Making MLSD Decisions by Thresholding the Matched Filter Output

Per Ödling, *Associate Member, IEEE*, Håkan B. Eriksson, *Member, IEEE*, and Per Ola Börjesson, *Member, IEEE*

***Abstract*—This paper presents an iterative detector that, by thresholding the output of the matched filter, gives maximum-likelihood sequence detector (MLSD) decisions on binary, antipodally modulated symbols that have been corrupted by intersymbol interference and additive Gaussian noise. The detector will make decisions on *some*, but often not all, of the symbols in a transmitted sequence, and those decisions will be the *same* decisions as the MLSD would have made. The number of symbols that are detected is stochastic, varying from sequence to sequence. The basis of the detector is a bound on a crossterm of the quadratic form in the log-likelihood function for the transmitted sequence. The detector is simple in structure, consisting of a matched filter and two variable threshold values for each symbol.**

***Index Terms*—Digital communication, equalizers, intersymbol interference, iterative methods, receiver.**

I. INTRODUCTION

IN THIS paper, we present a simple detector of independent, antipodally modulated, binary symbols transmitted in blocks over a discrete-time additive Gaussian channel with intersymbol interference (ISI) [1], [2]. We show that by accepting that some symbols are left undetermined, it is possible to derive the *same* decisions on the remaining symbols as the maximum-likelihood sequence detector (MLSD) [3] would, and in a computationally efficient fashion. It is not known beforehand how many, or which, symbols in any particular block will be left undetermined. The proposed detection device is not intended as a stand-alone detector, but as an efficient means of extracting information in the form of MLSD decisions from the received data. It can, for instance, be used in a scenario as described by Fig. 1. Some of the symbols are detected as MLSD decisions by the proposed detector. These decisions are then, together with the received signal, fed to a complementary receiver that decides on the remaining symbols.

Paper approved by E. Panayirci, the Editor for Synchronization and Equalization of the IEEE Communications Society. Manuscript received January 15, 1998; revised March 10, 1999. This work was supported in part by the Telecommunications Program of the Swedish National Board for Industrial and Technical Development (NUTEK) and in part by the Austrian K-Plus Program. This paper was presented in part at the International Symposium on Information Theory (ISIT), Ulm Germany, 1997.

P. Ödling is with the Lund Institute of Technology, Department of Applied Electronics, SE-221 00 Lund, Sweden (e-mail: per.odling@tde.lth.se). He is also with the Telecommunications Research Center Vienna (FTW), A-1040 Vienna, Austria.

H. B. Eriksson is with Ericsson Mobile Communications AB, SE-221 83 Lund, Sweden (e-mail: hakan.eriksson@ecs.ericsson.se).

P. O. Börjesson is with the Lund Institute of Technology, Department of Applied Electronics, SE-221 00 Lund, Sweden (e-mail: per.ola.borjesson@tde.lth.se).

Publisher Item Identifier S 0090-6778(00)01580-4.

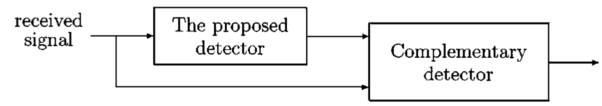

Fig. 1. Example of how the proposed detector can be used in a receiver.

The detector is simple in structure and consists of a matched filter and two variable thresholds for each symbol, where the thresholds are dependent on the received signal. In this paper, we assume that the transmitted symbols are binary, and consequently the output alphabet of the detector becomes ternary, "+1," "–1," and "unknown." The detector is derived by expanding, with respect to a single symbol, the quadratic form of the log-likelihood function for the entire transmitted sequence. This expansion leads to an expression where only one term depends on the symbol to be detected. This term is a product of that symbol and a difference between the output from a matched filter and a function of the other symbols, a function that can be upper and lower bounded by two thresholds that are independent of the undetermined symbols. The symbol is detected if the corresponding output from the matched filter is either above the upper threshold or below the lower threshold. As symbols are detected, new and tighter thresholds can be calculated and more symbols may be detected. All symbols that are detected by this procedure are detected with the MLSD criterion, because they maximize the log-likelihood function for the transmitted sequence.

Our detector offers an efficient way of obtaining MLSD decisions on some symbols. Perhaps the detector's most important use is as a computationally attractive means of obtaining MLSD decisions when the MLSD itself is infeasible, for example, when communicating over channels with long delay spreads. As the model we use to derive the detector is quite general, there are other areas besides the ISI problem where the detector is applicable, for instance, in cochannel-interference and multiple-access interference situations [4], [5].

The paper proceeds as follows. In Section II, a model for block transmission systems is described and some auxiliary definitions are given. The proposed detector is developed in Section III, and an attempt at geometrical visualization of the detector's operation is made in terms of decision regions. An algorithm for calculating the thresholds of the detector is given in Section III-B, together with an example. Section IV examines the behavior of the detector, especially its probability of detection. Section V exemplifies how it can be used in a system together with a complementary detector. Simulations illustrating performance improvements are included. Finally, in Section VI, a discussion of the results is found.

II. Block Transmission System Model

Consider the transmission of blocks of binary data, typically interspersed with symbols known to the receiver, through a channel with additive Gaussian noise and *known* ISI.[1] We represent the resulting transmission system with the matrix notation

$$\mathbf{y} = \mathbf{H}\mathbf{b} + \mathbf{n} \tag{1}$$

where the transmitted message is coded in $\mathbf{b} \in \{-1,+1\}^N, \mathbf{y} \in \mathbf{R}^{N+L-1}$ is a vector of channel observables, $\mathbf{H}$ is a real-valued, deterministic and known $(N+L-1)\times N$ matrix representing the ISI, and $\mathbf{n}$ is a jointly Gaussian zero-mean random vector with an $N(\mathbf{0},\mathbf{R_n})$ distribution [6], [7]. We assume the symbols in $\mathbf{b}$ to be independently, identically distributed with an equal probability of –1 and +1 occurring.

If the matrix $\mathbf{H}$ represents a linear, time-invariant, and causal ISI channel, the structure of $\mathbf{H}$ becomes

$$\mathbf{H} = \begin{pmatrix} h_0 & 0 & \cdots & 0 \\ h_1 & h_0 & \ddots & \vdots \\ \vdots & h_1 & \ddots & 0 \\ h_{L-1} & \vdots & \ddots & h_0 \\ 0 & h_{L-1} & \ddots & h_1 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \ldots & 0 & h_{L-1} \end{pmatrix} \tag{2}$$

where $[h_0, h_1, \cdots, h_{L-1}]$ is the impulse response of the system. If $h_0 \neq 0$ and $h_{L-1} \neq 0$, then L is the length of the system memory.

An MLSD is a processor that, given the received signal $\mathbf{y}$ and model (1), finds the sequence $\mathbf{b}$ that most probably was transmitted, consequently minimizing the probability of choosing the wrong sequence [8]. The MLSD finds the sequence that *minimizes* the negative of the log-likelihood function

$$\Lambda(\mathbf{b},\mathbf{y}) \triangleq \|\mathbf{y} - \mathbf{H}\mathbf{b}\|^2_{\mathbf{R_n}^{-1}} \tag{3}$$

where the norm $\|\mathbf{x}\|_{\mathbf{A}} \triangleq \sqrt{\mathbf{x}^T\mathbf{A}\mathbf{x}}$ [6]. In its trivial form, the MLSD performs an exhaustive search over $\{-1,+1\}^N$, and hence its computational complexity is proportional to 2^N. With the Viterbi algorithm [3], the computational complexity is proportional to $N2^L$, where if the channel impulse response is time-invariant, L is the length of the system memory as described by (2).

We continue with a number of additional definitions that we will find useful in what follows. Let us define the matrix

$$\mathbf{M} \triangleq \mathbf{H}^T\mathbf{R_n}^{-1}\mathbf{H} \tag{4}$$

which determines the metric used by the MLSD. The detector developed in the next section operates on the vector

$$\mathbf{z} \triangleq \mathbf{H}^T\mathbf{R_n}^{-1}\mathbf{y} = \mathbf{M}\mathbf{b} + \mathbf{H}^T\mathbf{R_n}^{-1}\mathbf{n} \tag{5}$$

[1]For simplicity, we present our results with binary pulse-amplitude modulation and real-valued channels. An extention to QPSK and complex-valued channels is straightforward.

which is the output from the matched filter for the discrete-time model (1), cf. [6], [7], [2]. We define the signal-to-noise ratio (SNR) in this block transmission system as

$$\text{SNR} \triangleq \frac{\operatorname{tr}\{\mathbf{M}\}}{N} \tag{6}$$

where $\operatorname{tr}\{\mathbf{M}\}$ denotes the trace of the matrix $\mathbf{M}$, i.e., the sum of the elements on the main diagonal. This block SNR measure is adequate for stationary channels where the entire block is subject to the same ISI and noise conditions. Other SNR measures are necessary to reflect variations of the ISI and noise characteristics between the symbols in a block.

We define the average probability that a symbol is left undetermined after passing through the detector as

$$P_u \triangleq \frac{1}{N}\sum_{k=1}^{N} P_{u,k} \tag{7}$$

where $P_{u,k}$ is the probability that the proposed detector leaves symbol number k undetermined.

III. The Detector

In this section, we develop a detector making MLSD decisions on individual symbols in a block, a detector that consists of a matched filter and a threshold device. We develop the detector in a step-by-step manner that we hope is intuitively appealing to the reader. An explicit algorithm is given in Section III-B.

Let $\hat{\mathbf{b}}^{\text{MLSD}}$ denote the output of an MLSD given the observation $\mathbf{y}$. Furthermore, let z_i, b_i, $\hat{b}_i^{\text{MLSD}}$, and $m_{i,j}$ denote the elements in $\mathbf{z}$, $\mathbf{b}$, $\hat{\mathbf{b}}^{\text{MLSD}}$, and $\mathbf{M}$, respectively, where i is the row index and j is the column index. Focusing on the detection of the kth symbol b_k, we rewrite in the appendix the negative of the log-likelihood function (3) and obtain

$$\Lambda(\mathbf{b},\mathbf{y}) = 2b_k(\Delta(\mathbf{b},k) - z_k) + \sum_{i\neq k, j\neq k} b_i m_{i,j} b_j - 2 \cdot \sum_{i\neq k} b_i z_i + m_{k,k} + \mathbf{y}^T\mathbf{R_n}^{-1}\mathbf{y} \tag{8}$$

where

$$\Delta(\mathbf{b},k) \triangleq \sum_{i\neq k} b_i m_{i,k}. \tag{9}$$

Observe that $\Delta(\mathbf{b},k)$, as given by (9), and all but the first term on the right-hand side of (8), are *independent* of b_k. Note also that

$$|\Delta(\mathbf{b},k)| \leq \sum_{i\neq k} |b_i m_{i,k}| = \sum_{i\neq k} |m_{i,k}| \tag{10}$$

and recall that z_k is the output from the matched filter. Now, we are ready to make the crucial observation: *if* z_k *is above the possible maximum of* $\Delta(\mathbf{b},k)$, i.e., *if* $z_k > \sum_{i\neq k}|m_{i,k}|$, *the function* $\Lambda(\mathbf{b},\mathbf{y})$ *in* (8) *is minimized by* $b_k = +1$, *independently of the status of the other symbols*. Hence, b_k takes the value +1 where the function $\Lambda(\mathbf{b},\mathbf{y})$ in (8) has its global minimum, and thus the corresponding output of the MLSD is positive, $\hat{b}_k^{\text{MLSD}} = +1$. Similarly, if $z_k < -\sum_{i\neq k}|m_{i,k}|$, then

$\hat{b}_k^{\text{MLSD}} = -1$. This is true for all positions k, and the tests can be done independently for all symbols. These observations are sufficient for attempting to detect some symbols in a block with the MLSD criterion using only a matched filter and a threshold device.

Assume now that some of the symbols in $\mathbf{b}$ have been detected as described. We proceed by minimizing the log-likelihood function (8) with respect to the remaining undetermined symbols. With the same technique as above, the sum $\Delta(\mathbf{b}, k)$ in (9) can now be bounded from above by

$$\Delta_k^+ \triangleq \sum_{\{i|i\neq k, b_i \text{ detected}\}} m_{i,k}\hat{b}_i^{\text{MLSD}} + \sum_{\{i|i\neq k, b_i \text{ not detected}\}} |m_{i,k}| \tag{11}$$

and from below by

$$\Delta_k^- \triangleq \sum_{\{i|i\neq k, b_i \text{ detected}\}} m_{i,k}\hat{b}_i^{\text{MLSD}} - \sum_{\{i|i\neq k, b_i \text{ not detected}\}} |m_{i,k}| \tag{12}$$

where the difference between (11) and (12) is the sign preceding the second sum. We will henceforth refer to Δ_k^+ and Δ_k^- as the upper and lower threshold for symbol k, respectively. When a symbol is detected, the distances between upper and lower thresholds in its vicinity decreases, as is determined by the matrix $\mathbf{M}$. Again, we can check whether the output of the matched filter $z_k > \Delta_k^+$, and if it is, we know that $\hat{b}_k^{\text{MLSD}} = +1$. Analogously, if $z_k < \Delta_k^-$, then $\hat{b}_k^{\text{MLSD}} = -1$. Thus, by comparing the output of the matched filter z_k with these new and tighter thresholds, additional symbols could possibly be detected. This procedure can be repeated until no more symbols are detected. The output of the detector then consists of a sequence of MLSD decisions possibly interspersed with undetermined symbols. Because both the MLSD and the proposed detector finds symbols that maximize the log-likelihood function (3), we conclude that their decisions on the symbols are identical.

Note that the above-presented procedure for detection is independent of a scaling of the noise covariance matrix $\mathbf{R_n}$. Hence, given the correlation of the noise, the detector is independent of the SNR, which also conforms with the MLSD.

We view the proposed detector as a matched filter followed by a threshold device, as illustrated in Fig. 2. As is evident from the above, the thresholds are dependent on the received signal and, thus, have to be recalculated for each block of data.

Viewing the detector in Fig. 2 in the framework of the sampled, whitened matched filter front-end as described by Forney [3], the digital whitening filter and the matched filter of the proposed receiver are each other's inverses. Thus, z_k would then be the output of the sampler.

A. Geometrical Interpretation

We would like to give an intuitive feeling for how a detector consisting of a linear filter and a threshold device can provide MLSD decisions. For this purpose, we compare the decision regions of the MLSD with the decision regions of the proposed detector. We make this comparison using the binary hypercube to describe the transmitted data, so for a moment we change signal space. Consider the detector's decision regions for the sufficient statistics [6]

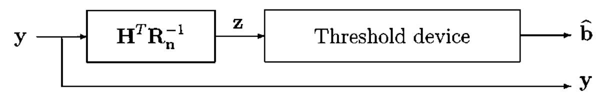

Fig. 2. Block description of the proposed detector.

$$\hat{\mathbf{b}}_u(\mathbf{y}) \triangleq \mathbf{M}^{-1}\mathbf{H}^T\mathbf{R_n}^{-1}\mathbf{y} = \mathbf{M}^{-1}\mathbf{z} \tag{13}$$

in $\mathbb{R}^N$, where $\mathbf{M}$ is given in (4).[2] Because both the MLSD and the proposed detector can be derived as operating on $\hat{\mathbf{b}}_u(\mathbf{y})$, they are completely described by their corresponding decision regions.

We plot decision regions for $\hat{\mathbf{b}}_u(\mathbf{y})$ when transmitting blocks of *two* symbols over a time-invariant channel with white stationary Gaussian noise $\mathbf{n}$ and with a channel impulse response $[h_0, h_1] = [1, 1/2]$, see (1) and (2). Fig. 3(a) contains the decision surface of the MLSD for the detection of the *first* symbol in the block. The four possible transmitted sequences are marked with circles. If $\hat{\mathbf{b}}_u \in \mathcal{D}_1^+$, then symbol one of $\hat{\mathbf{b}}^{\text{MLSD}}$ is equal to +1; otherwise, it is –1. In Fig. 3(b), the corresponding decision regions of the proposed detector are shown. If $\hat{\mathbf{b}}_u \in \mathcal{O}_1^+$ or $\hat{\mathbf{b}}_u \in \mathcal{O}_1^-$, the output for symbol one is equal to +1 and –1, respectively. If $\hat{\mathbf{b}}_u \in \mathcal{O}_1^0$, the detector is unable to make a decision. As in Fig. 3(a) and (b), the decision regions $\mathcal{O}_k^+$ and $\mathcal{O}_k^-$ of the proposed detector, will always be subsets of the corresponding decision regions of the MLSD $\mathcal{D}_k^+$ and $\mathcal{D}_k^-$. Consequently, whenever decisions are made, they are the same as the decisions made by the MLSD. Although the example does not cover higher dimensions or the iterative tightening of the thresholds, we hope that it illustrates how decisions are made by slicing the signal space into three parts.

B. Matlab™ Style Algorithm

In this section, we will give a formal description of the iterative algorithm that was derived in Section III. The algorithm is one example of a possible Matlab™ style algorithm, which stresses ease of understanding rather than computational efficiency.

Let $\hat{\mathbf{b}} = [\hat{b}_1, \cdots, \hat{b}_N]^T$ denote the output of the proposed detector. Furthermore, let Δ_k^+ and Δ_k^- denote the upper and the lower threshold, respectively. For convenience, let the output of the detector $\hat{\mathbf{b}}$ belong to $\{-1, 0, +1\}^N$, where a zero signifies that the detector does not make a decision on the corresponding bit.

[2]The matrix operation $\mathbf{M}^{-1}\mathbf{H}^T\mathbf{R_n}^{-1}$ is an instance of the Moore–Penrose pseudoinverse and can be seen as the *zero-forcing equalizer* for block transmission systems [6], [7]. This can be illustrated by inserting (1) in (13) giving $\hat{\mathbf{b}}_u(\mathbf{y}) = \mathbf{b} + \mathbf{M}^{-1}\mathbf{H}^T\mathbf{R_n}^{-1}\mathbf{n}$.

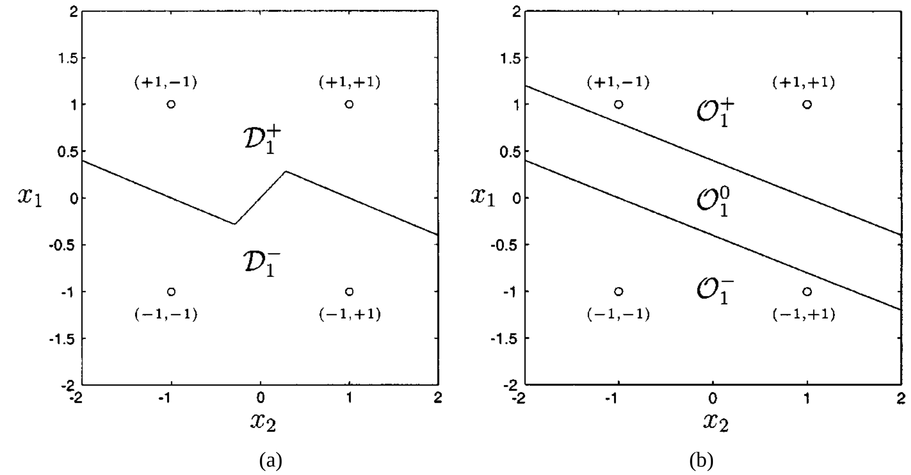

Fig. 3. Examples of two-dimensional decision regions for the statistics $\hat{\mathbf{b}}_u(\mathbf{y})$ and for the detection of the first symbol in a two-symbol block. (a) Decision regions of the MLSD and (b)the proposed detector, with the $[x_1, x_2]$ as coordinates in R^2.

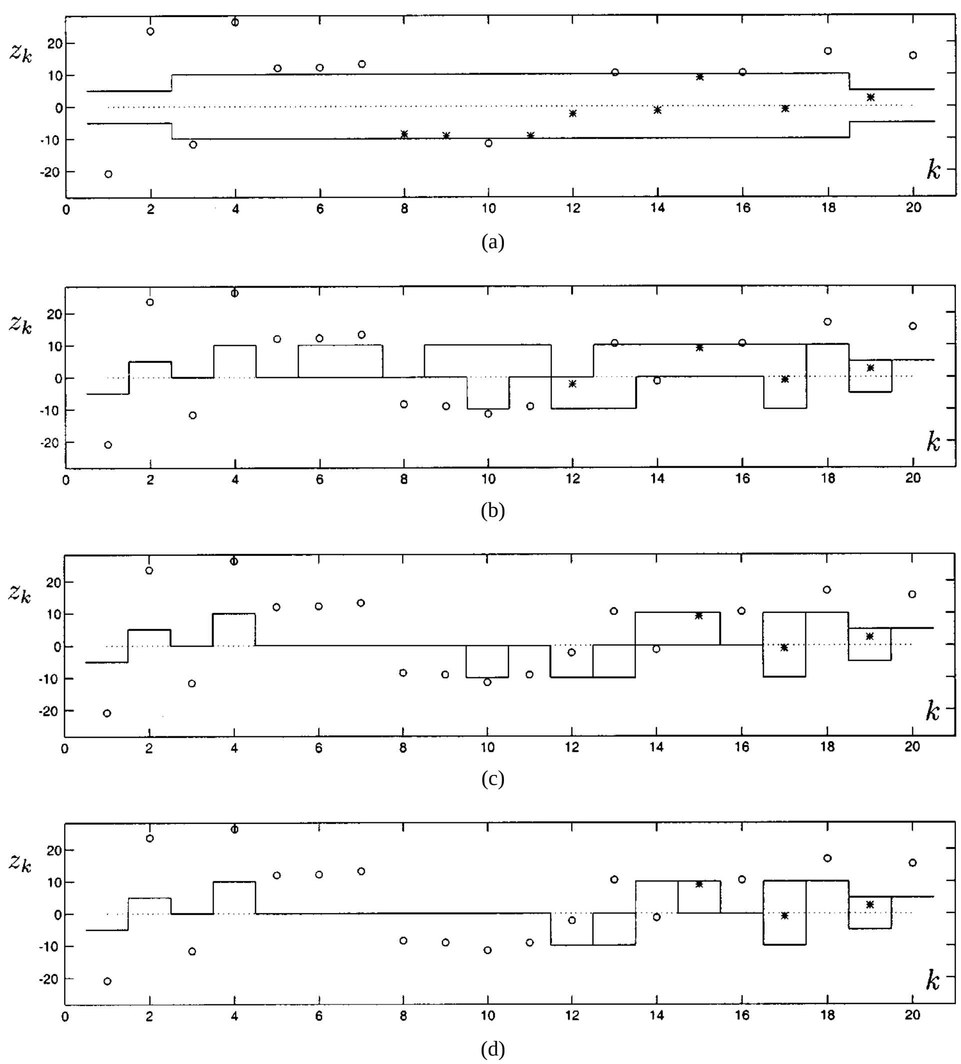

Fig. 4. The output of the matched filter plotted together with the thresholds of the iterative algorithm in an example where four iterations are needed to obtain the final thresholds. (a) First iteration, (b) second iteration, (c) third iteration, and (d) fourth iteration. Symbols that are detected are indicated with "o" and undetected symbols are with "*."

One algorithm describing the detector is as follows.

1) Initiate:

$$\hat{b}_k := 0, \quad \text{for all } k.$$

REPEAT

2) Store previously detected symbols:

$$a_k := \hat{b}_k, \quad \text{for all } k.$$

3) Calculate the upper threshold:

$$\Delta_k^+ := \sum_{\{i|i\neq k,\, \hat{b}_i\neq 0\}} m_{i,k}\hat{b}_i + \sum_{\{i|i\neq k,\, \hat{b}_i=0\}} |m_{i,k}|, \quad \text{for all } k.$$

4) Calculate the lower threshold:

$$\Delta_k^- := \sum_{\{i|i\neq k,\, \hat{b}_i\neq 0\}} m_{i,k}\hat{b}_i - \sum_{\{i|i\neq k,\, \hat{b}_i=0\}} |m_{i,k}|, \quad \text{for all } k.$$

5) Compare and detect:

$$\hat{b}_k := \begin{cases} +1, & z_k > \Delta_k^+ \\ -1, & z_k < \Delta_k^- \\ 0, & \text{otherwise} \end{cases} \qquad \text{for all } k.$$

UNTIL $\hat{b}_k = a_k$ for all k.

The algorithm will continue to iterate until none of the remaining symbols are detected or there are no more symbols to be detected. However, it is possible to perform a fixed number of iterations much lower than N without severe performance degradation, as indicated by simulations in Section IV.

C. Example of the Operation of the Detector

To illustrate how the algorithm presented above calculates the thresholds, we present an example where blocks of binary data of length $N = 20$ are transmitted over a time-invariant channel described by (1). The impulse response of the system, as in (2), is $[h_0, h_1, h_2] = [1, 0, 1]$, the noise $\mathbf{n}$ is white, stationary, and Gaussian with an SNR of 10 dB. In Fig. 4, an example of the iterations of the algorithm is given. The solid lines are the thresholds Δ_k^+ and Δ_k^-, while the stars and the circles are the output of the matched filter $\mathbf{z}$, with the circles indicating detected symbols, and the stars undetermined symbols. In the example, the upper and lower thresholds have merged for most of the symbols in the final iteration. The symbols at k equal to 15, 17, and 19 could not be detected. Note that symbol 12 was detected as positive although the output from the matched filter was negative.

IV. Simulation Examples—Properties

The potential of the proposed detector is dependent on its ability to make decisions. A natural approach would be to try to analytically determine the probability of decision $1 - P_u$. It seems, however, to be difficult to derive a closed-form expression, one obstacle being the stochastic behavior of the thresholds as they are dependent on the received signal $\mathbf{y}$. Here, simulations are used to reveal some of the important properties of the detector.

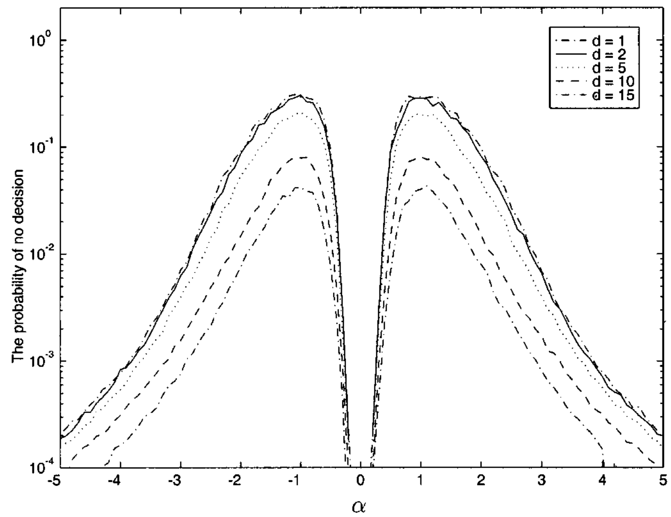

Fig. 5. Probability of a symbol being left undetermined, P_u, for the proposed detector plotted versus the strength of the second tap α, for a delay $d \in [1, 2, 5, 10, 15]$. SNR is 10 dB.

Simulations were carried out where the proposed detector was used in a two-path ISI environment described by (1) and (2) with

$$h_k = s(k) + \alpha s(k - d) \tag{14}$$

where $s(\cdot)$ is the raised-cosine function ([9], pp. 535–536) with rolloff β, and where the symbol length is normalized to one. The constant α is the real-valued amplitude and d is the delay of the second tap. Note that when the delay d is not an integer value, the pulse shaping may introduce substantial ISI. Throughout these simulations, the noise $\mathbf{n}$ is white, stationary, zero-mean, and Gaussian. If not otherwise stated, the block length $N = 20$.

In Fig. 5, estimates of the probability that a symbol is left undetermined, P_u in (7), are plotted versus the strength of the second tap α, for integer delays $d \in [1, 2, 5, 10, 15]$, with $\beta = 0.5$ and an SNR of 10 dB. The average probability, given a certain d, has global maxima at $\alpha = 1$ and $\alpha = -1$; it is low if $|\alpha|$ is small or large enough, and it is also nonincreasing with d. If $d \geq N$, the two received echoes are separable, effectively an ISI-free case, and consequently the detector is able to detect all symbols.

Fig. 6 shows the probability of decision $(1 - P_u)$, minimized by varying α in the interval $[0 \ldots 1]$, versus the delay d. Examples are given for $\beta = 0.2$, $\beta = 0.3$, and $\beta = 0.4$. The minimum probability of decision in Fig. 6 is about 20%, but it lies mostly between 30% and 80% for moderate delays d. One important factor for the probability of decision is the amount of ISI, or rather the autocorrelation of the channel impulse response. A channel with all channel taps positive gives a much worse probability of decision than the same channel would have done if the signs of the taps had varied randomly. The sum of the absolute values of the channel correlations determines the thresholds and, thus, the detection rate.

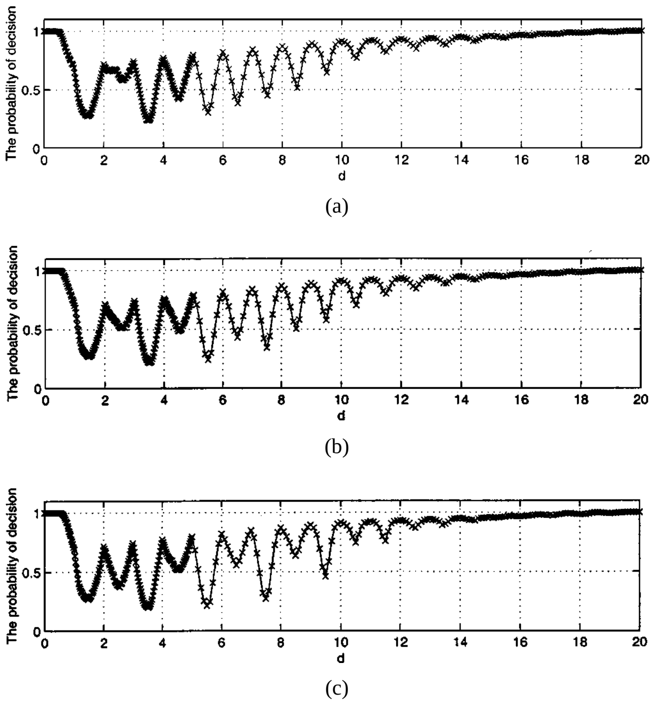

Fig. 6. Probability of decision $1 - P_u$ versus the delay $d \in [0 \cdots 20]$ for three different rolloff factors: (a) $\beta = 0.4$, (b) $\beta = 0.3$, and (c) $\beta = 0.2$. The probability of decision has been minimized by varying $\alpha \in [0 \cdots 1]$ with a step size of 0.05. SNR is 10 dB.

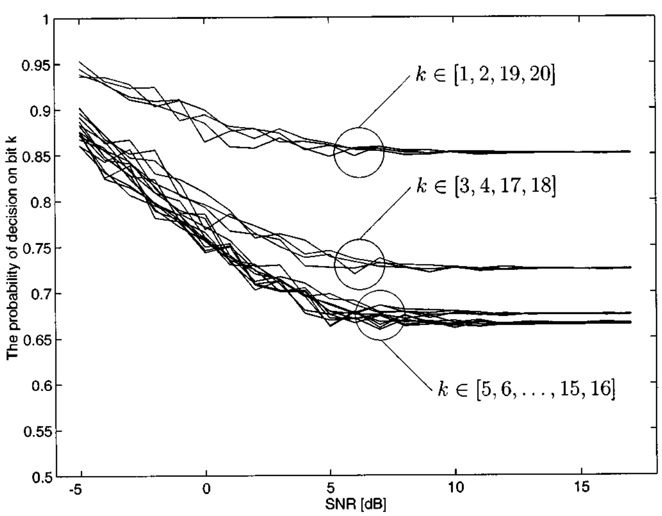

Fig. 7. Probability of detection for each individual symbol k, $P_{u,k}$ versus SNR.

Figs. 7 and 8 show data from a simulation varying the SNR in a two-tap channel model with impulse response $[h_0, h_1, h_2] = [1, 0, 1]$, as described by the model (14) with $d = 2$ and $\alpha = 1$. Fig. 7 shows the estimated probability that symbol number k is detected versus SNR. The plot shows the probability of decision for each individual symbol $k \in [1, 2, \cdots, 20]$ separately. The four symbols at the edges of the block, i.e., the symbols corresponding to $k \in [1, 2, 19, 20]$, are the most likely to be detected. This is because these symbols are subjected to less ISI than the symbols in the center part of the block. As these symbols are detected more frequently, their neighbors, the symbols at $k \in [3, 4, 17, 18]$, are also detected more frequently. It can be observed that the probability of detection in general decreases with increasing SNR, which may at first seem counterintuitive. However, as the SNR increases or, equivalently, the noise power decreases, the probability that any z_k exceeds the thresholds also decreases: the quality of the decisions increases with the SNR, while the number of decisions made decreases.

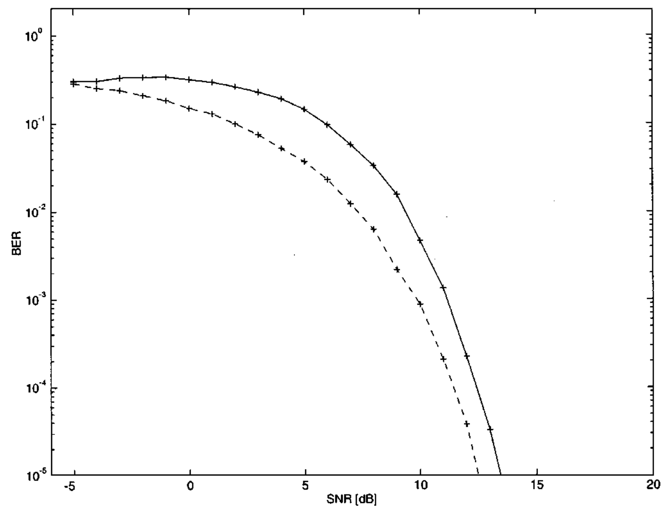

Fig. 8. BER of the MLSD for the symbols that could have been detected by the proposed detector *(dashed line)* and for the symbols that would have been left undetermined *(solid line)* plotted versus SNR.

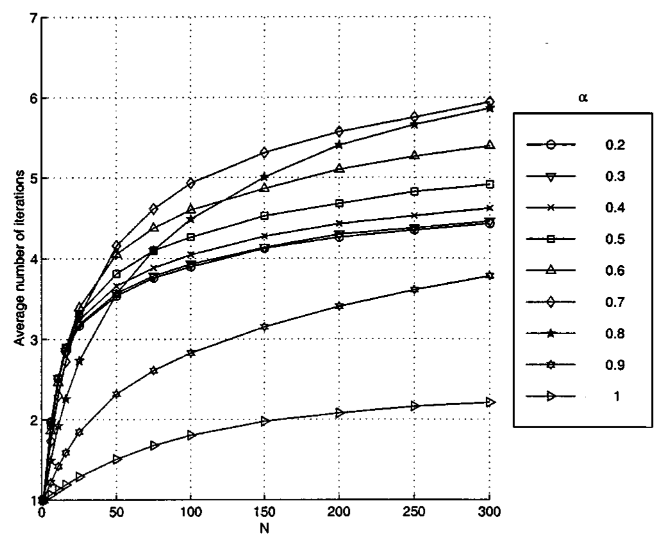

Fig. 9. Average number of iterations versus the block length N, for $\alpha \in \{0.2, 0.3, 0.4, 0.6, 0.8, 1.0\}$ with $d = 2$. Note the scales on the axes as well as the order of the curves.

In Fig. 8, the bit-error rate (BER) of the MLSD is plotted versus SNR, with the BER divided into two parts as follows: 1) BER of the symbols that would have been detected by the proposed detector and 2) BER of the symbols that would have been left undetected. It is observed that the symbols that would have been detected have a BER that is much lower than the BER of the other symbols. Apparently, the detector at an average finds symbols that are “easy” to make correct decisions on. In this sense, the detector can also be seen as a means to obtain binary soft information about the reliability of MLSD decisions.

In Fig. 9, the average number of iterations needed for the algorithm to terminate is plotted versus the block length N for varying strength of the second tap α $(d = 2)$ in the two-tap channel model (14). It can be observed that the average number of iterations only increases slowly with N, especially for small and large values of $|\alpha|$.

Fig. 10 shows the estimated probability that a symbol is left undetermined versus the block length N for different values of α in (14), but with $d = 2$. For large enough block lengths N, the probability that a symbol is left undetermined seems to be independent of N. This is reasonable as the influence of edge effects should decrease.

V. Simulation Examples—Performance

We present two examples of using the proposed detector in combination with other receivers. An extensive discussion of combination techniques and performance results is beyond the scope of this presentation, but we hope that the short example below will provide a working knowledge on the topic. This is one of many ways of combining receivers. We have chosen to show it here because it has favorable performance and because it is less obvious than, for instance, just substituting the detected symbols.

Say that the proposed detector is used in a system modeled with (1). In order to detect the remaining bits, we assume that the detected bits are correct and subtract their influence from the received signal, as is illustrated in Fig. 11. Consider the random vector resulting from this subtraction

$$\mathbf{x} = \mathbf{y} - \mathbf{H}\hat{\mathbf{b}}_\Delta \tag{15}$$

where $\mathbf{y}$ is the channel output from (1) and $\hat{\mathbf{b}}_\Delta \in \{-1, 0, +1\}^N$ denotes the ternary output of the proposed detector. Let $\mathbf{G}$ be a matrix containing the columns of $\mathbf{H}$ corresponding to those bits that were *not* detected. It follows from the structure of $\mathbf{G}$ and (15) that the random vector $\mathbf{x}$, given $\mathbf{b}$ (and given correct decisions) is Gaussian with $\mathbf{x} \sim N(\mathbf{Gc}, \mathbf{R_n})$, where $\mathbf{c}$ is a column vector containing the undetermined bits. Thus, we model $\mathbf{x}$ as

$$\mathbf{x} = \mathbf{Gc} + \mathbf{n} \tag{16}$$

where $\mathbf{n}$ is the same noise vector as in (1), thus $\mathbf{n} \sim N(\mathbf{0}, \mathbf{R_n})$. To detect the undetermined bits, we can give the vector $\mathbf{x}$ and the characteristics of the channel model (16), i.e., $\mathbf{G}$ and $\mathbf{R_n}$, to *any detector* intended for block transmission systems. Note that $\mathbf{G}$ in general will represent a time-variant system even when the original channel model $\mathbf{H}$ represented a time-invariant channel.

Naturally, when a decision made by the proposed detector is erroneous, the reduced system of (16) is an incorrect model of the signal $\mathbf{x}$. As a consequence, the complementary detector will be inclined to make additional erroneous decisions on the remaining bits. However, since the proposed detector makes MLSD decisions, the error propagation should be less serious than compared to the error propagation in, for example, a decision-feedback equalizer (DFE). Simulations such as the ones presented below provide some support for this conjecture.

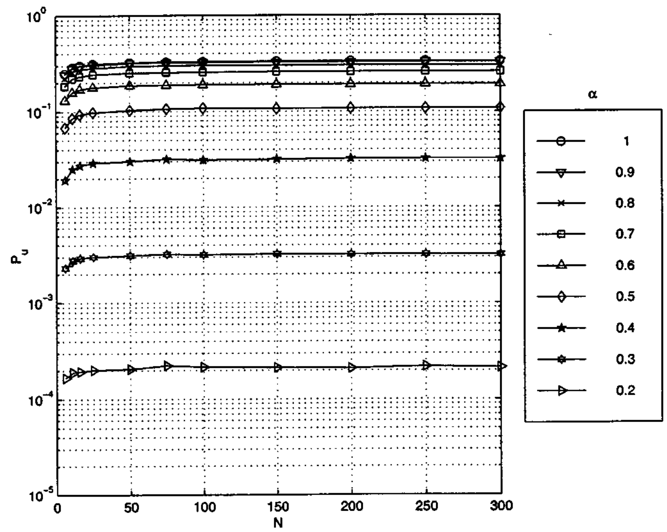

Fig. 10. Estimated probability that a symbol is left undetermined, P_u, versus the block length N, for $\alpha \in \{0.2, 0.3, 0.4, 0.6, 0.8, 1.0\}$ with $d = 2$.

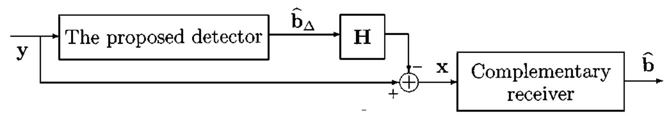

Fig. 11. Proposed detector as a preprocessor to a complementary receiver. The vector $\hat{\mathbf{b}}_\Delta$ denotes the ternary output of the proposed detector.

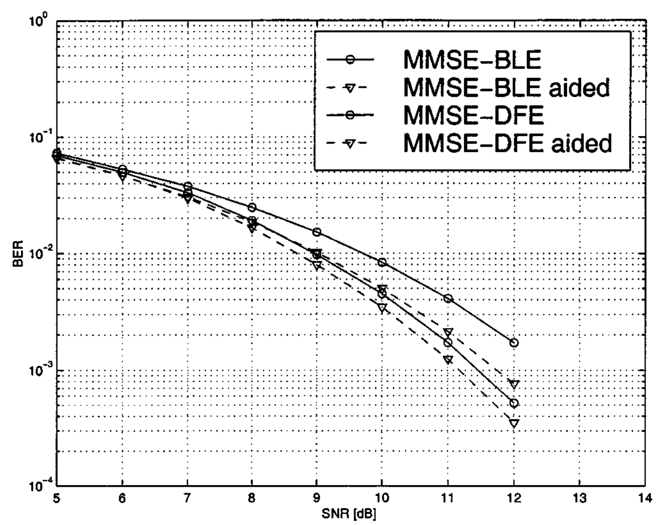

Fig. 12. BER of the aided and unaided receivers versus SNR for a channel with long delay spread.

We present results from simulations of block transmission systems with two different complementary receivers: the minimum mean-square-error (MMSE) linear equalizer and the MMSE DFE. They were both implemented as described by Kaleh in [7].[3]

In the simulations, blocks of data of length $N = 70$ are transmitted over a time-invariant dispersive channel with additive stationary white Gaussian noise, as described by (1). The impulse response of the channel is $[h_0, h_1, h_2, \cdots, h_{19}, h_{20}] =$

[3]These receivers were derived in [7] to operate on the output of the matched filter. Here, this corresponds to $\mathbf{G}^T\mathbf{R}_\mathbf{n}^{-1}\mathbf{x}$.

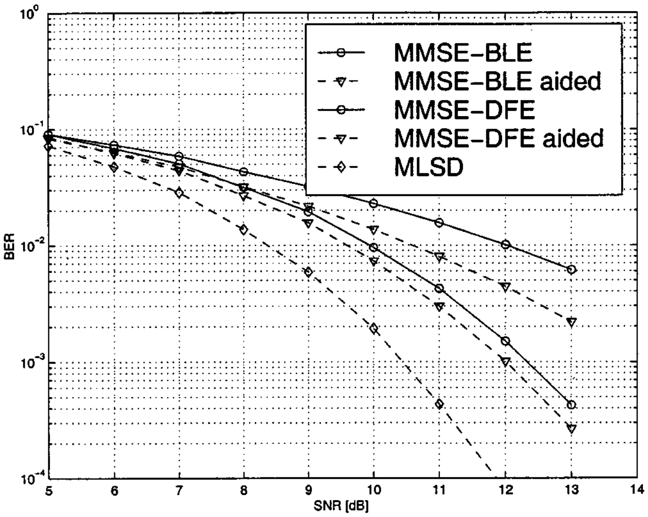

Fig. 13. BER of the aided and unaided receivers versus SNR for a channel with short delay spread. The BER of the MLSD is included as a reference.

$[1, 0, \cdots, 0, 1, 0.3]$. The simulation results in terms of the BER versus SNR are presented in Fig. 12.

Another example with a shorter channel enabling the MLSD to be included as a reference is given in Fig. 13. The previous channel has thus been shortened to $[h_0, h_1, h_2, h_3, h_4] = [1, 0, 0, 1, 0.3]$. Here, the performances of the combined receivers are compared with the performance of the MLSD. From these simulations, one can conclude that the proposed detector gives some, but not revolutionary, improvements of performance.

VI. Discussion

A unique approach to the problem of MLSD has been presented based on a novel decomposition of the likelihood function. A detector of low computational complexity is derived that makes MLSD decisions on a portion of the transmitted symbols. The key components of the detector are a matched filter and a threshold device with two variable thresholds per symbol. The thresholds for a specific symbol stem from an upper and lower bounding of a function of the other symbols, cf. (9)–(12). This function appears in the quadratic term of the log-likelihood function for the received sequence [see (8)]. We have given our presentation in a symbol-sampled block transmission environment. However, the results are generalizable, for instance, to continuous-transmission systems, fractionally-spaced complex-valued channels, and QPSK modulation.

We believe that the primary use of the proposed detector is as a preprocessor to suboptimal receivers inferior in performance to the MLSD. It can aid these receivers by providing MLSD decisions on some symbols.

In an implementation where the number of iterations is fixed, the detector can be implemented with a complexity proportional to the block length N times the length of the channel memory L, and due to the modulation being binary, without using multiplications (see [4]).

If the channel matrix $\mathbf{H}$ is unknown, an estimate of the channel can be used instead. If the estimate of the channel is close to the true channel, we expect that the receiver will perform close to optimum. No sensitivity analysis is given here, but it can be expected that the proposed detector and the MLSD have very similar characteristics.

An important property of the proposed detector, determining its practical potential, is its probability of making a decision. Typical detection rates lie between 30% and 80% of the symbols. Channels with strong ISI and high SNR leave many symbols undetermined. The average probability of making decisions is dependent on the ISI and the block length N, as well as on the covariance matrix of the noise.

Appendix
Log-Likelihood Function

Let b_i, z_i, and $m_{i,j}$ denote the elements in $\mathbf{b}$, $\mathbf{z} = \mathbf{H}^T\mathbf{R_n}^{-1}\mathbf{y}$ and $\mathbf{M} = \mathbf{H}^T\mathbf{R_n}^{-1}\mathbf{H}$, respectively, where i is the row index and j is the column index. Then, the log-likelihood function

$$\begin{aligned}\Lambda(\mathbf{b},\mathbf{y}) &= \|\mathbf{y}-\mathbf{Hb}\|^2_{\mathbf{R_n}^{-1}} \\ &= \mathbf{b}^T\mathbf{Mb} - 2\mathbf{b}^T\mathbf{z} + \mathbf{y}^T\mathbf{R_n}^{-1}\mathbf{y} \\ &= \sum_{i,j} b_i m_{i,j} b_j - 2\sum_i b_i z_i + \mathbf{y}^T\mathbf{R_n}^{-1}\mathbf{y} \\ &= \sum_{j\neq k} b_k m_{k,j} b_j + \sum_{i\neq k} b_i m_{i,k} b_k - 2 b_k z_k \\ &\quad + \sum_{i\neq k, j\neq k} b_i m_{i,j} b_j + b_k m_{k,k} b_k \\ &\quad - 2\sum_{i\neq k} b_i z_i + \mathbf{y}^T\mathbf{R_n}^{-1}\mathbf{y} \\ &= b_k\left(\sum_{i\neq k} b_i m_{i,k} + \sum_{j\neq k} m_{k,j} b_j - 2z_k\right) \\ &\quad + \sum_{i\neq k, j\neq k} b_i m_{i,j} b_j - 2\sum_{i\neq k} b_i z_i + |b_k|^2 m_{k,k} \\ &\quad + \mathbf{y}^T\mathbf{R_n}^{-1}\mathbf{y}. \end{aligned} \tag{A.1}$$

As $|b_k| = 1$ and $m_{i,j} = m_{j,i}$

$$\begin{aligned}\Lambda(\mathbf{b},\mathbf{y}) = 2b_k(\Delta(\mathbf{b},k) - z_k) &+ \sum_{i\neq k, j\neq k} b_i m_{i,j} b_j \\ &- 2\sum_{i\neq k} b_i z_i + m_{k,k} + \mathbf{y}^T\mathbf{R_n}^{-1}\mathbf{y}\end{aligned} \tag{A.2}$$

where

$$\Delta(\mathbf{b},k) \triangleq \sum_{i\neq k} b_i m_{i,k} \tag{A.3}$$

see (8).

References

[1] P. Ödling and H. B. Eriksson, "A Fast, Iterative Detector Making MLSD Decisions on Scattered Bits," Luleá Univ. of Technology, Luleá, Sweden, Res. Rep. TULEA 1995:12, May 1995.

[2] P. Ödling, H. B. Eriksson, and P. O. Börjesson, "Making MLSD-decisions on some individual symbols in a sequence by thresholding the matched filter output," in *Proc. 1997 IEEE Int. Symp. Information Theory*, Ulm, Germany, June 1997, p. 75.

[3] D. G. 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.

[4] R. Nilsson, F. Sjöberg, O. Edfors, P. Ödling, H. Eriksson, S. K. Wilson, and P. O. Börjesson, "A low complexity threshold detector making MLSD decisions in a multiuser environment," in *Proc. 48th IEEE Vehicular Technology Conf.*, May 1998, pp. 333–337.

[5] S. Verdú, "Minimum probability of error for asynchronous Gaussian multiple-access channels," *IEEE Trans. Inform. Theory*, vol. IT-32, pp. 85–96, Jan. 1986.

[6] C. Barbosa, "Maximum likelihood sequence estimators: A geometric view," *IEEE Trans. Inform. Theory*, vol. 35, pp. 419–427, Mar. 1989.

[7] G. K. Kaleh, "Channel equalization for block transmission systems," *IEEE J. Select. Areas Commun.*, vol. 13, pp. 110–121, Jan. 1995.

[8] R. W. Chang and J. C. Hancock, "On receiver structures for channels having memory," *IEEE Trans. Inform. Theory*, vol. IT-12, pp. 463–468, Oct. 1966.

[9] J. G. Proakis, *Digital Communications*. New York: McGraw-Hill, 1989.

Per Ödling (S'90–A'95) was born in Örnsköldsvik, Sweden, in 1966. He received the M.S.E.E. degree in 1989, the licentiate degree in 1993, and the Ph.D. degree in signal processing in 1995, all from Luleå University of Technology, Luleå, Sweden.

Currently, he is a Key Researcher at the Telecommunications Research Center Vienna, Austria, and an Adjoint Assistant Professor at the Department of Applied Electronics, Lund Institute of Technology, Lund, Sweden. Having researched cooperatively with industry over the past ten years, he is currently developing broad-band internet communication over radio and twisted copper pairs. His work includes contributions to the standardization of the Universal Mobile Telecommunications System (UMTS) and the Very high bit rate Digital Subscriber Loop (VDSL) within the European Telecommunications Standards Institute (ETSI), the American National Standards Institute (ANSI), and the International Telecommunication Union (ITU). He is also working as a consultant with the design and development of ADSL and VDSL modems and with regulatory aspects of the unbundling of the local loop.

Håkan B. Eriksson (S'90–M'95) was born in Stockholm, Sweden, in 1964. He received the M.S.E.E. degree in 1989 from Luleå University of Technology, Luleå, Sweden. Pursuing graduate studies in the field of signal processing, he received the licentiate degree in 1993, with biomedical signal processing and ultrasonic imaging as the topic. Shifting focus, he was awarded the Ph.D. degree in 1995 at Luleå University of Technology.

After a period as an Assistant Professor, he took a position as Senior Researcher at Ericsson Mobile Communications AB in Lund, Sweden, where he now is a technical manager. His research interests span from signal processing with efficient implementation of algorithms to information theory and its applications.

Per Ola Börjesson (S'70–M'80) was born in Karlshamn, Sweden, in 1945. He received the M.Sc. degree in electrical engineering in 1970 and the Ph.D. degree in telecommunication theory in 1980, both from the Lund Institute of Technology (LTH), Lund, Sweden. In 1983, he received the degree of Docent in telecommunication theory.

Since 1998, he has been with Lund University as a Professor of Signal Processing. Between 1988 and 1998, he was with Luleå University of Technology, Luleå, Sweden, as a Professor of Signal Processing. His primary research interest is in high performance communication systems, in particular, high data rate wireless and twisted pair systems. He is presently researching signal processing techniques in communication systems that use orthogonal frequency-division multiplexing (OFDM) or discrete multitone (DMT) modulation. He emphasizes the interaction between models and real systems, from the creation of application-oriented models based on system knowledge to the implementation and evaluation of algorithms.