MLSD, it has been shown that at high data densities, the error rate performance is dominated by the error sequence $e = \pm(-2,+2,-2)$, where $e$ is the difference between two valid sequences. An examination of the error sequences that correspond to $\beta_{min}$ for FDTS/DF show that the most likely error sequence consists of three or more consecutive non-zero values in the error sequence. The Euclidean distance between the various conflicting sequences in the tree search indicates that eliminating error sequences containing $\pm(\ldots,-2,+2,-2,\ldots)$ will yield a significant improvement in distance.

## 4.1 Coding Objective

Figure 6 shows the two pairs of conflicting patterns that can cause the error sequence $e = \pm(-2,+2,-2)$. The sequences shown are non-return-to-zero (NRZ) sequences, which correspond to the write current waveform in a magnetic recording system. One or both of the conflicting patterns can be eliminated by requiring that the valid sequences contain no more than two consecutive transitions. A transition corresponds to a change in the level of the NRZ sequence.

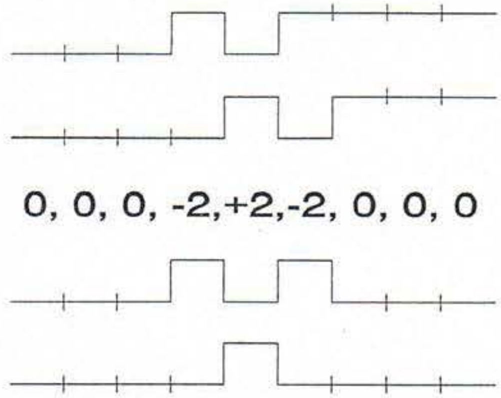


**Figure 6.** Sequences that cause $e=\pm(-2,+2,-2)$.

## 4.2 Maximum Transition Run Coding

In order to increase the minimum sequence distance, a new class of codes, designated *maximum transition run* (MTR) codes, is introduce. These codes limit the number of consecutive transitions that can occur in a recorded sequence. As noted before, eliminating three or more consecutive transitions results in a significant increase in minimum distance. Therefore, the use of MTR=2;$k$ codes is proposed for use in magnetic recording. The $k$ constraint is the same as the $k$ constraint used in RLL coding. The RLL $d$ constraint for the MTR codes is $d$=0. If the written data is considered as an NRZI (non-return-to-zero inversion) sequence, where a 1 corresponds to a change in the level of the corresponding NRZ sequence, and a 0 signifies no change, the MTR=2 constraint means that no more than two consecutive 1's can occur. For the remainder of this report, data and code words are assumed to be an NRZI representation.

The properties of a MTR=2 code are discussed in the context of code design. The characteristics of several codes that could be readily applied to data storage are examined. The discussion will focus on block codes, in which there is a one-to-one mapping between an $m$-bit block of user bits and an $n$-bit codeword. More complex and efficient codes can be developed by using a state machine as the encoder. While the design techniques for these codes is beyond the scope of this paper, they can be applied to develop codes which incorporate the MTR constraint.

DOCKET ALARM

# Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

## Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

## Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

## Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

## API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

**LAW FIRMS**

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

**FINANCIAL INSTITUTIONS**

Litigation and bankruptcy checks for companies and debtors.

**E-DISCOVERY AND LEGAL VENDORS**

Sync your system to PACER to automate legal marketing.

**WHAT WILL YOU BUILD?** | **sales@docketalarm.com** | **1-866-77-FASTCASE**

fastcase®
Smarter legal research.