IEEE TRANSACTIONS ON COMMUNICATIONS, VOL. 37, NO. 5, MAY 1989

Delayed Decision-Feedback Sequence Estimation

ALEXANDRA DUEL-HALLEN, MEMBER, IEEE, AND CHRIS HEEGARD, MEMBER, IEEE

Abstract—**Delayed decision-feedback sequence estimation (DDFSE) is a detection algorithm that provides a direct tradeoff between complexity and performance in digital communications over intersymbol interference channels. The complexity of the algorithm is controlled by a parameter μ and can be varied from zero to the memory of the channel (which can be infinite). The algorithm is based on a trellis with the number of states exponential in μ. When $\mu = 0$, DDFSE reduces to the decision-feedback detector. When the memory of the channel is finite, DDFSE with maximal complexity is equivalent to the Viterbi algorithm. Of course, if the channel has infinite memory, the Viterbi algorithm cannot be implemented. For the intermediate values of μ, the algorithm can be described as a reduced-state Viterbi algorithm with feedback incorporated into the structure of path metric computations.**

We first consider DDFSE for uncoded PAM signals. Estimates on the performance of the algorithm are given, and simulation results are provided for several examples. A more general form of DDFSE applicable to coded modulation systems is also presented. As an example, detection of trellis coded QPSK signals over intersymbol interference channels is discussed.

I. Introduction

CONSIDER the problem of detection of digital data in the presence of intersymbol interference (ISI) and additive noise. Assume that the receiver employs a matched filter which produces a sequence of sampled output values. Thus, a discrete time model of the channel is obtained. The objective of a detection algorithm is to produce a reliable decision of the input sequence given the received data.

Various approaches to data detection can be divided into symbol by symbol an sequence detection [4]. The first class contains linear and decision-feedback detectors. These schemes have low complexity and undesirably high error rates. Another approach to data detection is given by maximum-likelihood sequence estimation (MLSE) [5]. The trellis-based Viterbi algorithm solves the MLSE problem recursively when the memory of the channel is finite. The symbol error rate of the Viterbi algorithm is often much lower than error rates of the symbol by symbol detectors. However, the total storage (complexity) of the algorithm is proportional to the number of states of the trellis which grows exponentially with the channel memory length. When the channel memory becomes large (it can be infinite), the algorithm becomes impractical.

It is desirable to construct a detection method that reduces the complexity of the Viterbi algorithm by "shortening" the memory of the channel. Various approaches are described in [1]–[3], [8]–[10], [16], [18]–[22]. A popular method is the direct truncation of the channel response. In this scheme, analyzed by McLane [22], residual interference terms are not taken into account by the detector, and cause severe error propagation. Another approach is to use a linear (or a decision feedback) equalizer to estimate the input sequence and use these estimates to cancel the tail of the ISI in the received sequence prior to passing it to the Viterbi algorithm [8]. However, prefiltering still results in significant error propagation and high error rate.

Delayed decision-feedback sequence estimation (DDFSE) is another method to reduce the number of states. As in the Viterbi algorithm, at each step, the states describe all possible values taken on by a finite number μ of previous inputs. While this number μ is equal to the memory length of the channel for the Viterbi algorithm, in DDFSE it can be freely chosen; μ is finite when the channel memory is infinite. In the DDFSE algorithm, each state provides only partial information about the actual state of the channel. The required residual information is provided by an estimate associated with each state of the trellis. In principle, this information can be extracted from the path leading to each state. The channel state estimate and the trellis state are used in computing the branch metric. This operation is similar to decision-feedback equalization since the partial state estimate can be used to estimate the tail of the ISI in the received signal.

The complexity of DDFSE is determined by the number of trellis states, which is exponential in μ. At its minimal value ($\mu = 0$), the algorithm reduces to the decision-feedback detection. When the memory length of the channel η is finite, DDFSE with the maximal value of μ (given by η) is equivalent to the Viterbi algorithm. When the memory of the channel is infinite, the Viterbi algorithm cannot be implemented. The symbol error rate of DDFSE was estimated and shown to approach rapidly the symbol error rate of MLSE as μ grows large. This analysis is confirmed by computer simulations for several examples.

Large input alphabet sizes which arise in communications over bandlimited channels contribute to high complexity of the MLSE algorithm along with large memory lengths of ISI channels. This problem was addressed in the papers by Eyuboglu and Quereshi [9], [10]. They independently developed a reduced-state algorithm similar to DDFSE. In addition to introducing feedback into the structure of the path metric computations, they proposed to reduce complexity further by using ideas of set partitioning [12], [13]. The resulting algorithm, called RSSE, is applicable to finite memory channels with large input alphabets. On the other hand, DDFSE can handle infinite ISI channels with rational responses by using recursion in path metric computations. Thus, DDFSE is useful for a wider class of channels, whereas RSSE is more applicable for systems with large input alphabet sizes. The applications of RSSE considered in [9], [10] differ from the applications of DDFSE studied in this paper. In [9], [10], performance of RSSE is evaluated when it is used for detection of uncoded signals transmitted over narrow-band channels. We chose optical and magnetic recording channel models for the study of the DDFSE algorithm. As another important application, we consider the use of DDFSE for detection of trellis coded signals on ISI channels.

In Section II, we present the discrete time, white Gaussian noise channel model which arises at the output of a sampled, whitened matched filter receiver in pulse amplitude modula-

Paper approved by the Editor for Signal Design, Modulation, and Detection of the IEEE Communications Society. Manuscript received July 10, 1987; revised May 19, 1988. This work was supported by the National Science Foundation under Grant ECS-8352220, the IBM Faculty Development Award Program, and an AT&T Bell Laboratories Ph.D. Scholarship. This paper was presented in part at the 23rd Annual Allerton Conference on Communication, Control, and Computing, Monticello, IL, October 2-4, 1985.

The authors are with the School of Electrical Engineering, Cornell University, Ithaca, NY, 14853.

IEEE Log Number 8926992.

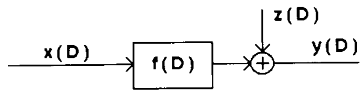


Fig. 1. The discrete time channel model.

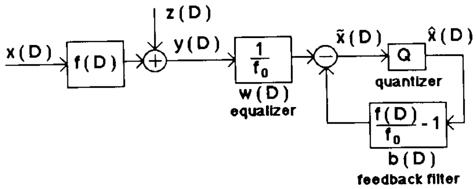


Fig. 2. The discrete time channel model and the zero-forcing decision feedback detector.

tion systems. Section III contains the description of the zero-forcing decision feedback detector and MLSE for this channel model. We develop DDFSE in Section IV. This section also contains examples of the DDFSE algorithm for a finite impulse response channel and for two infinite impulse response channel models arising in recording. Performance analysis of the DDFSE algorithm and simulation examples are given in Section V. In Section VI, the DDFSE algorithm is presented for trellis coded signals on ISI channels.

II. Discrete Time, White Gaussian Noise Model

Consider the following discrete time channel model. The input to the channel is given by a sequence of points drawn from a finite subset of real numbers X called the *modulation set*. The data sequence x_k is described by the D transform

$$x(D)=x_0+x_1D+\cdots+x_{n-1}D^{n-1}.$$

The output sequence $y_k = \Sigma_{i=0}^{\infty} f_i x_{k-i} + z_k$ where f_k is the channel transfer function, $f_0 > 0$ and z_k is an additive white Gaussian noise sequence (AWGN) (i.e., an i.i.d sequence of Gaussian random variables with zero mean and variance $Ez_k^2 = N_0$). Thus, in terms of D transforms (Fig. 1),

$$y(D)=f(D)x(D)+z(D). \quad (1)$$

We assume that the causal function $f(D)$ is rational, stable and can be factored as $f(D) = e(D)h(D)$ where $e(D)$ is a polynomial with all roots on the unit circle, and $h(D)$ is *minimum phase* (i.e., both $h(D)$ and $1/h(D)$ are causal and stable). The channel memory is defined as the degree of $f(D)$, $\eta = \deg(f(D))$, which is finite if $f(D)$ is a polynomial.

This discrete-time channel model arises in the pulse amplitude modulation (PAM) systems at the output of a sampled, whitened matched filter receiver [5], [16] (the receiver filter can be chosen so that $f(D)$ satisfies above assumptions). As pointed out in the following section, this receiver is optimal for zero-forcing decision-feedback detection. In addition, the performance of the DDFSE algorithm benefits from selecting $h(D)$ as a minimum phase filter, since the energy of the first μ terms is maximized [10], [26]. Note that the performance of MLSE does not depend on this assumption [5], [11].

A more general model arises when the signal y_k in (1) takes on complex values [11]. The extension of the results of the paper to the complex case is straightforward and is discussed in the example which deals with trellis coded quadrature amplitude modulated (QAM) systems [12], [13].

III. Detection Algorithms

A. Zero-Forcing Decision Feedback Detector

A *zero-forcing decision feedback (DF)* detector [17] is an example of a symbol by symbol detection method (Fig. 2). It involves two filters, an equalizer $w(D)$ and a feedback filter $b(D)$, chosen to eliminate all ISI and to maximize the signal-to-noise ratio (SNR) at the input to the quantizer. For the system (1) with the transfer function $f(D)$ specified in Section II, the filters are [16]

$$w(D)=1/f_0, \quad (2)$$

$$b(D)=f(D)/f_0-1. \quad (3)$$

When x_k is a binary equiprobable i.i.d sequence ($P(x_k = 0) = P(x_k = 1) = 1/2$), then the signal-to-noise ratio at the input to the quantizer is given by

$$\mathrm{SNR}=f_0^2/(4N_0). \quad (4)$$

In the absence of previous decision errors, the probability that the quantizer produces a symbol error is

$$Q(\sqrt{\mathrm{SNR}})=Q[f_0/(2\sqrt{N_0})] \quad (5)$$

where the Q function

$$Q(x)=\frac{1}{\sqrt{2\pi}}\int_x^{\infty} e^{-y^2/2}\,dy. \quad (6)$$

B. Maximum-Likelihood Sequence Estimation

Another approach to detection is illustrated by *maximum-likelihood sequence estimation* (MLSE) [5]. Given the output sequence $y(1)$, the objective of MLSE is to choose an estimate, $\hat{x}$, of the input sequence that maximizes the *likelihood function* $\phi(y|x)$ (the conditional probability density of the output y given the input x). Since the *metric M* of $x(D)$,

$$M=\|y(D)-f(D)x(D)\|^2=\sum_k (y_k-t_k)^2 \quad (7)$$

[where $t(D) = f(D)x(D)$], is proportional to the *log likelihood function*—log $\phi(y|x)$, the sequence $\hat{x}(D)$ that minimizes the metric M is the solution to the maximum-likelihood sequence estimation problem.

When the channel memory $\eta = \deg(f(D))$ is finite, the solution to the MLSE problem (7) can be computed recursively by the Viterbi algorithm [5]. This recursive solution follows from the finite-state machine (FSM) description of the channel. In such a FSM representation, each state s_k corresponds to the state of the channel at time k, given by the string of past inputs, $s_k = (x_{k-1}, \cdots x_{k-\eta})$. The transition or *branch* between states s_k and $s_{k+1} = (x_k, \cdots x_{k+1-\eta})$ is labeled with the input x_k and the filter output $t_k = \Sigma_{i=0}^{\eta} f_i x_{k-i}$. The *branch metric* is defined as $(y_k - t_k)^2$. When the FSM is described by a trellis with $|X|^{\eta}$ states, each input sequence $x(D)$ corresponds to a path through the trellis (i.e., a sequence of branches and states). Then the *path metric M* of $x(D)$, given by (7), is equal to the sum of the branch metrics associated with the path of $x(D)$. Thus, the path metric of $x(D)$ can be computed recursively as $M(s_{k+1}) = M(s_k) + (y_k - t_k)^2$ [where $s_0, s_1, \cdots$ is the state path of $x(D)$].

Since the Viterbi algorithm stores information for each possible state of the FSM, the amount of storage is proportional to the number of states $|X|^{\eta}$. Thus, when η is very large, the Viterbi algorithm becomes impractical; the algorithm breaks down when η is infinite.

IV. Delayed Decision-Feedback Sequence Estimation for Gaussian Channels

A. UV Decomposition of Channel

Any discrete-time channel specified by a causal, rational transfer function $f(D)$ can be described in terms of a state machine with the state space S chosen as a function of the obvious realization of the system; $|S| = \infty$ if $\eta = \deg(f(D))$ is infinite. The state space S can always be decomposed $S = U \times V$ where $|U| = |X|^\mu$ where μ is finite and $\mu \le \eta$. The decomposition $S = U \times V$ is obtained by representing the rational function $f(D) = \beta(D)/\gamma(D)$ [where $\beta(D)$ and $\gamma(D)$ are polynomials; $\gamma_0 = 1$ and $\gamma(D) = 1$ if η is finite] as the sum

$$f(D) = f_\mu(D) + D^{\mu+1} f^+(D) \tag{8}$$

where μ is the *reduced memory* of the channel ($\mu \le \eta$ and $\mu < \infty$),

$$f_\mu(D) = \sum_{i=0}^{\mu} f_i D^i, \; f^+(D) = \sum_{i=\mu+1}^{\eta} f_i D^{i-\mu-1}. \tag{9}$$

Note that $f^+(D) = (\beta(D) - f_\mu(D)\gamma(D))D^{-(\mu+1)}/\gamma(D) = \beta^+(D)/\gamma(D)$ is also rational. Let $\deg(\beta^+(D)) = n$, $\deg(\gamma(D)) = m$ and $w(D) = f^+(D)x(D)$. Then

$$w_k = \sum_{i=0}^{n} \beta_i^+ x_{k-i} - \sum_{i=1}^{m} \gamma_i w_{k-i} \text{ if } \eta = \infty \quad (m > 0), \tag{10a}$$

$$w_k = \sum_{i=0}^{\eta-\mu-1} f_{i+\mu+1} x_{k-i} \text{ if } \eta < \infty \quad (m = 0). \tag{10b}$$

The output of the channel is given by

$$y_k = \sum_{i=0}^{\mu} f_i x_{k-i} + w_{k-\mu-1} + z_k. \tag{10c}$$

From (10), the state of the system at time k can be chosen as

$$s_k = \begin{cases} (x_{k-1}, \cdots, x_{k-\mu}, x_{k-\mu-1}, \cdots x_{k-\mu-n}, w_{k-\mu-1}, \\ \quad \cdots w_{k-\mu-m}) \quad \eta = \infty \\ (x_{k-1}, \cdots x_{k-\mu}, x_{k-\mu-1}, \cdots x_{k-\eta}) \quad \eta < \infty. \end{cases}$$

The state is decomposed into

$$u_k = (x_{k-1}, \cdots x_{k-\mu}), \tag{11a}$$

$$v_k = \begin{cases} (x_{k-\mu-1}, \cdots x_{k-\mu-n}, w_{k-\mu-1}, \cdots w_{k-\mu-m}) \; \eta = \infty. \\ (x_{k-\mu-1}, \cdots x_{k-\eta}) \quad \eta < \infty. \end{cases} \tag{11b}$$

Note that the state v_{k+1} is a function of u_k and v_k if $\mu > 0$. (When $\mu = 0$, v_{k+1} is a function of x_k and v_k. In the rest of this section, we consider the case $\mu > 0$.) The partial state estimator $\hat{v}(u_k, v_k)$ is defined as the mapping that associates the value of v_{k+1} with states u_k and v_k.

B. Description of the Algorithm

The delayed decision-feedback sequence estimation algorithm (DDFSE) recursively finds an approximation to the maximum-likelihood sequence estimation problem. It is based on a trellis with a reduced number of states. The states of the trellis are given by the elements of the U-space (11a). At time $k + 1$, the algorithm stores for each possible state in U:

1) the best path leading to that state,
2) the metric of that path,
3) an estimate of the partial state in V.

The recursion step involves the following.

1) Computing for each (state, next state) pair the sum of the path metric plus the branch metric given by $(y_k - t_k)^2$ with $t_k = \sum_{i=0}^{\mu} f_i x_{k-i} + \tilde{w}_{k-\mu-1}$ where $(x_k, \cdots, x_{k-\mu})$ are determined by the (state, next state) pair. The estimate, $\tilde{w}_{k-\mu-1}$, of $w_{k-\mu-1}$ is obtained from the estimate of the partial state in V [stored with (state)] using (10, 11).

2) For each value of (next state) the best (smallest) metric sum is determined and the (state) which gives rise to the best (smallest sum) edge is selected.

3) For each value of (next state) an estimate of the partial state in V is made by applying the partial state estimator $\hat{v}$ to the (state) chosen in 2). As in the Viterbi algorithm, the path leading to each (next state) is found by extending the best path determined in 2).

The DDFSE algorithm requires an amount of storage proportional to the number of states in U (i.e., $|X|^\mu$). The algorithm complexity is proportional to the length of the sequence times $|X|^\mu$.

The DDFSE algorithm combines structures of the Viterbi algorithm and the decision feedback detector. As in the Viterbi algorithm, it uses a state machine description of the channel to recursively estimate the best path in the trellis while storing only one path for each state. But since each state of the DDFSE trellis provides only partial information about the full state of the channel, the algorithm also uses the best path leading to each state to compute the metric. An estimate of the partial state in V stores "feedback information" extracted from the best path. By analogy with the decision-feedback detector, this estimate is used to cancel interference from past inputs greater than μ samples in the past. Thus, the decision feedback operation is delayed by μ. Note that if $U = S$, the algorithm reduces to the Viterbi algorithm, If $V = S$, the algorithm is equivalent to the zero-forcing decision feedback detection.

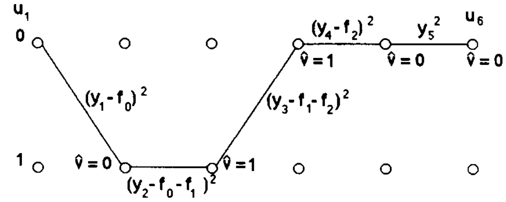


Fig. 3. A path through the trellis of the DDFSE algorithm ($\mu = 1$) for the FIR channel with response $f(D) = f_0 + f_1 D + F_2 D^2$.

C. Examples

Example 1) Finite Impulse Response Channel: Consider a finite impulse response (FIR) discrete time channel with transfer function

$$f(D) = f_0 + f_1 D + f_2 D^2. \tag{12}$$

The zero-forcing, DF detector for this channel (2, 3) is given by an equalizer $w(D) = 1/f_0$, and a feedback filter $b(D) = (f_1 D + f_2 D^2)/f_0$. The Viterbi algorithm is based on a trellis with the state space given by $S = X \times X$. At time k the state is given by (x_{k-1}, x_{k-2}).

The DDFSE algorithm with $\mu = 1$ decomposes the state space $S = U \times V$, with $U = V = X$. An estimate of $v_k = x_{k-2} \in V$ is stored with each state $u_k = x_{k-1} \in U$ in the DDFSE trellis. The branch metric is given by $(y_k - f_0 x_k - f_1 x_{k-1} - \tilde{w}_{k-2})^2$ where $\tilde{w}_{k-2}$ is the estimate of $f_2 x_{k-2}$ obtained from the estimate of $x_{k-2} \in V$. The partial state estimator is $v_{k+1} = \hat{v}(u_k, v_k) = u_k$. Fig. 3 shows a path through the trellis of the DDFSE ($\mu = 1$) with partial state estimates for states along the path and branch metrics.

Example 2) A One-Pole Channel Model: One simple model for optical recording arises when the channel filter in

(1) is chosen as the infinite series

$$f(D)=A\sum_{i=0}^{\infty}\alpha^k D^K=\frac{A}{1-\alpha D},\quad 0<\alpha<1. \qquad (13)$$

The response $f(D)$ is minimum phase; the zero-forcing decision-feedback equalizer (2, 3) results in the forward filter $w(D) = 1/A$ and the feedback filter $b(D) = \alpha D/(1 - \alpha D)$.

The Viterbi algorithm can not be applied to this channel since the ISI is infinite. For the DDFSE with complexity parameter μ (9), $f_\mu(D) = A(1 + \alpha D + \cdots + \alpha^\mu D^\mu), f^+(D) = A\alpha^{\mu+1}/(1 - \alpha D)$. Then

$$y_k=A(x_k+\alpha x_{k-1}+\cdots+\alpha^\mu x_{k-\mu})+w_{k-\mu-1}+z_k$$

where $w_k=A\alpha^{\mu+1}x_k+\alpha w_{k-1}$. The state of the channel factors into $u_k=(x_{k-1},\cdots,x_{k-\mu})$ and $v_k = w_{k-\mu-1}$ (11). The branch metric is thus given by

$$(y_k-A(x_k-\alpha x_{k-1}-\cdots\alpha^\mu x_{k-\mu})-\tilde{w}_{k-\mu-1})^2$$

where $\tilde{w}_{k-\mu-1}$ is an estimate of $w_{k-\mu-1}$ (given by values in V stored with states in U) and the partial state estimator is $\hat{v}(u_k, v_k) = A\alpha^{\mu+1}x_{k-\mu} + \alpha v_k$.

Example 3) A One-Pole, One-Zero Channel Model: A similar simple model for magnetic recording is obtained by choosing $f(D)$ to be

$$f(D)=\frac{A(1-D)}{1-\alpha D}. \qquad (14)$$

The $(1 - D)$ factor accounts for the differentiation in the read process of a magnetic recording channel.

The zero-forcing decision-feedback equalizer (2, 3) has the same $w(D)$ as the one-pole channel and

$$b(D)=D(\alpha-1)/(1-\alpha D).$$

Since the filter $f(D)$ is an infinite impulse response (IIR), the Viterbi algorithm does not exist for this channel. In DDFSE, (9) $f_\mu(D) = A(1 + \Sigma_{i=1}^{\mu}\alpha^i D^i(\alpha - 1)/\alpha), f^+(D) = A\alpha^\mu(\alpha - 1)/(1 - \alpha D)$. The states u_k and v_k are the same as in the one-pole example. The partial state estimator is $\hat{v}(u_k, v_k) = A\alpha(\alpha - 1)x_{k-\mu} + \alpha v_k$.

V. Analysis and Bounds

A. Derivation

Consider first a discrete time FIR channel (1) with $f(D) = f_0 + f_1 D + \cdots + f_\eta D^\eta$ where η is finite. The Viterbi algorithm for this channel has $|X|^\eta$ trellis states. Our error analysis of the DDFSE is based on error analysis for MLSE [5]. Thus, we first outline how an upper bound on the symbol error rate is found for the Viterbi algorithm. We assume that the input alphabet is $X = \{0, 1, \cdots m - 1\}$ where m is an integer. Let $x(D)$ and $\hat{x}(D)$ be the input sequence and the estimate produced by the algorithm. Define the input error sequence as $e(D) = x(D) - \hat{x}(D)$. Let $s(D)$ be the sequence of channel states in the path of $x(D)$, and let $\hat{s}(D)$ be the same for $\hat{x}(D)$. Define

$$t(D)=f(D)x(D),\quad \hat{t}(D)=f(D)\hat{x}(D) \qquad (15)$$

as the signal sequence and its estimate. Then an *error event* occurs between times k_1 and k_2 if states $s_{k_1} = \hat{s}_{k_1}$, $s_{k_2} = \hat{s}_{k_2}$, but $s_k \neq \hat{s}_k$ for $k_1 < k < k_2$. The squared *distance* of the error event is defined as $d^2 = \Sigma_{k=k_1}^{k_2-1}(t_k - \hat{t}_k)^2$. For moderate signal-to-noise ratios [5], an upper bound on the symbol error rate is dominated by the term

$$Q\left[\frac{1}{2}\frac{|d_{\min}|}{\sqrt{N_0}}\right]\sum_{\lambda\in\Lambda_{d_{\min}}} w(\lambda)\prod_{i=0}^{n-1}\frac{m-|e_i|}{m} \qquad (16)$$

where the Q function is given by (6), the *minimum distance* $d_{\min} = \min|d|$ (the minimum is taken over the set of all error events), and $\Lambda_{d_{\min}}$ is the set of error events which achieve the minimum distance. As error event λ is an element of this set, $w(\lambda)$, is the number of symbol errors entailed by an error event λ, and n is the duration of an error event λ (the assumption is made that all error events start at time 0).

The analysis of performance of DDFSE is also based on estimating probabilities of error events. Error events for DDFSE are defined in the same way as for the Viterbi algorithm, except channel states $s(D)$ are replaced by states $u(D)$ of the DDFSE trellis (11a). Thus, an error event occurs between times k_1 and k_2 if the elements of the state sequence u_k and its estimate $\hat{u}_k$ agree at times k_1 and k_2, but disagree at each time k for $k_1 < k < k_2$. Because of feedback, the path leading to the starting state u_{k_1} affects the metrics of the sequences associated with the error event and an upper bound on the probability of occurrence of the error event. Thus, as opposed to the Viterbi algorithm, error propagation affects performance of the DDFSE algorithm.

First consider the following upper bound on the probability that a particular error event occurs between times k_1 and k_2 given that no errors occur for η steps preceding the time k_1, i.e.,

$$(x_{k_1-1}, x_{k_1-2}, \cdots, x_{k_1-\eta})=(\hat{x}_{k_1-1}, \hat{x}_{k_1-2}, \cdots, \hat{x}_{k_1-\eta}). \qquad (17)$$

Note that the first μ symbols in (17) always agree since $u_{k_1} = \hat{u}_{k_1}$. For the error event to occur, the following inequality must hold

$$\sum_{k=k_1}^{k_2-1}(y_k-r_k)^2>\sum_{k=k_1}^{k_2-1}(y_k-\hat{r}_k)^2 \qquad (18)$$

where y_k is the received sequence (1), $y_k = t_k + z_k$. The estimates of the signal t_k (15) are

$$r_k=\sum_{i=0}^{\mu}f_i x_{k-i}+\tilde{w}_{k-\mu-1}(u_k),\quad \hat{r}_k=\sum_{i=0}^{\mu}f_i\hat{x}_{k-i}+\tilde{w}_{k-\mu-1}(\hat{u}_k)$$

where $\tilde{w}_{k-\mu-1}(\,)$ are the estimates of the residual interference $w_{k-\mu-1} = \Sigma_{i=\mu+1}^{\eta} f_i x_{k-i}$ derived from the paths leading to u_k and $\hat{u}_k$. Note that under the assumption (17) that no errors occur for η steps before k_1, $\tilde{w}_{k-\mu-1}(u_k) = w_{k-\mu-1}$ for $k_1 \leq k < k_2$, and so $r_k = t_k$. In general, this will not be true. Define the squared distance of the error event as $d^2 = \Sigma_{k=k_1}^{k_2-1} d_k^2$ where the signal error sequence

$$d_k=r_k-\hat{r}_k=\sum_{i=0}^{\min(k-k_1,\eta)} f_i e_{k-i},\quad k_1\leq k<k_2. \qquad (19)$$

The upper limit on the summation in (19) indicates that d_k does not depend on the error sequence prior to the time $k = k_1$. Under the assumption that the path leading to the state u_{k_1} is correct, $e_k = 0$ for $k_1 - \eta \leq k \leq k_1$, and the upper limit could be chosen simply as η without affecting the result. However, in the form (19), the definition of d_k will apply to more general treatment later in this section.

From (17), $t_k = r_k$ and $t_k - \hat{r}_k = d_k$. Thus, from (18), the conditional probability of this error event is upper bounded by [16]

$$\prod_{k=k_1}^{k_2-1}\frac{m-|e_i|}{m}\,Q\left[\frac{|d|}{2\sqrt{N_0}}\right]. \qquad (20)$$

Now consider the upper bound on the conditional probability of the same error event given an arbitrary path leading to the state u_{k_1}. Then $e_{k_1-1} = \cdots = e_{k_1-\mu} = 0$, but $e_{k_1-(\mu+1)}, \cdots, e_{k_1-\eta}$ are not necessarily zero. Define the residual signal

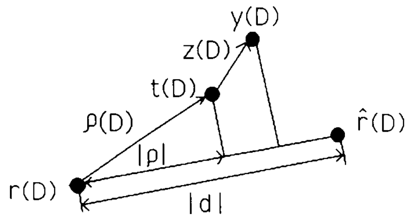


Fig. 4. Decision process in the signal space.

error sequence

$$\rho_k = \begin{cases} \sum_{i=k-k_1+1}^{\eta} f_i e_{k-i} & \text{if } k \le k_1 - 1 + \eta \\ 0 & \text{otherwise} \end{cases}. \tag{21}$$

Then $t_k - r_k = \rho_k$, $t_k - \hat{r}_k = d_k + \rho_k$ where d_k is given by (19). An error event occurs if the inequality (18) holds. It follows [16] that the conditional probability that this error event occurs given the path leading to the state u_{k_1} is upper bounded by

$$\prod_{k=k_1}^{k_2-1} \frac{m - |e_i|}{m} Q\left[\frac{\frac{|d|}{2} + \rho}{\sqrt{N_0}}\right], \tag{22}$$

$$\rho = \sum_{k=k_1}^{k_2-1} d_k \rho_k / |d| \,. \tag{23}$$

Note that the error sequences d_k and ρ_k depend on the input errors following the start of the error event k_1 and preceding it, respectively. The parameter ρ can be interpreted as a projection of the residual signal error vector with coordinates ρ_k on the signal error vector with coordinates d_k in the $(k_2 - k_1)$-dimensional signal space [8] (Fig. 4).

Expressions (20) and (22) are used to determine an upper bound on the probability that an error event λ occurs between times k_1 and k_2 [16]. An upper bound on the probability of occurrence of the error event λ is given by

$$\sum_{\hat{x}_k} \prod_{k=k_1}^{k_2-1} \frac{m - |e_i|}{m} Q\left[\frac{\frac{|d|}{2} + \rho_{\hat{x}_k}}{\sqrt{N_0}}\right] P_{\hat{x}_k}, \tag{24}$$

where $P_{\hat{x}_k}$ is the probability that $\hat{x}_k$ is the estimated sequence for $0 \le k < k_1$, the summation is over all possible sequences sequences $\hat{x}_k$, d is the distance of λ (19), and $\rho_{\hat{x}_k}$ is the projection determined for λ by $\hat{x}_k$ (23). The probability $P_{\hat{x}_k}$ depends on error events associated with $\hat{x}_k$ and can be computed by the chain rule [16]. The probability that an error event occurs at some time k and the symbol error probability can now by upper bounded by taking the union bound over the set of error events in the same way as in the Viterbi algorithm [5]. By considering the most significant contribution of error events preceded by a sequence of $\eta - \mu$ error-free steps (17), we obtain a term in an upper bound on the symbol error probability

$$Q\left[\frac{1}{2}\frac{|d^{\mu}_{\min}|}{\sqrt{N_0}}\right] \sum_{\lambda \in \Lambda_{d^{\mu}_{\min}}} w(\lambda) \prod_{k=0}^{n-1} \frac{m - |e_i|}{m} \tag{25}$$

where $|d^{\mu}_{\min}|$ is the minimum distance $|d|$ (19) achieved by an error event of the DDFSE trellis (when memory constraint is μ), and $\Lambda_{d^{\mu}_{\min}}$ is the set of error events which achieve the minimum distance. The term (25) can be thought of as an upper bound on the symbol error probability in the absence of decision errors preceding the start of an error event. In many cases, it dominates an upper bound [16].

Note that in (25) we made the assumption that all error events start at time 0. With this assumption, the distance of the error event can be computed as

$$\|[f(D)e(D)]_{m+\mu+1}\| = \|[f(D)e(D)]_n\| \tag{26}$$

where $e(D)$ is the error sequence associated with the error event of duration n, $e(D) = e_0 + e_1 D + \cdots + e_{n-1}D^{n-1}$, $m = \deg(e(D))$, $[a(D)]_p = a(D) \bmod D^p = a_0 + a_1 D + \cdots + a_{p-1}D^{p-1}$, and the squared norm of $a(D)$, $\|a(D)\|^2 = \Sigma_i a_i^2$. Thus, for $\mu = 0$ (for zero-forcing, DF detector), $|d^0_{\min}| = f_0$, and for $\mu = \eta < \infty$, $|d^{\eta}_{\min}|$ is the minimum distance of the Viterbi algorithm. An example of the above analysis for an FIR channel is given in the next section.

Similar estimates can be obtained for infinite impulse response channels (IIR) when we observe that the dependence between two error events is negligible if they are separated by a long sequence of error-free symbols. The term (25) also give an upper bound when decision errors preceding the start of an error event are ignored. Two IIR channels are also studied in the next section.

For i.i.d., equiprobable and binary input sequences, an upper bound on the symbol error probability of DDFSE (25) in the absence of decision errors preceding an error event (i.e., ignoring the effect of error propagation) can be directly compared to an upper bound on the symbol error rate of the Viterbi algorithm [16] (if η is finite) and to the symbol error rate of the zero-forcing decision-feedback equalizer in the absence of previous decision errors (5). All three estimates are expressed in terms of the Q-function (5) of the square root of the effective SNR given by

$$\sqrt{(\mathrm{SNR_{eff}})} = |d^{\mu}_{\min}|/(2\sqrt{N_0}) \tag{27}$$

where the values of $d^{\mu}_{\min}$ are given by the minimum distances of error events for the DDFSE and the Viterbi algorithm (26) (denote the corresponding $\mathrm{SNR_{eff}}$ as $\mathrm{SNR_{DDFSE}}$ and $\mathrm{SNR_{VA}}$), and by $|f_0|$ for the zero-forcing DF detector (denote its $\mathrm{SNR_{eff}}$ as $\mathrm{SNR_{DF}}$). Note that $\mathrm{SNR_{DF}} \le \mathrm{SNR_{DDFSE}} \le \mathrm{SNR_{VA}}$ where the first equality holds when $\mu = 0$ and the second equality holds when $\mu = \eta$. When η is infinite, the Viterbi algorithm cannot be implemented. In this case, it is still possible to find a lower bound on the performance of any detector which is proportional to the Q-function with an argument of the form (27), [15]. As μ grows to infinity, the value of $\mathrm{SNR_{DDFSE}}$ for such channels approaches to the value of $\mathrm{SNR_{eff}}$ in the lower bound (see the following section for an example). Note that the effect of error propagation is less severe for DDFSE then for the zero-forcing decision-feedback equalizer, since the decision feedback is delayed in DDFSE. Therefore, an estimate of symbol error rate computed ignoring the effect of error propagation (25) is closer to the actual error rate for DDFSE then for the zero-forcing decision-feedback detector.

B. Examples—Simulations Results and Comparison to Analysis

Example 1) Continued: Consider the specific FIR channel $f(D) = A(1 - 1.5D + 0.5D^2)$ and an i.i.d. data sequence with a binary modulation set $X = \{0, 1\}$.

First, observe that for $\mu = 0$ (the zero-forcing, DF detector), $(d^0_{\min})^2 = A^2$, for $\mu = 1$, $(d^1_{\min})^2 = A^2 9/4$, and for $\mu = 2$ (the Viterbi algorithm), $(d^2_{\min})^2 = A^2 5/2$. Consider the case $\mu = 1$. Assume that an error event starts at time zero. We find that for the input error sequences

$$e(D) = \pm(1 + D \cdots + D^n), \quad n \ge 1 \tag{28}$$

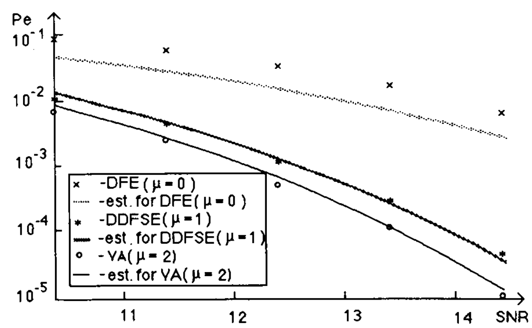


Fig. 5. Example 1. $A(1 - 1.5D + 0.5D^2)$ channel. SNR $= 7A^2/8N_0$.

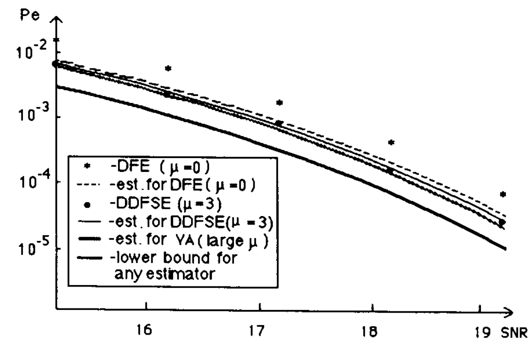


Fig. 6. Example 2. One pole channel. $A/(1 - 0.9D)$. SNR $= A^2/4N_0(1 - 0.9^2)$.

the squared distance is

$$d^2 = A^2 9/4. \tag{29}$$

By computing the distance of any other error event [16], it can be shown that (29) is the minimum squared distance.

An upper bound on the symbol error probability for $\mu = 1$ thus contains the term

$$\sum_{n=1}^{\infty} 2(n+1)\frac{1}{2^{n+1}} Q\left[\frac{3A}{4\sqrt{N_0}}\right] = 3Q\left[\frac{3A}{4\sqrt{N_0}}\right] \tag{30}$$

which is due to error events preceded by an error-free step. An upper bound on the bit error rate of the Viterbi algorithm, $\mu = 2$, is given by

$$3Q\left[\frac{A\sqrt{10}}{4\sqrt{N_0}}\right]. \tag{31}$$

An estimate of the bit error rate of the zero-forcing decision feedback equalizer, $\mu = 0$, made with the assumption that all previous decisions are correct is

$$Q(A/(2\sqrt{N_0})). \tag{32}$$

More complete performance analysis of the DDFSE involves computing joint probabilities of error events (24). It was carried out in [16]. The resulting upper bound was close to an estimate (30).

The estimates (30), (31), and (32) are plotted along with simulation results in the Fig. 5. The SNR denotes the signal-to-noise ratio (in db) for the output sequence given by $\Sigma_{i=0}^{\infty} f_i^2/4N_0 = 7A^2/8N_0$. Note that the bit error rates of DDFSE and the Viterbi algorithm are close and differ significantly from the bit error rate of the zero-forcing decision-feedback equalizer.

Example 2) Continued: Consider the analysis of the performance of DDFSE for the one-pole channel (13) $f(D) = A/(1 - \alpha D)$, where $\alpha = 0.9$ with binary inputs $X = \{0, 1\}$. A lower bound on the symbol error probability achieved by any estimator for this channel is given by [15]

$$\frac{1}{2} Q\left[\frac{A}{\sqrt{2N_0(1+\alpha)}}\right]. \tag{33}$$

The error rate of DDFSE is estimated by considering finite length error events. By inspection, we determine that the input error sequences $e(D) = \pm(1 - D)$ cause error events with minimum squared distance $|d_{\min}^{\mu}|^2 = A^2(1 + (1 - \alpha)(1 - \alpha^{2\mu+2})/(1 + \alpha))$. An upper bound on the bit error rate of the DDFSE for high signal-to noise ratio thus contains the term due to error events preceded by a long sequence of error-free bits

$$Q\left[\frac{A}{2}\left[\frac{1+\frac{1-\alpha}{1+\alpha}(1-\alpha^{2\mu+2})}{N_0}\right]^{1/2}\right].$$

For large μ this estimate approaches an upper bound on the performance of the Viterbi algorithm for the channel with

$$f(D) = A\sum_{i=0}^{\mu} \alpha^i D^i \tag{34}$$

which agrees in the limit with a lower bound (33) up to a constant factor. This factor usually appears when lower and upper bounds on symbol error rate of the Viterbi algorithm are computed [5]. It is possible to estimate other terms of the upper bound by taking into account joint probabilities of error events for this channel, but the effect of this analysis is not significant.

The bit error rate of the zero-forcing decision-feedback equalizer ($\mu = 0$) computed with the assumption that there is no previous decision errors is given by

$$Q(A/(2\sqrt{N_0})). \tag{35}$$

These bounds and simulation results are plotted in Fig. 6 against the output SNR $= A^2/(4N_0(1 - \alpha^2))$ for the channel with $\alpha = 0.9$. The estimate (35) differs significantly from the performance of the zero-forcing decision-feedback equalizer because of error propagation, but an estimate of the bit error rate for the DDFSE is close to the true bit error rate. The figure contains an upper bound on the bit error rate of the Viterbi algorithm for large μ for the channel with response (34). The bit error rate of the DDFSE approaches this upper bound as μ increases.

Example 3) Continued: For the one-zero, one-pole channel with response (14) $f(D) = A(1 - D)/(1 - \alpha D)$ a lower bound on bit error rate of any estimator is given by $Q(A/\sqrt{2N_0(1+\alpha)})$. The minimum distance is achieved by $e(D) = \pm 1$. An upper bound on the performance of DDFSE is

$$Q\left[\frac{A}{2}\left[\frac{1+\frac{1-\alpha}{1+\alpha}(1-\alpha^{2\mu})}{N_0}\right]^{1/2}\right].$$

The estimate of the performance of the zero-forcing decision feedback equalizer is the same as for the one-pole channel (35). The bounds and simulation results are plotted in the Fig. 7 against the output SNR $= A^2/(2N_0(1 + \alpha))$ for $\alpha = 0.6$.

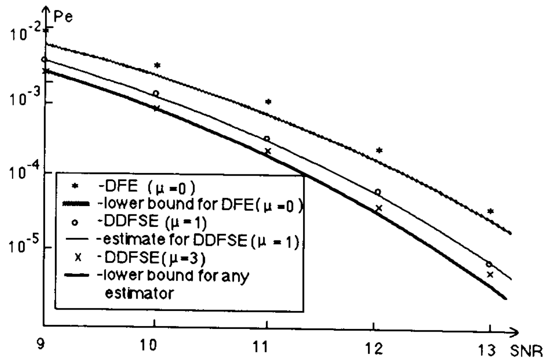


Fig. 7. Example 3. One zero, one pole channel. $A(1 - D)/(1 - 0.6D)$. SNR $= A^2/2N_0(1 + 0.6)$.

VI. DDFSE for Trellis Coded Modulation Systems

The DDFSE algorithm is applicable to a more general class of channels that those described by the discrete time model (1). In [1], the algorithm is presented for stationary Markov channels. The updated version appears in [16]. We consider trellis coded QPSK on a channel with intersymbol interference as one example.

In trellis coded modulation systems, the set of allowable data sequences is a proper subset of the set of sequences over the modulation set X, a finite subset of complex numbers [12]. Such a code is typically described by an (n, k) binary convolutional code (BCC) and an invertible mapping of binary n tuples onto the modulation set $q:\{0, 1\}^n \rightarrow X$ where $|X| = 2^n$. At time j, a random binary k tuple $m_j = (m_j^1, m_j^2, \cdots, m_j^k)$ is accepted by the BCC encoder. The corresponding encoder output $c_j = (c_j^1, c_j^2, \cdots, c_j^n)$, a binary n tuple, is then applied to the modulator resulting in the channel input $x_j = q(c)$.

The channel input is a function of the current input and past inputs $x_j = x(m_j^1, m_{j-1}^1, \cdots, m_{j-\nu_1}^1, \cdots, m_j^k, m_{j-1}^k, \cdots, m_{j-\nu_k}^k)$ where ν_i, $1 \leq i \leq k$, is called the memory length of the ith input. The Viterbi algorithm requires 2^ν states to decode such a code over a memoryless channel where the total memory [14] $\nu = \nu_1 + \cdots + \nu_k$.

If the trellis code is used over an ISI channel, then the number of states of the Viterbi algorithm grows (it can be infinite). The output of the channel is $y(D) = f(D)x(D) + z(D)$ where $z(D)$ is zero mean complex white Gaussian noise (i.e., its real and imaginary components are independent white Gaussian processes with mean zero and variance N_0), and $f(D)$ is complex and causal and satisfies assumptions of Section II. For the DDFSE algorithm, write the channel as in (8), then note that the term $f_\mu(D)x(D)$ can be computed by setting $u_k = (m_{j-1}^1, \cdots, m_{j-\nu_1-\mu}^1, \cdots, m_{j-1}^k, \cdots, m_{j-\nu_k-\mu}^k)$. This corresponds to a trellis with $2^{\nu+k\mu}$ states. The definition of the partial state v_j is given by (11b) and the partial state estimator, $\hat{v}(u_k, v_k)$ is determined from (10) and (11). The branch metric for DDFSE is $|y_k - t_k|^2$ where $t_k = \Sigma_{i=0}^{\mu} f_i x_{k-i} + \tilde{w}_{k-\mu-1}$, and $|x|$ denotes the amplitude of a complex number x. Note that the algorithm requires storage and computations in the field complex numbers. If $\mu = 0$, the trellis of the DDFSE has 2^ν states and coincides with the original trellis of the BCC. Note that in this case, if $\nu_i = 0$ for some i, the trellis of the DDFSE has multiple edges. The branch metric in this case is a function of the input m_k^i, which can not be determined solely from the state pair (s_{k-1}, s_k).

Examples

The following code and simple one-zero and one-pole channels illustrate issues involved in the implementation of the DDFSE algorithm in trellis coded, ISI systems.

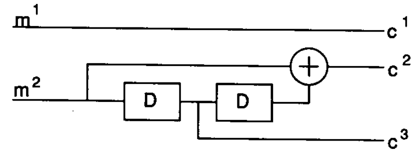


Fig. 8. The (3,2) BCC.

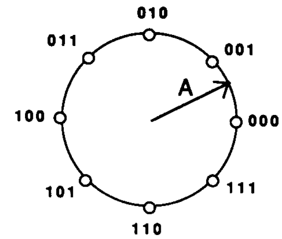


Fig. 9. The mapping from the outputs of the BCC to the modulation set.

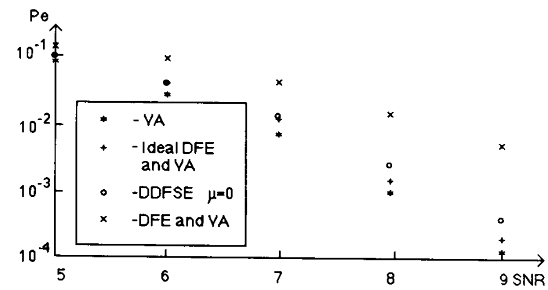


Fig. 10. Example 4. Rate 2/3, 8-PSK system followed by a channel with response $1 - 0.5D$. SNR $= A^2/N_0$.

Consider the trellis code based on the (3, 2) BCC [12], [13] encoder described by the circuit diagram given in Fig. 8. The modulation set, $X = \{Ae^{i\pi k/4}, 0 \leq k < 8\}$, of the trellis encoder lies on a circle of radius A in the complex plane (this is called the 8-PSK (phase-shift keyed) set). The mapping q of binary 3-tuples c is depicted in Fig. 9 and given by

$$x_j = x(m_j^1, m_j^2, m_{j-1}^2, m_{j-2}^2) = q(c_j) = Ae^{i(\pi/4)(4c_j^1 + 2c_j^2 + c_j^3)} \tag{36}$$

where $c_j^1 = m_j^1$, $c_j^2 = (m_j^2 + m_{j-2}^2)$, $c_j^3 = m_{j-1}^2$ (the last addition is modulo 2). The total memory of this trellis code is $\nu = 2(\nu_1 = 0, \nu_2 = 2)$.

A. Example 4

For the first trellis code example, suppose the ISI channel is given by $f(D) = 1 - \rho D$. Then the output of the channel $y_j = x_j - \rho x_{j-1} + z_j$. The dependence of x_j and x_{j-1} on the input is given by (36). The cascade of the encoder and the channel is described by a finite state machine with state $s_j = (m_{j-1}^1, m_{j-1}^2, m_{j-2}^2, m_{j-3}^2)$. This leads to a requirement of 16 states for the Viterbi algorithm. Consider DDFSE based on a trellis with 4 states ($\mu = 0$). Take $f(D) = f_0 + Df^+$ where $f_0 = 1, f^+ = -\rho$. Let $w(D) = f^+x(D) = -\rho x(D)$. Then $y(D) = x(D) + Dw(D) + z(D)$. This representation allows a decomposition of the state $s_j = (m_{j-1}^2, m_{j-2}^2, x_{j-1})$ into $u_j = (m_{j-1}^2, m_{j-2}^2)$ and $v_j = x_{j-1}$. The branch metric is $|y_j - t_j|^2$ where $t_j = x_j + \tilde{w}_{j-1}$, and from (10) and (11), the partial state estimator is given by $\hat{v}(u_j, m_j^1, m_j^2) = x_j(x_j = x(m_j^1, m_j^2, u_j)$ is determined by (36), and $\tilde{w}_{j-1} = -\rho v(u_j))$.

Fig. 10 shows results of computer simulations for this channel ($\rho = 1/2$). It gives error rates of the DDFSE and the

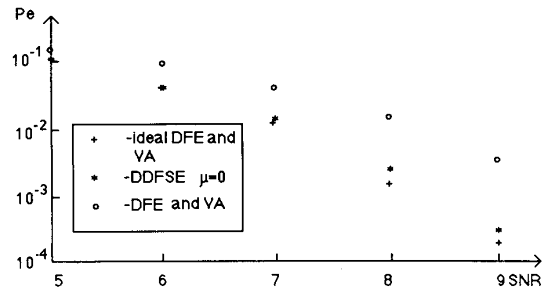


Fig. 11. Example 5. Rate 2/3, 8-PSK system followed by a channel with response $1/(1 - 0.5D)$. $SNR = A^2/N_0$.

Viterbi algorithm along with two other detection schemes involving decision feedback equalization (DFE). The SNR is defined as the ratio A^2/N_0 (db). The first scheme is a combination of a decision feedback equalizer with Viterbi decoding with 4 states. The DFE is applied to the output y_k and produces a decision $\hat{x}_k$ using previous decision $\hat{x}_{k-1}$. It has a forward filter $w(D) = 1$ and a feedback filter $b(D) = -\rho D$. The quantizer produces the decision $\hat{x}_k$ in X which is the closest to the estimate $\tilde{x}_k$ (on the complex plane). The estimate $\tilde{x}_k$ at the input to the quantizer is applied to the Viterbi algorithm which operates in the same way as for the memoryless channel (in the figure, this scheme is called "DFE and VA"). The second scheme is the same as the first, except decisions produced by the DFE are correct ("ideal DFE and VA"). The ideal DFE produces the same output as a memoryless channel with $f(D) = 1$, and thus has the same error rate. Note that performances of DDFSE and this scheme are close, and their trellises look the same.

B. Example 5

The second example corresponds to the same trellis code as in the previous section and to a channel model with the transfer function $f(D) = 1/(1 - \alpha D)$ where $0 < \alpha < 1$. The DDFSE algorithm with a memory constraint μ is described in the Section IV-C (in Example 2, set $A = 1$). For this channel, the state is $s_j = (m^1_{j-1}, \cdots, m^1_{j-\mu}, m^2_{j-1}, \cdots, m^2_{j-\mu-2}, w_{j-\mu-1})$. Thus, $u_j = (m^1_{j-1}, \cdots, m^1_{j-\mu}, m^2_{j-1}, \cdots, m^2_{j-\mu-2})$ and $v_j = w_{j-\mu-1}$. The partial state estimator is given by $\hat{v}(u_k, v_k) = \alpha^{\mu+1}x_{k-\mu} + \alpha v_k$. The DDFSE algorithm is based on trellis with $2^{2+2\mu}$ states.

The performance of DDFSE for $\mu = 0$, the DF equalizer (DFE) and ideal DFE (both combined with the Viterbi algorithm for the memoryless channel) for $\alpha = 1/2$ are plotted in the Fig. 11 against the SNR $= A^2/N_0$ (db). Again, note that the performance of ideal DFE (which is the same as for the trellis code on the memoryless channel) is close to the performance of DDFSE.

VII. Summary and Conclusions

We described the delayed decision-feedback sequence estimation (DDFSE) algorithm. The algorithm combines structures of the reduced-state Viterbi algorithm and a decision-feedback detector and provides a tradeoff between complexity and performance. We studied the application of the algorithm to the detection of signals transmitted over finite and infinite response intersymbol interference (ISI) channels. The performance of DDFE was estimated analytically and using computer simulations. The error rates of the algorithms for several channels were compared for various complexities. The performance of the DDFSE algorithm was shown to converge rapidly to the performance of the maximum-likelihood sequence estimator as its complexity grew.

We also considered the application of DDFSE to detection of coded signals over ISI channels. Computer simulations for trellis-coded QPSK system and two different channels showed that performances of the low complexity DDFSE algorithm and the ideal decision-feedback detector were close and signficantly exceeded the performance of the standard decision-feedback detector.

Acknowledgment

The authors would like to acknowledge Dr. J. W. Bergmans of Philips Research Laboratories for many helpful comments and suggestions.

References

[1] A. Duel and C. Heegard, "Delayed decision-feedback sequence estimation," presented at 23rd Annu. Allerton Conf. Commun., Contr. Comput., Oct. 2-4, 1985.
[2] ——, "Delayed decision-feedback sequence estimation for QAM and trellis coded systems," presented at 20th Conf. Inform. Sci. Syst., Princeton, NJ, Mar. 19-21, 1986.
[3] ——, "Signal processing in digital recording: The delayed decision-feedback sequence estimation algorithm," presented at SPIE's 30th Annu. Int. Tech. Symp. Optic. Optoelectron. Appl. Sci. Eng., San Diego, CA, Aug. 17-22, 1986.
[4] J. G. Proakis, *Digital Communications.* New York: McGraw-Hill, 1983.
[5] G. D. Forney, Jr., "Maximum-likelihood sequence estimation of digital sequences in the presence of intersymbol interference," *IEEE Trans. Inform. Theory,* vol. IT-18, pp. 363-378, May 1972.
[6] A. J. Viterbi and J. K. Omura, *Principles of Digital Communication and Coding.* New York: McGraw Hill, 1979.
[7] L. E. Franks, *Signal Theory.* Englewood Cliffs, NJ: Prentice-Hall, 1969.
[8] W. U. Lee and F. S. Hill, Jr., "A maximum likelihood sequence estimator with decision-feedback equalization," *IEEE Trans. Commun.,* vol. COM-25, pp. 971-980, Sept. 1977.
[9] V. M. Eyuboglu and S. U. Quereshi, "Reduced-state sequence estimation with decision feedback and set partitioning," presented at GLOBECOM '86, Houston, TX.
[10] ——, "Reduced-state sequence estimation with set partitioning and decision feedback," *IEEE Trans. Commun.,* vol. 36, pp. 13-20, Jan. 1988.
[11] G. Ungerboeck, "Adaptive maximum-likelihood receiver for carrier-modulated data-transmission systems," *IEEE Trans. Commun.,* vol. COM-22, pp. 624-636, May 1974.
[12] ——, "Channel coding with multilevel/phase signals," *IEEE Trans. Inform. Theory,* vol. IT-28, pp. 55-67, Jan. 1982.
[13] G. D. Forney, Jr., R. G. Gallager, G. R. Lang, F. M. Longstaff, and S. U. Quereshi, "Efficient modulation for band-limited channels," *IEEE J. Selected Areas Commun.,* vol. SAC-2, pp. 632-647, Sept. 1984.
[14] S. Lin and D. J. Costello, *Error Control Coding: Fundamentals and Applications.* Englewood Cliffs, NJ: Prentice-Hall, 1983.
[15] G. D. Forney, Jr., "Lower bounds on error probability in the presence of large intersymbol interference," *IEEE Trans. Commun.,* vol. COM-20, pp. 76-77, Feb. 1972.
[16] A. Duel-Hallen, "Detection for Channels with intersymbol interference," Ph.D. dissertation, Cornell Univ., Aug. 1987.
[17] R. Price, "Nonlinearly feedback-equalized PAM vs. capacity," in *Proc. 1972 IEEE Int. Conf. Commun.,* Philadelphia, PA, June 1972, pp. 22.12-22.17.
[18] C.-F. Lin and J. B. Anderson, "*M*-algorithm decoding of channel convolutional codes," presented at 20th Annu. Conf. Inform. Sci. Syst., Princeton, NJ, Mar. 19-21, 1986.
[19] G. J. Foschini, "A reduced state variant of maximum likelihood sequence detection attaining optimum performance for high signal-to-noise ratios," *IEEE Trans. Inform. Theory,* vol. IT-23, pp. 605-609, Sept. 1977.
[20] K. Wesolowski, "An efficient DFE & ML suboptimum receiver for data transmission over dispersive channels using two-dimensional signal constellations," presented at 1985 IEEE Int. Symp. Inform. Theory, Brighton, England, June 23-28, 1985.
[21] F. L. Vermeulen and M. E. Hellman, "Reduced state Viterbi decoders for channels with intersymbol interference," in *Proc. 1974 IEEE Int. Conf. Commun.,* pp. 37B-1-37B-4.

[22] P. J. McLane, "A residual interference error bound for truncated-state Viterbi detectors," *IEEE Trans. Inform. Theory,* vol. IT-26, pp. 549-553, Sept. 1980.

[23] H. Kobayashi, "Application of probabilistic decoding to digital magnetic recording systems," *IBM J. Res. Develop.,* pp. 64-74, Jan. 1971.

[24] S. U. H. Quereshi and E. E. Newhall, "An adaptive receiver for data transmission over time-dispersive channels," *IEEE Trans. Inform. Theory,* vol. IT-19, pp. 448-457, July 1973.

[25] ——, "An adaptive decision-feedback receiver using maximum-likelihood sequence estimation," presented at 1973 Int. Commun. Conf., Seattle, WA.

[26] A. V. Oppenheim and R. W. Schafer, *Digital Signal Processing.* Englewood Cliffs, NJ: Prentice-Hall, 1975.

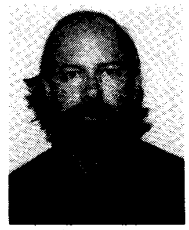
★

Alexandra Duel-Hallen (S'85-M'87) was born on December 28, 1959, in Kharkov, U.S.S.R. She received the B.S. degree in mathematics from Case Western Reserve University in 1982, the M.S. degree in computer, information and control engineering from the University of Michigan, Ann Arbor, in 1983 and the Ph.D. degree in electrical engineering from Cornell University in 1987.

She worked for AT&T Bell Laboratories in Columbus, OH, in the Summer of 1982 and participated in their One Year on Campus Program at the University of Michigan in the 1982-1983 academic year. She has been a Visiting Assistant Professor at Cornell University since 1987. Her research interests are in the areas of digital communications and information theory. In 1985 she was awarded an AT&T Bell Laboratories Ph.D. Scholarship.

Dr. Duel-Hallen is a member of IEEE Information Theory Group and Communication Society.

★



Chris Heegard (S'75-M'81) was born on October 4, 1953, in Pasadena, CA. He received the B.S. (1975) and M.S. (1976) degrees in electrical and computer engineering, from the University of Massachusetts, Amherst, and received the Ph.D. degree (1981) in electrical engineering from Stanford University.

For two years, 1976-1978, he was an Engineer at Linkabit Corporation, San Diego, CA. While there, he participated in the development of a digital, packet satellite modem as well as several sequential decoders. Since 1981, he has been on the faculty of the School of Electrical Engineering at Cornell University, Ithaca, NY, where he is currently an Associate Professor. During the 1987-1988 academic year, he was a visitor at the IBM Almaden Research Center, San Jose, CA; during the Spring of 1988 he was also a Visiting Associate at the California Institute of Technology, Pasadena, CA. He is also an active participant of the technical consulting community. His research interests are in the areas of communications, information and coding theory, and in the area of digital signal processing. He enjoys the application of these theories to communications and computer systems. In particular, his interests are in the areas of digital recording (magnetic and optical), reliability random access memories, local area networks, data compression, and digital audio.

Dr. Heegard is an active member of the IEEE publishing papers in the IEEE TRANSACTIONS OF INFORMATION THEORY, ON COMMUNICATIONS, and ON COMPUTERS. As a member of the Information Theory Group, he has participated in the organization of several meetings, was the Group's secretary for two years (1983-1985) and was elected to and actively served on the Board of Governors (1985-1988). He is currently organizing the 1989 Information Theory Workshop to be held at Cornell University, Ithaca, NY, June 26-29.