

The Finite-Length Multi-Input Multi-Output MMSE-DFE

Naofal Al-Dhahir, *Senior Member, IEEE*, and Ali H. Sayed, *Senior Member, IEEE*

Abstract—A new theoretical framework is introduced for analyzing the performance of a finite-length minimum-mean-square-error decision feedback equalizer (MMSE-DFE) in a multi-input multi-output (MIMO) environment. The framework includes transmit and receive diversity systems as special cases and quantifies the diversity performance improvement as a function of the number of transmit/receive antennas and equalizer taps. Fast and parallelizable algorithms for computing the finite-length MIMO MMSE-DFE are presented for three common multi-user detection scenarios.

Index Terms—Equalizers, FIR digital filters, mean square error methods, MIMO systems.

I. INTRODUCTION

IN MULTI-USER communication over linear, dispersive, and noisy channels, the received signal is composed of the sum of several transmitted signals corrupted by intersymbol interference (ISI), interuser interference (IUI), and noise. Examples include TDMA digital cellular systems with multiple transmit/receive antennas [4], [25], wideband asynchronous CDMA systems [3], where IUI is also known as multiple access interference (MAI), wideband transmission over digital subscriber lines (DSL) [1], where IUI takes the form of near-end and far-end crosstalk between adjacent twisted pairs, and high-density digital magnetic recording, where IUI is due to interference from adjacent tracks [15].

This work is motivated by the desire to increase the capacity of digital TDMA wireless networks by allowing multiple transmissions sharing the same time slot and frequency band and separating them spatio-temporally at the receiver. On the uplink, this corresponds to, for example, multiple synchronous co-channel and co-cell users, where each is equipped with single (or multiple) antennas transmitting in the same time slot and received using an antenna array at the base station. On the downlink, the base station antenna array provides transmitter spatial diversity (and possibly additional coding gain using space-time coding technology [25]), and spatio-temporal processing is performed at the mobiles to detect the multiple-input signals.¹

Manuscript received October 6, 1999; revised June 2, 2000. This work was supported in part by NSF Grant CCR-9732376.

N. Al-Dhahir is with AT&T Shannon Labs, Florham Park, NJ 07932 USA (e-mail: naofal@research.att.com; URL: <http://www.research.att.com/~naofal>).

A. H. Sayed is with the Electrical Engineering Department, University of California, Los Angeles, CA 90095 USA (e-mail: sayed@ee.ucla.edu).

Publisher Item Identifier S 1053-587X(00)07681-9.

¹In this paper, an input signal corresponds to a signal transmitted from a single antenna. Hence, multiple inputs could correspond to multiple distinct users, where each is equipped with a single antenna or a single user (e.g., a base station) equipped with multiple antennas or combinations thereof.

Multuser detection techniques for MIMO systems have been shown to offer significant performance advantages over single-user detection techniques that treat IUI as additive colored noise and lump its effects with background noise. Recently, it has been shown that the presence of ISI in these MIMO systems could enhance overall system capacity significantly, provided that effective multuser detection techniques are employed [12], [13].

The optimum maximum likelihood sequence estimation (MLSE) receiver for MIMO channels was developed in [26]; however, its exponential complexity increase with the number of users and channel memory makes its implementation costly for multuser detection on severe-ISI channels, especially as the input signal constellation size increases to improve spectral efficiency.² Two alternative lower complexity transceiver structures, which are widely used in practice for single-input single-output (SISO) dispersive channels, namely, discrete multitone (DMT) and minimum-mean-square-error decision feedback equalizer (MMSE-DFE), have been recently proposed for MIMO dispersive channels as well [2], [3], [12], [16], [17].

In this paper, we present a new analytical framework for analyzing the MIMO MMSE-DFE that extends the work in [6] to the MIMO case in a manner that is distinct from the work in [2], [3], [16], and [17] in three key aspects. First, the MIMO MMSE-DFE feedforward and feedback matrix filters are restricted to be finite impulse response (FIR) for practical implementation,³ and the decision delay is optimized, thus establishing finite-length analogs of the results in [2], [3], [16], and [17]. Second, the assumption of an equal number of channel inputs and outputs made in [2] and [3] is relaxed.⁴ Third, the special structure of the problem is exploited to derive fast and parallelizable MIMO MMSE-DFE computation algorithms suitable for real-time implementation. As shown in [3], computing the MIMO MMSE-DFE for the infinite-length case requires computationally intense spectral factorizations of matrix rational spectra.

The organization of this paper is as follows. We start in Section II by developing the input-output model for the MIMO MMSE-DFE and deriving closed-form expressions for its optimal filter settings. Several special cases of the MIMO MMSE-DFE structure are discussed in Section III. In Section

²For the third-generation wireless TDMA proposal [11], 8-PSK modulation is used. A typical urban EDGE channel (including the transmit pulse shape) has a memory of at least 3. Even with two transmit antennas, the number of MLSE states required is $8^6 = 262\,144$, which is clearly not practical.

³FIR filters are preferred over IIR filters due to their inherent stability, suitability for adaptive implementations, better numerical properties under finite-precision effects, and attractiveness for DSP and VLSI implementations.

⁴In some scenarios, this asymmetry is dictated by cost considerations. For example, in digital cellular networks, it is much more economical to deploy more antennas at the base station than at the mobile.

IV, we derive fast parallelizable algorithms for computing these optimal filter settings. A detailed complexity analysis and simulation results are presented in Section V, and the paper is concluded in Section VI.

Notation

The notation adopted in this paper conforms to the following convention

- Scalars are denoted in lower case: a .
- Unless otherwise stated, vectors are column vectors and are denoted in lower case bold: \mathbf{x} .
- \mathbf{e}_i denotes the i th unit vector (it has a one in the i th position and zeros everywhere else).
- In situations where the components of the vectors are to be emphasized, the first and last components, separated by a colon, are given as a subscript to the vector: $\mathbf{x}_{k+N_f-1:k-\nu}$.
- Matrices are upper case bold: \mathbf{A} .
- \mathbf{I}_N denotes the identity matrix of size N .
- $\mathbf{0}_{N \times M}$ denotes an all-zeros matrix with N rows and M columns.
- $|\mathbf{A}|$ denotes the determinant of matrix \mathbf{A} .
- $\text{trace}(\mathbf{A})$ denotes the trace of matrix \mathbf{A} .
- $E[\cdot]$ denotes the expected value operator.
- A diagonal matrix with elements $\{d_0, d_1, \dots, d_{N_f-1}\}$ on the main diagonal will be denoted by $\text{diag}(d_0, d_1, \dots, d_{N_f-1})$.
- The symbol $(\cdot)^*$ will be used to denote the complex-conjugate transpose of a matrix or a vector and the complex conjugate of a scalar.
- The symbol $(\cdot)^t$ will be used to denote the transpose of a matrix or a vector.
- When a submatrix of a given matrix needs to be specified, indices of the first and last rows (and columns) of the submatrix, separated by colons, are used as the first (and second) component of an argument to the matrix: $\mathbf{R}_{xx}(i_1:j_1, i_2:j_2)$.

For convenience, we summarize in Table I the key matrices used in this paper and their sizes.

II. FIR MIMO MMSE-DFE

We start in this section by describing the input–output model assumed throughout the paper. Then, we derive closed-form expressions for the optimum filter settings of the finite-length MIMO MMSE-DFE under three multiuser detection scenarios. In the first scenario, only previous decisions of other users are available at the input of the feedback filter for any user. In the second scenario, previous decisions as well as *current decisions* of lower indexed users are available. This scenario assumes that decisions are made sequentially, starting with lower indexed users. In the third scenario, previous and current decisions of all others users are assumed available (e.g., from a previous detection stage).

A. Input–Output Model

We consider the general case of a linear, dispersive, and noisy digital communication system with n_i inputs and n_o outputs.

TABLE I
SUMMARY OF KEY MATRICES USED IN THE PAPER AND THEIR SIZES

Matrix	Name	Dimensions
\mathbf{H}	Channel Matrix	$l n_o N_f \times n_i(N_f + \nu)$
\mathbf{R}_{xx}	Input Auto-Correlation Matrix	$n_i(N_f + \nu) \times n_i(N_f + \nu)$
\mathbf{R}_{nn}	Noise Auto-Correlation Matrix	$l n_o N_f \times l n_o N_f$
\mathbf{R}_{xy}	Input–Output Cross-Correlation Matrix	$n_i(N_f + \nu) \times l n_o N_f$
\mathbf{R}_{yy}	Output Auto-Correlation Matrix	$l n_o N_f \times l n_o N_f$
\mathbf{W}	Feedforward Filter Matrix	$n_o l N_f \times n_i$
$\tilde{\mathbf{B}}$	Augmented Feedback Filter Matrix	$n_i(N_f + \nu) \times n_i$
\mathbf{R}_{ee}	Error Auto-Correlation Matrix	$n_i \times n_i$
\mathbf{R}	$\stackrel{\text{def}}{=} \mathbf{R}_{xx}^{-1} + \mathbf{H}^* \mathbf{R}_{nn}^{-1} \mathbf{H}$	$n_i(N_f + \nu) \times n_i(N_f + \nu)$

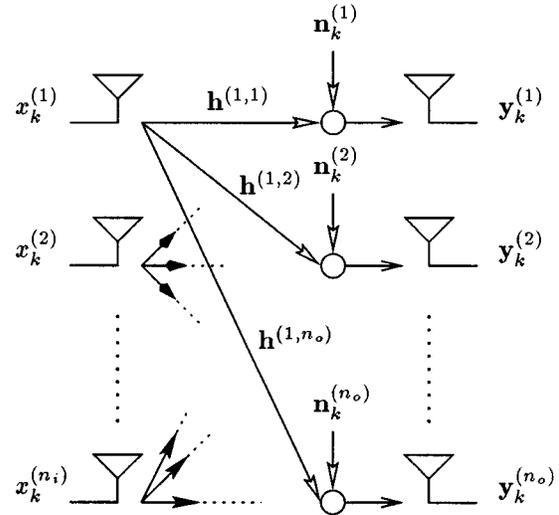


Fig. 1. Block diagram of the multi-input multi-output channel model.

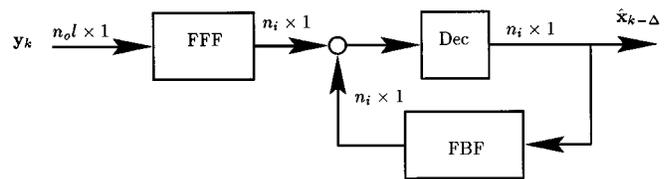


Fig. 2. Block diagram of the multi-input multi-output decision feedback equalizer.

We use the standard complex-valued baseband equivalent signal model. Assuming an oversampling factor of l , the samples at the j th channel output ($1 \leq j \leq n_o$) have the standard form (see Figs. 1 and 2)

$$\mathbf{y}_k^{(j)} = \sum_{i=1}^{n_i} \sum_{m=0}^{\nu^{(i,j)}} \mathbf{h}_m^{(i,j)} x_{k-m}^{(i)} + \mathbf{n}_k^{(j)}$$

$$\stackrel{\text{def}}{=} \sum_{i=1}^{n_i} \mathbf{h}^{(i,j)} [x_k^{(i)} \cdots x_{k-\nu^{(i,j)}}^{(i)}]^t + \mathbf{n}_k^{(j)} \quad (1)$$

where

- $\mathbf{y}_k^{(j)}$ j th channel output vector;
- $\mathbf{h}_m^{(i,j)}$ channel impulse response between the i th input and the j th output, whose memory is denoted by $\nu^{(i,j)}$;
- $\mathbf{n}_k^{(j)}$ noise vector at the j th output.

All of these three quantities are $l \times 1$ column vectors corresponding to the l time samples per symbol in the assumed

temporally oversampled channel model. Furthermore, the overall channel impulse response between the i th input and the j th output is represented by the vector $\mathbf{h}^{(i,j)} \stackrel{\text{def}}{=} [\mathbf{h}_0^{(i,j)} \ \mathbf{h}_1^{(i,j)} \ \dots \ \mathbf{h}_{\nu^{(i,j)}}^{(i,j)}]$. By grouping the received samples from all n_o channel outputs at symbol time k into an $n_o \times 1$ column vector \mathbf{y}_k , we can relate \mathbf{y}_k to the corresponding $n_i \times 1$ column vector of input samples as follows:

$$\mathbf{y}_k = \sum_{m=0}^{\nu} \mathbf{H}_m \mathbf{x}_{k-m} + \mathbf{n}_k \quad (2)$$

where \mathbf{H}_m is the $n_o \times n_i$ m th MIMO channel matrix coefficient, and \mathbf{x}_{k-m} is a size $n_i \times 1$ input vector at time $k-m$. The parameter ν is the maximum length of all of the $n_o n_i$ channel impulse responses, i.e., $\nu = \max_{i,j} \nu^{(i,j)}$.

Over a block of N_f symbol periods, (2) can be expressed in matrix notation as follows:

$$\begin{bmatrix} \mathbf{y}_{k+N_f-1} \\ \mathbf{y}_{k+N_f-2} \\ \vdots \\ \mathbf{y}_k \end{bmatrix} = \begin{bmatrix} \mathbf{H}_0 & \mathbf{H}_1 & \dots & \mathbf{H}_{\nu} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{H}_0 & \mathbf{H}_1 & \dots & \mathbf{H}_{\nu} & \mathbf{0} & \dots \\ \vdots & & & \ddots & & & \vdots \\ \mathbf{0} & \dots & \mathbf{0} & \mathbf{H}_0 & \mathbf{H}_1 & \dots & \mathbf{H}_{\nu} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{x}_{k+N_f-1} \\ \mathbf{x}_{k+N_f-2} \\ \vdots \\ \mathbf{x}_{k-\nu} \end{bmatrix} + \begin{bmatrix} \mathbf{n}_{k+N_f-1} \\ \mathbf{n}_{k+N_f-2} \\ \vdots \\ \mathbf{n}_k \end{bmatrix} \quad (3)$$

or more compactly

$$\mathbf{y}_{k+N_f-1:k} = \mathbf{H} \mathbf{x}_{k+N_f-1:k-\nu} + \mathbf{n}_{k+N_f-1:k} \quad (4)$$

By defining the $n_i(N_f + \nu) \times n_i(N_f + \nu)$ input auto-correlation matrix

$$\mathbf{R}_{xx} \stackrel{\text{def}}{=} E[\mathbf{x}_{k+N_f-1:k-\nu} \mathbf{x}_{k+N_f-1:k-\nu}^*]$$

and the $(n_o l N_f) \times (n_o l N_f)$ noise auto-correlation matrix

$$\mathbf{R}_{nn} \stackrel{\text{def}}{=} E[\mathbf{n}_{k+N_f-1:k} \mathbf{n}_{k+N_f-1:k}^*]$$

the input-output cross-correlation and the output auto-correlation matrices are given by

$$\mathbf{R}_{xy} \stackrel{\text{def}}{=} E[\mathbf{x}_{k+N_f-1:k-\nu} \mathbf{y}_{k+N_f-1:k}^*] = \mathbf{R}_{xx} \mathbf{H}^* \quad (5)$$

$$\mathbf{R}_{yy} \stackrel{\text{def}}{=} E[\mathbf{y}_{k+N_f-1:k} \mathbf{y}_{k+N_f-1:k}^*] = \mathbf{H} \mathbf{R}_{xx} \mathbf{H}^* + \mathbf{R}_{nn} \quad (6)$$

B. Performance Analysis

The FIR MIMO MMSE-DFE consists of a feedforward filter matrix

$$\mathbf{W}^* \stackrel{\text{def}}{=} [\mathbf{W}_0^* \ \mathbf{W}_1^* \ \dots \ \mathbf{W}_{N_f-1}^*]$$

with N_f matrix taps \mathbf{W}_i , each of size $(n_o l \times n_i)$, and a feedback filter matrix equal to

$$[\mathbf{I}_{n_i} \ \mathbf{0}_{n_i \times n_i N_b}] - \mathbf{B}^* \stackrel{\text{def}}{=} [(\mathbf{I}_{n_i} - \mathbf{B}_0^*) \ -\mathbf{B}_1^* \ \dots \ -\mathbf{B}_{N_b}^*]$$

Furthermore, we define the matrix

$$\mathbf{B}^* \stackrel{\text{def}}{=} [\mathbf{B}_0^* \ \mathbf{B}_1^* \ \dots \ \mathbf{B}_{N_b}^*]$$

with $(N_b + 1)$ matrix taps \mathbf{B}_i , each of size $n_i \times n_i$. Therefore, \mathbf{W}_i and \mathbf{B}_i have the forms

$$\mathbf{W}_i = \begin{bmatrix} \mathbf{w}_i^{(1,1)} & \dots & \mathbf{w}_i^{(1,n_i)} \\ \vdots & \dots & \vdots \\ \mathbf{w}_i^{(n_o,1)} & \dots & \mathbf{w}_i^{(n_o,n_i)} \end{bmatrix} \quad \mathbf{B}_i = \begin{bmatrix} b_i^{(1,1)} & \dots & b_i^{(1,n_i)} \\ \vdots & \dots & \vdots \\ b_i^{(n_i,1)} & \dots & b_i^{(n_i,n_i)} \end{bmatrix} \quad (7)$$

where each entry in \mathbf{W}_i is an $l \times 1$ vector corresponding to the l output samples per symbol.

If we define the size $n_i \times n_i(N_f + \nu)$ matrix $\tilde{\mathbf{B}}^* \stackrel{\text{def}}{=} [\mathbf{0}_{n_i \times n_i \Delta} \ \mathbf{B}^*]$, where $0 \leq \Delta \leq N_f + \nu - 1$ is the decision delay that satisfies the condition $(\Delta + N_b + 1) = (N_f + \nu)$, then it can be shown that the MIMO MMSE-DFE error vector at time k is given by (assuming correct previous decisions)

$$\mathbf{E}_k = \tilde{\mathbf{B}}^* \mathbf{x}_{k+N_f-1:k-\nu} - \mathbf{W}^* \mathbf{y}_{k+N_f-1:k} \quad (8)$$

Therefore, the $n_i \times n_i$ error auto-correlation matrix of the MIMO MMSE-DFE is equal to

$$\begin{aligned} \mathbf{R}_{ee} &\stackrel{\text{def}}{=} E[\mathbf{E}_k^* \mathbf{E}_k] \\ &= \tilde{\mathbf{B}}^* \mathbf{R}_{xx} \tilde{\mathbf{B}} - \tilde{\mathbf{B}}^* \mathbf{R}_{xy} \mathbf{W} - \mathbf{W}^* \mathbf{R}_{yx} \tilde{\mathbf{B}} + \mathbf{W}^* \mathbf{R}_{yy} \mathbf{W} \\ &= \tilde{\mathbf{B}}^* (\mathbf{R}_{xx} - \mathbf{R}_{xy} \mathbf{R}_{yy}^{-1} \mathbf{R}_{yx}) \tilde{\mathbf{B}} + (\mathbf{W}^* - \tilde{\mathbf{B}}^* \mathbf{R}_{xy} \mathbf{R}_{yy}^{-1}) \\ &\quad \cdot \mathbf{R}_{yy} (\mathbf{W}^* - \tilde{\mathbf{B}}^* \mathbf{R}_{xy} \mathbf{R}_{yy}^{-1})^* \\ &\stackrel{\text{def}}{=} \tilde{\mathbf{B}}^* \mathbf{R}^{\perp} \tilde{\mathbf{B}} + \mathbf{G}^* \mathbf{R}_{yy} \mathbf{G} \end{aligned} \quad (9)$$

where we introduced the matrices $\{\mathbf{R}^{\perp}, \mathbf{G}\}$. When dealing with multidimensional error random processes, either the trace or the determinant of \mathbf{R}_{ee} can be used as a mean square error measure. The trace measure is equal to the *arithmetic average* of the eigenvalues of \mathbf{R}_{ee} , whereas the determinant measure is related to their *geometric average*. For the MIMO MMSE-DFE, it turns out that the same feedforward and feedback coefficients minimize both measures; hence, either one can be adopted. Corresponding to these two measures, we define the following two decision-point SNR performance measures for the MIMO MMSE-DFE

- Arithmetic SNR defined by

$$\text{ASNR} \stackrel{\text{def}}{=} \frac{1}{n_i(N_f + \nu)} \frac{\text{trace}(\mathbf{R}_{xx})}{\frac{1}{n_i} \text{trace}(\mathbf{R}_{ee, \min})} \quad (10)$$

- Geometric SNR defined by

$$\text{GSNR} \stackrel{\text{def}}{=} \frac{|\mathbf{R}_{xx}|^{(1/n_i)(N_f + \nu)}}{|\mathbf{R}_{ee, \min}|^{(1/n_i)}} \quad (11)$$

In this paper, we adopt ASNR as a performance measure for two main reasons. First, it is more widely used in the literature back to the early work on multidimensional linear estimation. Second, it simplifies the proofs of some key results given in Appendices C–F.

C. Optimum MIMO MMSE-DFE Coefficients

In this subsection, we derive closed-form expressions for computing the optimal filter settings of the MIMO MMSE-DFE from estimates of the MIMO channel matrix \mathbf{H} and the noise auto-correlation matrix \mathbf{R}_{nn} . These estimates can be obtained using training sequences as described in [4] or estimated blindly using second-order statistics of the received signals as described in [28].

Applying the *orthogonality principle*, which states that $E[\mathbf{E}_k \mathbf{y}_{k+N_f-1:k}^*] = \mathbf{0}$, it can be shown that the optimum matrix feedforward and feedback filters are related by

$$\mathbf{W}_{opt}^* = \tilde{\mathbf{B}}_{opt}^* \mathbf{R}_{xy} \mathbf{R}_{yy}^{-1}. \quad (12)$$

We show in Appendix A that this solution minimizes both the trace and the determinant of \mathbf{R}_{ee} .

Using the optimum feedforward matrix filter settings of (12) in (9) and applying the matrix inversion lemma,⁵ we can write

$$\begin{aligned} \mathbf{R}_{ee} &= \tilde{\mathbf{B}}^* \mathbf{R}^\perp \tilde{\mathbf{B}} \\ &= \tilde{\mathbf{B}}^* (\mathbf{R}_{xx} - \mathbf{R}_{xy} \mathbf{R}_{yy}^{-1} \mathbf{R}_{yx}) \tilde{\mathbf{B}} \\ &= \tilde{\mathbf{B}}^* (\mathbf{R}_{xx} - \mathbf{R}_{xx} \mathbf{H}^* (\mathbf{H} \mathbf{R}_{xx} \mathbf{H}^* + \mathbf{R}_{nn})^{-1} \mathbf{H} \mathbf{R}_{xx}) \tilde{\mathbf{B}} \\ &= \tilde{\mathbf{B}}^* (\mathbf{R}_{xx}^{-1} + \mathbf{H}^* \mathbf{R}_{nn}^{-1} \mathbf{H})^{-1} \tilde{\mathbf{B}} \\ &\stackrel{\text{def}}{=} \tilde{\mathbf{B}}^* \mathbf{R}^{-1} \tilde{\mathbf{B}} \end{aligned} \quad (13)$$

where we defined the matrix \mathbf{R} . We consider three detection scenarios.

Scenario 1: Only previous decisions of other users are available for any user at the present time (and, hence, their interfering effect can be suppressed), i.e., $\mathbf{B}_0 = \mathbf{I}_{n_i}$. To determine the optimum matrix feedback filter coefficients, we need to solve the following constrained optimization problem:

$$\min_{\tilde{\mathbf{B}}} \text{trace}(\mathbf{R}_{ee}) = \min_{\tilde{\mathbf{B}}} \text{trace}(\tilde{\mathbf{B}}^* \mathbf{R}^{-1} \tilde{\mathbf{B}})$$

subject to

$$\tilde{\mathbf{B}}^* \Phi = \mathbf{C}^*$$

where

$$\Phi \stackrel{\text{def}}{=} \begin{bmatrix} \mathbf{I}_{n_i \Delta \times n_i (\Delta+1)} \\ \mathbf{0}_{n_i \times n_i (\Delta+1)} \end{bmatrix}$$

and

$$\mathbf{C}^* \stackrel{\text{def}}{=} [\mathbf{0}_{n_i \times n_i \Delta} \quad \mathbf{I}_{n_i}].$$

It can be shown that the solution is given by (see, e.g., [20])

$$\tilde{\mathbf{B}}_{opt} = \mathbf{R} \Phi (\Phi^* \mathbf{R} \Phi)^{-1} \mathbf{C} \quad (15)$$

$$\mathbf{R}_{ee, \min} = \mathbf{C}^* (\Phi^* \mathbf{R} \Phi)^{-1} \mathbf{C}. \quad (16)$$

If we define the partitioning $\mathbf{R} \stackrel{\text{def}}{=} \begin{bmatrix} \mathbf{R}_{11} & \mathbf{R}_{12} \\ \mathbf{R}_{12}^* & \mathbf{R}_{22} \end{bmatrix}$, where \mathbf{R}_{11} is of size $n_i(\Delta+1) \times n_i(\Delta+1)$, then

$$\tilde{\mathbf{B}}_{opt} = \begin{bmatrix} \mathbf{R}_{11} \\ \mathbf{R}_{12}^* \end{bmatrix} \mathbf{R}_{11}^{-1} \mathbf{C} = \begin{bmatrix} \mathbf{I}_{n_i(\Delta+1)} \\ \mathbf{R}_{12}^* \mathbf{R}_{11}^{-1} \end{bmatrix} \mathbf{C} \quad (17)$$

$$\mathbf{R}_{ee, \min} = \mathbf{C}^* \mathbf{R}_{11}^{-1} \mathbf{C} \quad (18)$$

⁵ $(\mathbf{A} + \mathbf{BCD})^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1} \mathbf{B} (\mathbf{D} \mathbf{A}^{-1} \mathbf{B} + \mathbf{C}^{-1})^{-1} \mathbf{D} \mathbf{A}^{-1}$.

where the decision delay parameter Δ is chosen to minimize the trace of $\mathbf{R}_{ee, \min}$.

A second approach for computing $\tilde{\mathbf{B}}_{opt}$ and $\mathbf{R}_{ee, \min}$ utilizes the *block* Cholesky factorization

$$\begin{aligned} \mathbf{R} &= \mathbf{R}_{xx}^{-1} + \mathbf{H}^* \mathbf{R}_{nn}^{-1} \mathbf{H} \\ &= \begin{bmatrix} \mathbf{L}_1 & \mathbf{0} \\ \mathbf{L}_2 & \mathbf{L}_3 \end{bmatrix} \begin{bmatrix} \mathbf{D}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_2 \end{bmatrix} \begin{bmatrix} \mathbf{L}_1^* & \mathbf{L}_2^* \\ \mathbf{0} & \mathbf{L}_3^* \end{bmatrix} \\ &\stackrel{\text{def}}{=} \mathbf{L} \mathbf{D} \mathbf{L}^* \end{aligned} \quad (19)$$

where \mathbf{L}_1 is of size $n_i(\Delta+1) \times n_i(\Delta+1)$, and \mathbf{L} has $n_i \times n_i$ identity block entries on the diagonal. In addition, \mathbf{D} is formed from $n_i \times n_i$ blocks. Using the result in (15) and (16), we get

$$\begin{aligned} \tilde{\mathbf{B}}_{opt} &= \begin{bmatrix} \mathbf{I}_{n_i(\Delta+1)} \\ \mathbf{L}_2 \mathbf{L}_1^{-1} \end{bmatrix} \quad \mathbf{C} = \begin{bmatrix} \mathbf{I}_{n_i} \\ \mathbf{L}_2 \mathbf{L}_1^{-1} \mathbf{C} \end{bmatrix} \\ &= \mathbf{L} [\mathbf{e}_{n_i \Delta_{opt}} \quad \cdots \quad \mathbf{e}_{n_i(\Delta_{opt}+1)-1}] \end{aligned} \quad (20)$$

$$\begin{aligned} \mathbf{R}_{ee, \min} &= \mathbf{C}^* \mathbf{L}_1^{-*} \mathbf{D}_1^{-1} \mathbf{L}_1^{-1} \mathbf{C} \\ &= \text{diag}(d_{n_i \Delta_{opt}}^{-1}, \dots, d_{n_i(\Delta_{opt}+1)-1}^{-1}) \end{aligned} \quad (21)$$

where, as mentioned before, the decision delay Δ_{opt} ($0 \leq \Delta_{opt} \leq N_f + \nu - 1$) is chosen to minimize the trace of $\mathbf{R}_{ee, \min}$. Using (20) along with (5) and (6), the feedforward matrix taps of (12) can be expressed as follows:

$$\begin{aligned} \mathbf{W}_{opt}^* &= \tilde{\mathbf{B}}_{opt}^* \mathbf{R}_{xx} \mathbf{H}^* (\mathbf{H} \mathbf{R}_{xx} \mathbf{H}^* + \mathbf{R}_{nn})^{-1} \\ &= \tilde{\mathbf{B}}_{opt}^* (\mathbf{R}_{xx}^{-1} + \mathbf{H}^* \mathbf{R}_{nn}^{-1} \mathbf{H})^{-1} \mathbf{H}^* \mathbf{R}_{nn}^{-1} \\ &= \begin{bmatrix} d_{n_i \Delta_{opt}}^{-1} \mathbf{e}_{n_i \Delta_{opt}}^* \\ \vdots \\ d_{n_i(\Delta_{opt}+1)-1}^{-1} \mathbf{e}_{n_i(\Delta_{opt}+1)-1}^* \end{bmatrix} \mathbf{L}^{-1} \mathbf{H}^* \mathbf{R}_{nn}^{-1}. \end{aligned} \quad (22)$$

In Section III, we present fast algorithms for performing the block Cholesky factorization in (19).

Yet a third approach for computing $\tilde{\mathbf{B}}_{opt}$ and $\mathbf{R}_{ee, \min}$ is by partitioning \mathbf{R}^\perp as follows:

$$\begin{aligned} \mathbf{R}_{ee} &= \tilde{\mathbf{B}}^* \mathbf{R}^\perp \tilde{\mathbf{B}} \\ &\stackrel{\text{def}}{=} [\mathbf{C}^* \quad \bar{\mathbf{B}}^*] \begin{bmatrix} \mathbf{R}_{11}^\perp & \mathbf{R}_{12}^\perp \\ \mathbf{R}_{12}^{*\perp} & \mathbf{R}_{22}^\perp \end{bmatrix} \begin{bmatrix} \mathbf{C} \\ \bar{\mathbf{B}} \end{bmatrix} \\ &\stackrel{\text{def}}{=} [\mathbf{I}_{n_i} \quad \bar{\mathbf{B}}^*] \begin{bmatrix} \bar{\mathbf{R}}_{11}^\perp & \bar{\mathbf{R}}_{12}^\perp \\ \bar{\mathbf{R}}_{12}^{*\perp} & \bar{\mathbf{R}}_{22}^\perp \end{bmatrix} \begin{bmatrix} \mathbf{I}_{n_i} \\ \bar{\mathbf{B}} \end{bmatrix} \\ &= (\bar{\mathbf{R}}_{11}^\perp - \bar{\mathbf{R}}_{12}^\perp (\bar{\mathbf{R}}_{22}^\perp)^{-1} \bar{\mathbf{R}}_{12}^{*\perp}) + (\bar{\mathbf{B}}^* + \bar{\mathbf{R}}_{12}^\perp (\bar{\mathbf{R}}_{22}^\perp)^{-1}) \\ &\quad \cdot \bar{\mathbf{R}}_{22}^\perp (\bar{\mathbf{B}}^* + \bar{\mathbf{R}}_{12}^\perp (\bar{\mathbf{R}}_{22}^\perp)^{-1})^* \end{aligned}$$

where $\bar{\mathbf{B}}^* \stackrel{\text{def}}{=} [\mathbf{B}_1^* \quad \cdots \quad \mathbf{B}_{N_b}^*]$, $\bar{\mathbf{R}}_{11}^\perp \stackrel{\text{def}}{=} \mathbf{C}^* \mathbf{R}_{11}^\perp \mathbf{C}$, and $\bar{\mathbf{R}}_{12}^\perp = \mathbf{C}^* \mathbf{R}_{12}^\perp$. Therefore

$$\mathbf{B}_{opt}^* = -\bar{\mathbf{R}}_{12}^\perp (\bar{\mathbf{R}}_{22}^\perp)^{-1} \quad (24)$$

$$\begin{aligned} \mathbf{W}_{opt}^* &= [\mathbf{0}_{n_i \times n_i \Delta} \quad \mathbf{I}_{n_i} \quad -\bar{\mathbf{R}}_{12}^\perp (\bar{\mathbf{R}}_{22}^\perp)^{-1}] \\ &\quad \cdot \mathbf{R}_{xx} \mathbf{H}^* (\mathbf{H} \mathbf{R}_{xx} \mathbf{H}^* + \mathbf{R}_{nn})^{-1} \\ \mathbf{R}_{ee, \min} &= \bar{\mathbf{R}}_{11}^\perp - \bar{\mathbf{R}}_{12}^\perp (\bar{\mathbf{R}}_{22}^\perp)^{-1} \bar{\mathbf{R}}_{12}^{*\perp}. \end{aligned} \quad (25)$$

We prove in Appendix B that (17) and (18) are equivalent to (24) and (25), respectively.

Scenario 2: Assume that users are ordered so that lower indexed users are detected first, more specifically, that *current decisions from lower indexed users are used by higher indexed users in making their decisions*, i.e., \mathbf{B}_0 is now a monic⁶ lower triangular matrix. The general results in (17) and (18) can still be applied in this case by setting $\mathbf{C}^* = [\mathbf{0}_{n_i \times n_i \Delta} \ \mathbf{B}_0^*]$, where \mathbf{B}_0 is an $n_i \times n_i$ monic lower triangular matrix whose entries are optimized to minimize $\text{trace}(\mathbf{R}_{ee, \min})$. Toward this end, define the partitioning⁷

$$\mathbf{R}_{11}^{-1} = \begin{bmatrix} \mathbf{R}_1 & \mathbf{R}_2 \\ \mathbf{R}_2^* & \mathbf{R}_3 \end{bmatrix} \quad (26)$$

where \mathbf{R}_1 is $n_i \Delta \times n_i \Delta$, and \mathbf{R}_3 is $n_i \times n_i$. Therefore, (18) simplifies to

$$\mathbf{R}_{ee, \min} = \mathbf{B}_0^* \mathbf{R}_3 \mathbf{B}_0. \quad (27)$$

It can be readily checked that the optimum monic lower triangular \mathbf{B}_0 that minimizes $\text{trace}(\mathbf{R}_{ee, \min})$ is given by the monic lower triangular Cholesky factor of \mathbf{R}_3^{-1} , i.e.,

$$\begin{aligned} \mathbf{R}_3^{-1} &\stackrel{\text{def}}{=} \mathbf{L}_3 \mathbf{D}_3 \mathbf{L}_3^* \\ \mathbf{B}_0^{\text{opt}} &= \mathbf{L}_3 \\ \mathbf{R}_{ee, \min} &= \mathbf{D}_3^{-1}. \end{aligned} \quad (28)$$

We show in Appendix D that Scenario 2 achieves lower mean-square-error (which is defined as the trace of $\mathbf{R}_{ee, \min}$) than Scenario 1.

A second approach for computing the optimum FIR MIMO MMSE-DFE filter settings under Scenario 2 is by performing a *standard* (i.e., not block) Cholesky factorization of the matrix $\mathbf{R} = \mathbf{R}_{xx}^{-1} + \mathbf{H}^* \mathbf{R}_{nn}^{-1} \mathbf{H}$ in the form $\mathbf{L} \mathbf{D} \mathbf{L}^*$. Then, the feedback filter matrix is given by the n_i *adjacent* columns of \mathbf{L} that correspond to a diagonal matrix with the smallest trace. Therefore, (20) and (23) are used to compute MIMO MMSE-DFE filter settings *with the understanding that \mathbf{L} is now a lower triangular matrix and not a block lower triangular matrix*. This result is shown in Appendix C.

The equivalence of the two approaches can be easily shown using the *nesting property* of Cholesky factorization. This property states that if $\mathbf{R} = \mathbf{L} \mathbf{D} \mathbf{L}^*$, then the lower triangular and diagonal Cholesky factors of the $k \times k$ submatrix of \mathbf{R} formed from its *first* k rows and columns are equal to the *first* k rows and columns of \mathbf{L} and \mathbf{D} , respectively. Conversely, if $\mathbf{R} = \mathbf{L}^* \mathbf{D} \mathbf{L}$, then the upper-triangular and diagonal Cholesky factors of the $k \times k$ submatrix of \mathbf{R} formed from its *last* k rows and columns are equal to the *last* k rows and columns of \mathbf{L}^* and \mathbf{D} , respectively.

Scenario 3: When multistage detectors [27] are employed, current decisions from all other users, which are obtained from a previous detection stage,⁸ are available for the detection of the user of interest. Therefore, suppressing their interfering effects would improve the performance of the MIMO MMSE-DFE. This detection scenario has the same mathematical formulation

⁶A monic matrix has diagonal elements equal to one.

⁷Note that \mathbf{R}_{11} is a Hermitian matrix.

⁸In asynchronous CDMA, the previous detection stage could be, for example, a bank of single-user matched filters or a decorrelating stage [27].

as Scenarios 1 and 2, *except* that \mathbf{B}_0 is now only constrained to be monic, i.e., $\mathbf{e}_i^* \mathbf{B}_0 \mathbf{e}_i = 1$ for all $0 \leq i \leq n_i - 1$. The general results in (17) and (18) still apply with $\mathbf{C}^* = [\mathbf{0}_{n_i \times n_i \Delta} \ \mathbf{B}_0^*]$, where \mathbf{B}_0 is optimized to minimize $\text{trace}(\mathbf{R}_{ee, \min})$. In summary, under Scenario 3, we solve the optimization problem

$$\min_{\mathbf{B}_0} \text{trace}(\mathbf{B}_0^* \mathbf{R}_3 \mathbf{B}_0)$$

subject to

$$\mathbf{e}_i^* \mathbf{B}_0 \mathbf{e}_i = 1 \text{ for all } 0 \leq i \leq n_i - 1$$

where \mathbf{R}_3 was defined in (26). Using Lagrange multiplier technique, it is straightforward to show that the optimum monic \mathbf{B}_0 and the corresponding MMSE are given by

$$\mathbf{B}_0^{\text{opt}} \mathbf{e}_i = \frac{\mathbf{R}_3^{-1} \mathbf{e}_i}{\mathbf{e}_i^* \mathbf{R}_3^{-1} \mathbf{e}_i} : 0 \leq i \leq n_i - 1 \quad (29)$$

$$\text{MMSE}_3 \stackrel{\text{def}}{=} \text{trace}(\mathbf{R}_{ee, \min}) = \sum_{i=0}^{n_i-1} \frac{1}{\mathbf{e}_i^* \mathbf{R}_3^{-1} \mathbf{e}_i}. \quad (30)$$

We show in Appendix E that Scenario 3 achieves lower MMSE than Scenario 2.

Scenario 3 was presented for completeness of the analysis; however, our main focus in this paper will be on Scenarios 1 and 2, and multistage detection will not be discussed any further.

Remark: In concluding this section, we would like to emphasize the importance of optimizing the decision delay parameter Δ in all three scenarios, especially for short feedforward filters. Using a suboptimum Δ can result in significant performance degradation, as it will be demonstrated in the simulation results of Section V-C. This is also the case for the SISO MMSE-DFE, as shown in [7].

For the MIMO case, allowing the different users to have different decision delays could result in improved decision-point SNR for the MIMO MMSE-DFE. However, it might violate the assumption of available previous decisions from all other users in Scenario 1 and the additional assumption of availability of current decisions from lower indexed users made in Scenario 2. Other drawbacks of allowing variable user decision delays include the increased computational complexity in optimizing these variable delays and computing the optimum MIMO MMSE-DFE settings and the fact that the resulting feedback filters will be of different lengths in general. For all of the above reasons, we do not explore this variable-delay detection strategy in this paper.

III. SOME SPECIAL CASES

In this section, we derive several special and limiting cases of the general results of Section II.

A. Uncorrelated Input and Noise

When the input and noise processes are uncorrelated between different time samples and from one input–output channel to the other, the matrices \mathbf{R}_{xx} and \mathbf{R}_{nn} become block diagonal and take the simple form

$$\begin{aligned} \mathbf{R}_{xx} &= \text{diag}(\text{diag}(\sigma_{x,1}^2, \sigma_{x,2}^2, \dots, \sigma_{x,n_i}^2), \dots \\ &\quad \text{diag}(\sigma_{x,1}^2, \sigma_{x,2}^2, \dots, \sigma_{x,n_i}^2)) \end{aligned}$$

$$\mathbf{R}_{nn} = \text{diag}(\text{diag}(\sigma_{n,1}^2 \mathbf{I}_l, \sigma_{n,2}^2 \mathbf{I}_l, \dots, \sigma_{n,n_o}^2 \mathbf{I}_l), \dots, \text{diag}(\sigma_{n,1}^2 \mathbf{I}_l, \sigma_{n,2}^2 \mathbf{I}_l, \dots, \sigma_{n,n_o}^2 \mathbf{I}_l)).$$

The SNR at the output of the (i, j) channel is defined by

$$\text{SNR}^{(i,j)} = \frac{\sigma_{x,i}^2 \sum_{k=0}^{\nu} |e_{i-1}^* \mathbf{H}_k e_{j-1}|^2}{\sigma_{n,j}^2} : 1 \leq i \leq n_i; 1 \leq j \leq n_o. \quad (31)$$

B. MIMO Zero-Forcing (ZF) DFE

The optimal settings of the FIR MIMO ZF-DFE follow as a special case of the MMSE-DFE by letting the noise variances go to zero. The optimal matrix feedback filter is computed from the lower triangular Cholesky factor of the matrix $\mathbf{H}^* \mathbf{H}$. The matrix feedforward filter is given by $\mathbf{W}_{opt}^* = \tilde{\mathbf{B}}^* \mathbf{R}_{xx} \mathbf{H}^* (\mathbf{H} \mathbf{R}_{xx} \mathbf{H}^*)^{-1}$. Note that a *pseudo inverse* is used to compute \mathbf{W}_{opt} when $N_f \geq (\nu / (l(n_o/n_i) - 1))$.

C. FIR MIMO MMSE-LE

The FIR MIMO MMSE-LE follows as a special case by setting $\mathbf{B}_0 = \mathbf{I}_{n_i}$ and all other \mathbf{B}_i to zero. The error auto-correlation matrix is given by

$$\begin{aligned} \mathbf{R}_{ee, \text{MMSE-LE}} &= [\mathbf{0}_{n_i \times n_i \Delta} \quad \mathbf{I}_{n_i} \quad \mathbf{0}_{n_i \times n_i(N_f + \nu - \Delta - 1)}] \\ &\cdot \mathbf{R}^{-1} \begin{bmatrix} \mathbf{0}_{n_i \Delta \times n_i} \\ \mathbf{I}_{n_i} \\ \mathbf{0}_{n_i(N_f + \nu - \Delta - 1) \times n_i} \end{bmatrix} \\ &= \mathbf{R}^{-1}(n_i \Delta + 1; n_i(\Delta + 1), n_i \Delta + 1; n_i(\Delta + 1)). \end{aligned} \quad (32)$$

The delay parameter Δ is chosen to minimize the trace of $\mathbf{R}_{ee, \text{MMSE-LE}}$. The optimum FIR MIMO MMSE-LE settings are given by

$$\begin{aligned} \mathbf{W}_{\text{MMSE-LE}} &= \mathbf{R}_{yy}^{-1} \mathbf{R}_{yx} \begin{bmatrix} \mathbf{0}_{n_i \Delta \times n_i} \\ \mathbf{I}_{n_i} \\ \mathbf{0}_{n_i(N_f + \nu - \Delta - 1) \times n_i} \end{bmatrix} \\ &= (\mathbf{H} \mathbf{R}_{xx} \mathbf{H}^* + \mathbf{R}_{nn})^{-1} \mathbf{H} \mathbf{R}_{xx} \begin{bmatrix} \mathbf{0}_{n_i \Delta \times n_i} \\ \mathbf{I}_{n_i} \\ \mathbf{0}_{n_i(N_f + \nu - \Delta - 1) \times n_i} \end{bmatrix}. \end{aligned} \quad (33)$$

In Appendix F, we prove that the MIMO MMSE-LE results in a higher MMSE than the MIMO MMSE-DFE. This result is well known in the SISO case (see, e.g., [23]).

D. FIR SIMO MMSE-DFE

By setting $n_i = 1$, we get the FIR version of the single-input multi-output (SIMO) MMSE-DFE studied in [8], *assuming infinite-length filters*. One scenario where the SIMO MMSE-DFE can be implemented is when multiple receive antennas are used to achieve a diversity gain when detecting a single input. In this case, \mathbf{H}_i is a column vector with $l n_o$ elements, and $\tilde{\mathbf{B}}^* = [\mathbf{0}_{1 \times \Delta} \quad 1 \quad b_1^* \quad \dots \quad b_{N_b}^*]$.

E. FIR MISO MMSE-DFE

The multi-input single-output (MISO) MMSE-DFE is obtained by setting $n_o = 1$. Each feedforward tap \mathbf{W}_i in this case is of size $l \times n_i$. This is an effective receiver structure in at least two important transmission scenarios. In the first scenario, the multiple inputs correspond to distinct users, and hence, the MISO MMSE-DFE suppresses co-channel interference by detecting the desired and the interfering user signals as received by the single output channel. In the second scenario, the multiple inputs correspond to transmit diversity paths for the same user (e.g., multiple transmit antennas at the base station and a single receive antenna at the mobile).

F. FIR SISO Fractionally Spaced MMSE-DFE

In this case, $n_i = n_o = 1$, and each \mathbf{H}_i is an $l \times 1$ column vector. This is the standard fractionally spaced finite-length MMSE-DFE studied in detail in [6]. In the presence of multiple inputs, the SISO MMSE-DFE treats other input signals as colored noise when detecting the desired input signal [5]. As will be demonstrated by the simulations of Section V, this results in significant performance degradation from the MIMO and MISO MMSE-DFE structures, which detect all n_i inputs.

IV. FAST COMPUTATION ALGORITHMS

We see from (20) and (23) for the optimal weight matrices $\tilde{\mathbf{B}}_{opt}$ and \mathbf{W}_{opt} that the fundamental step in evaluating these optimal coefficients is the computation of the Cholesky factor \mathbf{L} of \mathbf{R} in (19) or, equivalently, the factor $\bar{\mathbf{L}}$ in

$$\mathbf{R} = \bar{\mathbf{L}} \bar{\mathbf{L}}^*; \quad \bar{\mathbf{L}} = \mathbf{L} \mathbf{D}^{(1/2)}$$

where $\mathbf{D}^{1/2}$ denotes a square-root factor for \mathbf{D} (which is defined as lower triangular with positive-diagonal entries on the diagonal). Actually, two Cholesky factorizations are required, depending on whether \mathbf{B}_0 is taken to be the identity matrix \mathbf{I}_{n_i} or a lower triangular matrix. In the former case, $\bar{\mathbf{L}}$ is required to be a *block* lower triangular Cholesky factor with $n_i \times n_i$ block matrices along its diagonal. In the latter case, $\bar{\mathbf{L}}$ is required to be lower triangular with scalar entries along its diagonal, viz.

$$\bar{\mathbf{L}} = \begin{bmatrix} (n_i \times n_i) & & \\ \times & (n_i \times n_i) & \\ \times & \times & (n_i \times n_i) \end{bmatrix}$$

or

$$\bar{\mathbf{L}} = \begin{bmatrix} l_{00} & & \\ \times & l_{11} & \\ \times & \times & l_{22} \end{bmatrix}.$$

The inverse of \mathbf{L} need not be computed explicitly in order to evaluate \mathbf{W}_{opt} in (23). Instead, one could solve, via back-substitution, the linear system of equations

$$\mathbf{L} \mathbf{A} = \mathbf{H}^* \mathbf{R}_{nn}^{-1}$$

determine \mathbf{A} , and then set

$$\mathbf{W}_{opt}^* = \left[\begin{array}{c} d_{n_i \Delta_{opt}}^{-1} \mathbf{e}_{n_i \Delta_{opt}}^* \\ \vdots \\ d_{n_i(\Delta_{opt}+1)-1}^{-1} \mathbf{e}_{n_i(\Delta_{opt}+1)-1}^* \end{array} \right] \mathbf{A}.$$

Hence, we will focus on the computation of $\bar{\mathbf{L}}$. Actually, we study the case of uncorrelated input and noise as described in Section III-A. Now, since \mathbf{R} has dimensions $n_i(N_f + \nu) \times n_i(N_f + \nu)$, these computations would generally require $O[n_i^3(N_f + \nu)^3]$ operations by using the classical Gaussian elimination technique.

However, we will show in the sequel that by exploiting the block-Toeplitz structure of the matrix \mathbf{H} in (3), in addition to the block-diagonal structure of the matrices $\{\mathbf{R}_{xx}, \mathbf{R}_{nn}\}$, and by using a fast so-called generalized Schur algorithm (see, e.g., [19], [21], and [24]), this step can be performed efficiently in $O[n_i^2(N_f + \nu)^2]$ operations.

Introduce the block-diagonal matrices with input and noise variances

$$\mathbf{D}_x = \begin{bmatrix} \sigma_{x,1}^2 & & & \\ & \ddots & & \\ & & \ddots & \\ & & & \sigma_{x,n_i}^2 \end{bmatrix}$$

$$\mathbf{D}_n = \begin{bmatrix} \sigma_{n,1}^2 \mathbf{I}_l & & & \\ & \ddots & & \\ & & \ddots & \\ & & & \sigma_{n,n_o}^2 \mathbf{I}_l \end{bmatrix}$$

where \mathbf{D}_x is $n_i \times n_i$, and \mathbf{D}_n is $n_{ol} \times n_{ol}$. Then, $\{\mathbf{R}_{xx}, \mathbf{R}_{nn}\}$ can be expressed in terms of $\{\mathbf{D}_n, \mathbf{D}_x\}$ as

$$\mathbf{R}_{xx} = \begin{bmatrix} \mathbf{D}_x & & \\ & \ddots & \\ & & \mathbf{D}_x \end{bmatrix}$$

$$\mathbf{R}_{nn} = \begin{bmatrix} \mathbf{D}_n & & \\ & \ddots & \\ & & \mathbf{D}_n \end{bmatrix}$$

where \mathbf{R}_{xx} consists of $(N_f + \nu)$ block-diagonal blocks, whereas \mathbf{R}_{nn} has N_f block-diagonal entries. Moreover, \mathbf{H} is given by

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_0 & \mathbf{H}_1 & \dots & \mathbf{H}_\nu & & \\ & \mathbf{H}_0 & \mathbf{H}_1 & \dots & \mathbf{H}_\nu & \\ & & \ddots & \ddots & \ddots & \ddots \\ & & & \mathbf{H}_0 & \mathbf{H}_1 & \dots & \mathbf{H}_\nu \end{bmatrix}$$

where each \mathbf{H}_i is $n_{ol} \times n_i$, and where \mathbf{H} has N_f block rows and $(N_f + \nu)$ block columns. We will denote the nonzero entries of the top block row of \mathbf{H} by \mathbf{T}

$$\mathbf{T} = [\mathbf{H}_0 \quad \mathbf{H}_1 \quad \dots \quad \mathbf{H}_\nu]$$

which is $n_{ol} \times n_i \nu$.

A. Exploiting Matrix Structure

A powerful and convenient framework for exploiting the structure of the matrix \mathbf{R} is displacement structure theory [19], [21], [24]. In the sequel, we will first show that because of the block-Toeplitz structure of the matrices $\{\mathbf{H}, \mathbf{R}_{xx}, \mathbf{R}_{nn}\}$, the

matrix \mathbf{R} itself exhibits displacement structure. Once this fact is established, we will then show how it can be exploited for the purpose of fast Cholesky factorization.

To begin with, define the following block shift matrices

$$\mathbf{Z}_{n_o} = \begin{bmatrix} 0 & & & \\ \mathbf{I}_{n_{ol}} & 0 & & \\ & \ddots & \ddots & \\ & & \mathbf{I}_{n_{ol}} & 0 \end{bmatrix}$$

$$\mathbf{Z}_{n_i} = \begin{bmatrix} 0 & & & \\ \mathbf{I}_{n_i} & 0 & & \\ & \ddots & \ddots & \\ & & \mathbf{I}_{n_i} & 0 \end{bmatrix}$$

$$\mathcal{F} = \begin{bmatrix} \mathbf{Z}_{n_o} & 0 \\ 0 & \mathbf{Z}_{n_i} \end{bmatrix}$$

where \mathbf{Z}_{n_o} has N_f block columns and rows, whereas \mathbf{Z}_{n_i} has $(N_f + \nu)$ block columns and rows. Then, it is easy to verify that the matrices $\{\mathbf{H}, \mathbf{R}_{xx}, \mathbf{R}_{nn}\}$ satisfy

$$\mathbf{H} - \mathbf{Z}_{n_o} \mathbf{H} \mathbf{Z}_{n_i}^* = \begin{bmatrix} \mathbf{T} & 0 \\ 0 & 0 \end{bmatrix}$$

$$\mathbf{R}_{nn} - \mathbf{Z}_{n_o} \mathbf{R}_{nn} \mathbf{Z}_{n_o}^* = \begin{bmatrix} \mathbf{D}_n & \\ & 0 \end{bmatrix}$$

$$\mathbf{R}_{xx}^{-1} - \mathbf{Z}_{n_i} \mathbf{R}_{xx}^{-1} \mathbf{Z}_{n_i}^* = \begin{bmatrix} \mathbf{D}_x^{-1} & \\ & 0 \end{bmatrix}.$$

Now, observe that the negative of the matrix \mathbf{R} in (19) can be obtained as the Schur complement of the block-Hermitian matrix

$$\mathcal{M} \triangleq \begin{bmatrix} \mathbf{R}_{nn} & \mathbf{H} \\ \mathbf{H}^* & -\mathbf{R}_{xx}^{-1} \end{bmatrix}$$

with respect to its leading block entry. Moreover, \mathcal{M} satisfies

$$\mathcal{M} - \mathcal{F} \mathcal{M} \mathcal{F}^* = \begin{bmatrix} \mathbf{D}_n & 0 & \mathbf{T} & \circ & \circ \\ 0 & 0 & 0 & \circ & \circ \\ \mathbf{T}^* & 0 & -\mathbf{D}_x^{-1} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

If we further introduce the *normalized* quantities (i.e., we normalize the rows of each \mathbf{H}_i)

$$\bar{\mathbf{H}}_i \triangleq \mathbf{D}_n^{-1/2} \mathbf{H}_i$$

then it is easy to verify that

$$\mathcal{M} - \mathcal{F} \mathcal{M} \mathcal{F}^* = \mathcal{G} \mathcal{J} \mathcal{G}^*$$

where \mathcal{J} is a signature matrix given by

$$\mathcal{J} = \begin{bmatrix} \mathbf{I}_{n_{ol}} & & \\ & -\mathbf{I}_{n_{ol}} & \\ & & -\mathbf{I}_{n_i} \end{bmatrix}$$

and \mathcal{G} is a so-called generator matrix that is given by

$$\mathcal{G} = \begin{bmatrix} \mathbf{D}_n^{1/2} & 0 & 0 \\ 0_{n_o l} & 0_{n_o l} & 0_{n_o l} \\ \vdots & \vdots & \vdots \\ 0_{n_o l} & 0_{n_o l} & 0_{n_o l} \\ \overline{\mathbf{H}}_0^* & \overline{\mathbf{H}}_0^* & \mathbf{D}_x^{-1/2} \\ \overline{\mathbf{H}}_1^* & \overline{\mathbf{H}}_1^* & 0_{n_i} \\ \vdots & \vdots & \vdots \\ \overline{\mathbf{H}}_\nu^* & \overline{\mathbf{H}}_\nu^* & 0_{n_i} \\ 0 & 0 & 0 \\ \vdots & \vdots & \vdots \\ 0 & 0 & 0 \end{bmatrix}$$

where the number of block zero rows between $\mathbf{D}_n^{1/2}$ and $\overline{\mathbf{H}}_0^*$ is $N_f - 1$, whereas the number of block zeros in the last column underneath $\mathbf{D}_x^{-1/2}$ is $N_f + \nu - 1$.

Matrices \mathcal{M} that satisfy equations of the above form are called structured matrices, or matrices with displacement structure (see, e.g., [19] and [24]). Two notable properties of such matrices are the following:

- 1) The structure is preserved under Schur complementation. Therefore, since $-\mathbf{R}$ can be obtained as a Schur complement of \mathcal{M} , it will follow that $-\mathbf{R}$ itself has a similar displacement structure (for some other matrices $\{\mathcal{F}, \mathcal{G}\}$ of appropriate dimensions and for the same signature matrix \mathcal{J}).
- 2) There exists a fast (efficient) algorithm for the computation of the triangular factors of such structured matrices \mathcal{M} , which is known as the generalized Schur algorithm. The algorithm also provides the triangular factors of Schur complements of the matrix. Thus, by computing the triangular factors of \mathcal{M} , we will be able to obtain the desired triangular factors of \mathbf{R} directly from those of \mathcal{M} . In so doing, we would have derived an efficient procedure for factoring \mathbf{R} and, therefore, for determining the optimal coefficients $\{\tilde{\mathbf{B}}_{opt}, \mathbf{W}_{opt}\}$.

B. Fast Standard Factorization Algorithm

We first show how to obtain the *standard* Cholesky factorization of \mathbf{R} , viz., $\mathbf{R} = \overline{\mathbf{L}}\overline{\mathbf{L}}^*$, where $\overline{\mathbf{L}}$ is lower triangular with scalar entries along its diagonal.

First, note that the leading block matrix of \mathcal{M} is positive-definite, whereas the Schur complement of \mathcal{M} with respect to this entry is negative definite (and equal to $-\mathbf{R}$). Hence, \mathcal{M} admits a triangular factorization of the following form:

$$\mathcal{M} = \mathcal{L} \begin{bmatrix} \mathbf{I}_{N_f n_o l} & \\ & -\mathbf{I}_{(N_f + \nu)n_i} \end{bmatrix} \mathcal{L}^* \triangleq \mathcal{L} \mathcal{I}_M \mathcal{L}^*$$

where \mathcal{L} is lower triangular with positive diagonal entries, and \mathcal{I}_M is a signature matrix. We will soon explain how to determine \mathcal{L} efficiently by using the matrices $\{\mathcal{F}, \mathcal{G}, \mathcal{J}\}$ above. Assume for now that this has been done, and partition \mathcal{L} as follows:

$$\mathcal{L} = \begin{bmatrix} \mathcal{L}_1 & \\ \mathcal{L}_2 & \mathcal{L}_3 \end{bmatrix}$$

where \mathcal{L}_3 has the same dimensions as \mathbf{R} . Then, it can be verified that $-\mathbf{R} = -\mathcal{L}_3 \mathcal{L}_3^*$ so that the desired lower triangular factor $\overline{\mathbf{L}}$ is given by $\overline{\mathbf{L}} = \mathcal{L}_3$.

The so-called generalized Schur algorithm (see, e.g., [19], [21], and [24]) is an efficient procedure for the computation of the triangular factor \mathcal{L} of matrices \mathcal{M} that satisfy displacement equations of the form

$$\mathcal{M} - \mathcal{F}\mathcal{M}\mathcal{F}^* = \mathcal{G}\mathcal{J}\mathcal{G}^*.$$

It starts with $\{\mathcal{F}, \mathcal{G}, \mathcal{J}\}$ and iterates for a number of steps that is equal to the size of \mathcal{M} . At each iteration, it produces a column of \mathcal{L} .

There are various ways to describe the algorithm. Here, we state it in a so-called proper form, which is suitable for parallel implementations. Let us denote, for convenience, the entries of \mathcal{J} by

$$\mathcal{J} = \begin{bmatrix} \mathbf{I}_\alpha & \\ & -\mathbf{I}_\beta \end{bmatrix}.$$

That is, $\alpha = n_o l$, and $\beta = n_i + n_o l$. In addition, let $r = \alpha + \beta$.

Algorithm 1 (Fast Algorithm)

Let $\mathcal{F}_0 = \mathcal{F}$, $\mathcal{G}_0 = \mathcal{G}$, and repeat for $i \geq 0$

- 1) At step i , we have the matrices $\{\mathcal{F}_i, \mathcal{G}_i\}$.

Let g_i denote the top row of \mathcal{G}_i .

- 2) Starting from the top entry of the inertia matrix \mathcal{I}_M , we associate with each step of the algorithm an inertia entry. We will say that the step is positive if the corresponding entry is +1 in \mathcal{I}_M , and the step is negative if the corresponding entry is -1 in \mathcal{I}_M .
- 3) Implement a sequence of $(\alpha - 1)$ Givens rotations (or a single Householder transformation [14]) that annihilates the entries 2 through α of g_i , say

$$g_i = \begin{bmatrix} \times & \times & \times & \times & | & \times & \times & \times \\ \xrightarrow{\text{rotations}} & \times' & 0 & 0 & 0 & | & \times & \times & \times \end{bmatrix}.$$

- 4) Implement a sequence of $(\beta - 1)$ Givens rotations (or a single Householder transformation) that annihilates the last $(\beta - 1)$ entries of the transformed vector, say

$$\begin{bmatrix} \times' & 0 & 0 & 0 & | & \times & \times & \times \\ \xrightarrow{\text{rotations}} & \times' & 0 & 0 & 0 & | & \times' & 0 & 0 \end{bmatrix}.$$

- 5a) If the step is *positive*, do the following. [Otherwise move to step 5b).] Use one final hyperbolic rotation to annihilate the entry in position $\alpha + 1$,

$$\begin{bmatrix} \times' & 0 & 0 & 0 & | & \times' & 0 & 0 \\ \xrightarrow{\text{hyperbolic}} & \delta_i & 0 & 0 & 0 & | & 0 & 0 & 0 \end{bmatrix}.$$

The sequence of rotations in steps 3), 4), and 5a) should also be applied to all the other rows of \mathcal{G}_i . This leads to a new matrix, say $\overline{\mathcal{G}}_i = \mathcal{G}_i \Theta_i$, where Θ_i

denotes the combined effect of all rotations, and the top row of $\bar{\mathcal{G}}_i$ will thus have the special form

$$\text{top row of } \bar{\mathcal{G}}_i = [\delta_i \ 0 \ 0 \ 0 \ | \ 0 \ 0 \ 0].$$

5b) If the step is *negative*, then instead of step 5a), we use one final hyperbolic rotation to annihilate the entry in position 1

$$\begin{array}{c} [\times' \ 0 \ 0 \ 0 \ | \ \times' \ 0 \ 0] \\ \xrightarrow{\text{hyperbolic}} [0 \ 0 \ 0 \ 0 \ | \ \delta_i \ 0 \ 0]. \end{array}$$

In this case, the sequence of rotations of steps 3), 4), and 5b) should also be applied to all the other rows of \mathcal{G}_i . This leads to a new matrix, say, $\bar{\mathcal{G}}_i = \mathcal{G}_i \Theta_i$, where Θ_i denotes the combined effect of all rotations, and the top row of $\bar{\mathcal{G}}_i$ will thus have the special form

$$\text{top row of } \bar{\mathcal{G}}_i = [0 \ 0 \ 0 \ 0 \ | \ \delta_i \ 0 \ 0].$$

6) The (nonzero part of the) i th column of \mathcal{L} , which is denoted by l_i , is given by the first column of $\bar{\mathcal{G}}_i$ (if the step is positive) or by the $(\alpha + 1)$ th column of $\bar{\mathcal{G}}_i$ (if the step is negative)

$$l_i = \bar{\mathcal{G}}_i \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad (\text{for a positive step}),$$

$$l_i = \bar{\mathcal{G}}_i \begin{bmatrix} 0_{\alpha \times 1} \\ 1 \\ 0 \end{bmatrix} \quad (\text{for a negative step}).$$

7) To obtain \mathcal{G}_{i+1} , we multiply l_i by \mathcal{F}_i , and use the result to replace the column corresponding to l_i in $\bar{\mathcal{G}}_i$. All other columns of $\bar{\mathcal{G}}_i$ remain unchanged. This results in a new matrix whose top row is zero. We then delete the top row to get \mathcal{G}_{i+1} .

For example, for a positive step, this construction corresponds to

$$\begin{bmatrix} 0 \\ \mathcal{G}_{i+1} \end{bmatrix} = [\mathcal{F}_i l_i \quad 0] + \bar{\mathcal{G}}_i \begin{bmatrix} 0 \\ I_{r-1} \end{bmatrix}$$

whereas for a negative step, we perform

$$\begin{bmatrix} 0 \\ \mathcal{G}_{i+1} \end{bmatrix} = [0 \quad \mathcal{F}_i l_i \quad 0] + \bar{\mathcal{G}}_i \begin{bmatrix} I_\alpha & & \\ & 0 & \\ & & I_{\beta-1} \end{bmatrix}.$$

8) To get \mathcal{F}_{i+1} , we delete the first row and column of \mathcal{F}_i .

9. Return to step 1), and repeat the procedure. \blacksquare

At the end of the algorithm, we have available all the columns of \mathcal{L} from which we can obtain the desired submatrix matrix \mathcal{L}_3 (or $\bar{\mathcal{L}}$).

We may remark that the above algorithm is parallelizable. This is because the rotations can be applied to all rows of \mathcal{G}_i

simultaneously. Moreover, the numerical stability of this factorization procedure in finite-precision implementations has been established in [10] for the generalized form of the algorithm, as well as in [9] and [22] for more specialized forms. One particular conclusion from these works is that numerical stability requires that the hyperbolic rotations of steps 5a) and 5b) be implemented with care. There are several ways to do so, e.g., by using the so-called OD or H procedures of [10] or by using a mixed downdating form [9], [22].

C. Fast Block Factorization Algorithm

We now show how to obtain a block Cholesky factor $\bar{\mathbf{L}}$ with $n_i \times n_i$ block entries along its diagonal. For this purpose, note again that the first N_{fnol} steps of the above algorithm are positive steps, at the end of which we obtain a generator matrix $\mathcal{G}_{N_{fnol}}$ for \mathbf{R} . More explicitly, we obtain

$$\mathbf{R} - \mathbf{Z}_{n_i} \mathbf{R} \mathbf{Z}_{n_i} = \mathcal{G}_{N_{fnol}} \mathcal{J} \mathcal{G}_{N_{fnol}}^*.$$

We will assume that these initial N_{fnol} steps have been performed and that we have available a generator matrix $\mathcal{G}_{N_{fnol}}$.

With $\{\mathbf{Z}_{n_i}, \mathcal{G}_{N_{fnol}}, \mathcal{J}\}$ at hand, we can now obtain the desired block factor $\bar{\mathbf{L}}$ as follows.

Algorithm 2 (Block Fast Algorithm)

Let $\mathcal{F}_0 = \mathbf{Z}_{n_i}$, $\mathcal{G}_0 = \mathcal{G}_{N_{fnol}}$, and repeat for $i \geq 0$

- 1) At step i , we have the matrices $\{\mathcal{F}_i, \mathcal{G}_i\}$. Let G_i denote the top n_i rows of \mathcal{G}_i .
- 2) Implement a sequence of \mathcal{J} -unitary rotations that reduces G_i to the form

$$G_i = \begin{bmatrix} \times & \times & \times & \times & | & \times & \times & \times \\ \xrightarrow{\text{rotations}} [\Delta_i & 0 & 0 & 0 & | & 0 & 0 & 0] \end{bmatrix}$$

where Δ_i is now lower triangular. This can be accomplished in a number of ways. For example, we can first transform the top row of \mathcal{G}_i to the form

$$\begin{bmatrix} \times & \times & \times & \times & | & \times & \times & \times \\ \xrightarrow{\text{rotations}} [\times' & 0 & 0 & 0 & | & 0 & 0 & 0] \end{bmatrix}$$

and apply these transformations to all other rows of \mathcal{G}_i . Then, we transform the second row of the resulting \mathcal{G}_i to the form

$$\begin{bmatrix} \times & \times & \times & \times & | & \times & \times & \times \\ \xrightarrow{\text{rotations}} [\times' & \times' & 0 & 0 & | & 0 & 0 & 0] \end{bmatrix}$$

and apply the transformations again to all other rows of \mathcal{G}_i . We continue in this fashion until the top n_i rows of \mathcal{G}_i have been transformed to the form

$$G_i \rightarrow [\Delta_i \ 0 \ 0 \ 0 \ | \ 0 \ 0 \ 0]$$

where Δ_i is lower triangular. After all the transformations are applied to all rows of \mathcal{G}_i , we obtain a new matrix, say, $\bar{\mathcal{G}}_i = \mathcal{G}_i \Theta_i$, where Θ_i denotes the combined effect of all rotations, and the top n_i

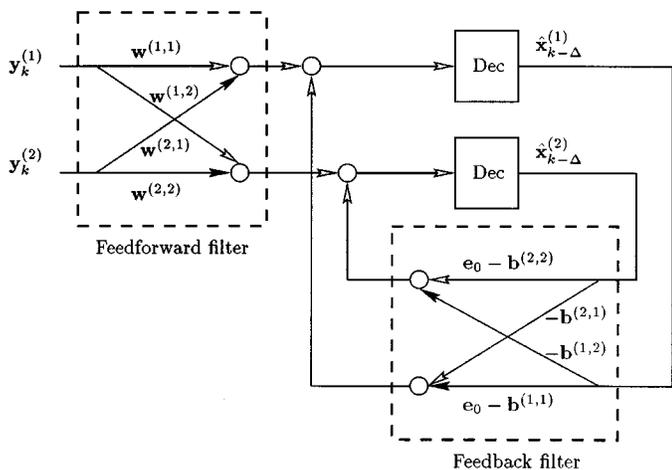


Fig. 3. Filter connections for the MIMO MMSE-DFE computed in the numerical example.

Note that the four feedback filters of the MIMO MMSE-DFE are *strictly causal*, *except* for the filter $\mathbf{b}^{(2,1)}$ in the case of a lower triangular \mathbf{B}_0 , where $\mathbf{b}^{(2,1)}$ is causal filter, i.e., its zeroth-order tap is not constrained to be zero. This extra tap results in the 0.52 dB ASNR improvement calculated above.

Scenario 3: \mathbf{B}_0 Is Monic: For completeness, we also computed the optimum MIMO MMSE-DFE coefficients and ASNR under Scenario 3. Using (17), (18), (22), (26), (29), and (30), we get

$$\begin{aligned} \Delta_{opt} &= 2 \\ \text{ASNR} &= 20.49 \text{ dB} \\ \mathbf{B}_0 &= \begin{bmatrix} 1 & 0.464 \\ 0.478 & 1 \end{bmatrix} \\ \mathbf{B}_1 &= \begin{bmatrix} 0.0438 & 0.0288 \\ -0.115 & -0.1028 \end{bmatrix} \\ \mathbf{W}_0 &= \begin{bmatrix} -0.0084 & 0.0083 \\ 0.0073 & -0.0072 \end{bmatrix} \\ \mathbf{W}_1 &= \begin{bmatrix} 0.2922 & -0.2462 \\ -0.2453 & 0.2051 \end{bmatrix} \\ \mathbf{W}_2 &= \begin{bmatrix} 0.3893 & 0.3778 \\ 0.4459 & 0.4634 \end{bmatrix}. \end{aligned}$$

Under this scenario, the extra nonzero tap in \mathbf{B}_0 results in 0.57 dB improvement in ASNR over Scenario 2. It is also interesting to note that the first-column entries of \mathbf{B}_i and \mathbf{W}_i are identical under Scenarios 2 and 3.

B. Computational Complexity

In this section, we give a rough order-of-magnitude estimate of the computational complexity involved in computing the optimum filter coefficients of the MIMO MMSE-DFE using (20) and (22).⁹ Several assumptions are made. First, we use the number of instructions per coefficient update as a complexity measure. Second, we only count complex multiplies and assume that each complex multiply is equivalent to six instructions (four real multiplies and two real adds).¹⁰ Third, we assume white input and noise sequences. Finally, the complexity incurred in estimating the MIMO channel matrix is not included. We evaluate the computational complexity as a function of the number of transmit and receive antennas, number of equalizer taps, and the MIMO channel memory. The results are shown in Figs. 4 and 5. The instruction counts given in these figures can be readily translated into MIPS estimates through multiplication by the MIMO MMSE-DFE coefficient update rate, which is a design parameter set according to the dynamics of channel time variations. As an example, assuming a 2-ms update rate,¹¹ then computing the coefficients of a $T/2$ -spaced MIMO MMSE-DFE with $N_f = 8$ (i.e., a total of 16 feedforward taps), four feedback taps for a channel memory of 4 and two antennas at each of the transmit and receive ends requires around 100 MIPS. This is well within the processing power of state-of-the-art programmable DSP chips like the TMS320C6X family of processors from Texas Instruments.

C. Computer Simulations

The channel impulse responses used in our computer simulations are unit-energy four-tap FIR filters. The four taps are randomly generated complex zero-mean uncorrelated Gaussian random variables. The input and noise processes are assumed to be uncorrelated. The performance results are calculated by averaging over 100 channel realizations.

Fig. 6 shows the variation of the ASNR and GSNR performance measures, as defined by (10) and (11), respectively, with the decision delay parameter Δ for $N_f = 3$ and $n_i = n_o = 2$. It can be seen that both ASNR and GSNR are maximized by the same optimum delay and that a suboptimum delay setting could

⁹The computational complexity can be reduced by an order-of-magnitude by using the fast algorithms of Section IV.

¹⁰This is a pessimistic assumption since state-of-the-art DSP's can typically perform a complex multiply and add operation in a single instruction.

¹¹A GSM mobile travelling at 55 mi/hr sees a channel coherence time of ≈ 13.6 ms. Hence, the channel can be assumed time-invariant over a 2-ms time window.

$$\mathbf{R} = \begin{bmatrix} 14.1983 & 14.6654 & 0.8780 & -2.3080 & 0 & 0 & 0 & 0 \\ 14.6654 & 16.2726 & 0.5945 & -2.1258 & 0 & 0 & 0 & 0 \\ 0.8780 & 0.5945 & 20.1010 & 9.4776 & 0.8780 & -2.3080 & 0 & 0 \\ -2.3080 & -2.1258 & 9.4776 & 21.0010 & 0.5945 & -2.1258 & 0 & 0 \\ 0 & 0 & 0.8780 & 0.5945 & 20.1010 & 9.4776 & 0.8780 & -2.3080 \\ 0 & 0 & -2.3080 & -2.1258 & 9.4776 & 21.0010 & 0.5945 & -2.1258 \\ 0 & 0 & 0 & 0 & 0.8780 & 0.5945 & 6.0027 & -5.1877 \\ 0 & 0 & 0 & 0 & -2.3080 & -2.1258 & -5.1877 & 5.7283 \end{bmatrix}$$

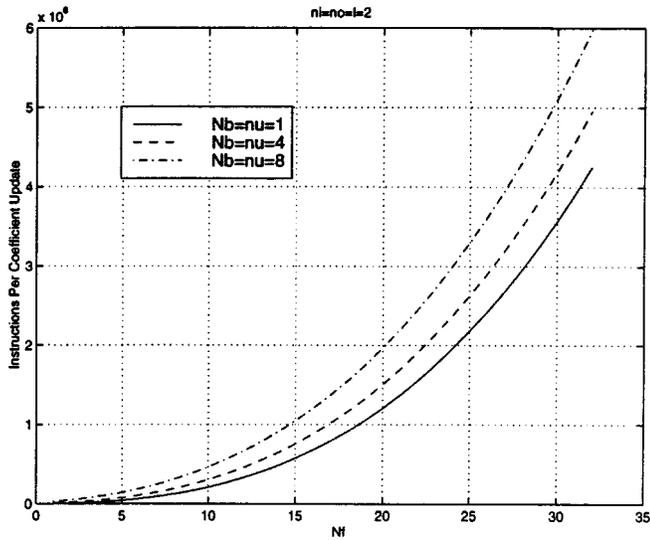


Fig. 4. Instructions count for computing MIMO MMSE-DFE coefficients versus N_f and N_b .

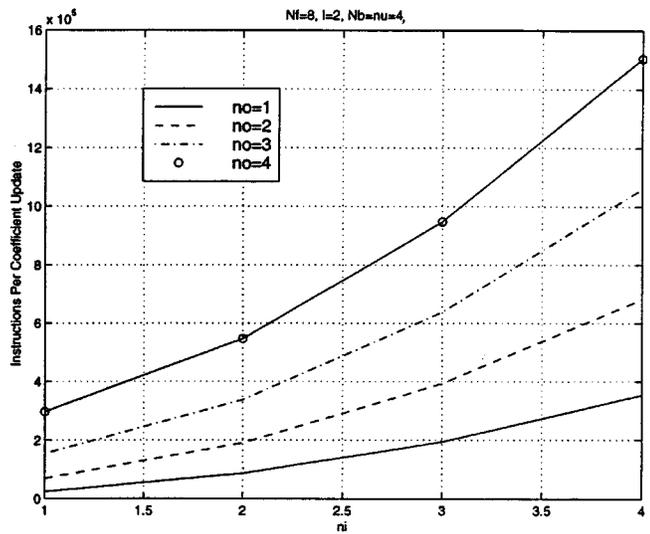


Fig. 5. Instructions count for computing MIMO MMSE-DFE coefficients versus n_i and n_o .

result in significant performance degradation. In Fig. 7, we plot the MIMO MMSE-DFE ASNR difference between Scenarios 1 and 2. We assume $n_i = n_o = 2$, $N_f = 3$, and set the SNR of the second user (*the higher indexed user*) at 10 dB while increasing the SNR of the first user (*the lower indexed user*) from -10 dB to 30 dB. It can be seen from the figure that constraining B_0 to be lower triangular (as in Scenario 2) always results in better performance than the case $B_0 = I$ (as in Scenario 1). As expected, this performance improvement increases as the SNR of the lower indexed user (whose current decisions are also fed back and used in detecting the higher indexed user) is increased.

For the rest of the simulations presented in this section, we

- adopt ASNR as a decision-point SNR performance measure;
- optimize the decision delay;
- constrain B_0 to be a lower triangular matrix.

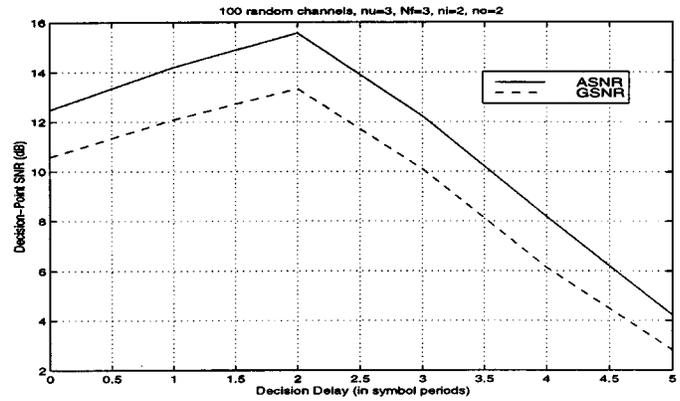


Fig. 6. Variation of ASNR and GSNR with decision delay for the MIMO MMSE-DFE with $\nu = 3$, $n_i = n_o = 2$.

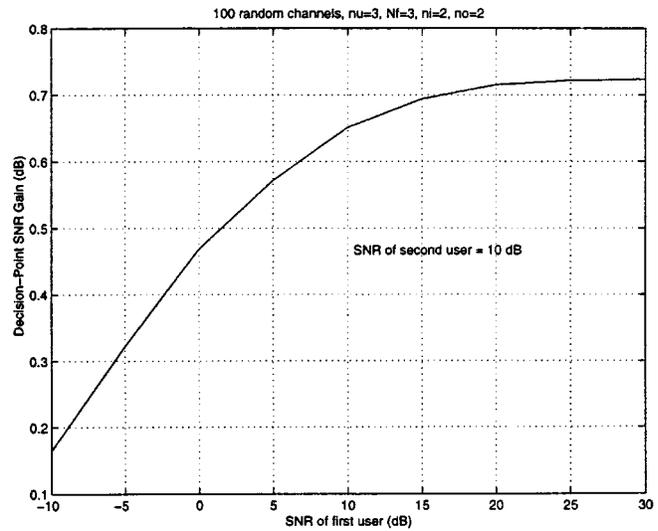


Fig. 7. MIMO MMSE-DFE ASNR gain of Scenario 2 over Scenario 1 versus input SNR of first user assuming second user at 10 dB SNR and $N_f = \nu = 3$.

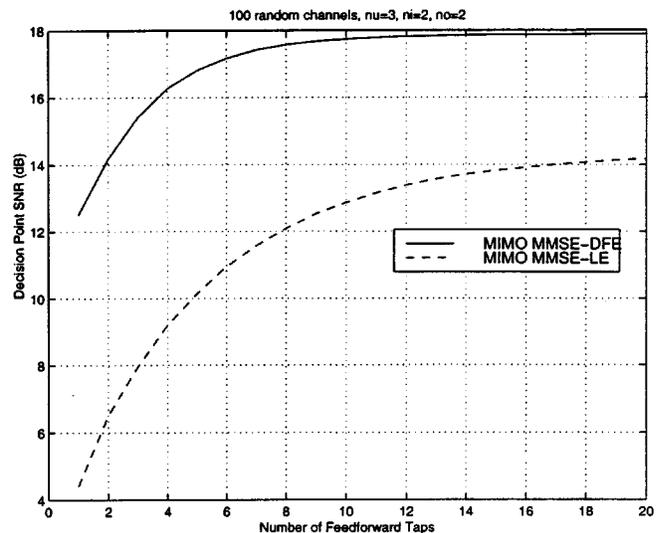


Fig. 8. Variation of ASNR of the MIMO MMSE-DFE and MMSE-LE with N_f for $\nu = 3$ and $n_i = n_o = 2$. Input SNR of first and second users equal 20 dB and 10 dB, respectively.

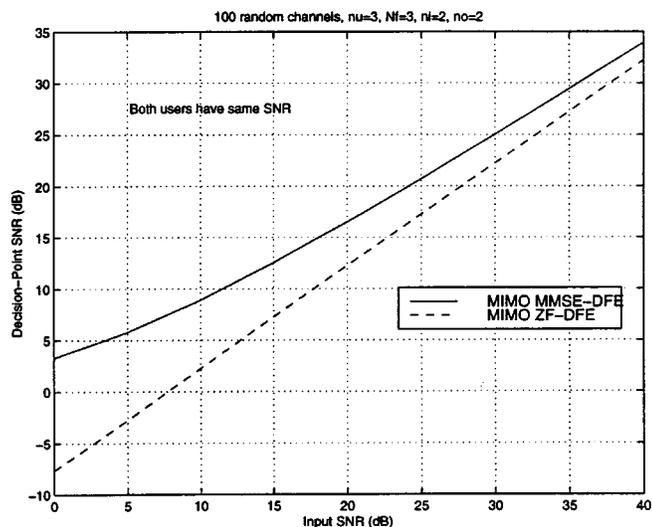


Fig. 9. Variation of ASNR of the MIMO MMSE-DFE and MIMO ZF-DFE versus input SNR of both users (assumed equal) for $N_f = \nu = 3$ and $n_i = n_o = 2$.

In Fig. 8, we compare the decision-point of the MIMO MMSE-DFE and the MIMO MMSE-LE as N_f is increased for $n_i = n_o = 2$. SNR of the first and second users is equal to 20 dB and 10 dB, respectively. Therefore, according to the SNR definition in (31), $\text{SNR}^{(1,1)} = \text{SNR}^{(1,2)} = 20$ dB, and $\text{SNR}^{(2,1)} = \text{SNR}^{(2,2)} = 10$ dB. It is clear from Fig. 8 that the MIMO MMSE-DFE offers significant performance advantage over the MIMO MMSE-LE at the expense of n_e^2 additional feedback filters. The decision-point SNR of the MIMO MMSE-DFE and the MIMO ZF-DFE are depicted in Fig. 9 for $n_i = n_o = 2$ and an equal input SNR on all four channels that is increased from 0 to 40 dB. As expected, use of the MMSE criterion results in a higher decision-point SNR, especially at low input SNR levels. Therefore, for the rest of the simulation results, we consider only the MIMO MMSE-DFE structure.

In Fig. 10, we investigate the performance improvement attained by increasing the number of output channels and implementing a SIMO MMSE-DFE over the case of a SISO MMSE-DFE. The SNR of the first output channel is set at 20 dB, whereas the SNR's of the other output channels are all equal and increase simultaneously in value from -10 dB to 20 dB. It is intuitively appealing to see that the performance advantage of the SIMO MMSE-DFE over the SISO MMSE-DFE increases as the SNR of the output diversity channels increases.

Next, we examine the effectiveness of the MIMO MMSE-DFE structure in suppressing co-channel interference from other users. In Fig. 11, we assume two simultaneous users having an SNR of 20 dB. We compare three detection strategies.

- 1) SISO MMSE-DFE that treats co-channel interference from the other user as colored noise;
- 2) MISO MMSE-DFE with $n_i = 2$ and $n_o = 1$ that detects both users using a single output channel;
- 3) MIMO MMSE-DFE with $n_i = n_o = 2$ that detects both users by processing two output channels.

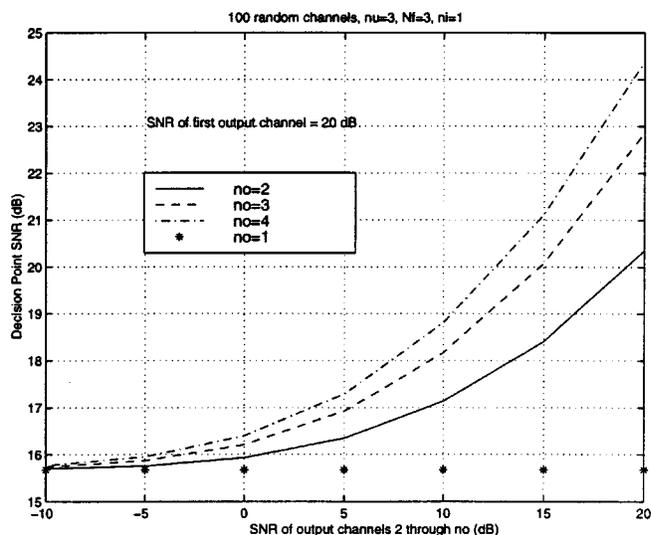


Fig. 10. Variation of ASNR of the SIMO MMSE-DFE with n_o and SNR of output channels 2 through n_o for $N_f = \nu = 3$.

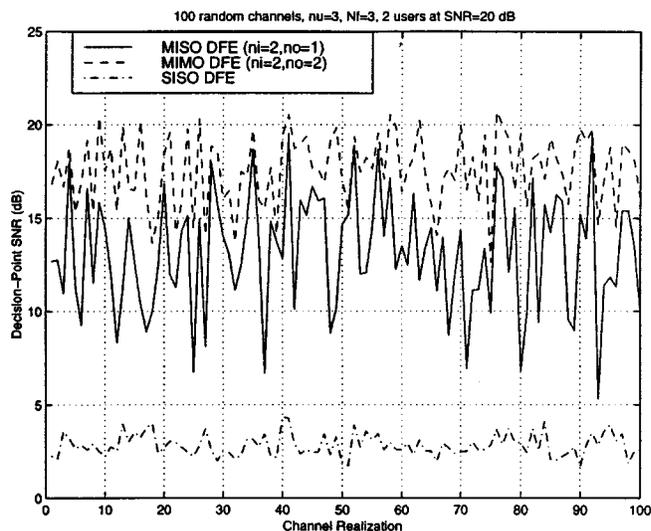


Fig. 11. Variation of ASNR of the MIMO, MISO, and SISO MMSE-DFE in presence of two users at 20 dB SNR for different channel realizations and $N_f = \nu = 3$.

It is clear from Fig. 11 that the MIMO structure is always superior in performance. In Fig. 12, we fix the SNR of the first user at 20 dB while increasing the SNR of the second user from -10 dB to 30 dB. In agreement with intuition, when the channel of the second (co-channel) user is very noisy, the performance of the SISO MMSE-DFE approaches that of the MISO MMSE-DFE with $n_i = 2$ and $n_o = 1$. However, as the SNR of the co-channel user improves, using a MISO MMSE-DFE results in very significant performance improvement over the SISO MMSE-DFE. Even better performance is achieved by processing two output channels for the two users, as in the MIMO MMSE-DFE structure.

VI. CONCLUSIONS

New closed-form expressions for the optimum finite-length MIMO MMSE-DFE filters and decision-point SNR were de-

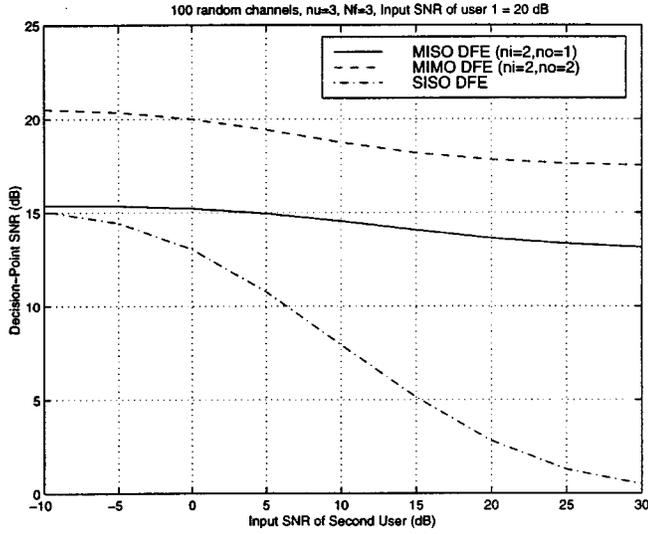


Fig. 12. Variation of ASNR of the MIMO, MISO, and SISO MMSE-DFE in the presence of two users versus input SNR of second user, assuming the first user at 20 dB SNR and $N_f = \nu = 3$

rived under the three most common multiuser detection scenarios. Fast and parallelizable algorithms for computing these filters were derived by exploiting the block-Toeplitz structure of the MIMO channel matrix. The presented analytical framework allows for quick MIMO MMSE-DFE performance evaluation as a function of the number of transmit antennas, receive antennas, and filter taps.

APPENDIX A

OPTIMUM MIMO MMSE-DFE SETTINGS MINIMIZE TRACE AND DETERMINANT OF \mathbf{R}_{ee}

We show that the solution in (12) minimizes both the trace and determinant of \mathbf{R}_{ee} . Define \mathbf{e}_i to be a unit vector with an i th element of 1 and zeros everywhere else. Using (9)

$$\begin{aligned} \mathbf{e}_i^* \mathbf{R}_{ee} \mathbf{e}_i &= \mathbf{e}_i^* \tilde{\mathbf{B}}^* \mathbf{R}^\perp \tilde{\mathbf{B}} \mathbf{e}_i + \mathbf{e}_i^* \mathbf{G}^* \mathbf{R}_{yy} \mathbf{G} \mathbf{e}_i \\ &\geq \mathbf{e}_i^* \tilde{\mathbf{B}}^* \mathbf{R}^\perp \tilde{\mathbf{B}} \mathbf{e}_i \\ \Rightarrow \sum_{i=0}^{n_i-1} \mathbf{e}_i^* \mathbf{R}_{ee} \mathbf{e}_i &\geq \sum_{i=0}^{n_i-1} \mathbf{e}_i^* \tilde{\mathbf{B}}^* \mathbf{R}^\perp \tilde{\mathbf{B}} \mathbf{e}_i \\ \Rightarrow \text{trace}(\mathbf{R}_{ee}) &\geq \text{trace}(\tilde{\mathbf{B}}^* \mathbf{R}^\perp \tilde{\mathbf{B}}) \end{aligned}$$

with equality if and only if $\mathbf{G} = \mathbf{0}$, which occurs when (12) is satisfied. Similarly

$$\begin{aligned} |\mathbf{R}_{ee}| &= |\tilde{\mathbf{B}}^* \mathbf{R}^\perp \tilde{\mathbf{B}}| \\ &\quad \cdot |\mathbf{I}_{n_i} + (\tilde{\mathbf{B}}^* \mathbf{R}^\perp \tilde{\mathbf{B}})^{-(1/2)} \mathbf{G}^* \mathbf{R}_{yy} \mathbf{G} (\tilde{\mathbf{B}}^* \mathbf{R}^\perp \tilde{\mathbf{B}})^{-(*/2)}| \\ &\geq |\tilde{\mathbf{B}}^* \mathbf{R}^\perp \tilde{\mathbf{B}}|. \end{aligned}$$

The last inequality follows from the positive semi-definiteness of $(\tilde{\mathbf{B}}^* \mathbf{R}^\perp \tilde{\mathbf{B}})^{-(1/2)} \mathbf{G}^* \mathbf{R}_{yy} \mathbf{G} (\tilde{\mathbf{B}}^* \mathbf{R}^\perp \tilde{\mathbf{B}})^{-(*/2)}$.

APPENDIX B

PROOF THAT (17) AND (18) ARE EQUIVALENT TO (24) AND (25), RESPECTIVELY

Starting with (24) and using the formula for the inverse of a block matrix given on [18, p. 656], we get

$$\begin{aligned} \bar{\mathbf{R}}_{12}^\perp &= \mathbf{C}^* \mathbf{R}_{12}^\perp \stackrel{\text{def}}{=} -\mathbf{C}^* \mathbf{R}_{11}^{-1} \mathbf{R}_{12} \mathbf{Q}^{-1} \\ (\mathbf{R}_{22}^\perp)^{-1} &\stackrel{\text{def}}{=} \mathbf{Q} \end{aligned}$$

where $\mathbf{Q} \stackrel{\text{def}}{=} \mathbf{R}_{22} - \mathbf{R}_{12}^* \mathbf{R}_{11}^{-1} \mathbf{R}_{12}$. Starting from (24)

$$\mathbf{B}_{opt}^* = -\bar{\mathbf{R}}_{12}^\perp (\mathbf{R}_{22}^\perp)^{-1} = \mathbf{C}^* \mathbf{R}_{11}^{-1} \mathbf{R}_{12}$$

which is identical to (17). Similarly, starting from (25)

$$\begin{aligned} \mathbf{R}_{ee, \min} &= \bar{\mathbf{R}}_{11}^\perp - \bar{\mathbf{R}}_{12}^\perp (\mathbf{R}_{22}^\perp)^{-1} \bar{\mathbf{R}}_{12}^{\perp*} \\ &= \mathbf{C}^* (\mathbf{R}_{11}^\perp - \mathbf{R}_{12}^\perp (\mathbf{R}_{22}^\perp)^{-1} \mathbf{R}_{12}^{\perp*}) \mathbf{C} = \mathbf{C}^* \mathbf{R}_{11}^{-1} \mathbf{C} \end{aligned}$$

which is identical to (18).

APPENDIX C

OPTIMUM MIMO MMSE-DFE SETTINGS FOR SCENARIO 2

Starting from (14) and (19), we have

$$\begin{aligned} \text{trace}(\mathbf{R}_{ee}) &= \sum_{k=0}^{n_i-1} \mathbf{e}_k^* \tilde{\mathbf{B}}^* \mathbf{L}^{-*} \mathbf{D}^{-1} \mathbf{L}^{-1} \tilde{\mathbf{B}} \mathbf{e}_k \\ &\stackrel{\text{def}}{=} \sum_{k=0}^{n_i-1} \tilde{\mathbf{b}}_k^* \mathbf{L}^{-*} \mathbf{D}^{-1} \mathbf{L}^{-1} \tilde{\mathbf{b}}_k \end{aligned}$$

where $\tilde{\mathbf{b}}_k^* \stackrel{\text{def}}{=} \tilde{\mathbf{B}} \mathbf{e}_k = [\mathbf{0}_{1 \times (n_i \Delta + k)} \quad 1 \quad \mathbf{b}_k^*]$. Therefore

$$\begin{aligned} \text{trace}(\mathbf{R}_{ee}) &= \sum_{k=0}^{n_i-1} \sum_{i=0}^{n_i(N_f+\nu)-1} d_i^{-1} \tilde{\mathbf{b}}_k^* \mathbf{L}^{-*} \mathbf{e}_i \mathbf{e}_i^* \mathbf{L}^{-1} \tilde{\mathbf{b}}_k \\ &= \sum_{k=0}^{n_i-1} \sum_{i=0}^{n_i(N_f+\nu)-1} d_i^{-1} |\mathbf{e}_i^* \mathbf{L}^{-1} \tilde{\mathbf{b}}_k|^2 \\ &\stackrel{\text{def}}{=} \sum_{k=0}^{n_i-1} \text{MSE}_k. \end{aligned}$$

To minimize this sum, we need to choose $\tilde{\mathbf{b}}_k$ to minimize each term MSE_k . It can be readily checked that MSE_0 is minimized by searching for the maximum diagonal elements of \mathbf{D} (call it $d_{n_i \Delta_{opt}}$), where $(0 \leq \Delta_{opt} \leq N_f + \nu - 1)$, and setting $\tilde{\mathbf{b}}_0 = \mathbf{L} \mathbf{e}_{n_i \Delta_{opt}}$, which results in MSE_0 equal to its minimum value of $d_{n_i \Delta_{opt}}^{-1}$. Since we are restricting all n_i users to have the same decision delay, it is clear that $\text{trace}(\mathbf{R}_{ee})$ is minimized by setting $\tilde{\mathbf{b}}_k = \mathbf{L} \mathbf{e}_{(n_i \Delta_{opt})+k}$ (where $0 \leq k \leq n_i - 1$), which is equivalent to (20) and results in

$$\begin{aligned} \text{trace}(\mathbf{R}_{ee, \min}) &= \sum_{k=0}^{n_i-1} \sum_{m=0}^{n_i(N_f+\nu)-1} d_{n_i m}^{-1} |\mathbf{e}_m^* \mathbf{e}_{\Delta_{opt}+k}|^2 \\ &= \sum_{k=0}^{n_i-1} d_{(n_i \Delta_{opt})+k}^{-1} \end{aligned} \quad (34)$$

which is identical to (21).

APPENDIX D

PROOF THAT SCENARIO 2 ACHIEVES LOWER MMSE THAN SCENARIO 1

The achievable MMSE under both scenarios can be expressed as

$$\text{MMSE} = \text{trace}(\mathbf{B}_0^* \mathbf{R}_3 \mathbf{B}_0). \quad (35)$$

Under Scenario 1, $\mathbf{B}_0 = \mathbf{I}_{n_i}$; therefore, the resulting MMSE, which is denoted by MMSE_1 , is given by

$$\text{MMSE}_1 = \text{trace}(\mathbf{R}_3). \quad (36)$$

Under Scenario 2, $\mathbf{B}_0 = \mathbf{L}_3$, where \mathbf{L}_3 is computed from the Cholesky factorization

$$\mathbf{R}_3^{-1} = \mathbf{L}_3 \mathbf{D}_3 \mathbf{L}_3^*. \quad (37)$$

Therefore, the achievable MMSE under Scenario 2, which is denoted by MMSE_2 , is

$$\text{MMSE}_2 = \text{trace}(\mathbf{L}_3^* \mathbf{R}_3 \mathbf{L}_3) = \text{trace}(\mathbf{D}_3^{-1}). \quad (38)$$

From (36) and (37), we get

$$\begin{aligned} \text{MMSE}_1 &= \text{trace}(\mathbf{L}_3^{-*} \mathbf{D}_3^{-1} \mathbf{L}_3^{-1}) \\ &= \sum_{k=0}^{n_i-1} \mathbf{e}_k^* \mathbf{L}_3^{-*} \mathbf{D}_3^{-1} \mathbf{L}_3^{-1} \mathbf{e}_k \\ &= \sum_{k=0}^{n_i-1} d_{3,k}^{-1} |\mathbf{L}_3^{-1} \mathbf{e}_k|^2 \\ &\geq \sum_{k=0}^{n_i-1} d_{3,k}^{-1} \cdot |\mathbf{L}_3^{-1} \mathbf{e}_k|^2 \geq 1 \\ &\quad \text{since } \mathbf{L}_3 \text{ is a lower triangular monic matrix} \\ &= \text{trace}(\mathbf{D}_3^{-1}) = \text{MMSE}_2. \end{aligned}$$

APPENDIX E

PROOF THAT SCENARIO 3 ACHIEVES LOWER MMSE THAN SCENARIO 2

Under Scenario 3, the achievable MMSE is given by [cf. (30)]

$$\text{MMSE}_3 = \sum_{i=0}^{n_i-1} \frac{1}{\mathbf{e}_i^* \mathbf{R}_3^{-1} \mathbf{e}_i}.$$

Starting from (37), we have

$$\begin{aligned} \mathbf{e}_i^* \mathbf{R}_3^{-1} \mathbf{e}_i &= \mathbf{e}_i^* \mathbf{L}_3 \mathbf{D}_3 \mathbf{L}_3^* \mathbf{e}_i \\ &\geq \mathbf{e}_i^* \mathbf{D}_3 \mathbf{e}_i \\ \Rightarrow \frac{1}{\mathbf{e}_i^* \mathbf{R}_3^{-1} \mathbf{e}_i} &\leq \mathbf{e}_i^* \mathbf{D}_3^{-1} \mathbf{e}_i \\ \Rightarrow \text{MMSE}_3 &= \sum_{i=0}^{n_i-1} \frac{1}{\mathbf{e}_i^* \mathbf{R}_3^{-1} \mathbf{e}_i} \leq \text{trace}(\mathbf{D}_3^{-1}) = \text{MMSE}_2. \end{aligned}$$

APPENDIX F

PROOF THAT MIMO MMSE-LE IS INFERIOR TO MIMO MMSE-DFE

Starting with (32)

$$\begin{aligned} \text{trace}(\mathbf{R}_{ee, \text{MMSE-LE}}) &= \sum_{k=0}^{n_i-1} \mathbf{e}_k^* \mathbf{R}_{ee, \text{MMSE-LE}} \mathbf{e}_k \\ &= \sum_{k=0}^{n_i-1} \mathbf{e}_{n_i \Delta_{opt} + k}^* \mathbf{L}^{-*} \mathbf{D}^{-1} \mathbf{L}^{-1} \mathbf{e}_{n_i \Delta_{opt} + k} \\ &\geq \sum_{k=0}^{n_i-1} d_{n_i \Delta_{opt} + k}^{-1} \\ &= \text{trace}(\mathbf{R}_{ee, \text{MMSE-DFE}}) \end{aligned}$$

where the last line follows from (21).

ACKNOWLEDGMENT

The authors would like to thank the Associate Editor for pointing out detection scenario 3.

REFERENCES

- [1] "Special Issue on High-Speed Digital Subscriber Lines," *IEEE J. Select. Areas Commun.*, vol. 9, Aug. 1991.
- [2] A. Duel-Hallen, "Equalizers for multiple input multiple output channels and PAM systems with cyclostationary input sequences," *IEEE J. Select. Areas Commun.*, vol. 10, pp. 630–639, Apr. 1992.
- [3] —, "A family of multiuser decision-feedback detectors for asynchronous code-division multiple-access channels," *IEEE Trans. Commun.*, vol. 43, pp. 421–434, Feb./Mar./Apr. 1995.
- [4] A. Naguib, V. Tarokh, N. Seshadri, and A. R. Calderbank, "A space-time coding modem for high-data-rate wireless communications," *IEEE J. Select. Areas Commun.*, vol. 16, pp. 1459–1477, Oct. 1998.
- [5] N. Al-Dhahir, "A computationally-efficient FIR MMSE-DFE for CCI-impaired dispersive channels," *IEEE Trans. Signal Processing*, vol. 45, pp. 32–41, Jan. 1997.
- [6] N. Al-Dhahir and J. M. Cioffi, "MMSE decision feedback equalizers: Finite-length results," *IEEE Trans. Inform. Theory*, vol. 41, pp. 961–976, July 1995.
- [7] —, "Efficient computation of the delay-optimized finite-length MMSE-DFE," *IEEE Trans. Signal Processing*, vol. 44, pp. 1288–1292, May 1996.
- [8] P. Balaban and J. Salz, "Optimum diversity combining and equalization in digital data transmission with applications to cellular mobile radio—Part I: Theoretical considerations," *IEEE Trans. Commun.*, vol. 40, pp. 885–894, May 1992.
- [9] A. W. Bojanczyk, R. P. Brent, F. R. de Hoog, and D. R. Sweet, "On the stability of the babess and related toeplitz factorization algorithms," *SIAM J. Matrix Anal. Applicat.*, vol. 16, pp. 40–57, Jan. 1995.
- [10] S. Chandrasekaran and A. H. Sayed, "Stabilizing the generalized Schur algorithm," *SIAM J. Matrix Anal. Applicat.*, vol. 17, no. 4, pp. 950–983, Oct. 1996.
- [11] A. Furuskar, S. Mazur, F. Muller, and H. Olofsson, "EDGE: Enhanced data rates for GSM and TDMA/136 evolution," *IEEE Personal Commun. Mag.*, June 1999.
- [12] G. Raleigh and J. Cioffi, "Spatio-temporal coding for wireless communication," *IEEE Trans. Commun.*, vol. 46, pp. 357–366, Mar. 1998.
- [13] G. J. Foschini and M. J. Gans, "On limits of wireless communication in a fading environment when using multiple antennas," *Wireless Personal Commun.*, pp. 311–335, Mar. 1998.
- [14] G. Golub and C. Van Loan, *Matrix Computations*, 2nd ed. Baltimore, MD: Johns Hopkins University Press, 1989.
- [15] J. Cioffi, W. Abbott, H. Thapar, C. Melas, and K. Fisher, "Adaptive equalization for magnetic disk storage channels," *IEEE Commun. Mag.*, pp. 14–29, Feb. 1990.

- [16] J. Yang and S. Roy, "Joint transmitter and receiver optimization for multiple-input-multiple-output systems with decision feedback," *IEEE Trans. Inform. Theory*, vol. 40, pp. 1334–1347, Sept. 1994.
- [17] —, "On joint transmitter and receiver optimization for multiple-input-multiple-output (MIMO) transmission systems," *IEEE Trans. Commun.*, vol. 42, pp. 3221–3231, Dec. 1994.
- [18] T. Kailath, *Linear Systems*. Englewood Cliffs, NJ: Prentice-Hall, 1980.
- [19] T. Kailath and A. H. Sayed, "Displacement structure: Theory and applications," *SIAM Rev.*, vol. 37, no. 3, pp. 297–386, Sept. 1995.
- [20] S. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Englewood Cliffs, NJ: Prentice-Hall, 1993.
- [21] H. Lev-Ari and T. Kailath, "Triangular factorization of structured hermitian matrices," in *Oper. Theory: Adv. Appl.*, I. Gohberg, Ed., 1986, vol. 18, pp. 301–324.
- [22] M. Stewart and P. Van Dooren, "Stability issues in the factorization of structured matrices," *SIAM J. Matrix Anal. Appl.*, pp. 104–118, 1997.
- [23] J. Proakis, *Digital Communications*, 3rd ed. New York: McGraw-Hill, 1995.
- [24] T. Kailath and A. H. Sayed, Eds., *Fast Reliable Algorithms for Matrices with Structure*. Philadelphia, PA: SIAM, 1999.
- [25] V. Tarokh, N. Seshadri, and A. R. Calderbank, "Space-time codes for high data rate wireless communications: Performance criterion and code construction," *IEEE Trans. Inform. Theory*, vol. 44, pp. 744–765, Mar. 1998.
- [26] S. Verdú, "Minimum probability of error for asynchronous Gaussian multiple-access channels," *IEEE Trans. Inform. Theory*, vol. 42, pp. 85–96, Jan. 1996.
- [27] —, *Multiuser Detection*. Cambridge, U.K.: Cambridge University Press, 1998.
- [28] Y. Li and K. Liu, "On blind MIMO channel identification using second-order statistics," in *Proc. Conf. Inform. Sci. Syst.*, 1996, pp. 1166–1171.

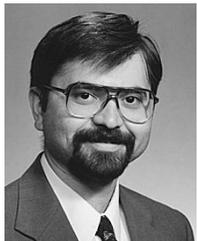


Ali H. Sayed (SM'99) received the Ph.D. degree in electrical engineering in 1992 from Stanford University, Stanford, CA.

He is Associate Professor of electrical engineering at the University of California, Los Angeles (UCLA). He has over 135 journal and conference publications, is co-author of the research monograph *Indefinite Quadratic Estimation and Control* (Philadelphia, PA: SIAM, 1999) and of the graduate-level textbook *Linear Estimation* (Englewood Cliffs, NJ: Prentice-Hall, 2000). He is also co-editor of the volume

Fast Reliable Algorithms for Matrices with Structure (Philadelphia, PA: SIAM, 1999). He is a member of the editorial boards of the *SIAM Journal on Matrix Analysis and Its Applications* and the *International Journal of Adaptive Control and Signal Processing*, has served as Co-Editor of special issues of the journal *Linear Algebra and Its Applications*. He has contributed several articles to engineering and mathematical encyclopedias and handbooks. His research interests span several areas including adaptive and statistical signal processing, filtering and estimation theories, equalization techniques for communications, interplays between signal processing and control methodologies, and fast algorithms for large-scale problems.

Dr. Sayed has served on the program committees of several international meetings. He is also a recipient of the 1996 IEEE Donald G. Fink Award. He is Associate Editor of the IEEE TRANSACTIONS ON SIGNAL PROCESSING.



Naofal Al-Dhahir (SM'98) received the M.S. and Ph.D. degrees from Stanford University, Stanford, CA, in 1990 and 1994, respectively, in electrical engineering.

He was an Instructor at Stanford University during the Winter of 1993. From August 1994 to July 1999, he was Member of Technical Staff at the Communications Program, GE Corporate R&D Center, Schenectady, NY, where he worked on various aspects of satellite communication systems design and anti-jam GPS receivers. Since August

1999, he has been a Principal Member of Technical Staff at AT&T Shannon Laboratory, Florham Park, NJ. His current research interests include equalization schemes, spatio-temporal processing, digital subscriber line technology, and third-generation cellular systems. He has authored 30 journal papers and holds three U.S. patents in the area of satellite communications.

Dr. Al-Dhahir is an Associate Editor of the IEEE TRANSACTIONS ON SIGNAL PROCESSING and IEEE COMMUNICATIONS LETTERS.