Signal Space Detectors for MTR-Coded Magnetic Recording Channels

Hamid Shafiee, Bernardo Rub and Robert Kost
Seagate Technology, 8001 E. Bloomington Freeway, Bloomington, MN 55420

***Abstract* –In this paper, we present signal space detectors for use with maximum transition run (MTR) codes. A three-dimensional signal space detector is first derived for an MTR=2 coded channel. The bit error rate performance of this detector is close to MTR-coded FDTS/DF(2) throughout the user density range of interest. The detector is then modified to be used with a time-variant MTR code. Simulation as well as experimental results are presented.**

I. INTRODUCTION

To improve bit error rate performance, or to increase linear recording density, the application of maximum likelihood sequence detection (MLSD) to digital magnetic recording has been investigated in recent years. It is observed that with binary input bits, for MLSD at high recording densities and for certain high-order partial response channels such as E^2PRML, the dominant error events are of the form ±{+2, -2, +2}. A new class of codes called maximum transition run (MTR) codes have recently been proposed as a way of removing such dominant error events and, hence, increasing the minimum Euclidean distance [1]. An MTR=2 code limits the maximum number of consecutive transitions to two, and consequently, removes all patterns which cause the dominant error events (Fig. 1). MTR codes of rate 6/7 have been developed in [2] and [3].

Utilizing the MTR constraint, Brickner and Moon have developed an efficient detector called 3D-110 whose performance is comparable to fixed delay tree search with decision feedback of depth 2 (FDTS/DF(2)) at high symbol densities [4]. The detector is constructed by considering vectors of received samples in a 3-dimensional space. Using three planer boundaries, the signal space is divided into two regions each of which correspond to a decision of +1 or -1. The 3D-110 forward filter removes the precursor intersymbol interference (ISI) terms and forces the two post-cursor ISI terms to be 1 and 0, respectively, where the cursor is also normalized to 1. The feedback filter removes all but two post-cursor ISI terms. With no error propagation, the equivalent discrete-time channel pulse response can be denoted as "110". Such a constraint on the channel response is used to further simplify the detector structure.

While the magnetic channel "natural" response is close to the "110" target at high recording densities, it deviates from the desired target at lower densities. Constraining the pulse response to this particular target will then result in performance degradation compared to FDTS/DF (2). Even at high densities, other factors such as the use of constrained-length finite impulse response (FIR) filters may cause deviation of the channel response from the "110" target. In the next section, we extend the geometrical detection approach in [4] and develop a sub-optimal three-dimensional signal space detector referred to as 3D-SSD, which does not constrain the channel response to any specific target. However, it uses the MTR constraint as well as modified signal space decision boundaries to simplify the detector structure. The new detector provides marginal improvements over 3D-110 at higher densities, but provides considerable gains at lower densities.

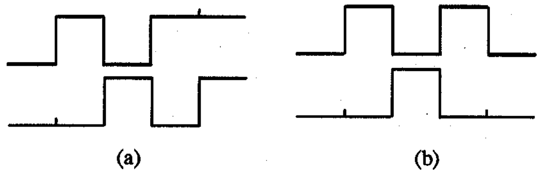


Fig. 1: Error events of the form ±(2, -2, 2) are caused when (a) a tribit is shifted or (b) when a quadbit is mistaken as a dibit or vice versa. MTR=2 codes remove all patterns containing 2 or more transitions. Time-variant MTR codes allow tribits to start at alternate time intervals.

The dominant error events mentioned above can also be removed using a time-variant transition run constraint that allows tribits to only start at even- (or odd-) numbered time intervals (Fig. 1) [5][6]. Such a relaxed constraint would then allow the development of codes with higher rates. Both 3D-110 and 3D-SSD channels are derived based on the assumption that no tribits are allowed in the input sequence. In Section III, we develop modified 3D detectors for use with a time-variant MTR code. Bit error rate (BER) results from an experimental set-up are included in Section IV.

II. DERIVATION OF THE 3D-SSD CHANNEL

We first consider an MTR=2 coded channel. In Fig. 2, a generic block diagram of a digital recording channel which uses decision feedback is depicted. Like other decision feedback techniques, 3D-SSD uses a whitened matched filter to remove all the ISI terms and to whiten the noise at the detector input. Unlike 3D-110, no constraints are enforced on the channel coefficients and, hence, the post-cursor ISI

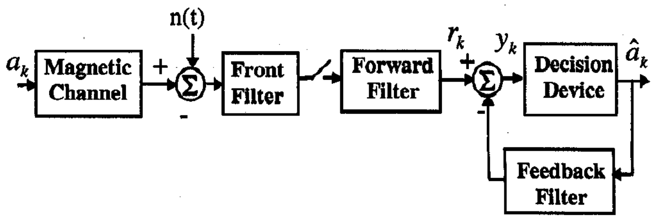


Fig. 2: Schematic diagram of a decision feedback read channel

Manuscript received June 16, 1997.

H. Shafiee, 612-806-2562, Hamid_R_Shafiee@notes.seagate.com; B. Rub, 612-806-2933, Bernardo_Rub@notes.seagate.com; R. Kost, 612-806-2771, Robert_E_Kost@notes.seagate.com.

terms are allowed to take on their "natural" values. The feedback filter removes all but two, post-cursor ISI terms. Assuming all the previous decisions are correct, the equivalent discrete-time channel response includes three terms and is denoted as $(1, f_1, f_2)$ where, without any loss of generality, the main tap is normalized to one. At time *k*, the noiseless input to the decision device, y_k, could be written as:

$$y_k = a_k + f_1 a_{k-1} + f_2 a_{k-2} \qquad (1)$$

where a_k is the input data bit at time *k*.

The 3D-SSD detector is designed by first considering the symbol constellation in a three-dimensional space. The detector decision depends on the symbol which is closest to the vector of observation samples at each time interval. This is analogous to finding the path with the minimum Euclidean distance between the observed and desired samples values for fixed-delay detectors such as FDTS/DF or look-ahead partial response channels [7,8]. In principle, each pair of symbols which point to different detector decisions need to be separated by a boundary plane. The planar boundaries are combined by a logic rule so that the signal space is partitioned into two regions, one corresponding to a decision of +1 and the other -1. Depending on where the vector of the received symbols falls in the vector space, a binary decision is released as the detector output. The detector structure is simplified by eliminating planes which are redundant or separate symbols which are much farther apart than the minimum Euclidean distance.

The 3-dimensional observation vector falls in the (y_k, y_{k-1}, y_{k-2}) signal space. The derivation of the planer boundaries is simplified with a linear vector space transformation [4]. Let us define the following parameters which comprise the axes of the new vector space:

$$y_k = a_k + f_1 a_{k-1} + f_2 a_{k-2} \qquad (2)$$

$$y'_{k-1} = a_{k-1} + f_1 a_{k-2} \qquad (3)$$

$$y''_{k-2} = a_{k-2} \qquad (4)$$

Here, y'_{k-1} and y''_{k-2} denote the detector inputs at times *k*-1 and *k*-2 with the ISI due to the available decisions (i.e., $\hat{a}_{k-3}$ and $\hat{a}_{k-4}$ at time *k*) canceled. Notice that at each time *k* in the detection process, the detector needs to make a decision on the input bit, a_{k-2}.

Index	(a_{k-2}, a_{k-1}, a_k)	$(y_k, y'_{k-1}, y''_{k-2})$
0	(+1, +1, +1)	$(+1 + f_1 + f_2, +1 + f_1, +1)$
1	(+1, +1, -1)	$(-1 + f_1 + f_2, +1 + f_1, +1)$
2*	(+1, -1, +1)	$(+1 - f_1 + f_2, -1 + f_1, +1)$
3	(+1, -1, -1)	$(-1 - f_1 + f_2, -1 + f_1, +1)$
4	(-1, +1, +1)	$(+1 + f_1 - f_2, +1 - f_1, -1)$
5*.	(-1, +1, -1)	$(-1 + f_1 - f_2, +1 - f_1, -1)$
6	(-1, -1, +1)	$(+1 - f_1 - f_2, -1 - f_1, -1)$
7	(-1, -1, -1)	$(-1 - f_1 - f_2, -1 - f_1, -1)$

Table 1: Input data patterns and the corresponding symbols in the signal space. *Cases 2 and 5 violate the MTR constraint with $\hat{a}_{k-3} = -1$ and +1, respectively.

Table 1 lists input write currents and the corresponding noiseless points in the $(y_k, y'_{k-1}, y''_{k-2})$ signal space. Notice that, depending on the value of $\hat{a}_{k-3}$, either symbol 2 or 5 is disallowed since it represents the present of a tribit, which is disallowed by the MTR=2 code. For example, with $\hat{a}_{k-3} = +1$, symbol 5 corresponds to a sequence of the form (+1, -1, +1, -1) which contains three consecutive transitions. Fig. 3(a) and (b) show the symbol constellation for a Lorentzian channel at a symbol density of 2.25 with $\hat{a}_{k-3} = +1$ and $\hat{a}_{k-3} = -1$, respectively. The y''_{k-2} axis extends out of the surface of the paper. Symbols which corresponds to $y''_{k-2} = +1$ and -1 are denoted by x's and o's, respectively. The index below the symbol marker points to the corresponding input data patterns listed in Table 1.

To keep the detector structure simple, we limit the number of slicer planes in 3D-SSD to three. The directions of these planes are also constrained to further simplify the detector

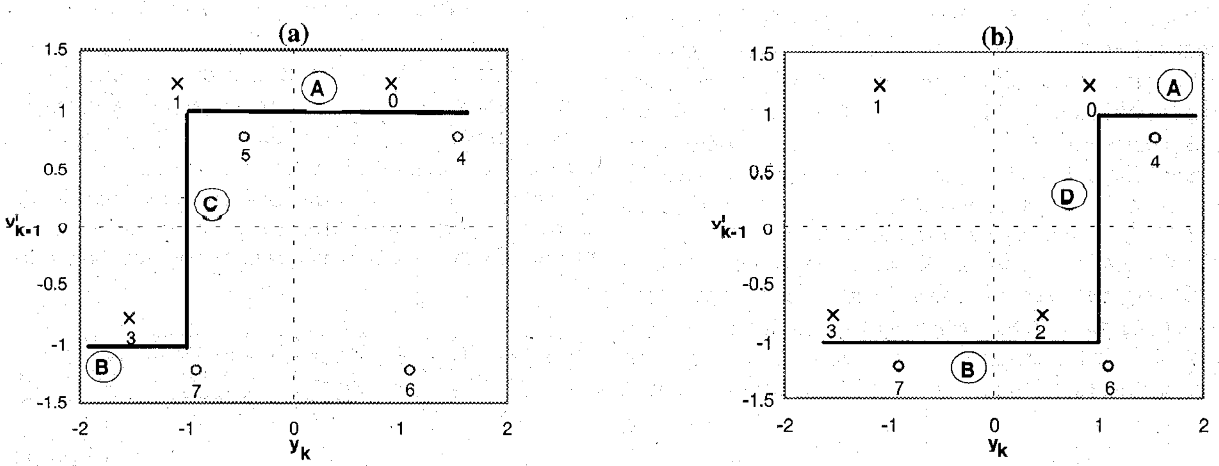


Fig. 3: (a) Symbol constellation with $\hat{a}_{k-3} = -1$, (b) symbol constellation with $\hat{a}_{k-3} = +1$. The dark lines show the intersection of the boundary planes with the $y_k y'_{k-1}$ surface.

structure. Initially, four decision planes denoted as A, B, C and D, are considered. (Later on, C and D are combined to form E.) Let us first consider plane A which separates symbols 0 and 4 (as well as 1 and 5 in Fig. 3(a)). Optimal decision boundaries are planes bisecting the line which connects pairs of symbols of interest. However, the constrained optimization here, locates a plane which separates, not the two symbols, but their projections on the $y'_{k-1}\, y''_{k-2}$ surface. The intent is to pick the two coordinates which contribute the most to the distance between the two symbols. Clearly, the y''_{k-2} coordinate needs to be retained, since the two symbols which correspond to different decisions on a_{k-2} , are well separated on this axis. Of the two remaining, except for very low symbol densities (Ds<1.6), y'_{k-1} contributes more significantly to the distance than y_k.

The slicer plane A, therefore, is constrained to only rotate perpendicular to the $y'_{k-1}\, y''_{k-2}$ surface. The projection of this plane onto the $y'_{k-1}\, y''_{k-2}$ surface will be a line whose direction changes as the slicer plane is allowed to rotate. All points on the desired line have the same distance from the projection of the pair of symbols. Since the coordinates of the projections of symbols 0 and 4 on the $y'_{k-1}\, y''_{k-2}$ surface are given by $(1+f_1,+1)$ and $(1-f_1,-1)$, respectively, the equation of the plane A can be obtained by writing:

$$(y'_{k-1}-(1+f_1))^2+(y''_{k-2}-1)^2=(y'_{k-1}-(1-f_1))^2+(y''_{k-2}+1)^2$$

This expression could be simplified to yield:

$$y''_{k-2}+f_1 y'_{k-1}-f_1=0$$

Using a similar procedure, the equation for slicer B which separates symbols 3 and 7 (as well as 2 and 6 in Fig. 3(b)), can be found to be:

$$y''_{k-2}+f_1 y'_{k-1}+f_1=0$$

Plane C separates symbols 3 and 5 when $\hat{a}_{k-3}=-1$. Here, this plane is constrained to only rotate perpendicular to the $y_k\, y''_{k-2}$ surface since the two coordinates which contribute more significantly to the distance in this case are y_k and y''_{k-2}. The plane equation can be derived by finding the line that bisects the projections of the two symbols on the $y_k\, y''_{k-2}$ surface. The operation is repeated for plane D which separates symbols 2 and 4 with $\hat{a}_{k-3}=+1$. Applying the procedure outlined above results in the following four boundary equations:

$$A\!: \operatorname{sgn}(y''_{k-2}+f_1 y'_{k-1}-f_1)$$

$$B\!: \operatorname{sgn}(y''_{k-2}+f_1 y'_{k-1}+f_1)$$

$$C\!: \operatorname{sgn}(y''_{k-2}-(f_1-f_2)y_k-(f_1-f_2)), \qquad \hat{a}_{k-3}=-1$$

$$D\!: \operatorname{sgn}(y''_{k-2}-(f_1-f_2)y_k+(f_1-f_2)), \qquad \hat{a}_{k-3}=+1$$

Boundaries C and D could be combined to give:

$$E\!: \operatorname{sgn}(y''_{k-2}-(f_1-f_2)y_k+(f_1-f_2)\hat{a}_{k-3})$$

The above equation can be further simplified by setting (f_1-f_2) equal to 1. This simplification has a negligible effect on the detector performance since at lower channel densities of interest, the two symbols to be separated by this plane are farther apart than those separated by planes A and B. Therefore, a slight change in the plane orientation and position would not impact the relative location of the received samples with respect to this plane. The new slicer plane becomes:

$$E\!:\operatorname{sgn}(y''_{k-2}-y_k+\hat{a}_{k-3})$$

Substituting for y'_{k-1} and y''_{k-2} using Equations (3) and (4), the following relations are obtained for the three decision planes:

$$A\!:\operatorname{sgn}(y_{k-2}+f_1 y_{k-1}+\Delta A) \tag{5}$$

$$B\!:\operatorname{sgn}(y_{k-2}+f_1 y_{k-1}+\Delta B) \tag{6}$$

$$E\!:\operatorname{sgn}(y_{k-2}-y_k+\Delta E) \tag{7}$$

where offset values ΔA, ΔB and ΔE are given by

$$\Delta A=(-f_1-f_1 f_2)\hat{a}_{k-3}-f_2\,\hat{a}_{k-4}-f_1 \tag{8}$$

$$\Delta B=(-f_1-f_1 f_2)\hat{a}_{k-3}-f_2\,\hat{a}_{k-4}+f_1 \tag{9}$$

$$\Delta E=(-f_1+1)\hat{a}_{k-3}-f_2\,\hat{a}_{k-4} \tag{10}$$

The offset levels can, in general, be implemented as short FIR filters with binary inputs, 2-input multiplexers or look-up tables.

To arrive at the decision logic, one can move a test point through the signal space and record the relative position of the point with respect to the planes. The corresponding detector output is also noted by finding the closest symbol in the constellation to the test point. A logic rule is found by combining the cases which result in the same output decision. For the three-dimensional case considered here, however, the logic rule can be written by inspection (Fig. 3). Mapping boundary decisions -1 to 0, the logic rule can be written as:

$$\hat{a}_{k-2}=B.E+A \tag{11}$$

The 3D-SSD architecture is shown in Fig. 4. With offsets implemented using multiplxers, the detector shown in Fig. 4 uses one multiplier, three slicers, three adders and three two-input multiplexers. The 3D-110 detector, on the other hand, can be implemented using three slicers, three adders and two

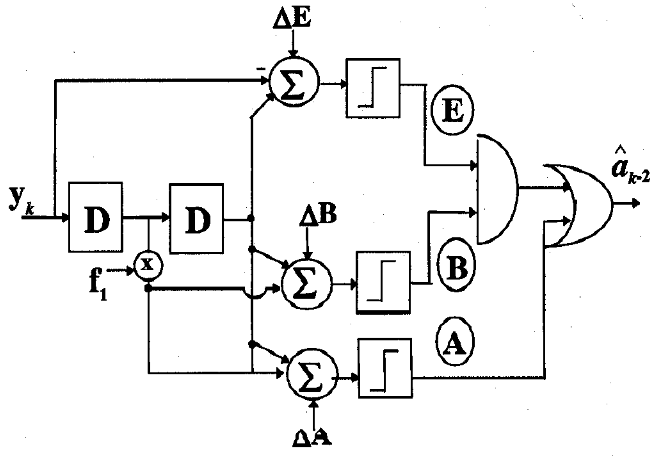


Fig. 4: 3D-SSD detector

two-input multiplexers [3].

In Fig. 5, the simulation results which compare the required SNR for a BER of 1e-4 for a Lorentzian channel with additive white Gaussian noise (AWGN) are depicted for several detectors. The noisy readback waveform is run through a 4th-order Butterworth low-pass front filter. The low-pass filter cut-off frequency is set at the Nyquist frequency. To remove the influence of filters lengths, both forward and feedback filters have a sufficient number of taps. The feedback detectors use an MTR=2 code of rate 6/7. Results for a (0, 4/4) RLL-coded PRML channel are also plotted for comparison. The plot shows that the performance of 3D-110 approaches that of FDTS/DF(2) as linear density increases. At low densities, however, constraining the channel response to the "110" target results in performance degradation compared to FDTS/DF due to noise enhancement and coloration. The 3D-SSD detector performs close to FDTS/DF throughout the user density range. At a user density of 2, 3D-SSD comes within 0.3 dB of FDTS/DF while 3D-110 has about 1.5 dB degradation. As the user density is lowered, the dominant error events for MLSD starts to change from tribit patterns to single bit events. Therefore, the coding gain of the MTR code is reduced.

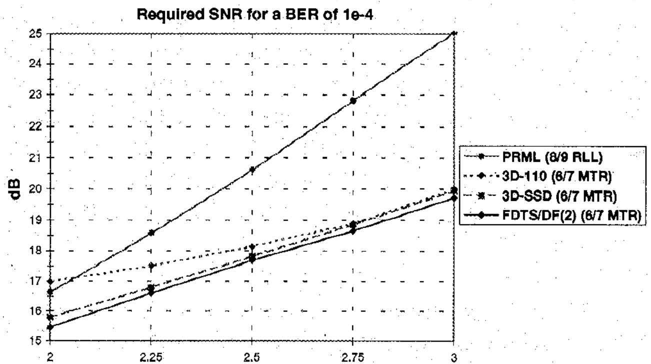


Fig. 5: Required SNR for a BER of 1e-4 vs. user density

III. 3D DETECTOR WITH TIME-VARIANT MTR CODE

3D-110 and 3D-SSD detectors are both constructed by taking advantage of the fact that at each time interval, only one of the two symbols 2 or 5 in Table 1 is present in the signal space. This is because MTR=2 codes remove one of these two symbols at all times. With the time-variant MTR code, however, it is possible to have both symbols present in the signal constellation at every other time interval. The structure of the 3D detectors would have to be modified to accommodate the change in the code constraint.

To design signal space detectors which utilize the new code, let us consider the FDTS/DF tree of depth two. In Fig. 6(a) and (b), assuming $\hat{a}_{k-3} = +1$, the detection tree is shown when the root is at an odd or even time interval, respectively. Here, without any loss of generality, it is assumed that the tribits are only allowed to start at even time intervals. Notice that in Fig. 6(a), as in the previous case, either path 2 or 5 is disallowed since it violates the code constraint. For example, branch 5 is pruned since it points to tribit pattern {+1, -1, +1, -1} which starts at an odd time interval.. Therefore, at odd times, the situation is identical to the MTR=2 case. On the other hand, when the root is an even time interval as in Fig. 6(b), both branches 2 and 5 are legal.

To realize the coding gain for an FDTS/DF(2) detector with a time-variant MTR code, one can remove the illegal path at odd times and restore it at even times. This would prevent shifted tribit errors from occurring since the erroneous tribit has to start at a forbidden time interval. But the presence of both paths 2 and 5 increases the chance of the erroneous section of the tree to be selected. In fact, as the density increases, these errors start to wipe out the code-rate benefit of the time-variant MTR code.

Let us extend the branches 2 and 5 one step further as shown in Fig. 6. This does not affect the situation in Fig. 6(a) since the extended branches of path 2 are allowed whereas those of 5 are disallowed. However, in Fig. 6(b), the top branch for path 2 denoted as 2A, is allowed while the bottom branch (2B) is disallowed since it corresponds to a tribit which start at an odd position. Similarly, only the bottom branch of path 5 (e.g., 5B) is allowed. The two symbols 2A and 5B correspond to error events of the form ±{2, -2, 2, 2}. Therefore, the distance between the two should be considerably greater than the minimum Euclidean distance. If the distances from the 4-dimensional observation vector to these two symbols are used to select the closer symbol, then either path 2 or 5 could be completely pruned from the tree. Removal of one of the two symbols (or paths) would then make the signal constellation resemble those shown in Fig. 3.

To come up with a boundary decision for the selection of either path 2A or 5B, let us first write the sample at time k+1 as:

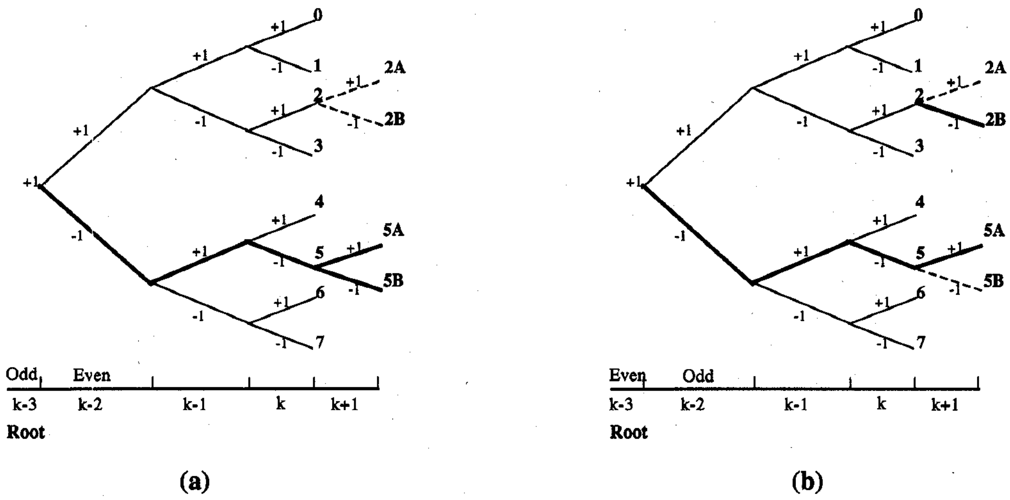
$$y_{k+1} = a_{k+1} + f_1 a_k + f_2 a_{k-1} \tag{12}$$

In the signal space $(y_{k+1}, y_k, y'_{k-1}, y''_{k-2})$, symbols 2A and 5B are given by $(+1+f_1-f_2, +1-f_1+f_2, -1+f_1, +1)$ and $(-1-f_1+f_2, -1+f_1-f_2, 1-f_1, -1)$, respectively. The new boundary plane P bisects the projection of these points in the $y_{k+1}y''_{k-2}$ surface. Assuming that sample y_{k+1} is available at time k, the equation of the plane can be obtained by writing the distance of the point (y_{k+1}, y''_{k-2}) from the two points as given below:

$$\begin{aligned}(y_{k+1} - (1+f_1-f_2))^2 + (y''_{k-2} - 1)^2 &= \\ ((y_{k+1} - (-1-f_1+f_2))^2 + (y''_{k-2}+1)^2\end{aligned} \tag{13}$$

The decision due to plane P impacts the symbol constellation at even times (i.e., when k-3 is even), since it removes either symbol 2 or 5. One way to alleviate the computational delay due to this boundary is to come up with the decision one time interval early. Toward this end, let us write Equation (13) one bit interval early. This gives:

$$\begin{aligned}(y_{k+2} - (1+f_1-f_2))^2 + (y''_{k-1} - 1)^2 &= \\ ((y_{k+2} - (-1-f_1+f_2))^2 + (y''_{k-1}+1)^2\end{aligned} \tag{14}$$



(a)

(b)

Figure 6: FDTS/DF tree: (a) root is an odd time interval; (b) root at an even time interval

Computation of y_{k+2} at time k, requires the availability of the sequence $(\hat{a}_{k-1}, \hat{a}_{k-2}, \ldots, \hat{a}_{k-N})$ for feedback cancellation, where N is the length of the feedback filter. However, $\hat{a}_{k-1}$ and $\hat{a}_{k-2}$ decisions are not yet available at time k. (Recall that the output of FDTS/DF(2) at time k is $\hat{a}_{k-2}$.) We will, hence, define a new parameter z_k where the ISI due to the feedback taps f_3 and f_4 are not yet subtracted, i.e.,

$$z_{k+2} = y_{k+2} + f_3 \hat{a}_{k-1} + f_4 \hat{a}_{k-2} \qquad (15)$$

For each symbol, the values of $\hat{a}_{k-1}$ and $\hat{a}_{k-2}$ can be taken from its own path. For example, the LHS of Equation (13) denotes the distance from symbol 2A, for which $\hat{a}_{k-1} = -1$ and $\hat{a}_{k-2} = +1$ as seen in Fig. 6(b). On the other hand, for symbol 5B (i.e., RHS of Equation (13)), $\hat{a}_{k-1} = +1$ and $\hat{a}_{k-2} = -1$. The operation above is similar to local feedback cancellation in channels such as reduced state sequence estimator (RSSE) [9]. Similarly, local feedback is used for the value of $\hat{a}_{k-2}$ when y''_{k-1} is transformed back to y_{k-1}. Substituting for y_{k+2} in Equation (14) and using local feedback values for $\hat{a}_{k-1}$ and $\hat{a}_{k-2}$ yield the following planer equation:

$$(-1 + (f_2 + f_3 - f_4) / (1 + f_1)) z_{k+2} - y_{k-1} + f_2 \hat{a}_{k-3} = 0$$

Since the distance between symbols 2A and 5B is much greater than the minimum Euclidean distance, the multiplicative factor of z_{k+2} can be set at -1 with negligible impact on performance. Therefore, slicer plane P is given by

$$P: \operatorname{sgn}(-z_{k+2} - y_{k-1} + f_2 \hat{a}_{k-3})$$

As mentioned before, with the decision due to plane P available at even times, the symbol constellation is made similar to those in Fig. 3 and the modified 3D-SSD detector can now be constructed. While the equations for all four planes, A, B, C and D remain unchanged, the conditions under which planes C and D are applied need to be modified. The constellation of Fig. 3(a) and the boundary plane C apply when $\hat{a}_{k-3} = -1$ with the root at an odd time interval or when $P = -1$ with the root at an even time interval. On the other hand, the constellation of Fig. 3(b) and the plane D are applicable when $\hat{a}_{k-3} = +1$ at odd times or when $P = +1$ at even times. This is summarized below:

$$\begin{cases} C: \operatorname{sgn}(y''_{k-2} - y_k - 1), \hat{a}_{k-3} = -1 \; (\mathit{odd}) \; \mathit{or} \; P = -1 \; (\mathit{even}) \\ D: \operatorname{sgn}(y''_{k-2} - y_k + 1), \hat{a}_{k-3} = +1 \; (\mathit{odd}) \; \mathit{or} \; P = +1 \; (\mathit{even}) \end{cases}$$

The two planes can be combined to give plane E' as shown below:

$$E': \begin{cases} y_{k-2} - y_k - \Delta E' + \hat{a}_{k-3}, & (k-3) \; \mathit{odd} \\ y_{k-2} - y_k - \Delta E' + P, & (k-3) \; \mathit{even} \end{cases}$$

where $\Delta E' = -f_1 \hat{a}_{k-3} - f_2 \hat{a}_{k-4}$, which again could be implemented as an FIR filter or a two-input multiplexer. The final decision is again given by Equation (11). Fig. 7 shows the architecture of the modified 3D-SSD detector for a time-variant MTR code.

In Fig. 8, the performance of 3D-110 and 3D-SSD detectors of rates 6/7 and 8/9 are compared. The code-rate gain of the time-variant MTR code is seen by the lower SNR

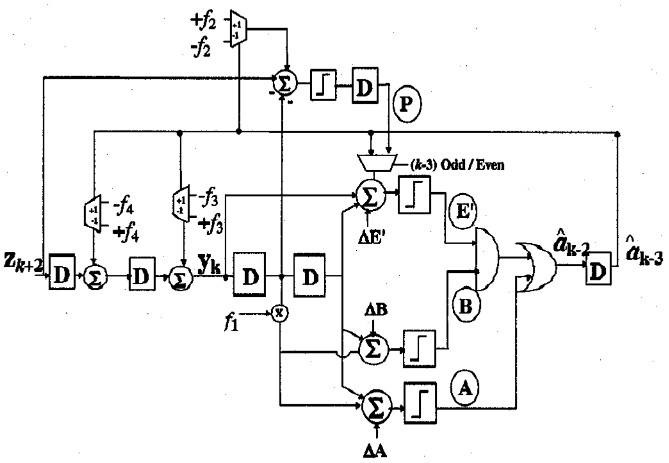


Fig. 7: 3D-SSD detector for use with the time-variant MTR code

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.