SURVEY PAPER   *Special Section on Information Theory and Its Applications*

Towards Efficient Detection of Two-Dimensional Intersymbol Interference Channels*

Brian M. KURKOSKI[†], *Member*

SUMMARY   This paper gives a survey and comparison of algorithms for the detection of binary data in the presence of two-dimensional (2-D) intersymbol interference. This is a general problem of communication theory, because it can be applied to various practical problems in data storage and transmission. Major results on trellis-based detection algorithms, previously disparate are drawn together, and placed into a common framework. All algorithms have better complexity than optimal detection, and complexity is compared. On the one hand, many algorithms perform within 1.0 dB or better of optimal performance. On the other hand, none of these proposed algorithms can find the optimal solution at high SNR, which is surprising. Extensive discussion outlines further open problems.
key words: *intersymbol interference, detection, 2-D detection, signal processing for storage*

1.   Introduction

Data storage devices and transmission systems continue to rapidly increase in capacity and data rates. Many systems increase capacity by allowing one-dimensional intersymbol interference, and then compensating by using signal processing, for example, by using Viterbi detection. However, the pressure to further increase data rates may lead to systems where accommodating intersymbol interference in two dimensions is rapidly becoming necessary.

In both optical recording and magnetic recording, there have been proposals to increase capacity by eliminating inter-track spacing, and to increase data rates by reading multiple tracks simultaneously. However, such dense spacing of tracks leads to not just the usual in-track interference, but to complete 2-D intersymbol interference. In holographic storage systems, data is stored in a 2-D array, and naturally suffers from 2-D intersymbol interference.

While recent research in 2-D detection has been driven by data storage applications, it is worthwhile to note that the problem of 2-D intersymbol interference arises in data transmission systems as well. There is a close relationship between 2-D storage and multiple-input, multiple-output (MIMO) transmissions systems [1]. Cellular networks, for example arranged on a hexagon, exhibit 2-D interference, and studies yield information-theoretic insight into such systems [2]. Even two-dimensional bar codes, such as QR-
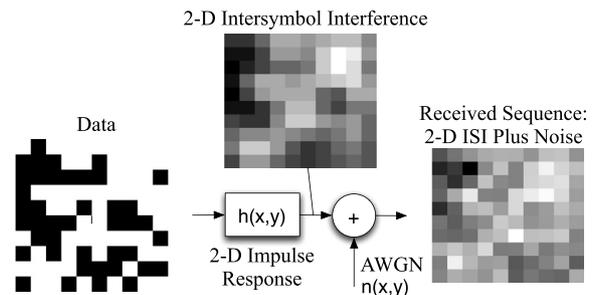
Fig. 1   Model for the two-dimensional intersymbol interference channel considered in this paper.

Codes, are a type of storage system which suffer from 2-D intersymbol interference [3].

The literature is rich with proposals for reducing the effects of 2-D ISI; while many are motivated by just one of the above applications, in fact the problem is a general one. Accordingly, this survey paper considers the following. An array of 2-D binary data is convolved with a 2-D channel response, of finite duration in both dimensions. To this, add additive white Gaussian noise, to obtain the noisy received sequence. The problem is illustrated in Fig. 1. This survey considers detection algorithms which estimate the original data, given the noisy received sequence.

Minimum mean-square error filtering is a prevalent technique for reducing the effects of intersymbol interference in communication systems, and is closely related to deconvolution in signal processing [4, Ch. 6]. Wakabayashi et al. proposed a two-dimensional transversal filter which showed significant SNR improvement for the elimination of ISI in optical disk systems [5]. Good results can also be obtained by applying minimum mean-squared error filters to holographic storage [6]. Deconvolution is a valuable tool in processing of images, which are 2-D arrays of real numbers, however there is additional performance gain to be had by considering the binary nature of the data.

An advantage of linear filtering is that it has reasonably low complexity, but unfortunately these techniques can suffer from noise enhancement, particularly on channels with nulls. For one-dimensional detection, the Viterbi algorithm is optimal, and has complexity linear in the number of channel input bits (although exponential in the channel memory) [7]. As a result, this algorithm, or one of its derivatives, is widely used in practice.

However, for 2-D detection, it is widely regarded that optimal 2-D detection is not computationally tractable,

and only recently a technical report appeared with a proof that maximum-likelihood sequence detection for two-dimensional ISI channels is NP-complete [8]. One possible technique for optimal 2-D detection is to construct a trellis which completely describes the intersymbol interference and apply the Viterbi algorithm to it. Since optimal techniques do not often appear in the literature, a tutorial description of optimal detection is given in Sect. 2. Because of the computational complexity of optimal detection, suboptimal methods must be used for these detection problems.

This survey paper gives an outline and comparison of a number of suboptimal detection algorithms which have emerged in recent years for the detection of 2-D intersymbol interference. Here, these algorithms are categorized as either 1. using multiple locally optimal algorithms which, when combined, are globally suboptimal or 2. applying a suboptimal decoding algorithm to the complete trellis.

In Sect. 3, the first category of algorithms is described, which use "strip-wise" detection or belief-propagation detection. Two types of strip-wise algorithms are included: the hard output multitrack Viterbi algorithm, and the soft-output iterative multistrip algorithm, based upon the BCJR algorithm [9]. Belief propagation techniques, which have received substantial attention for decoding error correcting codes [10], and have been successfully considered for 2-D detection [11], [12].

In Sect. 4, the second category of detection algorithms is considered, which simply constructs the complete trellis, and then applies a sub-optimal decoding algorithm, such as the M-Viterbi algorithm.

For most of these algorithms, most authors give examples of their algorithms coming within 0.2 to 1.0 dB of the performance of an ideal detector operating on an uncoded channel, at high signal-to-noise ratio. These algorithms have complexity that is linear in the number of data bits, although excellent performance is usually achieved by setting some tuning parameter.

For coded intersymbol-interference channels in one dimension, turbo equalization appears to be a practical way to approach the channel capacity [13]. In turbo equalization, channel detection and error correction is performed jointly, with soft information shared iteratively between the detector and the decoder [14]. Soft-output algorithms are covered by this survey because they are needed for turbo equalization. Turbo equalization for 2-D channels has been considered, with generally favorable results, see Singla et al. and references therein [12].

Finally, in Sect. 5 we engage in some discussion, to identify some of the open problems that are being faced, and give the conclusion.

2. Model and Assumptions

2.1 Problem under Consideration

The following model and assumptions are used to describe the problem of data detection in the presence of 2-D inter-
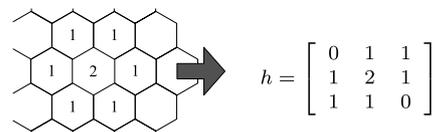
Fig. 2 Hexagonal recording model and corresponding rectangular response.

symbol interference and additive white Gaussian noise. Let $a[x, y] \in \{-1, +1\}$ denote binary data distributed in an array with M rows and N columns, with $x = 1, 2, \dots, M$ and $y = 1, 2, \dots, N$. Assume that $M \leq N$, because the field can be rotated 90 degrees to satisfy this condition.

Let h denote the discrete-space channel response (intersymbol interference), which has finite extent L_M-by-L_N:

$$h = \begin{bmatrix} h_{0,0} & \cdots & h_{0,L_N} \\ \vdots & & \vdots \\ h_{L_M,0} & \cdots & h_{L_M,L_N} \end{bmatrix} \tag{1}$$

The channel response is assumed to be linear space invariant (in one dimension, this would be linear time invariant).

The noiseless channel output $c[x, y]$ is the 2-D convolution of the binary data and the channel response:

$$c[x, y] = \sum_{p=0}^{L_M} \sum_{q=0}^{L_N} h_{p,q} a[x - p + x_{\text{off}}, y - q + y_{\text{off}}],$$
$$\text{for } 1 \leq x \leq M, 1 \leq y \leq N, \tag{2}$$

where x_{off} and $y_{\text{off}} \geq 0$ are non-causality offsets. For example, for channels with $L_M = L_N = 2$, it is convenient to assume that the central interferer is dominant, and so $x_{\text{off}} = y_{\text{off}} = 1$. For channels with $L_M = L_N = 1$, if "time" is increasing in the southeast direction, then $x_{\text{off}} = y_{\text{off}} = 0$ makes the impulse response purely causal.

The AWGN $n[x, y]$ with mean 0 and variance σ^2 is added to obtain the channel output $y[x, y]$:

$$y[x, y] = c[x, y] + n[x, y]. \tag{3}$$

The channel response h, and the noise variance σ^2 is known to the receiver. There are known termination bits outside of the x-y plane, that is $a[x, y] = -1$ for all x and y for which $1 \leq x \leq M$, $1 \leq y \leq N$ does not hold.

Note that any regular non-rectangular arrangement can be converted to the rectangular arrangement described above. For example, Maeda suggested to improve the packing density for optical storage by placing data onto a hexagonal, rather than rectangular, lattice, which is the optimal packing in two dimensions [15]. In fact, the hexagonal response shown in Fig. 2 has been proposed as a first-order model for a two-dimensional optical storage (TwoDOS) system [16]. From here, the rectangular data pattern will be assumed without any loss of generality.

In practice, partial response equalization may be used to match the channel characteristics to the detector response. This survey paper concentrates on detection algorithms for the white-noise channel, and assumes that no equalization is necessary.
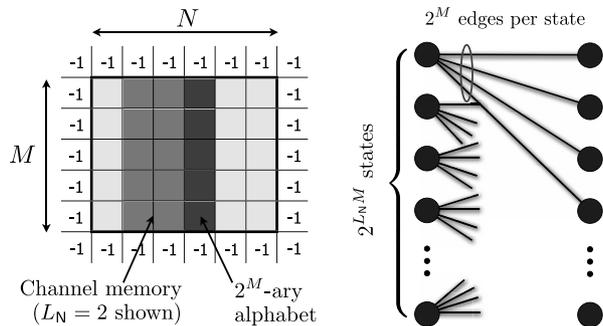
Fig. 3 Construction of a trellis for optimal 2-D detection.

2.2 Optimal Detection

There are two senses of optimality for detection, one is maximum likelihood sequence detection optimality. The other is a posteriori probability computation, which is used to compute the maximum a posteriori (MAP) decisions on the bit. However, in recent practice, the a posteriori probabilities, rather than the MAP decisions, are of interest because they are the soft information used in turbo equalization.

The maximum likelihood sequence estimate is the data sequence $\widehat{\mathbf{a}}$ from the set of all possible sequences $\{-1, 1\}^{M \times N}$ which maximizes the likelihood of $\mathbf{y}$, the sequence of $y[x, y]$ for $1 \le x \le M, 1 \le y \le N$, as in (3). That is,

$$\widehat{\mathbf{a}} = \arg \max_{\mathbf{a} \in \{-1,1\}^{M \times N}} Pr(\mathbf{y}|\mathbf{a}). \tag{4}$$

On the other hand, the a posteriori probability for an individual bit is given by the marginal block posterior probability as:

$$Pr(a[x, y] = b|\mathbf{y}) = \sum_{t \in \{-1,+1\}^{M \times N-1}} Pr(\mathbf{a} = (b, t)|\mathbf{y}),$$

with $b \in \{-1, +1\}$.

In the one-dimensional case, a trellis can be constructed which represents the intersymbol interference. The Viterbi algorithm, applied to this trellis, produces the maximum-likelihood sequence estimate. The BCJR algorithm, applied to the same trellis, produces the a posteriori probabilities. If v is the 1-D channel memory, the number of trellis transitions (or branches) per information bit is proportional to 2^v. In this paper, the number of trellis branches to detect all data bits is used as a measure of algorithmic complexity, since most algorithms have a fixed number of computations per trellis branch.

In the 2-D case, one may also construct a trellis, by considering each column of binary symbols as a single symbol from an alphabet of 2^M symbols. In this way, the 2-D problem can be considered as an equivalent 1-D ISI channel over a higher-order alphabet. Such a trellis has 2^{ML_n} states, and 2^M transitions per state, see Fig. 3. Each time index corresponds to M bits, so the total number of trellis transitions to detect MN bits is $N2^{M(L_n+1)}$. Thus, there is a significant

problem: while optimal 1-D detection has linear complexity in the number of bits, 2-D detection is exponential in the number of rows M.

3. Locally Optimal Algorithms

3.1 Overview

This section describes approaches to 2-D detection where the large detection problem is partitioned into several smaller problems which are less complex. However, because the smaller algorithms must share information, the result is usually globally suboptimal.

One method to partition the problem is to separate the 2-D field into overlapping strips. Section 3.2 describes the idea of a strip-wise trellis, and establishes notation. Using this trellis, hard-output Viterbi-like algorithms, called multitrack Viterbi algorithm (MVA), are described in Sect. 3.3. Soft-output BCJR-like algorithms, also using this trellis, are described in Sect. 3.4.

Another approach is to use belief propagation, which amounts to partitioning the area into small regions, detecting each region optimally, and iteratively sharing soft information between the adjoining regions. This is described in Sect. 3.5.

3.2 Strip-Wise Trellis

Strip-wise algorithms specify a strip width S, specify R rows of received symbols to be used as inputs, and specify A rows of bits which are to be estimated. The M-by-N field is separated into horizontal strips of size S-by-N, where S is the strip width. These strips may overlap, and in the most general case, adjacent strips have $S - 1$ rows in common. The topmost strip includes only the first row of information bits, $a[1, y], y = 1, \ldots, N$, and the remainder is the termination bits. On the other hand, the bottom-most strip includes only the last row of information bits, $a[M, y]$, and similarly the rest of the strip covers the bottom termination bits. The strip numbers range from $k = 1$ to $k = M + L_M$.

Now, a trellis can be constructed for the strip. The trellis state at time t consists of S-by-L_N symbols:

$$\mathbf{s}_t = \begin{bmatrix} s_t[1, 1] & \cdots & s_t[1, L_N] \\ \vdots & & \vdots \\ s_t[S, 1] & \cdots & s_t[S, L_N] \end{bmatrix}, \tag{5}$$

for $t = 1, 2, \ldots, N + L_N$. The relationship state bits are the data bits in strip k at time t is given by:

$$s_t[i, j] = a[i + k - S, j + t - 1] \tag{6}$$

Thus, the trellis has $2^{S L_N}$ states.

Each state $\mathbf{s}_t$ has 2^S transitions, leading to states $\mathbf{s}_{t+1}$. The trellis has R output labels for c, for specified rows within that strip. Similarly, the trellis has A input labels for a, for specified rows within that strip. The output label for row r is:
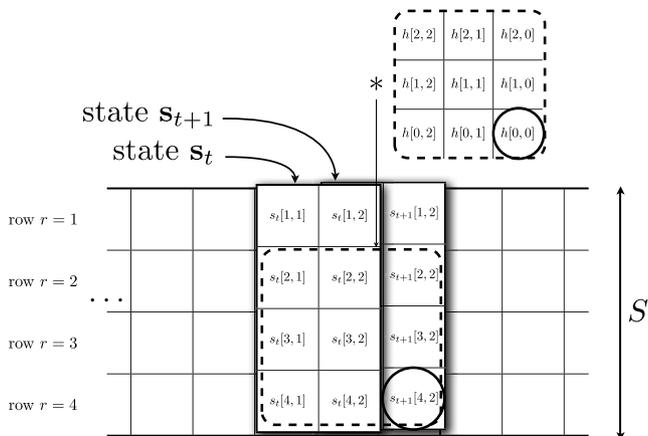
Fig. 4 Example of finding the 2-D strip trellis, with $S = 4$, $L_M = L_N = 2$. Shown, as circled, is how to compute $c(r, \mathbf{s}_t, \mathbf{s}_{t+1})$, for row $r = 4$. The symbols in the two dashed boxes are convolved to find the trellis output label for row $r = 4$.

$$c[r, \mathbf{s}_t, \mathbf{s}_{t+1}] = \sum_{q=1}^{L_N} \sum_{p=0}^{L_M} h[p, q] s_t[r - p, L_N - q + 1]$$
$$+ \sum_{p=0}^{L_M} h[p, 0] s_{t+1}[r - p, L_N]. \tag{7}$$

Note that not all rows are necessarily selected to generate an output label, and it will be an important difference between two algorithms. An example finding the output labels is shown in Fig. 4. The example uses $S = 4$ and $L_M = 2$ for clarity, however practical strip-wise algorithms use $S \leq L_M + 1$.

The trellis transitions also have A input labels, which can be found in a straightforward manner from the state labels $\mathbf{s}_t$.

3.3 The Multitrack Viterbi Algorithm

The multitrack Viterbi algorithm (MVA) applies the Viterbi algorithm to strips. The MVA appears to have its beginnings in a 1989 paper, which concentrated on single-track detection in the presence of multi-track ISI [17]. True multitrack detection was accomplished by using decision feedback, where hard outputs from detecting the first track used as inputs on the input to detection of the second track, and was referred to as the decision feedback Viterbi algorithm [18].

Greater detail was developed in two dissertations, where the algorithm was named "MVA" [19], [20]. Starting at the top edge, the Viterbi algorithm operates over a row of width $S = L_M$ and using all received symbols, $R = S$, and makes hard decisions for bits in the top row of width one (recall that termination bits above the top row are known). The Viterbi algorithm operates again in the same manner, shifted down by one row, but using the prior hard decisions. This proceeds row-by-row from top to bottom. The MVA algorithm is illustrated by considering just the processors V0,
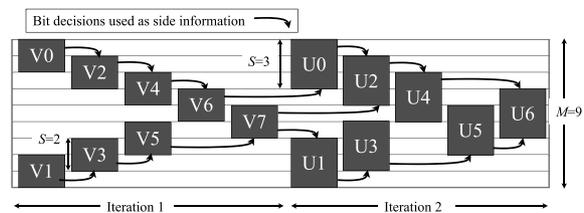
Fig. 5 Implementation of the refined MVA. The arrows show how hard inputs from one detector are used as side information at another detector.

V2, V4, ... of the first iteration, in Fig. 5.

For applications optical storage, several improvements were suggested for use with the TwoDOS channel, resulting in the "refined MVA" algorithm [21]. The refined MVA uses strip-wise detectors in two iterations. In the first iteration, $M - 1$ detectors have strip width $S = 2$, where each detector uses $R = 2$ received symbol rows and $A = 1$ bit rows (the outermost bit row is output, except the last stage which outputs two bit rows). In the second iteration, $M - 2$ detectors use $S = 3$, $R = 3$ and $A = 1$. Figure 5 illustrates the various detectors. The detectors operate in left-to-right order, and bottom-to-top processing is also used.

Significantly, the MVA uses $R = S$. That is, it uses all the received symbols in the strip. However, some trellis transition output labels cannot be computed, because the required information bits, used to form the state, are from rows above and below the strip and therefore are unknown; refer to Eq. (7). To solve this, refined MVA gives heuristic weighs the Euclidean branch metrics.

Using hard decisions is a form of decision feedback, which have a problem with error propagation. If the hard decisions made by processor V0 is incorrect, for example, this could have an unusually bad effect on the output of V2, which would propagate to V4, etc. This results in suboptimal performance. Increasing the width of each strip S will improve the performance of the detector, but this comes as the expense of complexity, which is exponential in S.

3.4 Iterative Multistrip (IMS) Algorithm

The iterative multistrip (IMS) algorithm uses strip-wise trellises as well, however applies the BCJR algorithm instead of the Viterbi algorithm. The BCJR algorithm is a soft-input, soft-output algorithm capable of producing and accepting probabilistic information, and this soft information is used to connect the multiple strip decoders together.

The IMS algorithm was proposed by Marrow and Wolf [22], [23], with the goal of producing a soft-output detector algorithm, which would be suitable for turbo equalization. A similar soft-output decoding algorithm was described independently [24, Ch. 5].

The IMS algorithm is an iterative algorithm. Generally, the strip width S is chosen to be equal to the channel memory, $S = L_M$. Each strip detector uses received symbols from only one row $R = 1$, but produces soft estimates for all bits within the strip, $A = S$. Note that it is possible to view
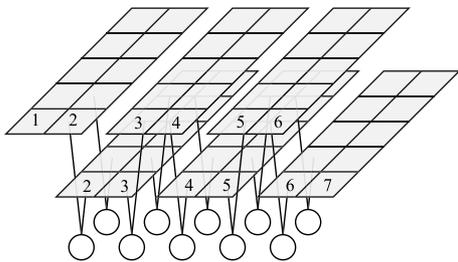
Fig. 6 Iterative multistrip algorithm. Circles connect bit cell which are the same.

Fig. 7 Bipartite graph for $L_M = L_N = 1$ detection. Circles are bit nodes corresponding to $a[x, y]$, and the triangles are sum-product nodes, Eq. (2).

the bits of a strip as the transitions of a finite state machine [24], but because the bits are memoryless this does not improve performance. Figure 6 illustrates the IMS algorithm for $S = L_M + 1 = 2$. A circle connects two bits which are the same.

In the IMS algorithm, the BCJR algorithm operates on each strip. The number of strips is equal to the number of rows M, and with each detector working with one row of received symbols. Each of the M detectors operates one time, and outputs soft a posteriori information to its neighbors. This is uses as a priori information input when the detectors run a second time. This is repeated for a fixed number of iterations. It is also possible to alternately operate on rows and columns of the 2-D array, although it is not clear how to efficiently share soft information between the various detectors [24].

As the number of iterations increases, the performance of this algorithm will improve, but this comes at the expense of increasing complexity. As with the MVA algorithm, increasing the width of each strip will favor performance at the expense of complexity. When $L_M = 1$ and $S = 2$, the graph, as shown in Fig. 6 appears to be loop-free, and when the iterations equal to the number of rows M, the IMS detection algorithm appears optimal.

3.5 Belief-Propagation Detection

Belief-propagation techniques, also called sum-product decoding, are a probabilistic decoding algorithm applied to a bipartite graph, and have been extremely successful in decoding of low-density parity-check codes [10], [25]. Such techniques have also been extended to the detection of 1-D intersymbol interference channels. As a stand-alone detector, the bipartite graphs have a large number of short cycles, which degrades performance, but when used with an outer error-correcting code, the algorithm appears promising [26].

For 2-D detection, a bipartite graph has a factor node (triangle) which represents the sum-product relationship between the bits, Eq. (2). The graph has a variable node (circle) which represents the bits, refer to Fig. 7. As given in [10], the factor node f computes an output message $m_{f \to i}(x_i)$ to send to bit i:

$$m_{f \to i}(x_i) = \sum_{\mathbf{x} \setminus x_i} Pr(y | \mathbf{x}, x_i) \prod_{j \in N(f) \setminus i} n_{j \to f}(x_j), \quad (8)$$
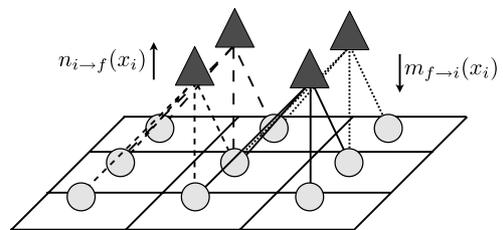
where $\mathbf{x}$ is the set of bits that the factor node is connected to, and $N(i)$ is the all the neighbors of the node except for i. The variable node for x_i computes an output message $n_{i \to f}(x_i)$ to send to factor node:

$$n_{i \to f}(x_i) = \prod_{c \in N(i) \setminus f} m_{c \to i}(x_i). \quad (9)$$

Messages are passed iteratively between the two types of nodes until a stopping condition, usually a fixed number of iterations, is reached.

Because the belief-propagation graph has a large number of loops, 2-D detection is not expected to perform well. The primary source of complexity is due the factor node computation, (8), which is order $O(2^{(L_M+1)(L_N+1)})$; in addition, complexity is linear in the number of iterations. However, this algorithm was combined with the a belief-propagation decoder for an outer low-density parity check code in a turbo equalization scheme, and noise thresholds were obtained [12].

While a stand-alone belief-propagation detector is not expected to perform well, generalized belief propagation [10] was successfully used in a slightly different application, the computation of the symmetric information rate, which is similar to the channel capacity, of an intersymbol interference channel [11]. Single belief-propagation nodes are grouped in to clusters, or regions of nodes. Within each cluster, optimal detection (or exact inference) is performed, and message are shared between adjoining regions. Shental et al. used cluster sizes of three-by-three on a channel with two-by-two interference ($L_M = L_N = 1$), which was sufficient to accurately compute the channel capacity. By using clusters, the effects of loops in the graph were minimized enough to obtain good performance. Using generalized belief-propagation detection on a three-by-three region has some resemblance to the IMS algorithm with a strip width of $S = 3$.

4. Suboptimal Algorithms on the Optimal Trellis

4.1 M-Viterbi Algorithm on 1-D Equivalent Trellis

A promising approach constructs an optimal trellis, and applies the M-Viterbi algorithm to it [27], [28]. In this approach, rather than the 2-D trellis described in Sect. 2, an equivalent 1-D trellis is constructed. This 1-D trellis first scans top-to-bottom, and then left-to-right, as shown in
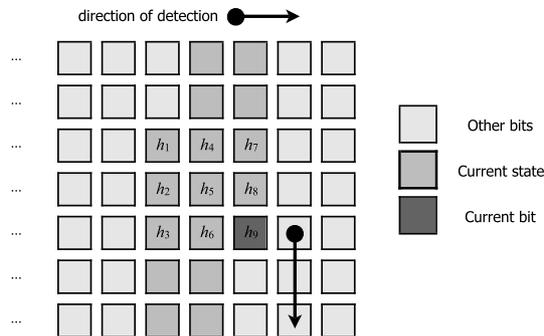
Fig. 8 Symbols used to construct the 1-D equivalent trellis.

Fig. 8. At each time instance, only one bit ("current bit") is detected, so each state has only two outgoing branches. However, the trellis memory ("current state") must be unwrapped to cover the entire channel memory, so that the trellis has $2^{ML_N+L_M+1}$ states. There are, however, NM trellis sections.

The M-Viterbi algorithm stores only the m best paths (m to avoid confusion with the number of rows M), and provides good performance-complexity tradeoff compared to the full Viterbi algorithm [29]. After each add-compare-select operation, $2m$ paths remain, and the main computational complexity is due to some sorting operation required to select the best m of these $2m$ paths. One such algorithm has requires about $2m(\log_2 2m)^2$ compare-select operations [30].

This algorithm was considered on the so-called 2PR4 channel, with:

$$h = \begin{bmatrix} \alpha & 0 & -\alpha \\ 1 & 0 & -1 \\ \alpha & 0 & -\alpha \end{bmatrix}, \tag{10}$$

where α is the intertrack-interference magnitude. For a system with $M = 8$ tracks, and $m = 16$ traceback paths, a 0.5 dB loss with respect to full-complexity Viterbi detection was found [27, Fig. 6] (BER not given by authors), which represents a favorable complexity-performance tradeoff. Changing the value of $\alpha = 0.25, 0.5, 1$ had little effect on the performance loss.

4.2 Reduced-Complexity Detection Algorithms

Besides the M-Viterbi algorithm, other reduced-complexity-type Viterbi algorithms for detection generally employ lists to select the best paths [31]. Such algorithms are particularly important when the trellis is non-binary.

In fact, the full 2-D detection trellis is non-binary, and reduced-complexity detection algorithms (RCDA) have been considered as a suboptimal detection algorithm, for a 2-D multitrack magnetic recording application [32]. This approach was evaluated for the following channel:

$$h = \begin{bmatrix} \alpha & \alpha & -\alpha & -\alpha \\ 1 & 1 & -1 & -1 \\ \alpha & \alpha & -\alpha & -\alpha \end{bmatrix}, \tag{11}$$

which represents an EPR4 channel with intertrack-interference of α. For the case of $M = 2$ tracks, the various parameters for the RCDA algorithm had a strong influence on the performance-complexity tradeoff. However, while the best performance found was only a 0.3 dB loss from the performance of the full Viterbi at a BER of 10^{-4} [32, Table II], the inter-track interference α was only 0.1, which currently is considered to be relatively weak.

5. Discussion and Conclusion

5.1 Complexity and Performance Comparisons

The complexity of all of the surveyed algorithms is linear in the number of bits, an important improvement over optimal detection, which has exponential complexity. The multitrack Viterbi algorithm (MVA) and the iterative multistrip (IMS) algorithm are based upon a strip trellis which has complexity which is exponential in only the channel memory. However, both the MVA and IMS algorithms are iterative, and the complexity is further linear in the number of iterations. Note also that the big-O complexity of the MVA/IMS algorithms and the belief-propagation algorithms are the same.

Complexity of the reduced complexity algorithms of Sect. 4 are difficult to compare because the main source of complexity lies in sorting operations. Good performance was found for the M-Viterbi algorithm when the number of tracks $M = 8$ and the number of traceback paths $m \geq 16$. However, it is reasonable to assume that if the number of tracks increases, that the number of required traceback paths, and thus the complexity, will increase worse than linearly.

The complexity of the various algorithms is compared in Table 1, and the performance comparison is made in Table 2.

5.2 Non-linearities

Linearity was assumed in this paper, but practical systems are non-linear and any serious implementation must account for this. In optical recording, examples include de-focus, tracking offset and recording fluctuations. For magnetic recording, media jitter and non-linear transition shifts are a serious problem. As compared to using linear filtering alone, an advantage of the trellis-based approaches in this paper is that the trellis labels can be modified to explicit

Table 1 Complexity comparison. The data field is N-by-M, the channel memory is L_M-by-L_N and m is the number of traceback paths of the M-Viterbi algorithm, I is number of iterations.

Trellis/Algorithm	Complexity $O(\cdot)$
Full Trellis	$N2^{M(L_N+1)}$
Full 1-D Trellis [27]	$NM2^{ML_N+L_M+1}$
M-Viterbi on 1-D Trellis [27]	$NMm(\log m)^2$
Strip Trellis [21] [23]	$NMI2^{(L_N+1)(L_M+1)}$
Belief-Propagation [12]	$NMI2^{(L_M+1)(L_N+1)}$

Table 2 Performance Comparisons. Number of tracks M, With respect to = "w.r.t."

Algorithm	Performance	Channel
Refined MVA [21]	1 dB loss w.r.t. Viterbi	TwoDOS channel, $M = 7$
IMS [23]	0.5 dB loss w.r.t union bound	TwoDOS, 10 iterations
M-Viterbi on 1-D Trellis [27]	> 0.8 dB loss w.r.t. Viterbi ($m = 8$) > 0.2 dB loss w.r.t. Viterbi ($m = 32$)	2PR4 Channel, $M = 8$
RCDA Viterbi [32]	0.3 dB loss w.r.t. Viterbi	$M = 2$ with EPR4

take account of certain types of non-linearities. The Two-DOS recording system has data-dependent non-linearities [33], and these were included in a detector design [12].

Trellis-based approaches have other advantages. For example, when an outer run-length limited code or convolutional code is used, both can be decoded together, if the trellis has been suitably designed [34], [35].

5.3 Suggestions for Future Research

The MVA and IMS algorithms are "window" algorithms, in the sense that after I iterations, only samples from a window consisting of $2I + 1$ rows have been contributed to the final decision. For a 1-D detection algorithm, when the number of iterations is small, this can seriously degrade the error rate performance; however, the use of a precoder was shown to fix this problem [26]. It would be reasonable to expect that a similar benefit could be obtained by using a 2-D precoder. Note that multidimensional convolutional codes could be used as the basis of a 2-D precoder [36].

For the MVA algorithm, the complexity-performance tradeoff was improved by using weak but low-complexity detector (strip width $S = 2$) in the first iteration, and using a higher complexity decoder ($S = 3$) in the second iteration. Also, each iteration of the MVA used all rows of received symbols ($R = S$), and obtained good performance in just two iterations. On the other hand, the IMS algorithm uses just one row of received symbols ($R = 1$), and required far more iterations. It would be reasonable to expect that the performance-complexity tradeoff of the IMS algorithm could be significantly improved.

5.4 Conclusion

Researchers working in diverse areas such as optical, magnetic and holographic storage have considered various approaches, sometimes independently, to detection of 2-D intersymbol interference channels. Thus, the problem of 2-D detection is a fundamental one, and this paper has surveyed the trellis-based approaches to the problem by presenting them in a common framework. In this way, similarities and differences were explained, and suggestions for future research were made.

Acknowledgments

References

[1] Z. Zhang, T.M. Duman, and E.M. Kurtas, "Achievable information rates and coding for MIMO systems over ISI channels and frequency-selective fading channels," IEEE Trans. Commun., vol.52, no.10, pp.1698–1710, 2004.

[2] A.D. Wyner, "Shannon-theoretic approach to a gaussian cellular multiple-access channel," IEEE Trans. Inf. Theory, vol.40, no.6, pp.1713–1727, 1994.

[3] Japanese Industrial Standard, Two dimensional symbol — QR code — Basic specification, JIS X 0510:2004, 2004.

[4] D. Manolakis, V.K. Ingle, and S.M. Kogon, Statistical and Adaptive Signal Processing, McGraw Hill, 2000. ISBN 0070400512.

[5] K. Wakabayashi, H. Sugiyama, A. Saito, and T. Maeda, "High density optical disk system using two-dimensional recording," IEICE Trans. Electron., vol.E78-C, no.11, pp.1582–1590, Nov. 1995.

[6] K. Chugg, X. Chen, and M. Neifeld, "Two-dimensional equalization in coherent and incoherent page-oriented optimal memory," J. Opt. Soc. Am., vol.16, no.3, pp.549–562, 1999.

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

[8] E. Ordentlich and R.M. Roth, "On the computational complexity of 2D maximum-likelihood sequence detection," Technical Report HPL-2006-69, HP Laboratories Palo Alto, July 2006.

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

[10] J.S. Yedidia, W.T. Freeman, and Y. Weiss, "Constructing free-energy approximations and generalized belief propagation algorithms," IEEE Trans. Inf. Theory, vol.51, no.7, pp.2282–2312, July 2005.

[11] O. Shental, N. Shental, and S. Shamai, "On the achievable information rates of finite-state input two-dimensional channels with memory," Proc. International Symposium on Information Theory, ISIT 2005, pp.2354–2358, Sept. 2005.

[12] N. Singla and J.A. O'Sullivan, "Joint equalization and decoding for nonlinear two-dimensional intersymbol interference channels," Proc. IEEE International Symposium on Information Theory, Adelaide, Australia, 2005.

[13] A. Kavcic, X. Ma, and M. Mitzenmacher, "Binary intersymbol interference channels: Gallager codes, density evolution and code performance bounds," IEEE Trans. Inf. Theory, vol.49, no.7, pp.1636–1652, July 2003.

[14] R. Koetter, A. Singer, and M. Tuchler, "Turbo equalization," IEEE Signal Process. Mag., vol.21, no.1, pp.67–80, Jan. 2004.

[15] T. Maeda. Private communication, 2007.

[16] A. Immink, W. Coene, A. van der Lee, C. Busch, A. Hekstra, J. Bergmans, J. Riani, S. Beneden, and T. Conway, "Sig-

nal processing and coding for two-dimensional optical storage," Global Telecommunications Conference, 2003. GLOBECOM'03, pp.3904–3908, Dec. 2003.

[17] H. Burkhardt, "Optimal data retrieval for high density storage," Proc. Optimal Data Retrieval for High Density Storage, pp.43–48, Hamburg, West Germany, May 1989.

[18] J.F. Heanue, K. Gürkan, and L. Hesselink, "Signal detection for page-access optical memories with intersymbol interference," J. Appl. Opt., vol.35, no.14, pp.2431–2438, May 1996.

[19] R. Krishnamoorthi, Two-dimensional Viterbi-like algorithms, Master's thesis, University of Illinois at Urbana-Champaign, 1998.

[20] W. Weeks, Full-surface data storage, Ph.D. thesis, University of Illinois at Urbana-Champaign, 2000.

[21] A.P. Hekstra, W.M.J. Coene, and A.H.J. Immink, "Refinements of multi-track Viterbi bit-detection," IEEE Trans. Magn., vol.43, no.7, pp.3333–3339, 2007.

[22] M. Marrow and J. Wolf, "Iterative detection of 2-dimensional ISI channels," Proc. Information Theory Workshop, 2003, pp.131–134, March/April 2003.

[23] M. Marrow and J. Wolf, "Detection of 2-dimensional signals in the presence of ISI and noise," Proc. International Symposium on Information Theory and its Applications, pp.891–894, Oct. 2003.

[24] K. Chugg, A. Anastasopoulos, and X. Chen, Iterative Detection: Adaptivity, Complexity Reduction, and Applications, Kluwer, Norwell, MA, USA, 2001.

[25] R.G. Gallager, Low-Density Parity-Check Codes, M.I.T. Press, Cambridge, MA, USA, 1963.

[26] B.M. Kurkoski, P.H. Siegel, and J.K. Wolf, "Joint message-passing decoding of LDPC codes and partial-response channels," IEEE Trans. Inf. Theory, vol.48, no.6, pp.1410–1422, June 2002.

[27] T. Conway, R. Conway, and S. Tosi, "Signal processing for multitrack digital data storage," IEEE Trans. Magn., vol.41, no.4, pp.1333–1339, 2005.

[28] S. Tosi and T. Conway, "Detector target response optimization for multitrack digital data storage," IEEE Trans. Magn., vol.42, no.7, pp.1926–1928, 2006.

[29] J. Anderson and S. Mohan, "Sequential coding algorithms: A survey and cost analysis," IEEE Trans. Commun., vol.32, no.2, pp.169–176, 1984.

[30] K.E. Batcher, "Sorting networks and their applications," AFIPS Proc. Spring Joint Comput. Conf., pp.307–314, 1968.

[31] T. Hashimoto, "A list-type reduced-constraint generalization of the viterbi algorithm," IEEE Trans. Inf. Theory, vol.33, no.6, pp.866–876, 1987.

[32] E. Kurtas, J.G. Proakis, and M. Salehi, "Reduced complexity maximum likelihood sequence estimation for multitrack high-density magnetic recording channels," IEEE Trans. Magn., vol.35, no.4, pp.2187–2193, 1999.

[33] W.M.J. Coene, "Nonlinear signal-processing model for scalar diffraction in optical recording," Appl. Opt., no.32, Nov. 2003.

[34] T. Kato, S. Taira, T. Maeda, Y. Katayama, and T. Nishiya, "Two-dimensional run-length-limited code and partial response maximum likelihood system with multi-track recording," ISOM/ODS 2002, Joint Int. Symp. Optical Memory and Optical Data Storage, p.5153, 2002. Post-Deadline Paper WP.23.

[35] N. Kogo, N. Hirano, and R. Kohno, "Convolutional code and 2-dimensional PRML class IV for multi-track magnetic recording system," Proc. IEEE International Symposium on Information Theory, p.309, 2000.

[36] P. Weiner, Multidimensional Convolutional Codes, Ph.D. Thesis, University of Notre Dame, 1998.

Brian M. Kurkoski was born in 1970, in Portland, Oregon, USA. He received the B.S. degree from the California Institute of Technology in 1993, and worked in industry for several years. He received the M.S. and Ph.D. degrees from the University of California San Diego in 2000 and 2004, respectively. At the University of Electro-Communications, Tokyo he was a postdoctoral researcher from 2004 to 2006, a research assistant until 2007, and since 2007 has been an associate professor.