Minimum Mean Squared Error Equalization Using *A Priori* Information

Michael Tüchler, Andrew C. Singer, *Member, IEEE*, and Ralf Koetter, *Member, IEEE*

***Abstract*—A number of important advances have been made in the area of joint equalization and decoding of data transmitted over intersymbol interference (ISI) channels. Turbo equalization is an iterative approach to this problem, in which a maximum *a posteriori* probability (MAP) equalizer and a MAP decoder exchange soft information in the form of prior probabilities over the transmitted symbols. A number of reduced-complexity methods for turbo equalization have recently been introduced in which MAP equalization is replaced with suboptimal, low-complexity approaches. In this paper, we explore a number of low-complexity soft-input/soft-output (SISO) equalization algorithms based on the minimum mean square error (MMSE) criterion. This includes the extension of existing approaches to general signal constellations and the derivation of a novel approach requiring less complexity than the MMSE-optimal solution. All approaches are qualitatively analyzed by observing the mean-square error averaged over a sequence of equalized data. We show that for the turbo equalization application, the MMSE-based SISO equalizers perform well compared with a MAP equalizer while providing a tremendous complexity reduction.**

***Index Terms*—Equalization, iterative decoding, low complexity, minimum mean square error.**

I. INTRODUCTION

IN MANY practical communication systems, data is transmitted over a channel with intersymbol interference (ISI). At the transmitter, the data is often protected by the addition of a controlled amount of redundancy using forward error correction or an error-correction code (ECC). It is then the task of the receiver to exploit both the structure of the transmit symbol constellation (as viewed at the output of the channel) and the structure of the code to detect and decode the transmitted data sequence. Methods that exploit the structure of the transmitted symbol constellation are referred to as equalization, whereas those that exploit the structure of the code are termed decoding. A number of important advances have been made in the area of joint equalization and decoding in which traditional equalization methods and decoding methods exchange information in an iterative fashion until convergence is achieved. In its original form, turbo equalization [1] employed maximum *a posteriori* probability (MAP) equalization and decoding methods in such an iterative scheme. In this paper, we consider an approach to minimum mean square error (MMSE)-based equalization that enables significant reduction in the computational complexity of such turbo-equalization methods.

We will assume a coherent symbol-spaced receiver front-end, as well as precise knowledge of the signal phase and symbol timing, such that the channel can be approximated by an equivalent, discrete-time, baseband model, where the transmit filter, the channel, and the receive filter, are represented by a discrete-time linear filter, with finite-length impulse response

$$h[n] = \sum_{k=0}^{M-1} h_k \, \delta[n-k]$$

of length M. The coefficients h_k are assumed to be time-invariant and known to the receiver.

In a typical receiver, the data received from the channel is processed with an equalizer to mitigate the effects of ISI. The equalizer then produces estimates of the data, which are passed onto the decoder for the ECC. For complexity reasons, the equalization task typically consists of linear processing of the received channel output [linear equalizer (LE)] and possibly past symbol estimates [decision feedback equalizer (DFE)] [2], [3]. The parameters of these filters can be designed according to a variety of different optimization criteria, such as the zero forcing (ZF) or MMSE criteria [2], [3]. Optimal equalization methods for minimizing the bit error rate (BER) and the sequence error rate are also nonlinear and are based on maximum likelihood (ML) estimation, which turns into MAP estimation in presence of *a priori* information about the transmitted data. Reasonably efficient algorithms exist for MAP/ML sequence estimation, e.g., the Viterbi algorithm (VA) [2], [4], [5] and MAP/ML symbol estimation, e.g., the BCJR algorithm [6]. We refer to these estimation methods as MAP/ML equalization.

When the data has been protected with a convolutional code, improvements in the BER can be easily obtained through the use of a soft-input convolutional decoder, with negligible increase in computational complexity. There is an increase in hardware complexity, however, since these symbol estimates (soft information) must be passed to the decoder and will require higher precision than the quanitized (discrete-alphabet) symbol constellation points. Most practical communication systems also insert an interleaver after the encoder (in the transmitter) and a deinterleaver before the decoder (in the receiver) [7], [8]. The process of interleaving permutes the symbols within a given block of data and, therefore, tends to decorrelate error events

Manuscript received July 24, 2000; revised November 21, 2001. This work was supported in part by the National Science Foundation, under Grants CCR-0092598 (CAREER) and CCR-9984515 (CAREER) and by the Office of Naval Research under Award N000140110117. The associate editor coordinating the review of this paper and approving it for publication was Michail K. Tsatsanis.

M. Tüchler is with the Institute for Communications Engineering, Technische Universität München, München, Germany (e-mail: michael.tuechler@ei.tum.de).

A. C. Singer and R. Koetter are with the Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, Urbana, IL 61801 USA (e-mail: acsinger@uiuc.edu; koetter@uiuc.edu).

Publisher Item Identifier S 1053-587X(02)01336-3.

introduced by the equalizer between neighboring symbols. Convolutional decoders are often troubled by such error "bursts" if left unpermuted.

A BER-optimal receiver that jointly addresses equalization and decoding is usually impractically complex, in particular, in the presence of an interleaver. However, a number of iterative receiver algorithms have been developed that achieve near-optimal performance by repeating the equalization and decoding tasks on the same set of received data, using feedback information from the decoder in the equalization process. This method, which is called turbo equalization, was originally developed for concatenated convolutional codes (turbo coding [9]) and is now adapted to various communication problems, such as trellis-coded modulation (TCM) [10], [11] and code division multiple access (CDMA) [12]. We refer to standard references [13]–[15] for an overview of turbo coding. Turbo equalization systems were first proposed in [1] and developed further by a number of others [16], [17]. In each of these systems, MAP-based techniques, most often a soft-output Viterbi algorithm [18], are used exclusively for both equalization and decoding [1], [16]. The slightly more complex BCJR algorithm [6] was implemented in [16]. Combined turbo coding and equalization [19], [20] can include three or more layers, with two or more coding layers as in conventional turbo coding applications and an additional equalizer.

For channels with large delay spreads (long duration impulse responses) and for large constellation sizes, MAP/ML-based equalization suffers from impractically high computational complexity. This situation is only exacerbated in the context of turbo equalization, with the need to perform equalization and decoding several times for each block of data. An important area of active research is the development of low-complexity alternatives to such MAP/ML equalization methods for use in joint equalization and decoding. Ariyavisitakul and Li [21] proposed a joint approach that is distinct from turbo equalization, working with convolutional coding and a DFE. Here, within the DFE, soft information from the DFE forward filter and tentative (hard) decisions from the decoder using the VA are fed back. Wang and Poor [12] proposed a turbo equalization-like system as part of a multiuser detector for coded CDMA. This iterative scheme is based on turbo equalization using an LE to reduce ISI and MAP decoding. The MAP equalizer is thus replaced with a LE, whose filter parameters are updated for every output symbol of the equalizer. In [22], the MAP equalizer in the turbo equalization framework is exchanged with a soft interference canceler based on linear filters with low computational complexity, whose coefficients are obtained using a least-mean-square (LMS)-based update algorithm. This idea is enhanced in [23], where the filter coefficients are obtained using the LMS algorithm to match the output of a MAP equalizer. For various signal-to-noise ratios (SNRs) and feedback information states, a linear estimate of the MAP equalizer is stored in a table and used for equalization in the receiver. The approach in [24] is similar to that of [22] but assumes a (known) impulse response of a partial response channel occurring in magnetic recording applications. The equalizer filter output is assigned a reliability measure enabling the receiver to decide whether the linear algorithm should be used instead of MAP equalization. Another common technique to decrease the complexity of the MAP equalizer is to reduce the number of states in the underlying trellis, which was applied to turbo equalization in [25]. The approaches in [22]–[24], and those proposed in this paper, address a major shortcoming of the classical turbo equalization scheme [1], [16], [17], which is the exponentially increasing complexity of the equalizer for channels with a long impulse response or large signal alphabets.

In this paper, we replace the MAP equalizer with a linear equalizer, where the filter parameters are updated using the MMSE criterion. This differs from conventional MMSE-based equalization methods in that the MMSE criterion is evaluated over both the distribution of the noise as well as the distribution over the symbols. In the context of turbo equalization, the symbol distribution is no longer independent and identically distributed (i.i.d.), as is typically assumed for MMSE-based equalization, due to the information fed back to the equalizer from the error correction decoder. We show that, as a result, the coefficients of the equalizer change as a function of time and must be recomputed for each data symbol to be estimated. We address the additional computational complexity by developing a recursive algorithm for computing the equalizer coefficients, as well as a number of suboptimal, low-complexity methods whose performance is nearly as good as the MMSE-optimal approach. The performance of the derived algorithms is analyzed by observing the time averaged mean squared error (MSE) and verified by simulation results.

II. Basics

For the derivations of the algorithms, we consider the data transmission system depicted in Fig. 1. Length $L \cdot Q$ sequences $\mathbf{c} \triangleq [\mathbf{c}_1 \; \mathbf{c}_2 \cdots \mathbf{c}_L]$, partitioned into length Q subsequences $\mathbf{c}_n \triangleq [c_{n,1} \; c_{n,2} \cdots c_{n,Q}]$, of bits $c_{n,j} \in \{0, 1\}$, are subject to transmission over an ISI channel. We assume that the $c_{n,j}$ are independent and distributed according to the *a priori* information $L(c_{n,j})$ defined as the log-likelihood ratio (LLR)

$$L(c_{n,j}) \triangleq \ln[P(c_{n,j}=0)/P(c_{n,j}=1)].$$

For the turbo equalization application, the $c_{n,j}$ are the interleaved code bits from the encoder for the ECC, and the $L(c_{n,j})$ are the feedback information from the decoder for the ECC. Since neither the $c_{n,j}$ are independent nor the $L(c_{n,j})$ are true *a priori* information on the $c_{n,j}$, it is the task of the interleaver to assure that the two assumptions *approximately* hold at least locally and for several iterations.

The modulator maps each $\mathbf{c}_n$ to a symbol x_n from the 2^Q-ary symbol alphabet $\mathcal{S} = \{\alpha_1, \alpha_2, \ldots, \alpha_{2^Q}\}$, where $\alpha_i \in \mathbb{C}$ corresponds to the bit pattern $\mathbf{s}_i \triangleq [s_{i,1} \; s_{i,2} \cdots s_{i,Q}]$, $s_{i,j} \in \{0, 1\}$, and $\mathbb{C}$ denotes the complex numbers. We require that the alphabet has zero mean $\sum_{i=1}^{2^Q} \alpha_i = 0$ and unit energy $2^{-Q} \sum_{i=1}^{2^Q} |\alpha_i|^2 = 1$. To illustrate the algorithms to be derived, we consider the three alphabets in Table I.

The sequence $\mathbf{x} = [x_1 \; x_2 \cdots x_L]$, $x_n \in \mathcal{S}$ is transmitted. The receiving process is disturbed by complex-valued additive white Gaussian noise (AWGN), i.e., the noise samples w_n are i.i.d.

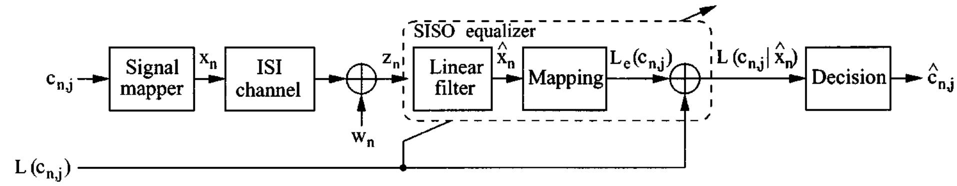


Fig. 1. Soft-in soft-out equalization using *a priori* information.

TABLE I
THREE SYMBOL ALPHABETS OVER THE COMPLEX NUMBERS ($\imath$ DENOTES $\sqrt{-1}$)

BPSK:

i	1	2
$s_{i,1}$	0	1
α_i	$+1$	-1

QPSK:

i	1	2	3	4
$s_{i,1}\,s_{i,2}$	00	10	01	11
α_i	$(+1+\imath)/\sqrt{2}$	$(-1+\imath)/\sqrt{2}$	$(+1-\imath)/\sqrt{2}$	$(-1-\imath)/\sqrt{2}$

8-PSK:

i	1	2	3	4	5	6	7	8
$s_{i,1}\,s_{i,2}\,s_{i,3}$	000	100	010	110	001	101	011	111
α_i	$(-1+\imath)/\sqrt{2}$	$-\imath$	$\imath$	$(+1-\imath)/\sqrt{2}$	-1	$(-1-\imath)/\sqrt{2}$	$(+1+\imath)/\sqrt{2}$	$+1$

with the probability density function (PDF) $\phi_{0,\,\sigma_w^2}(w)$ defined by

$$\phi_{\mu,\,\sigma^2}(w) \triangleq 1/(\pi\sigma^2)\cdot e^{-|w-\mu|^2/\sigma^2}, \quad w,\,\mu \in \mathbb{C},\ \ \sigma^2 \in \mathbb{R}^+.$$

The variance of the real and the imaginary part of w_n is $\sigma_w^2/2$. The receiver observes the length $L+M$ sequence $\mathbf{z} = [z_1\ z_2 \cdots z_{L+M-1}]$

$$z_n \triangleq \left(\sum_{k=0}^{M-1} h_k\, x_{n-k}\right) + w_n$$

which, together with the *a priori* information $L(c_{n,\,j})$ for each $c_{n,\,j}$, is input to the SISO equalizer.

A MAP-based equalizer computes the *a posteriori* probabilities $P(c_{n,\,j} = c|\mathbf{z})$, $c \in \{0,\,1\}$, or the *a posteriori* LLR

$$\begin{aligned} L(c_{n,\,j}|\mathbf{z}) &\triangleq \ln \frac{P(c_{n,j}=0|\mathbf{z})}{P(c_{n,j}=1|\mathbf{z})} \\ &= \ln \frac{\sum\limits_{\forall\,\mathbf{c}:c_{n,j}=0} p(\mathbf{z}|\mathbf{c})P(\mathbf{c})}{\sum\limits_{\forall\,\mathbf{c}:c_{n,j}=1} p(\mathbf{z}|\mathbf{c})P(\mathbf{c})} \end{aligned} \tag{1}$$

respectively, which can be broken up into the sum

$$\ln \frac{\sum\limits_{\forall\,\mathbf{c}:c_{n,\,j}=0} p(\mathbf{z}|\mathbf{c}) \prod\limits_{\forall\,n',\,j' \text{ except } j'=j,\,n'=n} P(c_{n',\,j'})}{\sum\limits_{\forall\,\mathbf{c}:c_{n,\,j}=1} p(\mathbf{z}|\mathbf{c}) \prod\limits_{\forall\,n',\,j' \text{ except } j'=j,\,n'=n} P(c_{n',\,j'})} + L(c_{n,\,j}).$$

The first term represents the information about $c_{n,\,j}$ contained in $\mathbf{z}$ (channel information) and in the bits $c_{n',\,j'}$, for all n', j' except $j'=j$, $n'=n$. Despite the independence assumption on the $c_{n,\,j}$, knowledge about the $c_{n',\,j'}$ improves the information about $c_{n,\,j}$ since the ISI is reduced, and thus, the channel information is improved. Estimates $\hat{c}_{n,\,j} \in \{0,\,1\}$ of the transmitted bits are obtained from the sign of $L(c_{n,\,j}|\mathbf{z})$. When there is a further receiver component, e.g., a decoder of an ECC, the soft information $L(c_{n,\,j}|\mathbf{z})$ should be delivered instead of $\hat{c}_{n,\,j}$. This can improve the performance of the decoder tremendously.

Rather than computing (1), which may be computationally expensive, the proposed SISO equalizer in Fig. 1 first computes estimates $\hat{x}_n$ of the transmitted symbols x_n using a linear filter, whose coefficients are determined with the MMSE criterion and, next, *a posteriori* LLRs

$$\begin{aligned} L(c_{n,\,j}|\hat{x}_n) &\triangleq \ln \frac{P(c_{n,j}=0|\hat{x}_n)}{P(c_{n,j}=1|\hat{x}_n)} \\ &= \ln \frac{\sum\limits_{\forall\,\mathbf{c}_n:c_{n,j}=0} p(\hat{x}_n|\mathbf{c}_n)P(\mathbf{c}_n)}{\sum\limits_{\forall\,\mathbf{c}_n:c_{n,j}=1} p(\hat{x}_n|\mathbf{c}_n)P(\mathbf{c}_n)} \end{aligned} \tag{2}$$

with respect to $\hat{x}_n$. These LLR's, which can be viewed as an approximation of $L(c_{n,\,j}|\mathbf{z})$, can be broken up into the sum

$$\underbrace{\ln \frac{\sum\limits_{\forall\,\mathbf{c}_n:c_{n,j}=0} p(\hat{x}_n|\mathbf{c}_n) \prod\limits_{\forall\,j':j'\neq j} P(c_{n,j'})}{\sum\limits_{\forall\,\mathbf{c}_n:c_{n,j}=1} p(\hat{x}_n|\mathbf{c}_n) \prod\limits_{\forall\,j':j'\neq j} P(c_{n,j'})}}_{L_e(c_{n,\,j})} + L(c_{n,\,j}).$$

We emphasize that the LLR $L_e(c_{n,\,j})$, which is a function of $\hat{x}_n$ and the *a priori* LLRs $L(c_{n,\,j'})$, for all $j' \neq j$ should not depend on $L(c_{n,\,j})$, which is added separately. Therefore, we require that $\hat{x}_n$ does not depend on $L(c_{n,\,j})$, $j = 1,\,2,\,\cdots,\,Q$, which affects the derivation of the MMSE equalization algorithms. Using this restriction on $\hat{x}_n$, only one $\hat{x}_n$ has to be computed to obtain $L_e(c_{n,\,j})$ for all j.

For the sequel, some frequently used notation is introduced. Vectors are written in bold letters, and matrices are written in bold capital letters. Time-varying quantities are augmented with a time index, e.g., n, as subscript. The $i \times j$ matrix $\mathbf{1}_{i\times j}$ contains all ones, and $\mathbf{0}_{i\times j}$ contains all zeros. The matrix $\mathbf{I}_i$ is an $i \times i$ identity matrix. The operator $\mathrm{E}(\cdot)$ is the expectation over the PDF of the noise w_n and that of the transmitted symbols x_n, taking into account the *a priori* information $L(c_{n,\,j})$. The covariance operator is given by $\mathrm{Cov}(\mathbf{x},\,\mathbf{y}) \triangleq \mathrm{E}(\mathbf{x}\mathbf{y}^{\mathrm{H}}) - \mathrm{E}(\mathbf{x})\mathrm{E}(\mathbf{y}^{\mathrm{H}})$, where $(\,)^{\mathrm{H}}$ is the Hermitian operator. A conditioned expectation is over the corresponding conditioned PDFs and is denoted $\mathrm{E}(\cdot \mid condition)$. The operator $\mathrm{diag}(\cdot)$ to be applied to a length i vector returns an $i \times i$ square matrix with the vector elements along the diagonal.

III. Linear Equalization Using *A Priori* Information

A. MMSE Equalization

A linear estimate $\hat{x}_n$ of the transmitted symbol x_n using the observation $\mathbf{z}_n \triangleq [z_{n-N_2} z_{n-N_2+1} \cdots z_{n+N_1}]^{\mathrm{T}}$ of $N = N_1 + N_2 + 1$ received symbols z_n is given by

$$\hat{x}_n = \mathbf{a}_n^{\mathrm{H}} \mathbf{z}_n + b_n$$

where $\mathbf{a}_n \triangleq [a_{n,N_2}^* a_{n,N_2-1}^* \cdots a_{n,-N_1}^*]^{\mathrm{T}}$, $a_{n,k}, b_n \in \mathbb{C}$ are the coefficients of the estimator. The parameters N_1 and N_2 specify the length of the noncausal and the causal part of the estimator filter, respectively, and N is the overall filter length. When we allow both $\mathbf{a}_n$ and b_n to vary with n, we find that the choice

$$\begin{aligned}\mathbf{a}_n &= \mathrm{Cov}(\mathbf{z}_n, \mathbf{z}_n)^{-1} \mathrm{Cov}(\mathbf{z}_n, x_n) \\ b_n &= \mathrm{E}(x_n) - \mathbf{a}_n^{\mathrm{H}} \mathrm{E}(\mathbf{z}_n)\end{aligned}$$

minimizes the cost $\mathrm{E}(|x_n - \hat{x}_n|^2)$ [26], the MSE, and that

$$\hat{x}_n = \mathrm{E}(x_n) + \mathrm{Cov}(x_n, \mathbf{z}_n) \mathrm{Cov}(\mathbf{z}_n, \mathbf{z}_n)^{-1} (\mathbf{z}_n - \mathrm{E}(\mathbf{z}_n)).$$

We call this the MMSE solution. The observation $\mathbf{z}_n$ is given by

$$\mathbf{z}_n = \mathbf{H}\mathbf{x}_n + [w_{n-N_2} w_{n-N_2+1} \cdots w_{n+N_1}]^{\mathrm{T}}$$

where $\mathbf{x}_n \triangleq [x_{n-N_2-M+1}\ x_{n-N_2-M+2} \cdots x_{n+N_1}]^{\mathrm{T}}$, and $\mathbf{H}$ is the $N \times (N+M-1)$ channel convolution matrix

$$\mathbf{H} \triangleq \begin{bmatrix} h_{M-1} & h_{M-2} & \cdots & h_0 & 0 & \cdots & & 0 \\ 0 & h_{M-1} & h_{M-2} & \cdots & h_0 & 0 & \cdots & 0 \\ & & & \ddots & & & & \\ 0 & & \cdots & 0 & h_{M-1} & h_{M-2} & \cdots & h_0 \end{bmatrix}.$$

Given the i.i.d. noise samples w_n, we find that

$$\begin{aligned}\mathrm{E}(\mathbf{z}_n) &= \mathbf{H}\,\mathrm{E}(\mathbf{x}_n) \\ \mathrm{Cov}(x_n, \mathbf{z}_n) &= \mathrm{Cov}(x_n, x_n)[\mathbf{0}_{1\times(N_2+M-1)}\ \ 1\ \ \mathbf{0}_{1\times N_1}]\mathbf{H}^{\mathrm{H}} \\ \mathrm{Cov}(\mathbf{z}_n, \mathbf{z}_n) &= \sigma_w^2 \mathbf{I}_N + \mathbf{H}\mathrm{Cov}(\mathbf{x}_n, \mathbf{x}_n)\mathbf{H}^{\mathrm{H}}.\end{aligned}$$

It follows from the independence of the bits $c_{n,j}$ that the symbols x_n are independent and that $\mathrm{Cov}(x_n, x_m) = 0$ for all n, m, $n \neq m$. The entries of the covariance matrix $\mathrm{Cov}(\mathbf{x}_n, \mathbf{x}_n)$ are, therefore, nonzero only on the main diagonal. To compute $\mathbf{a}_n$ and b_n, the mean and the variance

$$\begin{aligned}\overline{x} &\triangleq \mathrm{E}(x_n) = \sum_{\alpha_i \in \mathcal{S}} \alpha_i \cdot P(x_n = \alpha_i) \\ v_n &\triangleq \mathrm{Cov}(x_n, x_n) = \left(\sum_{\alpha_i \in \mathcal{S}} |\alpha_i|^2 \cdot P(x_n = \alpha_i) \right) - |\overline{x}_n|^2\end{aligned}$$

of the transmitted symbols x_n are required, which are functions of the *a priori* information $L(c_{n,j})$ since

$$\begin{aligned}P(x_n = \alpha_i) &= \prod_{j=1}^{Q} P(c_{n,j} = s_{i,j}) \\ &= \prod_{j=1}^{Q} 1/2 \cdot (1 + \tilde{s}_{i,j} \cdot \tanh(L(c_{n,j}/2))\end{aligned}$$

where

$$\tilde{s}_{i,j} \triangleq \begin{cases} +1, & s_{i,j} = 0 \\ -1, & s_{i,j} = 1. \end{cases}$$

Table II presents these functions for the three considered symbol alphabets. Using $\overline{x}_n$ and v_n to define

$$\begin{aligned}\overline{z}_n &\triangleq \mathrm{E}(z_n) = \sum_{k=0}^{M-1} h_k \overline{x}_{n-k} \\ \overline{\mathbf{z}}_n &\triangleq \mathrm{E}(\mathbf{z}_n) = [\overline{z}_{n-N_2} \overline{z}_{n-N_2+1} \cdots \overline{z}_{n+N_1}]^{\mathrm{T}} \\ \mathbf{V}_n &\triangleq \mathrm{Cov}(\mathbf{x}_n, \mathbf{x}_n) = \mathrm{diag}[v_{n-M-N_2+1} \cdots v_{n+N_1}] \\ \mathbf{s} &\triangleq \mathbf{H}[\mathbf{0}_{1\times(N_2+M-1)}\ \ 1\ \ \mathbf{0}_{1\times N_1}]^{\mathrm{T}} \\ \mathbf{\Sigma}_n &\triangleq \mathrm{Cov}(\mathbf{z}_n, \mathbf{z}_n) = \sigma_w^2 \mathbf{I}_N + \mathbf{H}\mathbf{V}_n\mathbf{H}^{\mathrm{H}}\end{aligned}$$

the estimate $\hat{x}_n$ is given by

$$\hat{x}_n = \overline{x}_n + \mathbf{a}_n^{\mathrm{H}}(\mathbf{z}_n - \overline{\mathbf{z}}_n) = \overline{x}_n + \sum_{k=-N_1}^{N_2} a_{n,k} \cdot (z_{n-k} - \overline{z}_{n-k})$$

where $\mathbf{a}_n = v_n\ \mathbf{\Sigma}_n^{-1}\mathbf{s}$. This equation is equivalent to filtering the difference $z_n - \overline{z}_n$ with a linear filter with N coefficients $f_{n,k}$, $k = -N_1, 1 - N_1, \cdots, N_2$ given by

$$\mathbf{f}_n = [f_{n,N_2}^* f_{n,N_2-1}^* \cdots f_{n,-N_1}^*]^{\mathrm{T}} \triangleq \mathbf{\Sigma}_n^{-1}\,\mathbf{s} \tag{3}$$

multiplying the result with v_n and adding $\overline{x}_n$

$$\hat{x}_n = \overline{x} + v_n \cdot \mathbf{f}_n^{\mathrm{H}}(\mathbf{z}_n - \overline{\mathbf{z}}_n). \tag{4}$$

However, $\hat{x}_n$ depends on $L(c_{n,j})$ via $\overline{x}_n$ and v_n. In order that $\hat{x}_n$ is independent from $L(c_{n,j})$, for all j, we set $L(c_{n,j})$, $j = 1, \cdots, Q$, to 0 while computing $\hat{x}_n$, yielding $\overline{x}_n = 0$ and $v_n = 1$. This changes (3) and (4) to

$$\begin{aligned}\mathbf{f}_n' &\triangleq \mathbf{f}_n|_{v_n=1} = \left(\mathbf{\Sigma}_n + (1 - v_n)\,\mathbf{s}\,\mathbf{s}^{\mathrm{H}}\right)^{-1}\mathbf{s} \\ \hat{x}_n &= 0 + 1 \cdot \mathbf{f}'^{\mathrm{H}}_n(\mathbf{z}_n - \overline{\mathbf{z}}_n + (\overline{x}_n - 0)\mathbf{s}).\end{aligned} \tag{5}$$

We can express $\mathbf{f}_n'$ as scaled version of $\mathbf{f}_n$ using the matrix inversion lemma [27]

$$\begin{aligned}\mathbf{f}_n' &\triangleq \left(\mathbf{\Sigma}_n^{-1} - \mathbf{\Sigma}_n^{-1}\mathbf{s}\left((1 - v_n)^{-1} + \mathbf{s}^{\mathrm{H}}\mathbf{\Sigma}_n^{-1}\mathbf{s}\right)^{-1}\mathbf{s}^{\mathrm{H}}\mathbf{\Sigma}_n^{-1}\right)\mathbf{s} \\ &= \mathbf{f}_n - \mathbf{f}_n\left((1 - v_n)^{-1} + \mathbf{f}_n^{\mathrm{H}}\mathbf{s}\right)^{-1}\mathbf{f}_n^{\mathrm{H}}\mathbf{s} \\ &= (1 + (1 - v_n)\,\mathbf{f}_n^{\mathrm{H}}\mathbf{s})^{-1}\mathbf{f}_n.\end{aligned}$$

Finally, the estimates $\hat{x}_n$ are computed as

$$\hat{x}_n = K_n \cdot \mathbf{f}_n^{\mathrm{H}}(\mathbf{z}_n - \overline{\mathbf{z}}_n + \overline{x}_n\,\mathbf{s}) \tag{6}$$

where $K_n \triangleq (1 + (1 - v_n)\,\mathbf{f}_n^{\mathrm{H}}\mathbf{s})^{-1}$.

We assume that the PDFs $p(\hat{x}_n|\mathbf{c}_n=\mathbf{s}_i)=p(\hat{x}_n|x_n=\alpha_i)$, $i=1,2,\cdots,2^Q$ are Gaussian with the mean $\mu_{n,i}\triangleq \mathrm{E}(\hat{x}_n|x_n=\alpha_i)$ and the variance $\sigma^2_{n,i}\triangleq \mathrm{Cov}(\hat{x}_n,\hat{x}_n|x_n=\alpha_i)$

$$p(\hat{x}_n|\mathbf{c}_n=\mathbf{s}_i)\approx \phi_{\mu_{n,i},\sigma^2_{n,i}}(\hat{x}_n). \tag{7}$$

This assumption, which is also made in [12], tremendously simplifies the computation of the LLRs $L_e(c_{n,j})$ in (2). The performance degradation is negligible since (7) is applied to derive a mapping between $\hat{x}_n$ and $L_e(c_{n,j})$ only, and we found that receiver components using $L_e(c_{n,j})$ or $L(c_{n,j}|\hat{x}_n)$ were very robust to small-scale deviations in this mapping. The statistics $\mu_{n,i}$ and $\sigma^2_{n,i}$ of $\hat{x}_n$ are given by

$$\begin{aligned}\mu_{n,i}&=K_n\cdot \mathbf{f}_n^{\mathrm{H}}(\mathrm{E}(\mathbf{z}_n|x_n=\alpha_i)-\overline{\mathbf{z}}_n+\overline{x}_n\,\mathbf{s})\\&=K_n\cdot \mathbf{f}_n^{\mathrm{H}}(\mathbf{H}\,\mathrm{E}(\mathbf{x}_n|x_n=\alpha_i)-\overline{\mathbf{z}}_n+\overline{x}_n\,\mathbf{s})\\&=K_n\cdot\alpha_i\cdot \mathbf{f}_n^{\mathrm{H}}\mathbf{s}\\ \sigma^2_{n,i}&=K_n^2\cdot \mathbf{f}_n^{\mathrm{H}}\mathrm{Cov}(\mathbf{z}_n,\mathbf{z}_n|x_n=\alpha_i)\mathbf{f}_n\\&=K_n^2\cdot \mathbf{f}_n^{\mathrm{H}}\left(\mathbf{\Sigma}_n-v_n\mathbf{s}\,\mathbf{s}^{\mathrm{H}}\right)\mathbf{f}_n\\&=K_n^2\cdot(\mathbf{f}_n^{\mathrm{H}}\mathbf{s}-v_n\,\mathbf{f}_n^{\mathrm{H}}\mathbf{s}\,\mathbf{s}^{\mathrm{H}}\mathbf{f}_n)\end{aligned}$$

yielding

$$\begin{aligned}&L_e(c_{n,j})\\&=\ln\frac{\sum\limits_{\forall\mathbf{s}_i:s_{i,j}=0}p(\hat{x}_n|\mathbf{c}_n=\mathbf{s}_i)\prod\limits_{\forall j':j'\neq j}P(c_{n,j'}=s_{i,j'})}{\sum\limits_{\forall\mathbf{s}_i:s_{i,j}=1}p(\hat{x}_n|\mathbf{c}_n=\mathbf{s}_i)\prod\limits_{\forall j':j'\neq j}P(c_{n,j'}=s_{i,j'})}\\&=\ln\frac{\sum\limits_{\forall\mathbf{s}_i:s_{i,j}=0}\exp\left(-\varrho_{n,i}+\sum\limits_{\forall j':j'\neq j}\tilde{s}_{i,j'}L(c_{n,j'})\Big/2\right)}{\sum\limits_{\forall\mathbf{s}_i:s_{i,j}=1}\exp\left(-\varrho_{n,i}+\sum\limits_{\forall j':j'\neq j}\tilde{s}_{i,j'}L(c_{n,j'})\Big/2\right)}\end{aligned}\tag{8}$$

where

$$\varrho_{n,i}\triangleq\frac{|\overline{x}_n-\mu_{n,i}|^2}{\sigma^2_{n,i}}=\frac{|\mathbf{f}_n^{\mathrm{H}}(\mathbf{z}_n-\overline{\mathbf{z}}_n+\overline{x}_n\,\mathbf{s})-\alpha_i\,\mathbf{f}_n^{\mathrm{H}}\mathbf{s}|^2}{\mathbf{f}_n^{\mathrm{H}}\mathbf{s}-v_n\,\mathbf{f}_n^{\mathrm{H}}\mathbf{s}\,\mathbf{s}^{\mathrm{H}}\mathbf{f}_n}.$$

Table III shows that computing $L_e(c_{n,j})$, $j=1,2,\cdots,Q$, from $\varrho_{n,i}$ becomes rather simple for the symbol alphabets considered.

Without the existence of *a priori* information, i.e., $L(c_{n,j})=0$, for all n, j, the coefficient vector $\mathbf{f}_n$ would be fixed, i.e., not change with time. From $\overline{x}_n=0$ and $v_n=1$, for all n, under this condition, it follows that $\mathbf{f}_n$ is equal to

$$\mathbf{f}_{\mathrm{NA}}\triangleq\ \mathbf{\Sigma}_n^{-1}\mathbf{s}\big|_{v_n=1,\forall n}=\left(\sigma_w^2\mathbf{I}_N+\mathbf{H}\mathbf{H}^{\mathrm{H}}\right)^{-1}\mathbf{s}$$

which is the usual solution for linear MMSE equalization [2]. The subscript NA denotes "no *a priori* information." However, $\mathbf{f}_n=\mathbf{\Sigma}_n^{-1}\mathbf{s}$ has to be computed for each n since in general, the *a priori* information $L(c_{n,j})$ vary with n. A direct implementation of this operation requires an order of complexity that is cubic in N. We exploit the structured time dependence of $\mathbf{\Sigma}_n$ on the LLRs $L(c_{n,j})$ to develop a fast recursive solution to compute $\mathbf{f}_n$ that requires an order of complexity that is N^2. There are a number of related fast algorithms in the adaptive filtering literature [27]–[31] that develop recursive update strategies similar in spirit to the one developed in this paper. Many of these approaches exploit the existence of common submatrices within a partitioned covariance matrix, using either the matrix inversion lemma, or other structured inversions of partitioned matrices. These algorithms typically focus on time- or order-recursive (or both) least-squares methods. The time-recursive update algorithm introduced here uses the following partitioning scheme:

$$\mathbf{\Sigma}_n\triangleq\begin{bmatrix}\sigma_{\mathrm{P}} & \boldsymbol{\sigma}_{\mathrm{P}}^{\mathrm{H}}\\ \boldsymbol{\sigma}_{\mathrm{P}} & \mathbf{\Sigma}_{\mathrm{P}}\end{bmatrix},\qquad \mathbf{\Sigma}_{n+1}\triangleq\begin{bmatrix}\mathbf{\Sigma}_{\mathrm{N}} & \boldsymbol{\sigma}_{\mathrm{N}}\\ \boldsymbol{\sigma}_{\mathrm{N}}^{\mathrm{H}} & \sigma_{\mathrm{N}}\end{bmatrix} \tag{9}$$

where the $\mathbf{\Sigma}_i$, $i\in\{\mathrm{P},\mathrm{N}\}$ are $(N-1)\times(N-1)$ matrices, the $\boldsymbol{\sigma}_i$ are length $(N-1)$ column vectors, and the σ_i are scalars. The subscript P denotes quantities at the "present" time step n and N at the "next" time step $n+1$. A similar partitioning scheme is introduced for the inverses of $\mathbf{\Sigma}_n$ and $\mathbf{\Sigma}_{n+1}$:

$$\begin{aligned}\mathbf{\Sigma}_n^{-1}&\triangleq\mathbf{U}_n\triangleq\begin{bmatrix}u_{\mathrm{P}} & \mathbf{u}_{\mathrm{P}}^{\mathrm{H}}\\ \mathbf{u}_{\mathrm{P}} & \mathbf{U}_{\mathrm{P}}\end{bmatrix}\\ \mathbf{\Sigma}_{n+1}^{-1}&\triangleq\mathbf{U}_{n+1}\triangleq\begin{bmatrix}\mathbf{U}_{\mathrm{N}} & \mathbf{u}_{\mathrm{N}}\\ \mathbf{u}_{\mathrm{N}}^{\mathrm{H}} & u_{\mathrm{N}}\end{bmatrix}\end{aligned} \tag{10}$$

TABLE II
CONVERSION FROM $L(c_{n,j})$ TO $\overline{x}_n\triangleq\mathrm{E}(x_n)$ AND $v_n\triangleq\mathrm{Cov}(x_n,x_n)$

BPSK:
$\bar{x}_n=\tanh(L(c_{n,1})/2)$,
$v_n=1-|\bar{x}_n|^2$,

QPSK:
$\bar{x}_n=1/\sqrt{2}\cdot(\tanh(L(c_{n,1})/2)+\tanh(L(c_{n,2})/2)\,\imath)$,
$v_n=1-|\bar{x}_n|^2$,

8-PSK:
$l_j=\tanh(L(c_{n,j})/2),\ \ j=1,2,3$,
$\bar{x}_n=((1+\sqrt{2})\imath-1)/4\cdot l_1-(1+\sqrt{2}+\imath)/4\cdot l_2$
$\ \ +l_3\cdot\left((1-\sqrt{2}+\imath)/4\cdot l_1+(1+(\sqrt{2}-1)\imath)/4\cdot l_2\right)$,
$v_n=1-|\bar{x}_n|^2$.

TABLE III
CONVERSION FROM THE LINEAR FILTER OUTPUT TO $L_e(c_{n,j})$ FOR THE MMSE SOLUTION

BPSK:
$L_e(c_{n,1})=4/(1-v_n\,\mathbf{s}^{\mathrm{H}}\mathbf{f}_n)\cdot\mathrm{Re}(\mathbf{f}_n^{\mathrm{H}}(\mathbf{z}_n-\bar{\mathbf{z}}_n+\bar{x}_n\mathbf{s}))$,

QPSK:
$L_e(c_{n,1})=\sqrt{8}/(1-v_n\,\mathbf{s}^{\mathrm{H}}\mathbf{f}_n)\cdot\mathrm{Re}(\mathbf{f}_n^{\mathrm{H}}(\mathbf{z}_n-\bar{\mathbf{z}}_n+\bar{x}_n\mathbf{s}))$,
$L_e(c_{n,2})=\sqrt{8}/(1-v_n\,\mathbf{s}^{\mathrm{H}}\mathbf{f}_n)\cdot\mathrm{Im}(\mathbf{f}_n^{\mathrm{H}}(\mathbf{z}_n-\bar{\mathbf{z}}_n+\bar{x}_n\mathbf{s}))$,

8-PSK:
$q_i=-2/(1-v_n\,\mathbf{s}^{\mathrm{H}}\mathbf{f}_n)\cdot\mathrm{Re}(\mathbf{f}_n^{\mathrm{H}}(\mathbf{z}_n-\bar{\mathbf{z}}_n+\bar{x}_n\mathbf{s})\alpha_i^*),\ \ i=1,\cdots,8$,
$l_j=L(c_{n,j})/2,\ \ j=1,2,3$,
$L_e(c_{n,1})=\ln\dfrac{e^{q_1+l_2+l_3}+e^{q_3-l_2+l_3}+e^{q_5+l_2-l_3}+e^{q_7-l_2-l_3}}{e^{q_2+l_2+l_3}+e^{q_4-l_2+l_3}+e^{q_6+l_2-l_3}+e^{q_8-l_2-l_3}}$,
$L_e(c_{n,2})=\ln\dfrac{e^{q_1+l_1+l_3}+e^{q_2-l_1+l_3}+e^{q_5+l_1-l_3}+e^{q_6-l_1-l_3}}{e^{q_3+l_1+l_3}+e^{q_4-l_1+l_3}+e^{q_7+l_1-l_3}+e^{q_8-l_1-l_3}}$,
$L_e(c_{n,3})=\ln\dfrac{e^{q_1+l_1+l_2}+e^{q_2-l_1+l_2}+e^{q_3+l_1-l_2}+e^{q_4-l_1-l_2}}{e^{q_5+l_1+l_2}+e^{q_6-l_1+l_2}+e^{q_7+l_1-l_2}+e^{q_8-l_1-l_2}}$.

where we use the relationship that the inverse of a Hermitian matrix is also Hermitian. The derivation of an efficient algorithm arises from noting that the submatrices $\boldsymbol{\Sigma}_{\mathrm{P}}$ and $\boldsymbol{\Sigma}_{\mathrm{N}}$ are identical.

$$\boldsymbol{\Sigma}_{\mathrm{P}} = \boldsymbol{\Sigma}_{\mathrm{N}} = \sigma_w^2 \mathbf{I}_{N-1} + \mathbf{H}' \mathrm{diag}[v_{n-M-N_2+2} \cdots v_{n+N_1}] \mathbf{H}'^{\mathrm{H}}$$

where $\mathbf{H}'$ is a $(N-1)\times(N+M-2)$ channel convolution matrix. Based on this fact, the recursive algorithm computes first $\boldsymbol{\Sigma}_{\mathrm{P}}^{-1}$ from $\mathbf{U}_n$, sets $\boldsymbol{\Sigma}_{\mathrm{P}}^{-1} = \boldsymbol{\Sigma}_{\mathrm{N}}^{-1}$, and computes $\mathbf{U}_{n+1}$ from $\boldsymbol{\Sigma}_{\mathrm{N}}^{-1}$.

The inverse $\boldsymbol{\Sigma}_{\mathrm{P}}^{-1}$ of the submatrix $\boldsymbol{\Sigma}_{\mathrm{P}}$ of $\boldsymbol{\Sigma}_n$ is expressed in terms of components of $\mathbf{U}_n$ by solving $\boldsymbol{\Sigma}_n \mathbf{U}_n = \mathbf{I}_N$ using (9) and (10):

$$\begin{aligned} \boldsymbol{\Sigma}_{\mathrm{P}} \mathbf{U}_{\mathrm{P}} + \boldsymbol{\sigma}_{\mathrm{P}} \mathbf{u}_{\mathrm{P}}^{\mathrm{H}} &= \mathbf{I}_{N-1} \\ \boldsymbol{\Sigma}_{\mathrm{P}} \mathbf{u}_{\mathrm{P}} + \boldsymbol{\sigma}_{\mathrm{P}} u_{\mathrm{P}} &= \mathbf{0}_{N-1} \\ \rightarrow\ \boldsymbol{\Sigma}_{\mathrm{P}}^{-1} &= \mathbf{U}_{\mathrm{P}} - \mathbf{u}_{\mathrm{P}} \mathbf{u}_{\mathrm{P}}^{\mathrm{H}} / u_{\mathrm{P}}. \end{aligned} \tag{11}$$

By solving $\boldsymbol{\Sigma}_{n+1} \mathbf{U}_{n+1} = \mathbf{I}_N$, we express $\mathbf{U}_{\mathrm{N}}$, $\mathbf{u}_{\mathrm{N}}$, and u_{N} in terms of $\boldsymbol{\Sigma}_{\mathrm{N}}$, $\boldsymbol{\sigma}_{\mathrm{N}}$, and σ_{N}:

$$\begin{aligned} \boldsymbol{\sigma}'_{\mathrm{N}} &\triangleq \boldsymbol{\Sigma}_{\mathrm{N}}^{-1} \boldsymbol{\sigma}_{\mathrm{N}} \\ u_{\mathrm{N}} &= 1/(\sigma_{\mathrm{N}} - \boldsymbol{\sigma}_{\mathrm{N}}^{\mathrm{H}} \boldsymbol{\sigma}'_{\mathrm{N}}) \\ \mathbf{u}_{\mathrm{N}} &= -u_{\mathrm{N}}\, \boldsymbol{\sigma}'_{\mathrm{N}} \\ \mathbf{U}_{\mathrm{N}} &= \boldsymbol{\Sigma}_{\mathrm{N}}^{-1} + u_{\mathrm{N}}\, \boldsymbol{\sigma}'_{\mathrm{N}} \boldsymbol{\sigma}'^{\mathrm{H}}_{\mathrm{N}} \end{aligned}$$

where we ordered the equations to optimize the computation by using the intermediate vector $\boldsymbol{\sigma}'_{\mathrm{N}}$. The matrix $\boldsymbol{\Sigma}_{\mathrm{N}}^{-1}$ is equal to $\boldsymbol{\Sigma}_{\mathrm{P}}^{-1}$ and the quantities $\boldsymbol{\sigma}_{\mathrm{N}}$ and σ_{N} are computed using (3) and (9):

$$\begin{bmatrix} \boldsymbol{\sigma}_{\mathrm{N}} \\ \sigma_{\mathrm{N}} \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{N-1} \\ \sigma_w^2 \end{bmatrix} + \mathbf{H} \mathbf{V}_{n+1} \mathbf{H}^{\mathrm{H}} \begin{bmatrix} \mathbf{0}_{N-1} \\ 1 \end{bmatrix}. \tag{12}$$

Assembling $\mathbf{U}_{n+1}$ from $\mathbf{U}_{\mathrm{N}}$, $\mathbf{u}_{\mathrm{N}}$, and u_{N} completes the recursive algorithm. The LLRs $L_e(c_{n+1,\,j})$, $j = 1, 2, \cdots, Q$ are given by (8) after $\mathbf{f}_{n+1} = \boldsymbol{\Sigma}_{n+1}^{-1} \mathbf{s}$ is computed.

To bootstrap the time-recursive update algorithm, an initialization of $\mathbf{U}_n$ at the starting time step $n = 1$ is required, e.g., by computing $\mathbf{U}_1 = (\sigma_w^2 \mathbf{I}_N + \mathbf{H} \mathbf{V}_1 \mathbf{H}^{\mathrm{H}})^{-1}$. This operation can be trivial when the block of transmitted symbols x_n starts with a preamble of at least $N + M$ symbols known to the receiver, yielding $\mathbf{V}_1 = 0$, and therefore, $\mathbf{U}_1 = \sigma_w^{-2} \mathbf{I}_N$.

The complete algorithm processing a sequence of received symbols z_n and *a priori* information $L(c_{n,\,j})$ to output $L \cdot Q$ LLRs $L_e(c_{n,\,j})$ is summarized in Table IV.

B. Low-Complexity Approximate MMSE Equalization

To further reduce the computational burden, we seek filter coefficients $\mathbf{f} \triangleq [f_{N_2}^* f_{N_2-1}^* \cdots f_{-N_1}^*]^{\mathrm{T}}$, which are not varying with n. Instead of using (5) to obtain the estimates $\hat{x}_n$, which requires computation of $(\boldsymbol{\Sigma}_n + (1 - v_n)\, \mathbf{s}\,\mathbf{s}^{\mathrm{H}})^{-1} \mathbf{s}$ for each time step n, we propose a low-complexity approximate solution to MMSE equalization, where the coefficients are computed using the time average of the expression $\boldsymbol{\Sigma}_n + (1 - v_n)\, \mathbf{s}\,\mathbf{s}^{\mathrm{H}}$:

$$\begin{aligned} \mathbf{f}' &\triangleq \left(\frac{1}{L} \sum_{n=1}^{L} \boldsymbol{\Sigma}_n + (1 - v_n)\, \mathbf{s}\,\mathbf{s}^{\mathrm{H}} \gamma \right)^{-1} \mathbf{s} \\ &= \left(\sigma_w^2 \mathbf{I}_N + \mathbf{H} \overline{\mathbf{V}} \mathbf{H}^{\mathrm{H}} \gamma\gamma + (1 - \overline{v})\, \mathbf{s}\,\mathbf{s}^{\mathrm{H}} \right)^{-1} \mathbf{s} \end{aligned}$$

TABLE IV
SISO EQUALIZER ALGORITHM: MMSE SOLUTION

INPUT:
- signal constellation $\mathcal{S} = \{\alpha_1, \cdots, \alpha_{2^Q}\}$,
- estimator filter parameters N_1 and N_2,
- channel characteristics h_k, $k = 0, \cdots, M-1$, and σ_w^2,
- received symbols z_n, $n = 1 - N_2, \cdots, L + N_1$,
- a-p. inf. $L(c_{n,j})$, $n = 1 - N_2 - M + 1, \cdots, L + N_1$, $j = 1, \cdots, Q$,

INITIALIZATION:
- define variables $\mathbf{f} = \mathbf{0}_N$, $\tilde{\mathbf{U}} = \mathbf{0}_{N\times N}$, $\mathbf{U} = \mathbf{0}_{(N-1)\times(N-1)}$, $\mathbf{u} = \mathbf{u}' = \mathbf{0}_{N-1}$, $x = u = \mu = \varrho_i = 0$, $i = 1, \cdots, 2^Q$,
- compute $\bar{x}_n$ and v_n, $n = 1 - N_2 - M + 1, \cdots, L + N_1$,
- compute $\bar{z}_n$, $n = 1 - N_2, \cdots, L + N_1$,
- compute $\tilde{\mathbf{U}} = (\sigma_w^2 \mathbf{I}_N + \mathbf{H} \mathbf{V}_1 \mathbf{H}^{\mathrm{H}})^{-1}$,

EQUALIZATION ALGORITHM:

FOR $n = 1$ TO L DO

$\mathbf{f} = \tilde{\mathbf{U}}\, \mathbf{s}$,
$\mu = \mathbf{f}^{\mathrm{H}}\, \mathbf{s}$,
$x = \mathbf{f}^{\mathrm{H}} (\mathbf{z}_n - \bar{\mathbf{z}}_n) + \bar{x}_n\, \mu$,

FOR $i = 1$ TO 2^Q DO

$\varrho_i = |x - \alpha_i\, \mu|^2 / (\mu - \mu^2)$,

END

FOR $j = 1$ TO Q DO

$$L_e(c_{n,j}) = \ln \frac{\displaystyle\sum_{\forall \mathbf{s}_i : s_{i,j} = 0} \exp(-\varrho_i + \sum_{\forall j' : j' \neq j} \tilde{s}_{i,j'} L(c_{n,j'})/2)}{\displaystyle\sum_{\forall \mathbf{s}_i : s_{i,j} = 1} \exp(-\varrho_i + \sum_{\forall j' : j' \neq j} \tilde{s}_{i,j'} L(c_{n,j'})/2)}.$$

END

IF $n < L$ THEN

$\begin{bmatrix} \mathbf{U} & \mathbf{u} \\ \mathbf{u}^{\mathrm{H}} & u \end{bmatrix} = \tilde{\mathbf{U}}$,
$\mathbf{U} = \mathbf{U} - \mathbf{u}\, \mathbf{u}^{\mathrm{H}} / u$,
$\begin{bmatrix} \mathbf{u} \\ u \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{N-1} \\ \sigma_w^2 \end{bmatrix} + \mathbf{H} \mathbf{V}_{n+1} \mathbf{H}^{\mathrm{H}} \begin{bmatrix} \mathbf{0}_{N-1} \\ 1 \end{bmatrix}$,
$\mathbf{u}' = \mathbf{U}\, \mathbf{u}$,
$u = 1/(u - \mathbf{u}^{\mathrm{H}} \mathbf{u}')$,
$\mathbf{u} = -u\, \mathbf{u}'$,
$\mathbf{U} = \mathbf{U} + u\, \mathbf{u}'\, \mathbf{u}'^{\mathrm{H}}$,
$\tilde{\mathbf{U}} = \begin{bmatrix} u & \mathbf{u}^{\mathrm{H}} \\ \mathbf{u} & \mathbf{U} \end{bmatrix}$,

END

END

where $\overline{\mathbf{V}} \triangleq (1/L) \sum_{n=1}^{L} \mathbf{V}_n$, and $\overline{v} \triangleq (1/L) \sum_{n=1}^{L} v_n$. Similar to (5), the estimates $\hat{x}_n$ are given by

$$\hat{x}_n = \mathbf{f}'^{\mathrm{H}} (\mathbf{z}_n - \overline{\mathbf{z}}_n + \overline{x}_n\, \mathbf{s}). \tag{13}$$

We call this the low-complexity (LC) solution. Even though this approach is *ad hoc*, we justify its use by referring to the analytical results in Section IV and the simulation results in Section V. Defining the vector

$$\mathbf{f} \triangleq \left(\sigma_w^2 \mathbf{I}_N + \mathbf{H} \overline{\mathbf{V}} \mathbf{H}^{\mathrm{H}} \right)^{-1} \mathbf{s}$$

$\mathbf{f}'$ is expressed in terms of $\mathbf{f}$ as it was done for $\mathbf{f}'_n$ and $\mathbf{f}_n$ in Section III-A:

$$\mathbf{f}' = (1 + (1 - \overline{v})\, \mathbf{f}^{\mathrm{H}} \mathbf{s})^{-1} \mathbf{f}. \tag{14}$$

The mean $\mu_{n,i}$ and the variance $\sigma^2_{n,i}$ of $\hat{x}_n$ are accordingly

$$\begin{aligned}\mu_{n,i} &= K \cdot \mathbf{f}^{\mathrm{H}}(\mathrm{E}(\mathbf{z}_n | x_n = \alpha_i) - \overline{\mathbf{z}}_n + \overline{x}_n \mathbf{s}) = K \cdot \alpha_i \cdot \mathbf{f}^{\mathrm{H}}\mathbf{s} \\ \sigma^2_{n,i} &= K^2 \cdot \mathbf{f}^{\mathrm{H}}\mathrm{Cov}(\mathbf{z}_n, \mathbf{z}_n | x_n = \alpha_i)\mathbf{f} \\ &= K^2 \cdot \mathbf{f}^{\mathrm{H}}(\mathbf{\Sigma}_n - v_n \mathbf{s}\,\mathbf{s}^{\mathrm{H}})\mathbf{f}\end{aligned}$$

where $K \triangleq (1 + (1 - \overline{v})\,\mathbf{s}^{\mathrm{H}}\mathbf{f})^{-1}$, yielding

$$\varrho_{n,i} = \frac{|\mathbf{f}^{\mathrm{H}}(\mathbf{z}_n - \overline{\mathbf{z}}_n + \overline{x}_n \mathbf{s}) - \alpha_i \mathbf{f}^{\mathrm{H}}\mathbf{s}|^2}{\mathbf{f}^{\mathrm{H}}(\sigma_w^2 \mathbf{I}_N + \mathbf{H}\mathbf{V}_n\mathbf{H}^{\mathrm{H}} - v_n \mathbf{s}\,\mathbf{s}^{\mathrm{H}})\,\mathbf{f}}.$$

The LLRs $L_e(c_{n,j})$ are computed as in (8).

The term $\overline{\mathbf{V}}$ can be approximated by $\overline{v}\,\mathbf{I}_{N+M-1}$, which we found does not degrade the performance of the SISO equalizer substantially, in particular for large L. This simplifies the computation of $\mathbf{f}$ to

$$\hat{\mathbf{f}} \triangleq \left(\sigma_w^2 \mathbf{I}_N + \overline{v}\,\mathbf{H}\mathbf{H}^{\mathrm{H}}\right)^{-1}\mathbf{s}.$$

The same can be said about approximating $\varrho_{n,i}$ with

$$\begin{aligned}&\frac{|\mathbf{f}^{\mathrm{H}}(\mathbf{z}_n - \overline{\mathbf{z}}_n + \overline{x}_n \mathbf{s}) - \alpha_i \mathbf{f}^{\mathrm{H}}\mathbf{s}|^2}{\mathbf{f}^{\mathrm{H}}\left(\sigma_w^2 \mathbf{I}_N + \mathbf{H}\overline{\mathbf{V}}\mathbf{H}^{\mathrm{H}} - \overline{v}\,\mathbf{s}\,\mathbf{s}^{\mathrm{H}}\right)\mathbf{f}} \\ &\quad\approx \frac{|\hat{\mathbf{f}}^{\mathrm{H}}(\mathbf{z}_n - \overline{\mathbf{z}}_n + \overline{x}_n \mathbf{s}) - \alpha_i \hat{\mathbf{f}}^{\mathrm{H}}\mathbf{s}|^2}{\hat{\mathbf{f}}^{\mathrm{H}}(\sigma_w^2 \mathbf{I}_N + \overline{v}\,(\mathbf{H}\mathbf{H}^{\mathrm{H}} - \mathbf{s}\,\mathbf{s}^{\mathrm{H}}))\,\hat{\mathbf{f}}} \\ &\quad= \frac{|\hat{\mathbf{f}}^{\mathrm{H}}(\mathbf{z}_n - \overline{\mathbf{z}}_n + \overline{x}_n \mathbf{s}) - \alpha_i \hat{\mathbf{f}}^{\mathrm{H}}\mathbf{s}|^2}{\hat{\mathbf{f}}^{\mathrm{H}}\mathbf{s}\,(1 - \mathbf{s}^{\mathrm{H}}\hat{\mathbf{f}})}.\end{aligned}$$

The MMSE and the LC solution coincide whenever all v_n are constant, yielding $v_n = \overline{v}$, e.g., if $|L(c_{n,j})| = \text{const}$, for all n, j. Table V shows how $L_e(c_{n,j})$, $j = 1, 2, \cdots, Q$ is computed from the estimator filter output for the symbol alphabets considered using the exact $\varrho_{n,i}$ with the coefficient vector $\hat{\mathbf{f}}$.

The complete algorithm processing a sequence of received symbols z_n and *a priori* information $L(c_{n,j})$ to output $L \cdot Q$ LLRs $L_e(c_{n,j})$ is summarized in Table VI.

TABLE V
CONVERSION FROM THE LINEAR FILTER OUTPUT TO $L_e(c_{n,j})$ FOR THE LC SOLUTION

BPSK:

$\mu = \hat{\mathbf{f}}^{\mathrm{H}}\mathbf{s},\ \mathbf{p} = \mathbf{H}^{\mathrm{H}}\hat{\mathbf{f}},\ K_f = \hat{\mathbf{f}}^{\mathrm{H}}\hat{\mathbf{f}},\ x = \hat{\mathbf{f}}^{\mathrm{H}}(\mathbf{z}_n - \bar{\mathbf{z}}_n + \bar{x}_n\mathbf{s}),$

$L_e(c_{n,1}) = 4 \cdot \mu \cdot \mathrm{Re}(x)/(\sigma_w^2 K_f + \mathbf{p}^{\mathrm{H}}\mathbf{V}_n\mathbf{p} - v_n\,|\mu|^2),$

QPSK:

$\mu = \hat{\mathbf{f}}^{\mathrm{H}}\mathbf{s},\ \mathbf{p} = \mathbf{H}^{\mathrm{H}}\hat{\mathbf{f}},\ K_f = \hat{\mathbf{f}}^{\mathrm{H}}\hat{\mathbf{f}},\ x = \hat{\mathbf{f}}^{\mathrm{H}}(\mathbf{z}_n - \bar{\mathbf{z}}_n + \bar{x}_n\mathbf{s}),$

$L_e(c_{n,1}) = \sqrt{8} \cdot \mu \cdot \mathrm{Re}(x)/(\sigma_w^2 K_f + \mathbf{p}^{\mathrm{H}}\mathbf{V}_n\mathbf{p} - v_n\,|\mu|^2),$

$L_e(c_{n,2}) = \sqrt{8} \cdot \mu \cdot \mathrm{Im}(x)/(\sigma_w^2 K_f + \mathbf{p}^{\mathrm{H}}\mathbf{V}_n\mathbf{p} - v_n\,|\mu|^2),$

8-PSK:

$\mu = \hat{\mathbf{f}}^{\mathrm{H}}\mathbf{s},\ \mathbf{p} = \mathbf{H}^{\mathrm{H}}\hat{\mathbf{f}},\ K_f = \hat{\mathbf{f}}^{\mathrm{H}}\hat{\mathbf{f}},\ x = \hat{\mathbf{f}}^{\mathrm{H}}(\mathbf{z}_n - \bar{\mathbf{z}}_n + \bar{x}_n\mathbf{s}),$

$q_i = -2 \cdot \mu \cdot \mathrm{Re}(x\,\alpha_i^*)/(\sigma_w^2 K_f + \mathbf{p}^{\mathrm{H}}\mathbf{V}_n\mathbf{p} - v_n\,|\mu|^2)),\quad i = 1, \cdots, 8,$

$l_j = L(c_{n,j})/2,\quad j = 1, 2, 3,$

$$L_e(c_{n,1}) = \ln\frac{e^{q_1+l_2+l_3} + e^{q_3-l_2+l_3} + e^{q_5+l_2-l_3} + e^{q_7-l_2-l_3}}{e^{q_2+l_2+l_3} + e^{q_4-l_2+l_3} + e^{q_6+l_2-l_3} + e^{q_8-l_2-l_3}},$$
$$L_e(c_{n,2}) = \ln\frac{e^{q_1+l_1+l_3} + e^{q_2-l_1+l_3} + e^{q_5+l_1-l_3} + e^{q_6-l_1-l_3}}{e^{q_3+l_1+l_3} + e^{q_4-l_1+l_3} + e^{q_7+l_1-l_3} + e^{q_8-l_1-l_3}},$$
$$L_e(c_{n,3}) = \ln\frac{e^{q_1+l_1+l_2} + e^{q_2-l_1+l_2} + e^{q_3+l_1-l_2} + e^{q_4-l_1-l_2}}{e^{q_5+l_1+l_2} + e^{q_6-l_1+l_2} + e^{q_7+l_1-l_2} + e^{q_8-l_1-l_2}}$$

TABLE VI
SISO EQUALIZER ALGORITHM: LC SOLUTION

INPUT:
- signal constellation $\mathcal{S} = \{\alpha_1, \cdots, \alpha_{2^Q}\}$,
- estimator filter parameters N_1 and N_2,
- channel characteristics h_k, $k = 0, \cdots, M-1$, and σ_w^2,
- received symbols z_n, $n = 1 - N_2, \cdots, L + N_1$,
- a-p. inf. $L(c_{n,j})$, $n = 1 - N_2 - M + 1, \cdots, L + N_1$, $j = 1, \cdots, Q$,

INITIALIZATION:
- define variables $\mathbf{f} = \mathbf{p} = \mathbf{0}_N$,
  $x = \bar{v} = K = \mu = \varrho_i = 0$, $i = 1, \cdots, 2^Q$,
- compute $\bar{x}_n$ and v_n, $n = 1 - N_2 - M + 1, \cdots, L + N_1$,
- compute $\bar{z}_n$, $n = 1 - N_2, \cdots, L + N_1$,
- compute

$$\begin{aligned}\bar{v} &= \tfrac{1}{L}\textstyle\sum_{n=1}^{L} v_n, \\ \mathbf{f} &= (\sigma_w^2 \mathbf{I}_N + \bar{v}\,\mathbf{H}\mathbf{H}^{\mathrm{H}})^{-1}\mathbf{s}, \\ \mathbf{p} &= \mathbf{H}^{\mathrm{H}}\mathbf{f}, \\ \mu &= \mathbf{f}^{\mathrm{H}}\mathbf{s}, \\ K &= \sigma_w^2 \mathbf{f}^{\mathrm{H}}\mathbf{f},\end{aligned}$$

EQUALIZATION ALGORITHM:

FOR $n = 1$ TO L DO
 $x = \mathbf{f}^{\mathrm{H}}(\mathbf{z}_n - \bar{\mathbf{z}}_n) + \bar{x}_n\,\mu$,
  FOR $i = 1$ TO 2^Q DO
   $\varrho_i = |x - \alpha_i\,\mu|^2/(K + \mathbf{p}^{\mathrm{H}}\mathbf{V}_n\mathbf{p} - v_n\mu^2)$,
  END
  FOR $j = 1$ TO Q DO

$$L_e(c_{n,j}) = \ln\frac{\displaystyle\sum_{\forall \mathbf{s}_i : s_{i,j}=0} \exp(-\varrho_i + \sum_{\forall j' : j' \neq j} \tilde{s}_{i,j'}\,L(c_{n,j'})/2)}{\displaystyle\sum_{\forall \mathbf{s}_i : s_{i,j}=1} \exp(-\varrho_i + \sum_{\forall j' : j' \neq j} \tilde{s}_{i,j'}\,L(c_{n,j'})/2)}.$$

  END
END

IV. ANALYSIS

We observe the time averaged MSE $J_n \triangleq \mathrm{E}(|x_n - \hat{x}_n|^2)$

$$\overline{J} \triangleq \frac{1}{L}\sum_{n=1}^{L} J_n = \frac{1}{L}\sum_{n=1}^{L} \mathrm{E}(|x_n - \hat{x}_n|^2)$$

of a length L sequence of estimates $\hat{x}_n$ to show that the incorporation of *a priori* information improves the performance of the equalizer. The MMSE and LC solutions, yielding the average costs $\overline{J}_{\mathrm{MMSE}}$ and $\overline{J}_{\mathrm{LC}}$, respectively, are compared with usual MMSE equalization without the use of *a priori* information. For the latter approach, the parameter vector $\mathbf{a}_n$ is computed, assuming that $L(c_{n,j}) = 0$ for all n, j, yielding $\mathbf{a}_n = \mathbf{f}_{\mathrm{NA}}$, as shown at the end of Section III-A. In order to have an unbiased estimator, we set b_n to $\mathrm{E}(x_n) - \mathbf{a}_n^{\mathrm{H}}\mathrm{E}(\mathbf{z}_n)$, yielding

$$\hat{x}_n = \overline{x}_n + \mathbf{f}_{\mathrm{NA}}^{\mathrm{H}}(\mathbf{z}_n - \overline{\mathbf{z}}_n).$$

Imposing the independence constraint for computing $\overline{x}_n$ ($\overline{x}_n = 0$, $v_n = 1$) results in

$$\hat{x}_n = \mathbf{f}_{\mathrm{NA}}^{\mathrm{H}}(\mathbf{z}_n - \overline{\mathbf{z}}_n + \overline{x}_n\,\mathbf{s}). \tag{15}$$

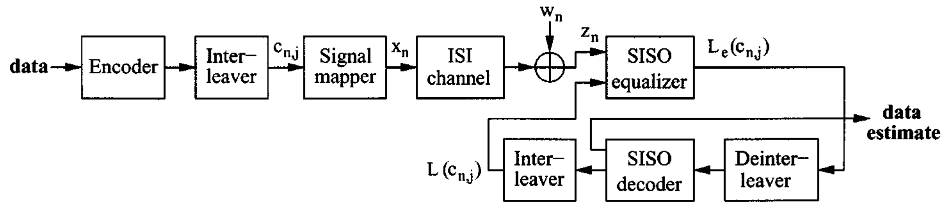


Fig. 2. Application of equalization using *a priori* information: turbo equalization.

We call this the no *a priori* information (NA) solution. Using $\mathbf{z}'_n = \mathbf{z}_n - \overline{\mathbf{z}}_n + \overline{x}_n\mathbf{s}$, the MSE for all three algorithms [MMSE (5), LC (13), and NA (15) solution] is given by

$$\begin{aligned} J_n &= \mathrm{E}(|x_n - \mathbf{a}_n^{\mathrm{H}}\mathbf{z}'_n|^2) \\ &= \mathrm{E}(|x_n|^2) - 2\mathrm{Re}(\mathbf{a}_n^{\mathrm{H}}\mathrm{E}(\mathbf{z}'_n x_n^*)) + \mathbf{a}_n^{\mathrm{H}}\mathrm{E}(\mathbf{z}'_n\mathbf{z}'_n)\mathbf{a}_n \\ &= \mathrm{E}(|x_n|^2)(1 - 2\,\mathbf{a}_n^{\mathrm{H}}\mathbf{s}) + \mathbf{a}_n^{\mathrm{H}}(\mathbf{\Sigma}_n + |\overline{x}_n|^2\mathbf{s}\,\mathbf{s}^{\mathrm{H}})\mathbf{a}_n \end{aligned} \tag{16}$$

where $\mathbf{a}_n = \mathbf{f}'_n$ for the MMSE, $\mathbf{a}_n = \mathbf{f}'$ for the LC, and $\mathbf{a}_n = \mathbf{f}_{\mathrm{NA}}$ for the NA solution. We also used that $\mathbf{a}_n^{\mathrm{H}}\mathbf{s}$ is always real and that $\mathrm{E}(|x_n|^2) = v_n + |\overline{x}_n|^2$. The average MSE of the NA solution is denoted $\overline{J}_{\mathrm{NA}}$. We show that for any *a priori* information constellation

$$\overline{J}_{\mathrm{MMSE}} \le \overline{J}_{\mathrm{LC}} \le \overline{J}_{\mathrm{NA}}$$

holds. From (5) and (16), it follows that for the MMSE solution

$$J_n = (1 - \mathbf{f}'^{\mathrm{H}}_n\mathbf{s})\left(\mathbf{f}'^{\mathrm{H}}_n\mathbf{s}(1 - \mathrm{E}(|x_n|^2)) + \mathrm{E}(|x_n|^2)\right).$$

The average MSE $\overline{J}_{\mathrm{MMSE}}$ is given by

$$\overline{J}_{\mathrm{MMSE}} = \left(1 - \overline{\mathbf{f}'^{\mathrm{H}}_n\mathbf{s}}\right)\left(\overline{\mathbf{f}'^{\mathrm{H}}_n\mathbf{s}}(1 - \overline{m}) + \overline{m}\right)$$

where $\overline{m} \triangleq (1/L)\sum_{n=1}^{L}\mathrm{E}(|x_n|^2)$ and $\overline{\mathbf{f}'^{\mathrm{H}}_n\mathbf{s}} \triangleq (1/L)\sum_{n=1}^{L}\mathbf{f}'^{\mathrm{H}}_n\mathbf{s}$. Since the transmitted symbols x_n are assumed to be equally likely $\alpha_1, \alpha_2, \cdots, \alpha_{2^Q}$ in the sequence, the average symbol energy $\overline{m}$ approaches 1 for a sufficiently large L, and thus

$$\overline{J}_{\mathrm{MMSE}} = 1 - \overline{\mathbf{f}'^{\mathrm{H}}_n\mathbf{s}} = 1 - \frac{1}{L}\sum_{n=1}^{L}\mathbf{s}^{\mathrm{H}}(\mathbf{\Sigma}_n + (1 - v_n)\mathbf{s}\,\mathbf{s}^{\mathrm{H}})^{-1}\mathbf{s}.$$

From (14) and (16), it follows that for the LC solution

$$\begin{aligned} \overline{J}_{\mathrm{LC}} &= \overline{m}(1 - 2\mathbf{f}'^{\mathrm{H}}\mathbf{s}) + \mathbf{f}'^{\mathrm{H}}\left(\sigma_w^2\mathbf{I}_N + \mathbf{H}\overline{\mathbf{V}}\mathbf{H}^{\mathrm{H}} + (\overline{m} - \overline{v})\mathbf{s}\mathbf{s}^{\mathrm{H}}\right)\mathbf{f}' \\ &= 1 - \mathbf{f}'^{\mathrm{H}}\mathbf{s} \\ &= 1 - \mathbf{s}^{\mathrm{H}}\left(\sigma_w^2\mathbf{I}_N + \mathbf{H}\overline{\mathbf{V}}\mathbf{H}^{\mathrm{H}} + (1 - \overline{v})\,\mathbf{s}\,\mathbf{s}^{\mathrm{H}}\right)^{-1}\mathbf{s}. \end{aligned}$$

Since the MMSE solution is optimal in the MMSE sense, we always have $\overline{J}_{\mathrm{MMSE}} \le \overline{J}_{\mathrm{LC}}$. Equality holds whenever all v_n are constant over n. From (15) and (16), it follows that for the NA solution

$$\begin{aligned} \overline{J}_{\mathrm{NA}} &= \overline{m}(1 - 2\mathbf{f}_{\mathrm{NA}}^{\mathrm{H}}\mathbf{s}) \\ &\quad + \mathbf{f}_{\mathrm{NA}}^{\mathrm{H}}\left(\sigma_w^2\mathbf{I}_N + \mathbf{H}\overline{\mathbf{V}}\mathbf{H}^{\mathrm{H}} + (\overline{m} - \overline{v})\mathbf{s}\mathbf{s}^{\mathrm{H}}\right)\mathbf{f}_{\mathrm{NA}} \\ &= 1 - 2\,\mathbf{f}_{\mathrm{NA}}^{\mathrm{H}}\mathbf{s} + \mathbf{f}_{\mathrm{NA}}^{\mathrm{H}}\left(\sigma_w^2\mathbf{I}_N + \mathbf{H}\overline{\mathbf{V}}\mathbf{H}^{\mathrm{H}} + (1 - \overline{v})\mathbf{s}\mathbf{s}^{\mathrm{H}}\right)\mathbf{f}_{\mathrm{NA}}. \end{aligned}$$

Using $\overline{\mathbf{\Sigma}} \triangleq (\sigma_w^2\mathbf{I}_N + \mathbf{H}\overline{\mathbf{V}}\mathbf{H}^{\mathrm{H}} + (1 - \overline{v})\,\mathbf{s}\,\mathbf{s}^{\mathrm{H}})$, the proof showing that $\overline{J}_{\mathrm{LC}} \le \overline{J}_{\mathrm{NA}}$ is

$$\begin{aligned} \overline{J}_{\mathrm{LC}} &\le \overline{J}_{\mathrm{NA}} \\ 1 - \mathbf{s}^{\mathrm{H}}\overline{\mathbf{\Sigma}}^{-1}\mathbf{s} &\le 1 - 2\,\mathbf{f}_{\mathrm{NA}}^{\mathrm{H}}\mathbf{s} + \mathbf{f}_{\mathrm{NA}}^{\mathrm{H}}\overline{\mathbf{\Sigma}}\,\mathbf{f}_{\mathrm{NA}} \\ 2\,\mathbf{f}_{\mathrm{NA}}^{\mathrm{H}}\mathbf{s} &\le \mathbf{s}^{\mathrm{H}}\overline{\mathbf{\Sigma}}^{-1}\mathbf{s} + \mathbf{f}_{\mathrm{NA}}^{\mathrm{H}}\overline{\mathbf{\Sigma}}\,\mathbf{f}_{\mathrm{NA}} \\ \mathbf{f}_{\mathrm{NA}}^{\mathrm{H}}\overline{\mathbf{\Sigma}}\overline{\mathbf{\Sigma}}^{-1}\mathbf{s} + \mathbf{s}^{\mathrm{H}}\overline{\mathbf{\Sigma}}^{-1}\overline{\mathbf{\Sigma}}\,\mathbf{f}_{\mathrm{NA}} &\le \mathbf{s}^{\mathrm{H}}\overline{\mathbf{\Sigma}}^{-1}\mathbf{s} + \mathbf{f}_{\mathrm{NA}}^{\mathrm{H}}\overline{\mathbf{\Sigma}}\overline{\mathbf{\Sigma}}^{-1}\overline{\mathbf{\Sigma}}\,\mathbf{f}_{\mathrm{NA}} \\ 0 &\le (\mathbf{s}^{\mathrm{H}} - \mathbf{f}_{\mathrm{NA}}^{\mathrm{H}}\overline{\mathbf{\Sigma}})\,\overline{\mathbf{\Sigma}}^{-1}(\mathbf{s} - \overline{\mathbf{\Sigma}}\,\mathbf{f}_{\mathrm{NA}}). \end{aligned}$$

Equality holds only for $\mathbf{s} = \overline{\mathbf{\Sigma}}\mathbf{f}_{\mathrm{NA}}$, which is true for $\overline{\mathbf{\Sigma}} = \sigma_w^2\mathbf{I}_N + \mathbf{H}\mathbf{H}^{\mathrm{H}}$, i.e., no *a priori* information is available, $L(c_{n,j}) = 0$, and $v_n = 1$, for all n, j.

V. Results

As mentioned in the introduction, the major motivation for developing algorithms for MMSE equalization using *a priori* information was to find low-complexity equalization techniques feasible to process and output soft information. Turbo equalization, or iterative equalization and decoding, is an application where such a scenario occurs. Fig. 2 depicts a system performing turbo equalization in the receiver [1], [32].

Binary data is encoded to code bits using a convolutional ECC, which are permuted with an interleaver to $L \cdot Q$ bits $c_{n,j}$. The interleaved code bits are mapped to L symbols x_n from $\mathcal{S}$. We chose a rate $R = 1/2$, memory 2 ECC with generator $(1 + D^2\ 1 + D + D^2)$, $L = 4048$, a random interleaver [15], and the 8-PSK symbol alphabet from Table I. The receiver performs turbo equalization, i.e., after an initial equalization step yielding L LLRs $L_e(c_{n,j})$, where $L(c_{n,j}) = 0$, for all n, j, decoding and equalization steps are repeated on the same received data while using the decoder feedback as *a priori* information $L(c_{n,j})$. The input to the decoder must be the difference $L(c_{n,j}|\hat{x}_n) - L(c_{n,j}) = L_e(c_{n,j})$ when a linear equalizer is used or $L(c_{n,j}|\mathbf{z}) - L(c_{n,j})$ when a MAP

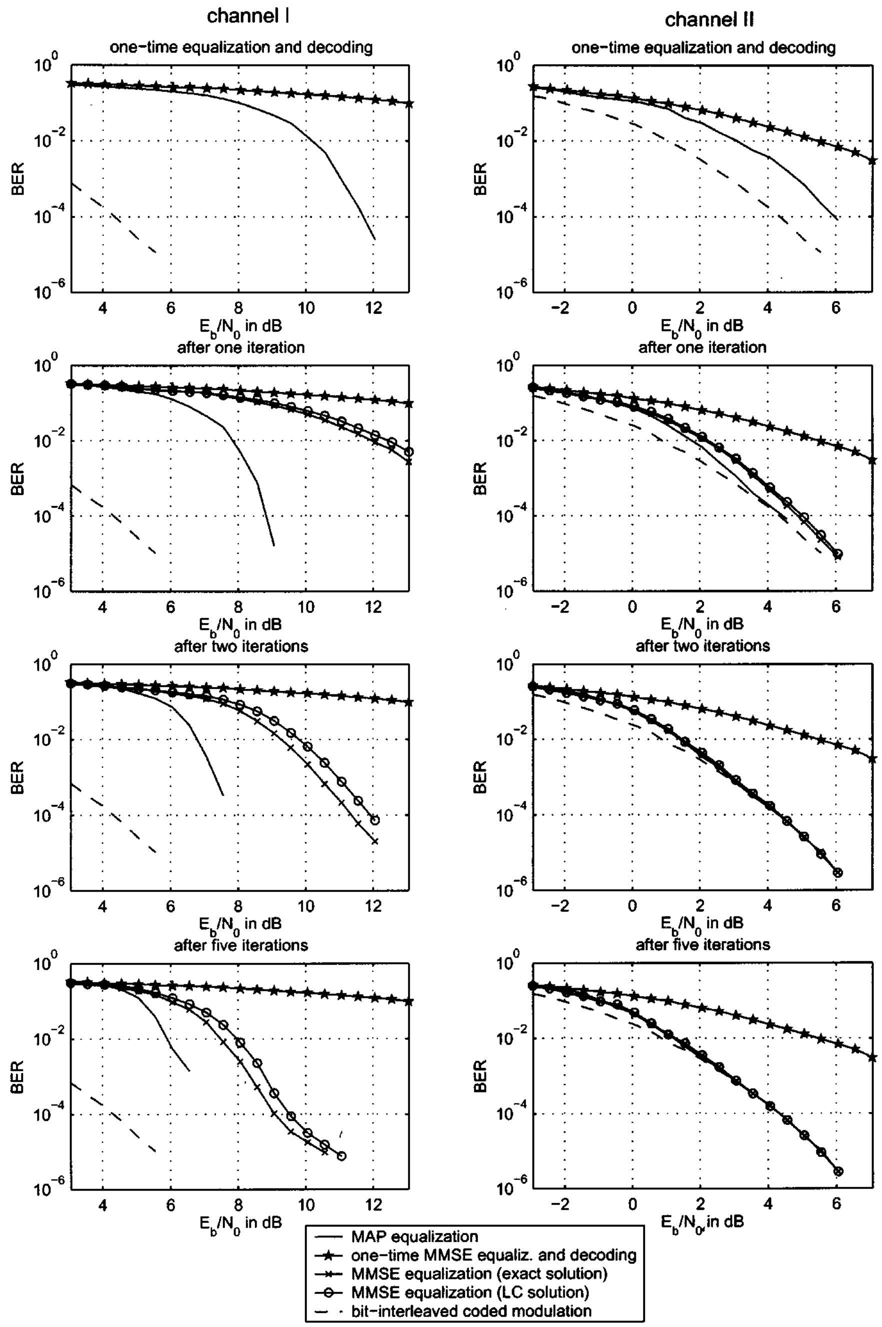


Fig. 3. BER performance of MMSE equalization using *a priori* information.

equalizer is used [1], [32]. We considered two ISI channels with the impulse responses

$$\begin{aligned} h_{\mathrm{I}}[n] =& 0.227\delta[n] + 0.46\delta[n-1] + 0.688\delta[n-2] \\ &+ 0.46\delta[n-3] + 0.227\delta[n-4] \\ h_{\mathrm{II}}[n] =& (2+0.4\imath)\delta[n] + (1.5+1.8\imath)\delta[n-1] + \delta[n-2] \\ &+ (1.2-1.3\imath)\delta[n-3] + (0.8+1.6\imath)\delta[n-4] \end{aligned}$$

causing severe and mild ISI, respectively, taken from [2] and [22]. The channel characteristics are precisely known to the receiver. The filter length parameters were set to $N_1 = 9$ and $N_2 = 5$. The noise variance σ_w^2 is determined according to the SNR

$$\frac{E_b}{N_0} \triangleq \frac{E_s}{N_0 R \log_2(|\mathcal{S}|)} = \frac{\mathrm{E}(|z_n|^2)}{N_0 Q R} = \frac{\sum_{k=0}^{M-1} |h_k|^2}{2\sigma_w^2 Q R}.$$

MAP equalization or MMSE equalization using the MMSE or the LC solution and MAP decoding are applied in the receiver.

Fig. 3 shows the BER performance with respect to the encoded information bits of the three considered receiver algorithms for both channels after initial equalization and decoding and after one, two, or five iterations. References provided are the BER of one-time MMSE equalization and decoding, where no *a priori* information is used, and the BER of bit-interleaved coded modulation (BICM) at the same SNR. The latter approach corresponds to a system in Fig. 2, where no ISI is present, i.e., $h[n] = \delta[n]$, and its performance is a lower bound for the systems investigated here.

For transmission over channel I, one-time MMSE equalization and decoding achieves a BER of 10^{-4} at 33 dB E_b/N_0. After five iterations, turbo equalization using the MMSE solution achieves that BER at 9 dB and the LC solution at 9.5 dB E_b/N_0. The gain is thus 24 or 23.5 dB, respectively. Using a BER-optimal MAP equalizer yields an extra 2-dB gain. The BICM system representing the ISI-free case achieves a BER of 10^{-4} at 4.3 dB. The BICM lower bound is not attained by the systems transmitting over channel I. This likely stems from the relatively short block length L, which limits the performance improvement over the iterations, as observed in [32].

For transmission over channel II, the system using one-time MMSE equalization and decoding achieves a BER of 10^{-4} at 14 dB E_b/N_0. After five iterations, the MMSE and the LC solution achieve that BER at 4.3 dB E_b/N_0. The gain is thus 9.7 dB, which is not increased using MAP equalization. The BICM lower bound is attained for E_b/N_0 larger then 2 dB since the BICM system achieves no better performance.

VI. Conclusion and Discussion

Several algorithms were proposed for linear MMSE equalization of symbols disturbed during transmission over an ISI channel. The introduced equalizers improve their performance by incorporating *a priori* information on these symbols, which was shown by observing the average MSE of the symbol estimates. Two instances of such algorithms are proposed: an exact and a low-complexity approximate approach to MMSE equalization using *a priori* information. Turbo equalization is an application where such equalizers can be used successfully. Simulation results show that the performance improvement is substantial. Moreover, the exact and the approximate solution show similar performance, i.e., the performance degradation is small even for channels with severe ISI.

Encouraged by the promising performance results, further research could extend the proposed algorithms to scenarios with unknown channel characteristics, e.g., combined channel estimation and equalization using *a priori* information. The qualitative MSE analysis could be extended to a quantitative analysis for given *a priori* information distributions occurring, e.g., in turbo equalization.

References

[1] C. Douillard *et al.*, "Iterative correction of intersymbol interference: Turbo equalization," *Eur. Trans. Telecomm.*, vol. 6, pp. 507–511, Sept./Oct. 1995.

[2] J. Proakis, *Digital Communications*, 3rd ed. Singapore: McGraw-Hill, 1995.

[3] S. Haykin, *Communication Systems*, 3rd ed. New York: Wiley, 1994.

[4] S. Lin and J. J. Costello, *Error Control Coding*. Englewood Cliffs, NJ: Prentice-Hall, 1983.

[5] G. Forney, "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] L. R. Bahl *et al.*, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. Inform. Theory*, vol. IT-20, pp. 284–287, Mar 1974.

[7] Y. Li, B. Vucetic, and Y. Sato, "Optimum soft-output detection for channels with intersymbol interference," *IEEE Trans. Inform. Theory*, vol. 41, pp. 704–713, May 1995.

[8] W. Koch and A. Baier, "Optimum and sub-optimum detection of coded data disturbed by time varying intersymbol interference," *Proc. IEEE Global Telecomm. Conf.*, pp. 1679–1684, Dec. 1990.

[9] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo codes," in *Proc. IEEE Int. Conf. Commun.*, Geneva, Switzerland, May 1993.

[10] S. Benedetto *et al.*, "Serial concatenated trellis coded modulation with iterative decoding: Design and performance," in *Proc. IEEE Global Telecomm. Conf.*, Nov. 1997.

[11] S. Benedetto *et al.*, "Parallel concatenated trellis coded modulation," in *Proc. IEEE Int. Conf. Commun.*, June 1996.

[12] X. Wang and H. Poor, "Iterative (turbo) soft interference cancellation and decoding for coded CDMA," *IEEE Trans. Commun.*, vol. 47, pp. 1046–1061, July 1999.

[13] J. Hagenauer, "The turbo principle: Tutorial introduction and state of the art," in *Proc. Int. Symp. Turbo Codes*, Brest, France, Sept. 1997, pp. 1–11.

[14] C. Berrou and A. Glavieux, "Near optimum error correcting coding and decoding: Turbo codes," *IEEE Trans. Commun.*, vol. 44, pp. 1261–1271, Oct. 1996.

[15] C. Heegard and S. Wicker, *Turbo Coding*. Boston, MA: Kluwer, 1999.

[16] G. Bauch and V. Franz, "A comparison of soft-in/soft-out algorithms for 'Turbo detection'," in *Proc. Int. Conf. Telecomm.*, June 1998, pp. 259–263.

[17] A. Anastasopoulos and K. Chugg, "Iterative equalization/decoding for TCM for frequency-selective fading channels," in *Conf. Rec. 31st Asilomar Conf. Signals, Syst. Comput.*, vol. 1, Nov. 1997, pp. 177–181.

[18] J. Hagenauer and P. Hoeher, "A Viterbi algorithm with soft-decision outputs and its applications," in *Proc. IEEE Global Telecomm. Conf.*, 1989, pp. 1680–1686.

[19] D. Raphaeli and Y. Zarai, "Combined turbo equalization and turbo decoding," in *Proc. IEEE Global Telecomm. Conf.*, vol. 2, Nov. 1997, pp. 639–643.

[20] M. Toegel, W. Pusch, and H. Weinrichter, "Combined serially concatenated codes and turbo-equalization," in *Proc. 2nd Int. Symp. Turbo Codes*, Brest, France, Sep. 2000, pp. 375–378.

[21] S. Ariyavisitakul and Y. Li, "Joint coding and decision feedback equalization for broadband wireless channels," *IEEE J. Select. Areas Commun.*, vol. 16, pp. 1670–1678, Dec. 1998.

[22] A. Glavieux, C. Laot, and J. Labat, "Turbo equalization over a frequency selective channel," in *Proc. Int. Symp. Turbo Codes*, Brest, France, Sept. 1997, pp. 96–102.

[23] D. Raphaeli and A. Saguy, "Linear equalizers for turbo equalization: A new optimization criterion for determining the equalizer taps," in *Proc. 2nd Int. Symp. Turbo Codes*, Brest, France, Sept. 2000, pp. 371–374.

[24] Z. Wu and J. Cioffi, "Turbo decision aided equalization for magnetic recording channels," in *Proc. Global Telecomm. Conf.*, Dec. 1999, pp. 733–738.

[25] A. Berthet, R. Visoz, and P. Tortelier, "Sub-optimal turbo-detection for coded 8-PSK signals over ISI channels with application to EDGE advanced mobile system," in *Proc. IEEE Veh. Technol. Conf.*, Sept. 2000.

[26] H. Poor, *An Introduction to Signal, Detection and Estimation*, 2nd ed. New York: Springer-Verlag, 1994.

[27] S. Haykin, *Adaptive Filter Theory*, 3rd ed. Upper Saddle River, NJ: Prentice-Hall, 1996.

[28] P. Strobach, "Pure order recursive least-squares ladder algorithms," *IEEE tRans. Acoust., Speech, Signal Processing*, vol. ASSP-36, pp. 880–897, Aug. 1986.

[29] ——, "New forms of Levinson and Schur algorithms," *IEEE Signal Processing Mag.*, pp. 12–36, Jan. 1991.

[30] B. Friedlander, "Lattice methods for spectral estimation," *Proc. IEEE*, vol. 70, pp. 990–1017, Sept. 1982.

[31] ——, "Lattice filters for adaptive processing," *Proc. IEEE*, vol. 70, pp. 829–867, Aug. 1982.

[32] M. Tüchler, R. Koetter, and A. Singer, "Turbo equalization: Principles and new results," *IEEE Trans. Commun.*, to be published.

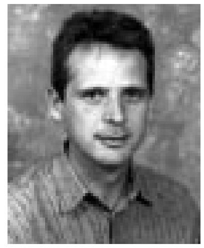
Michael Tüchler was born on October 6, 1975, in Hildburghausen, Germany. He studied electrical engineering from 1995 to 1997 at the Technical University Ilmenau, Ilmenau, Germany, and, from 1997 to 1998, at the Technical University Munich, Munich, Germany. He received the M.Sci. degree from the University of Illinois at Urbana-Champaign in 2000. Currently, he is pursuing the Ph.D. degree at the Institute for Communications Engineering, Technical University Munich.

In 2001, he was a visiting scientist at the IBM Zurich Research Laboratory, Zurich, Switzerland. His research interests include coding theory and equalization with application to wireless and data storage communication systems.

Andrew C. Singer (M'96) was born in Akron, OH, in 1967. He received the S.B., S.M., and Ph.D. degrees, all in electrical engineering and computer science, from the Massachusetts Institute of Technology, Cambridge, in 1990, 1992, and 1996, respectively.

Since 1998, he has been with the Faculty of the Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, where he is currently an Assistant Professor with the Electrical and Computer Engineering Department and a Research Assistant Professor with the Coordinated Science Laboratory. During 1996, he was a Postdoctoral Research Affiliate with the Research Laboratory of Electronics, MIT. From 1996 to 1998, he was a Research Scientist at Sanders, A Lockheed Martin Company, Manchester, NH. His research interests include statistical signal processing and communication, universal prediction and data compression, and machine learning.

Dr. Singer was a Hughes Aircraft Masters Fellow and was the recipient of the Harold L. Hazen Memorial Award for excellence in teaching in 1991. In 2000, he received the National Science Foundation CAREER Award, and in 2001, he received the Xerox Faculty Research Award. He is currently a member of the MIT Educational Council and of Eta Kappa Nu and Tau Beta Pi.



Ralf Koetter (M'95) was born on October 10, 1963, in Königstein, Germany. He received the Dipl.-Ing. degree from the Technical University Darmstadt, Darmstadt, Germany in 1990 and the Ph.D. degree from the Department of Electrical Engineering, Linköping University, Linköping, Sweden.

From 1996 to 1997, he was a visiting scientist at the IBM Almaden Research Lab., San Jose, CA. He was a Visiting Assistant Professor at the University of Illinois at Urbana-Champaign (UIUC) and Visiting Scientist at CNRS, Sophia Antipolis, France, from 1997 to 1998. He joined the Faculty of the UIUC in 1999 and is currently an Assistant Professor with the Coordinated Science Laboratory at the University. His research interests include coding and information theory and their application to communication systems.

Dr. Koetter served as associate editor for coding theory and techniques for the IEEE TRANSACTIONS ON COMMUNICATIONS from 1999 to 2001. In 2000, he started a term as associate editor for coding theory of the IEEE TRANSACTIONS ON INFORMATION THEORY. He received an IBM Invention Achievement Award in 1997, an NSF CAREER Award in 2000, and an IBM Partnership Award in 2001.