Multi-Input Multi-Output Fading Channel Tracking and Equalization Using Kalman Estimation

Christos Komninakis, *Student Member, IEEE*, Christina Fragouli, Ali H. Sayed, *Fellow, IEEE*, and Richard D. Wesel, *Senior Member, IEEE*

***Abstract*—This paper addresses the problem of channel tracking and equalization for multi-input multi-output (MIMO) time-varying frequency-selective channels. These channels model the effects of inter-symbol interference (ISI), co-channel interference (CCI), and noise. A low-order autoregressive model approximates the MIMO channel variation and facilitates tracking via a Kalman filter. Hard decisions to aid Kalman tracking come from a MIMO finite-length minimum-mean-squared-error decision-feedback equalizer (MMSE-DFE), which performs the equalization task. Since the optimum DFE for a wide range of channels produces decisions with a delay $\Delta > 0$, the Kalman filter tracks the channel with a delay. A channel prediction module bridges the time gap between the channel estimates produced by the Kalman filter and those needed for the DFE adaptation. The proposed algorithm offers good tracking behavior for multiuser fading ISI channels at the expense of higher complexity than conventional adaptive algorithms. Applications include synchronous multiuser detection of independent transmitters, as well as coordinated transmission through many transmitter/receiver antennas, for increased data rate.**

***Index Terms*—Equalization, MIMO systems, multichannel tracking.**

I. Introduction

THIS paper considers the problem of channel tracking and equalization of a multi-input multi-output (MIMO) (n_T, n_R) wireless system, where $n_T \geq 1$ is the number of transmitter antennas, and $n_R \geq 1$ is the number of receiver antennas. In most of the paper, the discussion is general enough to cover both the case of n_T independent users with one antenna each (which can be thought of as the multi-user detection problem, for instance, in wideband multiple access systems [1]) and the case of one user with coordinated transmissions through n_T antennas for higher data rate (which is akin to the systems of [2], [3]). We assume that the time variation of the MIMO channel within a packet is significant; thus, channel tracking is needed for the equalization to be effective. For MIMO tracking, we use a Kalman filter aided by staggered decisions from a finite-length MMSE-DFE, which performs the task of equalization and separation of the sources.

For the problem of adaptive equalization of a single fading dispersive channel $(n_T = n_R = 1)$, a comprehensive review of the extensive research is [4]. In [5], we present a special case of the results herein, which is applicable to this scenario. For array measurements at the receiver $(n_T = 1, n_R > 1)$, an adaptive approach based on per-survivor-processing (PSP) is explored in [6], but when more transmitters are sharing the bandwidth, there are two broad classes of techniques to combat co-channel interference (CCI) at the receiver. One is to suppress interference, possibly in an adaptive fashion, as in [7]. Another strategy is to decode all n_T data sequences simultaneously (e.g., [8]), possibly with a blind/adaptive approach [9]. The method we demonstrate here embraces the second paradigm, with a few key differences from previous approaches. First, we adopt the design of a finite-length MMSE-DFE from [10] for practical implementation. Then, we consider the channel taps to have significant time-variation from symbol to symbol but with largely invariant specular mean and Doppler, which can be identified during a training phase. After that, in tracking mode, the Kalman filter and this finite MIMO DFE cooperate to adapt to the rapid channel variations.

It should be noted that, at least for the single-user channel, Kalman-based estimation methods are quite common in the literature (e.g., [11] uses the extended Kalman filter to track a channel with unknown delays, and [12] discusses a special case of the problem herein for $n_T = n_R = 1$, first-order autoregressive channel modeling and no decision delay $\Delta = 0$ in the DFE). In addition, in [13], the Kalman approach is used to formulate extended forms of the recursive least-squares (RLS) algorithm, and the tracking superiority of those is demonstrated compared with the standard RLS and least mean-squares (LMS) algorithms. Here, we use a Kalman filter to track the time variation of the MIMO channel taps. These taps are typically modeled as mutually uncorrelated circular complex Gaussian random processes, having locally constant means, due to large scale path loss, reflections, and shadowing effects. We assume the tap means are known from a preceding training phase and concentrate on tracking their time-variant part, which has autocorrelation properties corresponding to the wide-sense stationary and uncorrelated scattering "WSSUS" model of Bello [14]. If the tap means are zero, the channel is said to introduce Rayleigh fading (worst case), whereas a nonzero mean tap corresponds to Ricean fading. The Kalman channel estimator is aided by previous hard decisions about

Manuscript received November 17, 2000; revised January 28, 2002. This work was supported by the National Science Foundation under Grants CCR-9732376 and ECS-9820765, NSF CAREER award CCR-9733089, Texas Instruments, Xetron Corporation, and the Defense Advanced Research Projects Agency under Contract N00014-99-C-0328. Part of this paper was presented at the 2000 ICC, New Orleans, LA, June 2000. The associate editor coordinating the review of this paper and approving it for publication was Dr. Olivier Besson.

C. Komninakis is with Broadcom Corporation, El Segundo, CA 90245 USA.

C. Fragouli is with the National Capodistrean University of Athens, Athens, Greece.

A. H. Sayed, and R. D. Wesel are with the Department of Electrical Engineering, University of California, Los Angeles CA 90095 USA (e-mail: sayed@ee.ucla.edu; wesel@ee.ucla.edu).

Publisher Item Identifier S 1053-587X(02)03285-3.

the transmitted symbols from all users produced by the MIMO equalizer.

Assuming perfect knowledge of the MIMO channel, the optimum receiver is a maximum likelihood sequence estimator (MLSE), but its complexity is prohibitive, even for low-order channels with a small number of inputs and outputs. Here, we use the MIMO finite-length minimum-mean-squared-error decision-feedback equalizer (MMSE-DFE), which was developed in [10] and optimized for decision delay $\Delta \geq 0$. The choice of $\Delta > 0$ improves performance for a wide range of channels, as shown in [5] for the single-user channel. However, this delay poses the problem of channel prediction when combined with the Kalman tracking procedure mentioned previously because there is a time gap of Δ between channel estimates produced by the Kalman filter (aided by the delayed DFE decisions) and the channel estimates needed for the current DFE adaptation. We discuss simple methods to bridge this time gap and show simulation results to demonstrate that the joint tracking and equalization algorithm proposed in this paper offers good performance. In fact, it outperforms conventional adaptive equalization algorithms such as LMS or RLS. These algorithms do not have an explicit mechanism for incorporating the largely invariant channel statistics, such as the Doppler rate and the channel mean, in case they are known to the receiver from a previous training phase.

The paper is organized as follows. Section II presents the channel model. Section III introduces the receiver block diagram and discusses the Kalman-based tracking, the channel prediction, and the delay-optimized adaptive DFE design. Section IV presents simulation results of the proposed algorithm, using the lower complexity LMS and RLS adaptive algorithms for the MIMO DFE as a useful baseline for performance comparison. Finally, Section V concludes the paper.

II. Channel Model

Each receiver antenna of the n_T-input, n_R-output MIMO channel observes a linear combination of all transmitted data sequences, each distorted by ISI, under white Gaussian noise. Specifically, the observable $y_t^{(j)}$ from receiver j (with $j = 1, \ldots, n_R$) at time t is

$$y_t^{(j)} = \sum_{i=1}^{n_T} \sum_{m=0}^{\nu^{(i,j)}} c_m^{(i,j)}(t) x_{t-m}^{(i)} + v_t^{(j)}, \tag{1}$$

where $c_m^{(i,j)}$ is the mth tap of the impulse response of order $\nu^{(i,j)}$ between the ith input $x^{(i)}$ and the jth output $y^{(j)}$ of the MIMO channel. The complex baseband constellation point $x_{t-m}^{(i)}$ is transmitted by the ith user at time $t-m$, and $v_t^{(j)}$ is the complex noise sample at the jth receiver. In essence, there exist a total of $n_T n_R$ interfering, time-varying ISI channels $\mathbf{c}^{(i,j)}(t)$

$$\mathbf{c}^{(i,j)}(t) = \left[c_0^{(i,j)}(t) \quad c_1^{(i,j)}(t) \quad \cdots \quad c_{\nu^{(i,j)}}^{(i,j)}(t) \right]. \tag{2}$$

Each of the taps in (2) can be written as

$$c_m^{(i,j)}(t) = \bar{c}_m^{(i,j)} + h_m^{(i,j)}(t), \quad m = 0, \ldots, \nu \tag{3}$$

where $\bar{c}_m^{(i,j)}$ is the tap mean, and we define $\nu = \max_{i,j} \nu^{(i,j)}$ and set $c_m^{(i,j)}(t) = 0$ for $m > \nu^{(i,j)}$.

A clear way to represent the time variation of the MIMO channel taps is to rewrite the input–output relationship of (1) in vector form, collecting the outputs $y_t^{(j)}$ from all receiver antennas at time t into an n_R-dimensional column vector $\mathbf{y}_t$:

$$\mathbf{y}_t = \mathbf{X}_t \cdot \mathbf{c}_t + \mathbf{v}_t \tag{4}$$

where the $n_R \times n_R n_T(\nu+1)$ data matrix $\mathbf{X}_t$ is a "wide" matrix with the transmitted symbols repeated diagonally, according to the Kronecker product

$$\mathbf{X}_t = \left[x_t^{(1)} \ldots x_t^{(n_T)} x_{t-1}^{(1)} \ldots x_{t-1}^{(n_T)} \ldots x_{t-\nu}^{(1)} \ldots x_{t-\nu}^{(n_T)} \right] \otimes \mathbf{I}_{n_R} \tag{5}$$

and $\mathbf{c}_t$ is a long vector of length $n_T n_R(\nu+1)$ containing all the channel taps at time t, $\mathbf{c}_t = \bar{\mathbf{c}} + \mathbf{h}_t$

$$\mathbf{c}_t = \Big[c_0^{(1,1)}(t) \ldots c_0^{(1,n_R)}(t) \ldots c_0^{(n_T,1)}(t) \ldots c_0^{(n_T,n_R)}(t) \ldots c_\nu^{(1,1)}(t) \ldots c_\nu^{(1,n_R)}(t) \ldots c_\nu^{(n_T,1)}(t) \ldots c_\nu^{(n_T,n_R)}(t) \Big]^T. \tag{6}$$

With this setup, the channel is a complex Gaussian vector process $\mathbf{c}_t$ with a constant mean vector $\bar{\mathbf{c}}$ and a time-variant part $\mathbf{h}_t$, all of dimensions $(n_T n_R(\nu+1)) \times 1$.

A. Bello's Model

According to the WSSUS model of Bello [14], all the channel taps are independent; therefore, the time-varying part $h_m^{(i,j)}(t)$ of each tap in (3) is a zero-mean, wide-sense-stationary complex Gaussian process, uncorrelated with any other $h_{m'}^{(i',j')}(t)$, and has time-autocorrelation properties governed by the Doppler rate $f_D T$ (T is the baud duration) as in [15]

$$E\left\{ h_m^{(i,j)}(t_1) \left[h_m^{(i,j)}(t_2) \right]^* \right\} \sim \mathcal{J}_o\left(2\pi f_D^{(i,j,m)} T |t_1 - t_2| \right) \tag{7}$$

where $\mathcal{J}_o(\cdot)$ is the zero-order Bessel function of the first kind. Each of the $n_T n_R(\nu+1)$ taps can change independently with a different Doppler rate, although this will not be essential in our development. This issue of tap independence is revisited in Section IV-C. In the long channel vector setup of (6), all the entries of the vector process $\{\mathbf{h}_t\}$ evolve independently, according to the autocorrelation model of (7)).

If we let index k enumerate all the taps $k = 1, \ldots, n_T n_R(\nu+1)$ and denote $f_D^{(k)} = f_D^{(i,j,m)}$ the Doppler of the mth tap of the channel from input i to output j, then the normalized spectrum for each tap $\{h_t^{(k)}\}$ is

$$S_k(f) = \begin{cases} \dfrac{1}{\pi f_D^{(k)} T} \dfrac{1}{\sqrt{1 - \left(\dfrac{f}{f_D^{(k)} T} \right)^2}}, & |f| < f_D^{(k)} T \\ 0, & \text{otherwise.} \end{cases} \tag{8}$$

B. Autoregressive Model

Exact modeling of the time evolution of the vector process $\{\mathbf{h}_t\}$ with an autoregressive moving-average (ARMA) model is impossible because the autocorrelation functions are nonrational. Accurate but large-order AR models for the fading

channel are presented in [16]. However, the first few correlation terms of (7) for small lag $|t_1 - t_2|$ are more important for the design of the receiver. Thus, even low-order autoregressive models, or even a simple Gauss-Markov model, matching the Bessel autocorrelation well for small lags, can capture most of the channel tap dynamics and lead to effective tracking algorithms, as demonstrated later. To this end, we approximate the MIMO channel variation $\{\mathbf{h}_t\}$ with the following multichannel AR process of order p, as done in [12] for $n_T = n_R = 1$:

$$\mathbf{h}_t = \sum_{l=1}^{p} \mathbf{A}(l)\mathbf{h}_{t-l} + \mathbf{G}_0\mathbf{w}_t \tag{9}$$

where $\mathbf{w}_t$ is a zero-mean i.i.d. circular complex Gaussian vector process with correlation matrices $\mathbf{R}_{ww}(j)$ for every lag j given by $\mathbf{R}_{ww}(j) = E\{\mathbf{w}_t\mathbf{w}_{t+j}^*\} = \mathbf{I}_{n_T n_R(\nu+1)}\delta(j)$. Due to the WSSUS assumption, the matrices $\mathbf{A}(l), l = 1, \ldots, p$, and $\mathbf{G}_0$ of the model (9) must be diagonal. For the selection of their diagonal entries, various criteria of optimality can be adopted, such that the AR(p) model of (9) would be a "best-fit" to the true channel autocorrelation of (7). One such criterion can be to require the process (9) to be such that 90% of the energy spectrum of each tap is contained in the frequency range $|f| < f_D^{(k)}T$, as indicated by (8). In this paper, we use the so-called "correlation-matching" property of the AR(p) model [17] and equate the autocorrelations of (9) and (7) for the first $p+1$ lags, deeming the matching of those to be more important than matching terms of larger time-lag. This method amounts to solving the linear system of the p Yule–Walker equations [18] with respect to the coefficients of the AR(p) model and provides good autocorrelation matching even for low orders p. Of course, for lags greater than p, the autocorrelation of the AR(p) model extrapolated by a recursive difference equation [18] differs from the true autocorrelation. Fig. 1 shows that for $p = 2$, the AR(2) autocorrelation matches the true Bessel autocorrelation well for lags less than 20, whereas for $p = 1$, the matching is satisfactory only for very few small lags.

The multichannel AR(p) model of (9) for the time varying part of the channel $\mathbf{h}_t$ can be written in matrix state-space form as

$$\mathbf{z}_t = \mathbf{F}\mathbf{z}_{t-1} + \mathbf{G}\mathbf{w}_t \tag{10}$$

where the long vector $\mathbf{z}_t$ of length $p(\nu+1)n_T n_R$ contains the realization of the time-varying part of the vector channel $\mathbf{h}_t$ for p consecutive times

$$\mathbf{z}_t = [\mathbf{h}_t^T \quad \mathbf{h}_{t-1}^T \quad \cdots \quad \mathbf{h}_{t-p+1}^T]^T \tag{11}$$

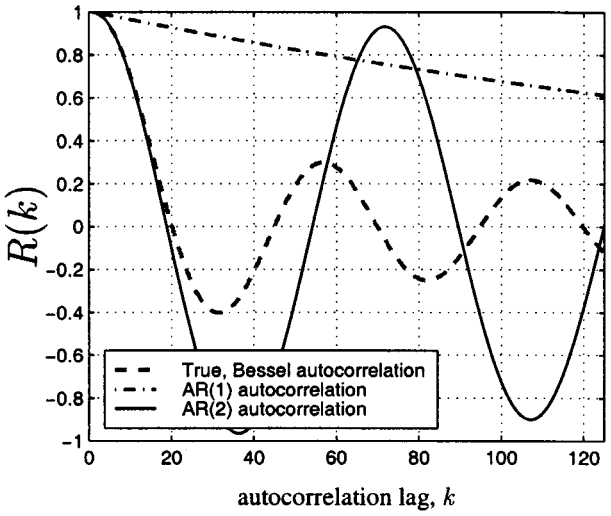

Fig. 1. Autocorrelation function $R(k)$ true (Bessel) and for the AR(p) model for $p = 1, 2$ and Doppler rate $f_D T = 0.02$. The second-order AR model autocorrelation matches the true expression for lag < 20, although only the first three terms are exactly equal.

and the matrices $\mathbf{F}$ and $\mathbf{G}$ are as in (12) and (13), shown at the bottom of the page. After choosing the order p for the AR model, we fix the entries of matrices $\mathbf{F}$ and $\mathbf{G}$ of (10), i.e., the diagonal entries of the matrices $\mathbf{A}(l), l = 1, \ldots, p$ and $\mathbf{G}_0$ of (9). Call those diagonal entries $a_k(l)$ and g_k, respectively, $k = 1, \ldots, n_T n_R(\nu+1)$ for each tap. When modeling the Ricean channel, the objective is to capture the most essential parts of the channel variation, namely, how "fast" and how "much" the time-varying part $h^{(k)}$ of each channel tap varies with respect to the known mean of that tap $\bar{c}^{(k)}$. The speed of variation is determined by the Doppler or, equivalently, by the relative velocity between the n_T antennas of the transmitter and the n_R antennas of the receiver. At least for the scenario of coordinated transmission, the assumption of equal Doppler rates $f_D^{(k)}T = f_D T, k = 1, \ldots, n_T n_R(\nu+1)$ makes intuitive sense; therefore, we adopt it in the simulations, although the algorithm derivation certainly does not rely on that. Different Doppler rates $f_D^{(k)}T$ for each tap can be accommodated. Larger Doppler rate $f_D T$ implies faster channel variation, hence, more diversity but a more difficult estimation task. In any case, the Doppler rate uniquely specifies a Bessel autocorrelation, as in (7). Then, the entries $a_k(l), k = 1, \ldots, n_T n_R(\nu+1), l = 1, \ldots, p$ of matrix $\mathbf{F}$ are determined from the Yule–Walker system, as explained previously. For example, for an AR(1) system, $\mathbf{z}_t = \mathbf{h}_t$, and $\mathbf{F} = \mathbf{A}(1)$ is diagonal with entries $a_k(1) = \mathcal{J}_o(2\pi f_D T)$, which makes the autocorrelation of the taps modeled by (9) equal the true autocorrelation at unit lag.

$$\mathbf{F} = \begin{bmatrix} \mathbf{A}(1) & \mathbf{A}(2) & \cdots \mathbf{A}(p-1) & \mathbf{A}(p) \\ \mathbf{I}_{(p-1)n_T n_R(\nu+1)} & & & \mathbf{0}_{(p-1)n_T n_R(\nu+1)\times n_T n_R(\nu+1)} \end{bmatrix} \tag{12}$$

$$\mathbf{G} = \begin{bmatrix} \mathbf{G}_0 \\ \mathbf{0}_{(p-1)n_T n_R(\nu+1)\times n_T n_R(\nu+1)} \end{bmatrix}. \tag{13}$$

Having fixed the *rate* of channel variation via $\mathbf{F}$, the magnitude of variation of the kth tap is then controlled by the diagonal entries g_k of $\mathbf{G}_0$ since the power of the time-variant part of each tap is $E|h^{(k)}|^2$, and it is proportional to g_k^2, $k = 1, \ldots, n_T n_R(\nu+1)$. The amount of variation of each tap with respect to the mean is expressed by the "specular-to-diffuse power ratio" K_k as

$$K_k \stackrel{\text{def}}{=} 10\log\left(\frac{|\bar{c}^{(k)}|^2}{E\left\{\left|h^{(k)}\right|^2\right\}}\right). \tag{14}$$

The ratio K_k indicates the ratio of power of the kth mean channel tap to the mean-squared power of the random, time-variant part of that tap, which is analogous to the Ricean factor defined in the wireless channel literature. Clearly, $K = -\infty$ dB corresponds to Rayleigh taps, whereas large positive values of K in decibels represent almost no channel variation at all. In the simulations, although, again, it is arbitrary, we choose equal values of $K_k = K$ for all the taps. By specifying the magnitude of change and the rate of change of the time-varying part of the channel $\mathbf{h}_t$ through the parameters K and $f_D T$, respectively, one can immediately construct an AR(p) model to approximate the true dynamics of the channel, as described in (7). Taking AR(1)—similarly for any order p—as an example, the ratio K_k uniquely specifies g_k because $E|h^{(k)}|^2 = g_k^2/(1-a_k(1))^2$, and $a_k(1)$ has already been determined from the Doppler rate. For perspective, in a 2.4-GHz transmission with baud rate of 40 kHz and Doppler frequency $f_D = 200$ Hz (corresponding to vehicular velocity of 90 Km/h or 56 mi/h), $f_D T = 0.02$. Hence, for an AR(1) model, $a_k(1) = 0.9961$, and for an AR(2) model, $a_k(1) = 1.9901$, and $a_k(2) = -0.9980$. In addition, a value of $K = 6$ dB implies that the average power of each tap variation is one fourth of the constant mean tap value. In the AR(1) case, this sets $g_k = |\bar{c}^{(k)}|\sqrt{(1-a_k^2(1))}10^{K/20} = 0.0445|\bar{c}^{(k)}|$, whereas in the AR(2) case, $g_k = 0.0018|\bar{c}^{(k)}|$.

The modeling inaccuracy of the AR(p) approximation can be made arbitrarily small by increasing the order p [as $p \to \infty$, the autocorrelation of the AR(p) model will equal (7) for all lags]. However, the complexity of the tracking algorithm described next increases, making this impractical, and actually unnecessary, because it turns out that first- or second-order approximations are enough to model the channel dynamics to the extent necessary for a receiver to operate. This is shown in the simulation section.

For the single-input single-ouput (SISO) case (i.e., $n_T = 1, n_R = 1$), a useful method to obtain the sequence of matrices $\mathbf{A}(l), l = 1, \ldots, p$ during a training mode is provided in [12] via higher than second-order statistics (HOS). For the single channel case, their method is effective and requires only reasonable assumptions about the transmitted sequence and the noise. An analogous training method can be adopted for general MIMO channels. However, the way we formulated the AR(p) model of (10) here, the information needed to construct the model is only the channel mean $\bar{\mathbf{c}}$, the Doppler rates, the ratios K_k, and the noise variance. Hence, we assume that these quantities are known from a training phase and focus on decision-aided tracking of the channel for relatively long time spans without retraining.

III. Receiver Structure

The receiver uses a Kalman filter to track the channel and an MMSE-DFE to equalize it. The Kalman filter assumes that the DFE hard decisions are correct and uses them to estimate the next channel value, whereas the DFE assumes correct Kalman filter channel estimates and uses them in turn to equalize the channel. In general, the optimum decision delay $\Delta \geq 0$ can be determined analytically given a channel (see [10]). For a wide range of channels (including, but not limited to, nonminimum-phase channels), it turns out that a DFE producing decisions with $\Delta > 0$ is optimal. Even for the few channels where $\Delta = 0$ is best, it does not degrade performance to use a DFE with $\Delta > 0$, provided that there are enough taps in the feedforward and feedback filters. Thus, it makes sense, particularly for time-varying channels like the ones treated here, to use decision delays $\Delta > 0$.

However, when $\Delta > 0$, a time gap is created. At time t, when the last received vector is $\mathbf{y}_t$, the DFE produces the hard-decision $\hat{\mathbf{x}}_{t-\Delta}$. The staggered decisions cause the Kalman filter to operate with delay, that is, operate at time $t-\Delta$ since it only has available hard decisions from the DFE up to then. However, the DFE design still needs channel estimates up to time t. Thus, the receiver needs to use channel prediction to bridge the time gap between the Kalman channel estimation and the channel estimates needed for the current DFE adaptation.

The proposed system block diagram of Fig. 2 shows the time succession of steps 1) through 4), which follow. The notation $\mathbf{r}_{n_1}^{n_2}$ (or $r_{n_1}^{n_2}$) means the collection of vector-valued (or scalar) variables $\mathbf{r}_{n_1} \ldots \mathbf{r}_{n_2}$ (or $r_{n_1} \ldots r_{n_2}$). In Fig. 2, the flow of new information is clockwise, starting from top left, with each of the blocks corresponding to one of the following actions:

1) $\hat{\mathbf{h}}_{t-\Delta} = \mathcal{K}(\hat{\mathbf{h}}_{t-\Delta-p}^{t-\Delta-1}, \mathbf{y}_{t-\Delta-1}, \hat{\mathbf{x}}_{t-\Delta-\nu-1}^{t-\Delta-1})$.
2) $\hat{\mathbf{h}}_{t-\Delta+1}^{t} = \mathcal{P}(\hat{\mathbf{h}}_{t-\Delta-p+1}^{t-\Delta}, \mathbf{y}_{t-\Delta}^{t})$.
3) $[\mathbf{W}_t^{\text{opt}}, \mathbf{B}_t^{\text{opt}}] =$ design DFE$(\hat{\mathbf{h}}_{t-N_f}^{t})$.
4) $\hat{\mathbf{x}}_{t-\Delta} =$ DFE$(\mathbf{W}_t^{\text{opt}}, \mathbf{B}_t^{\text{opt}})$.

The iteration starts with the well-known Kalman filter recursions denoted by $\mathcal{K}(\cdot)$, which at time t yield the optimum linear estimator [based on the AR(p) model of (9) or (10)] of the time-varying part of the channel $\hat{\mathbf{h}}_{t-\Delta}$ as it was at time $t-\Delta$. For that, the Kalman filter relies on the (assumed reliable) DFE decisions $\hat{\mathbf{x}}_{t-\Delta-1}, \ldots, \hat{\mathbf{x}}_{t-\Delta-\nu-1}$, the received vector $\mathbf{y}_{t-\Delta-1}$, and p previously estimated channel vectors. In the second step, $\mathcal{P}(\cdot)$ denotes a predictor that may exploit the additional received vectors $\mathbf{y}_t, \ldots \mathbf{y}_{t-\Delta}$, along with the most recent Kalman estimates $\hat{\mathbf{h}}_{t-\Delta-p+1}, \ldots, \hat{\mathbf{h}}_{t-\Delta}$ to compute the sequence of Δ predicted channels $\hat{\mathbf{h}}_{t-\Delta+1}, \ldots, \hat{\mathbf{h}}_t$.

Those Δ predicted channels, along with the $N_f - \Delta$ most recent channel estimates from the Kalman filter, are used by the DFE design module (see Section III-B) to design the optimum feedforward matrix filter $\mathbf{W}_t^{\text{opt}}$ and the feedback matrix filter $\mathbf{B}_t^{\text{opt}}$ of an MMSE-DFE. Finally, the newly designed DFE decodes one more n_T-dimensional symbol $\hat{\mathbf{x}}_{t-\Delta}$. This is added to

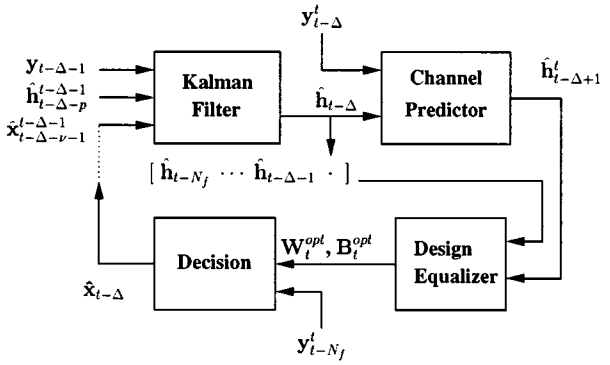

Fig. 2. Receiver block diagram.

the collection of past (assumed reliable) decisions, which will help the Kalman filter make a new channel estimate $\hat{\mathbf{h}}_{t-\Delta+1}$ at the next iteration that takes place at time instant $t+1$. In the following subsections, we look at the implementation of the two main receiver modules in greater detail.

A. Kalman Tracking and Channel Prediction

For notational convenience, we describe the Kalman tracking when the receiver uses an AR(1) channel model, where in (10), we have $\mathbf{z}_t = \mathbf{h}_t$. The extension to higher order AR models is straightforward. The channel at time t has a constant (assumed known) mean $\bar{\mathbf{c}}$ and a zero-mean time-varying part $\mathbf{h}_t$, which follows the AR(1) model

$$\mathbf{h}_{t+1} = \mathbf{F}\mathbf{h}_t + \mathbf{G}\mathbf{w}_t. \tag{15}$$

At time t, the (zero-mean) received vector $\mathbf{y}_t$ is given by

$$\mathbf{y}_t = \mathbf{X}_t \cdot (\bar{\mathbf{c}} + \mathbf{h}_t) + \mathbf{v}_t. \tag{16}$$

Assuming the matrices $\mathbf{F}$ and $\mathbf{G}$ and the mean channel vector $\bar{\mathbf{c}}$ are known from a preceding training phase and assuming the matrix of the most recent available decisions $\hat{\mathbf{X}}_{t-\Delta-1}$ to be equal to the true $\mathbf{X}_{t-\Delta-1}$ defined in (5), the receiver can use the Kalman filter to track the channel variation $\mathbf{h}_{t-\Delta}$, using as observable the vector $\mathbf{y}_{t-\Delta-1} - \hat{\mathbf{X}}_{t-\Delta-1}\bar{\mathbf{c}}$. The Kalman filter operating with a delay Δ is described at time t by the series of equations [19]

$$\begin{aligned}
\hat{\mathbf{h}}_{t-\Delta} &= \mathbf{F}\hat{\mathbf{h}}_{t-\Delta-1} + \mathbf{K}_{t-\Delta-1}\mathbf{e}_{t-\Delta-1} \\
\mathbf{e}_{t-\Delta-1} &= \mathbf{y}_{t-\Delta-1} - \hat{\mathbf{X}}_{t-\Delta-1}(\hat{\mathbf{h}}_{t-\Delta-1} + \bar{\mathbf{c}}) \\
\mathbf{K}_{t-\Delta-1} &= (\mathbf{F}\mathbf{P}_{t-\Delta-1}\hat{\mathbf{X}}^*_{t-\Delta-1})\mathbf{R}^{-1}_{e,t-\Delta-1} \\
\mathbf{R}_{e,t-\Delta-1} &= \mathbf{R}_{vv} + \hat{\mathbf{X}}_{t-\Delta-1}\mathbf{P}_{t-\Delta-1}\hat{\mathbf{X}}^*_{t-\Delta-1} \\
\mathbf{P}_{t-\Delta} &= \mathbf{F}\mathbf{P}_{t-\Delta-1}\mathbf{F}^* + \mathbf{G}\mathbf{G}^* \\
&\quad - \mathbf{K}_{t-\Delta-1}\mathbf{R}_{e,t-\Delta-1}\mathbf{K}^*_{t-\Delta-1}.
\end{aligned} \tag{17}$$

The above Kalman recursions implement the optimum linear estimator for the time-varying part of the channel $\mathbf{h}_{t-\Delta}$, assuming the taps follow an AR(1) model. For a general AR(p) channel model, we rewrite (15) and (16) as

$$\mathbf{z}_{t+1} = \mathbf{F}\mathbf{z}_t + \mathbf{G}\mathbf{w}_t \tag{18}$$

$$\mathbf{y}_t = \begin{bmatrix}\mathbf{X}_t & \mathbf{0}_{n_R\times(p-1)n_T n_R(\nu+1)}\end{bmatrix} \cdot (\bar{\mathbf{c}} + \mathbf{h}_t) + \mathbf{v}_t \tag{19}$$

and construct the Kalman recursion similarly. In this case, $\mathbf{h}_t$ consists of the first entries of $\mathbf{z}_t$, as shown in (11). Regardless of the order p of the channel model, the last reliable decision made by the DFE and used by the Kalman filter at time t is $\hat{\mathbf{x}}_{t-\Delta-1}$. For matrices $\mathbf{F}$ and $\mathbf{G}$ that are multiples of the identity (produced, for instance, by an AR(1) model of uncorrelated fading with the same Doppler and ratio K_k for all taps), fast algorithms for the above Kalman recursions can be pursued (see, e.g., [20]). For channels with nonindependent taps, either because of spatial correlation as in the uplink channel model of [21] or because of time-correlation induced by the shaping filters at the transmitter and receiver, the essence of the recursion does not change. An additional "shaping" matrix $\mathbf{S}$ is introduced after $\mathbf{X}$ (replace $\mathbf{X}$ with $\mathbf{X}\cdot\mathbf{S}$) to model the correlation of the channel taps, but the vector $\mathbf{h}_t$ containing the independent variations remains the objective of the tracking. The correlation introduced because of shaping is discussed further in Section IV.

Note that for the block-constant fading channel model adopted in the space-time literature [3], nothing else changes in the model of (15), except for setting $\mathbf{F} = \mathbf{I}$ and $\mathbf{G} = 0$, which simplifies the Kalman recursions significantly. This makes the Ricean factors K_k in (14) useless because if the channel remains unchanged for the whole block, the distinction between the mean $\bar{\mathbf{c}}$ and $\mathbf{h}$ in (6) is arbitrary. However, in this case of block constant channel, adaptation of the MIMO DFE at every time is not warranted. A few iterations of the algorithm at the beginning of each block should be enough to adapt the matrix coefficients to the constant channel. Then, their values can remain fixed for the rest of the block.

In channels with smaller coherence time (higher Doppler), adapting the MIMO DFE every time instant improves performance. For designing the DFE at time t (step 3), the N_f most recent channel estimates are needed, where N_f is the order of the matrix feedforward filter $\mathbf{W}^{\text{opt}}$ of the DFE. Up to time $t-\Delta$, channel estimates are available from the Kalman filter, but the last Δ channel vectors $\hat{\mathbf{h}}_t, \ldots, \hat{\mathbf{h}}_{t-\Delta+1}$ have to be predicted. The implementation of the prediction depends on the SNR of operation and how fast the channel varies. For the block-constant channel or a very slowly varying one, the simplest choice is to assume that the channel remains constant over Δ sampling periods, that is

$$\hat{\mathbf{h}}_t = \hat{\mathbf{h}}_{t-1} = \cdots = \hat{\mathbf{h}}_{t-\Delta} \tag{20}$$

where $\hat{\mathbf{h}}_{t-\Delta}$ is already provided by the Kalman filter.

More generally, the optimal linear predictions, given that the channel follows the AR(1) model of (15) but ignoring the additional received vectors $\mathbf{y}_t, \ldots, \mathbf{y}_{t-\Delta}$, are

$$\hat{\mathbf{h}}_t = \mathbf{F}^{\Delta}\hat{\mathbf{h}}_{t-\Delta}, \ldots, \hat{\mathbf{h}}_{t-\Delta+1} = \mathbf{F}\hat{\mathbf{h}}_{t-\Delta} \tag{21}$$

where again $\hat{\mathbf{h}}_{t-\Delta}$ is the last Kalman channel estimate. Again, for AR(p) models, the prediction formulas of (21) remain essentially unchanged but with $\mathbf{z}$ replacing $\mathbf{h}$.

The received vectors $\mathbf{y}_t, \ldots, \mathbf{y}_{t-\Delta}$, which are also available, can be used to improve the prediction for a fast-varying channel at high SNR. For example, one could formulate and optimize least-squares cost functions $J(\mathbf{h})$, forcing the predicted channel vectors $\hat{\mathbf{h}}_k, k = t, \ldots, t-\Delta+1$ to the vicinity of the values of

(21), weighted by how well a certain $\hat{\mathbf{h}}_k$ "justifies" the received vectors $\mathbf{y}_t, \ldots, \mathbf{y}_{t-\Delta}$ (see [5]). We do not pursue this prediction avenue in this paper, and all simulations are done using (21) for channel prediction.

B. MMSE-DFE Design

The design of the optimum MMSE feedforward and feedback matrix filters $\mathbf{W}^{\text{opt}}$ and $\mathbf{B}^{\text{opt}}$ of lengths N_f and N_b matrix taps, respectively, as well as the optimum selection of the delay Δ for any (n_T, n_R) system, is solved in [10] and will not be repeated here. It is an MMSE design in the sense that it minimizes both the trace and the determinant of the autocorrelation matrix $\mathbf{R}_{ee}$ of the error vector $\mathbf{e}_t = \tilde{\mathbf{x}}_t - \mathbf{x}_t$, where $\tilde{\mathbf{x}}_{t-\Delta}$ is the vector with the n_T equalized soft values at time t, as seen in Fig. 3. For the design, we assume that there is no error propagation, i.e., the hard decision vector $\hat{\mathbf{x}}_{t-\Delta}$ is the same as the transmitted vector $\mathbf{x}_{t-\Delta}$.

In [10], various design methodologies are given, depending on whether there is a feedback filter or not (in which case the design is that of an MMSE linear MIMO equalizer). The choice of oversampling is also available without significant changes to the derivation. For the DFE, an important design choice is whether *current* decisions of stronger users, or only past decisions from every user, are available. The former case would correspond to a successive cancellation scheme and would provide better performance at the cost of added complexity to order the n_T users according to their power.

Here, we avoid the extra complexity by designing a symbol-spaced MIMO DFE, where only past decisions for all users are available and go into the feedback matrix filter $\mathbf{B}^{\text{opt}}$. Clearly, this choice of strictly causal feedback filtering has a consequence. Although it permits almost perfect cancellation of the ISI and cross-ISI, it does not completely suppress residual cross-coupling. In other words, the ith entry $\tilde{x}_t^{(i)}$ of the equalized vector $\tilde{\mathbf{x}}_t$ is almost devoid of the corrupting presence of all $x_{t-k}^{(j)}$, $k \neq 0, j = 1, \ldots, n_T$, but the presence of current other symbols $x_t^{(j)}, j \neq i$ in it is nulled somewhat less effectively.

We do not replicate the equations for optimum design of $\mathbf{W}^{\text{opt}}$ and $\mathbf{B}^{\text{opt}}$ here. Analogous to a SISO system with $n_T = n_R = 1$, we can write the MIMO channel input–output relationship in vector form as a matrix FIR filter by collecting the outputs $y_t^{(j)}$ of (1) from all receiver antennas at time t into a n_R-dimensional column vector $\mathbf{y}_t$

$$\mathbf{y}_t = \sum_{m=0}^{\nu} \mathbf{C}_m(t)\mathbf{x}_{t-m} + \mathbf{v}_t \tag{22}$$

where $\mathbf{y}_t$ and $\mathbf{v}_t$ are column vectors of length n_R, $\mathbf{x}_{t-m}$ are column vectors of length n_T, and $\mathbf{C}_m(t)$ are $n_R \times n_T$ matrix channel taps (instead of the scalar SISO taps). Each of the matrices $\mathbf{C}_m(t), m = 0, \ldots, \nu$ contains the tap $c_m^{(i,j)}(t)$ of (1) and (3) in its (j,i)th position. The essential part of the DFE design at every time instant t is the formulation of the $n_R N_f \times n_T(N_f + \nu)$ block prewindowed channel matrix $\mathbf{H}$

$$\mathbf{H} = \begin{bmatrix} \hat{\mathbf{C}}_0^t & \cdots & \hat{\mathbf{C}}_\nu^t & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \hat{\mathbf{C}}_0^{t-1} & \cdots & \hat{\mathbf{C}}_\nu^{t-1} & \mathbf{0} & \cdots \\ \vdots & & & & & \vdots \\ \mathbf{0} & \cdots & \mathbf{0} & \hat{\mathbf{C}}_0^{t-N_f+1} & \cdots & \hat{\mathbf{C}}_\nu^{t-N_f+1} \end{bmatrix} \tag{23}$$

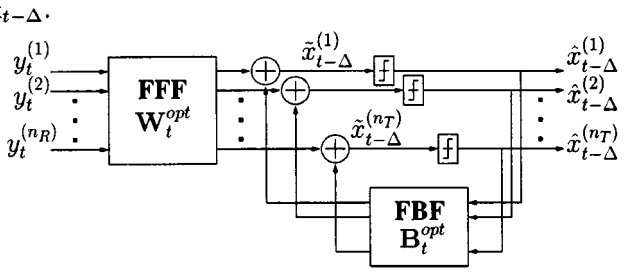

Fig. 3. MIMO DFE block diagram.

where $\hat{\mathbf{C}}_m^k, m = 0, 1, \ldots, \nu, k = t, t-1, \ldots, t-N_f+1$ are the estimates of the $n_R \times n_T$ channel matrices $\mathbf{C}_m(k)$ in (22). Constructing $\mathbf{H}$ of (23) merely involves adding the constant part $\bar{\mathbf{c}}$ to the long vector estimates $\hat{\mathbf{h}}_k, k = t, t-1, \ldots, t-N_f+1$ of the time-varying part of the channel (some of which are estimated via the Kalman filter and some are predicted) and then rearranging the resulting long vectors into the dimensions specified by (22). Once $\mathbf{H}$ of (23) has been formed, the MIMO DFE design procedure described in [10] is straightforward and provides the correlation matrix $\mathbf{R}_{ee}$ of the error vector as a by-product. Fast Cholesky factorization improves the computational efficiency of this finite-length MIMO DFE design, playing a role parallel to that of spectral factorization for matrix filters of infinite length.

IV. Simulation Results

Before the performance evaluation of the Kalman tracking and MIMO DFE equalization algorithm outlined previously, this section briefly presents conventional adaptive algorithms for the DFE, such as LMS and RLS for the case of n_T transmitters and n_R receivers.

A. Baseline Adaptive Systems

Unlike the Kalman-aided DFE presented above, the LMS and RLS adaptation mechanisms do not estimate the channel explicitly. Instead, they adapt the DFE matrix taps based on observed symbols from the time-varying channel and hard decisions from the equalizer. LMS and RLS will serve as baseline systems, the performance of which will be compared with the Kalman-aided DFE through simulations in Section IV-B. For both algorithms, consider the concatenation of the $n_T \times n_R N_f$ matrix FF filter and the $n_T \times n_T N_b$ matrix FB filter of the DFE into an $n_T \times (n_R N_f + n_T N_b)$ equalizer filter $\mathbf{Q} = [\mathbf{W}\ \mathbf{B}]$. At time t, this equalizer filter $\mathbf{Q}_t$ operates on the column "regressor" vector $\mathbf{u}_t$ of length $n_R N_f + n_T N_b$

$$\mathbf{u}_t = \left[\mathbf{y}_t^T\ \cdots\ \mathbf{y}_{t-N_f+1}^T\ \hat{\mathbf{x}}_{t-\Delta-1}^T\ \cdots\ \hat{\mathbf{x}}_{t-\Delta-N_b}^T\right]^T. \tag{24}$$

The operation produces a vector $\tilde{\mathbf{x}}_{t-\Delta}$ of n_T soft values, which are then fed to n_T slicers, as in Fig. 3, producing hard decisions $\hat{\mathbf{x}}_{t-\Delta}^T$. From [19], the following LMS recursions, which are separate for each transmitter $i = 1, \ldots, n_T$, iteratively approximate the least-mean-square solution

$$\mathbf{q}_i(t) = \mathbf{q}_i(t-1) + \mu\left[\hat{x}_{t-\Delta}^{(i)} - \mathbf{q}_i(t-1)\cdot\mathbf{u}_t\right]\mathbf{u}_t^*. \tag{25}$$

Those n_T simultaneous LMS recursions for the rows $\mathbf{q}_i$ of $\mathbf{Q}$ can be combined into one for the entire DFE $\mathbf{Q}$

$$\text{MIMO LMS: } \mathbf{Q}_t = \mathbf{Q}_{t-1} + \mu[\hat{\mathbf{x}}_{t-\Delta} - \mathbf{Q}_{t-1}\cdot\mathbf{u}_t]\mathbf{u}_t^*. \quad (26)$$

Similarly, considering n_T parallel RLS adaptations for each row and this time using instantaneous approximations for the necessary correlation matrices in a Newton recursion (see [19]), we get

$$\mathbf{q}_i(t) = \mathbf{q}_i(t-1) + \left[\hat{\mathbf{x}}_{t-\Delta}^{(i)} - \mathbf{q}_i(t-1)\cdot\mathbf{u}_t\right]\mathbf{u}_t^* P_t$$
$$\text{for } i = 1,\ldots,n_T, \quad \text{with}$$
$$P_t = \lambda^{-1}\cdot\left[P_{t-1} - \frac{\lambda^{-1}P_{t-1}\mathbf{u}_t\mathbf{u}_t^*P_{t-1}}{1+\lambda^{-1}\mathbf{u}_t^*P_{t-1}\mathbf{u}_t}\right]$$

which, again, can be combined into one matrix recursion for $\mathbf{Q}_t$

$$\text{MIMO RLS}: \quad P_t = \lambda^{-1}\cdot\left[P_{t-1} - \frac{\lambda^{-1}P_{t-1}\mathbf{u}_t\mathbf{u}_t^*P_{t-1}}{1+\lambda^{-1}\mathbf{u}_t^*P_{t-1}\mathbf{u}_t}\right]$$
$$\mathbf{Q}_t = \mathbf{Q}_{t-1} + [\hat{\mathbf{x}}_{t-\Delta} - \mathbf{Q}_{t-1}\cdot\mathbf{u}_t]\,\mathbf{u}_t^* P_t \quad (27)$$

with initial condition $P_{-1} = \pi_o\mathbf{I}$ and $0 \ll \lambda \leq 1$.

Note that the recursive adaptation of (27) above is also the exact solution of a weighted, regularized least-squares problem involving a block of observations and a single equalizer matrix $\mathbf{Q}$. In fact, with the obvious changes to accommodate the different dimensions of $\mathbf{Q}$, the RLS recursion of (27) can easily be shown to be exactly equivalent to the more compact RLS recursions given in [19, ch. 10] and [22, p. 569] with the update for P_t after the equalizer update.

B. Performance Evaluation

In all simulations presented in this section, we implement the Kalman-aided MIMO DFE algorithm outlined in Section III and compare the performance of the system, as measured by its symbol error rate (SER) when transmitting 4-PSK constellation points through (2, 2) and single-antenna (i.e., (1, 1)) Ricean fading channels. We assume the channel mean to be constant and known at the beginning of each block of N symbols per user. For comparison purposes, the plots include the SER performance (dashed curves) with the receiver having access to "genie-provided" perfect channel information, as well as that of the LMS and RLS of Section IV-A. Note that LMS and RLS are less computationally intensive than the tracking algorithm proposed in this paper.

The SNR is set to be the same for both users. In addition, all $n_T n_R$ mean interfering channels of (2) are normalized $||\bar{\mathbf{c}}^{(i,j)}||^2 = 1$, and all ratios K_k are chosen equal to a single K. Thus, with σ_v^2 being the noise variance at each receiver, the SNR plotted is the SNR of each interfering channel, neglecting co-channel interference

$$\text{SNR} = 10\log\frac{1+10^{K/10}}{\sigma_v^2} \quad (28)$$

since the input 4-PSK points are normalized to unit power. Given that all $n_T n_R$ (direct and interfering) ISI channels are normalized to have equal energy $1+10^{K/10}$, clearly, the optimum allocation of power among the n_T transmitters—without special shaping—is equal power to all. Of course, different channel energies and transmitter power allocation can be explored, but in this paper, we performed all simulations based on the above assumptions. Clearly, for the (2, 2) example, the above definition of SNR corresponds to 3 dB higher SNR (i.e., $10\log n_T$) than that commonly defined in the space-time literature (see [2]) because we allocate unit power to each transmitter antenna, thinking about individual uncoordinated users. For coordinated transmission, to maintain a fair comparison with conventional (1, 1) systems, input power must be split evenly among the n_T transmitter antennas, which lowers the SNR by $10\log n_T$ dB.

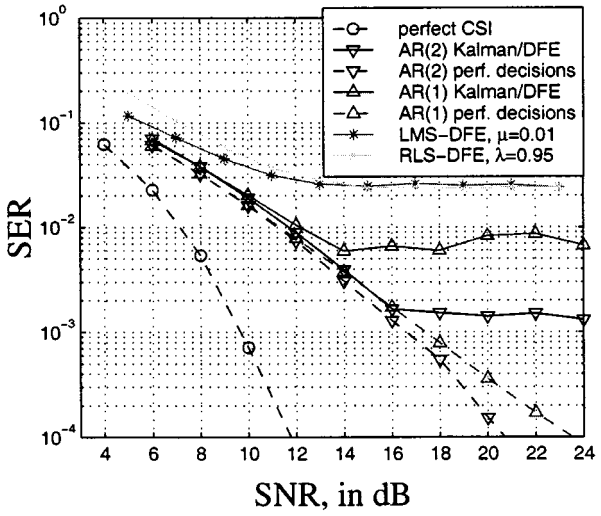

Fig. 4. Performance of (2, 2) system versus SNR. Blocklength $N = 500$, Bello's Ricean channel with $f_D T = 0.02, K = 10$ dB. Both AR(2)-based and AR(1)-based approximations at the receiver outperform plain LMS/RLS adaptations. Notice also the performance with *correct* decisions fed back, both in the DFE and into the Kalman estimator (dashed curves).

In Fig. 4, the mean ISI channels were the normalized versions of $(1+j)[1\ \ 0.8]$ and $(1+j)[1\ \ 0.3]$ for the direct paths, and $(1+j)[1\ \ -0.8]$ and $(1+j)[1\ \ -0.5]$ for the interfering paths; the Doppler rate of the variations was $f_D T = 0.02$, and the specular-to-diffuse power ratio was $K = 10$ dB. The indicated symbol-error rate (SER) performance represents unsupervised channel tracking for blocks of $N = 500$ symbols per user. The DFE has $N_f = 3$ and $N_b = 1$ matrix taps and decision delay $\Delta = 2$. We observe that performance with receivers employing Kalman tracking is generally significantly better than plain LMS/RLS adaptations when using either AR(1) or AR(2) approximations of the channel, which is varying according to Bello's model (7).

The superiority of our algorithm versus model-independent adaptive solutions such as LMS and RLS should come as no surprise, first of all because of the difference in complexity.The LMS adaptation costs $\mathcal{O}(n_T(n_R N_f + n_T N_b))$ complex operations, and RLS requires $\mathcal{O}((n_T(n_R N_f + n_T N_b))^2)$. The Kalman estimation requires $\mathcal{O}((pL)^3)$, where p is the order of the AR(p) approximation used by the receiver, and the total number of channel taps is $L = n_T n_R(\nu+1)$. Another $\mathcal{O}((N_f+N_b)^2)$ operations are required for the DFE design after Kalman estimation. Additionally, the memory required for the Kalman/DFE adaptation is larger because N_f estimated and predicted channels need to be stored to form the matrix

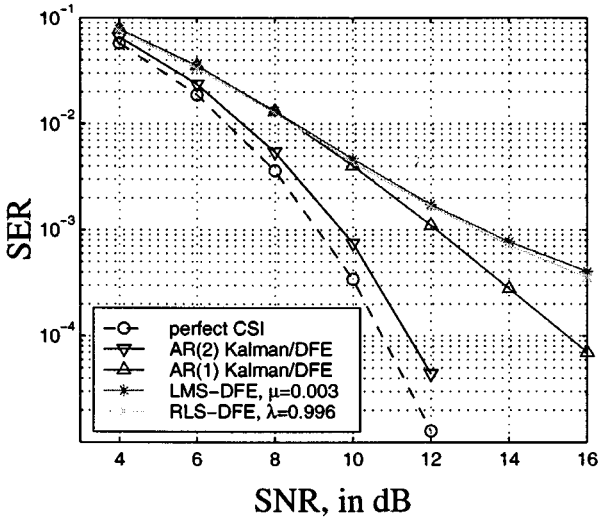

Fig. 5. Performance of (2, 2) system versus SNR. Same as in Fig. 4, but the channel is AR(2), which is derived from $f_DT = 0.02, K = 10$ dB, i.e., no modeling mismatch for AR(2) Kalman/DFE. Superiority versus LMS/RLS is retained, and the AR(2) receiver follows the perfect CSI (dashed) curve very closely.

$\mathbf{H}$ of (23) and the matrices of the Kalman recursion, whereas LMS/RLS require only the current regressor vector (24) to be stored. The fundamental reason for the improved performance with respect to LMS/RLS is that these algorithms do not not have an explicit mechanism of incorporating known channel statistics and only rely on careful selection of the parameters μ and λ, respectively, to perform the best tracking they can. Therefore, exploiting the knowledge of largely invariant channel parameters (i.e., the mean and Doppler) to form a low-order autoregressive AR(1) or AR(2) channel model offers a significant performance improvement.

The reason behind the significant distance between the SER curve of AR(2) and AR(1) Kalman tracking and the SER curve when the receiver has perfect channel-state information (CSI—dashed curve) is twofold. First, there is the modeling mismatch between the true Bessel time correlation of the channel taps and the AR models used by the Kalman/DFE receiver. This is further evidenced by Fig. 5, where we actually generated the channel using the AR(2) model corresponding to the $f_DT = 0.02$ and $K = 10$ dB, which was discussed previously. In other words, in Fig. 5, the AR(2) channel model at the receiver has no mismatch to the actual simulated channel. We observe that when the channel follows the AR(2) model used by the receiver, the distance from perfect CSI is less than 1 dB. However, when modeling mismatch exists, i.e., when the receiver uses an AR(1) model to track while the simulated channel is AR(2), then the distance from perfect CSI increases.

Comparing Figs. 4 and 5, one notices generally better performance on the AR(2) channel with parameters derived from Bello's channel (in Fig. 5) than on the more realistic Bello channel itself (Fig. 4). In addition, the gap between the Kalman receivers and RLS is more pronounced in Fig. 4. Both of these observations show that Bello's channel is a more difficult channel in which to perform well because the nonrational autocorrelation makes tracking a harder problem. Even the perfect CSI curve (no estimation complications) reaches SER $= 10^{-4}$ more than 1 dB earlier in the AR(2) channel, confirming that the AR channel is simply more benign and presents an easier tracking task to the receiver, thus allowing even the much simpler LMS/RLS algorithms to perform much closer to our proposed Kalman/DFE solution than they do in Bello's channel.

The second main source of performance degradation relative to perfect CSI is error propagation of the MIMO DFE. The exact effect of this can be shown by the difference between the solid performance curves and their dashed counterparts, which present simulations with ideal "error-free" feedback to both the Kalman filter and the feedback section of the MIMO DFE (see Figs. 4, 6, and 7 for a (1, 1) channel). Error propagation is also solely responsible for the apparent "floor" in error probability noticeable both in Figs. 4 and 7 and less so in the slower channel of Fig. 6. Notice that when correct decisions are fed back into the receiver, no error floor appears [dashed curves marked with "$\triangle$" for AR(1) and "$\bigtriangledown$" for AR(2) receivers]. In our simulations, the "error-floor" effect appears when the SNR is high enough that a single incorrect decision changes the tracking and DFE design enough in the wrong direction to compromise the rest of the block. The remedy for this "runaway" effect (recognized also in [12]) is to periodically insert known symbols—pilots—in the data stream, sacrificing part of the transmission rate to make the situation more similar to the ideal "error-free feedback" curves shown here.

Although error propagation is responsible for the error floor, higher order channel modeling helps lower it significantly. From the solid curves (true decisions fed back) in Figs. 4, 6, and 7, we conclude that increased order channel parameterization offers higher quality tracking, which makes the probability of a block of data being lost due to error propagation smaller. Therefore, increasing p in the AR(p) receiver model renders a more robust, albeit more complex receiver. In fact, this is a major advantage of the Kalman/DFE algorithm with respect to LMS and RLS, which make a small number of erroneous decisions (about 1%) in almost every block even with correct decisions fed back (not shown in the plots). In contrast, the Kalman/DFE algorithm at high SNR decodes almost all blocks correctly, with the exception of a few blocks, which are lost to catastrophic error propagation, causing the SER error floor.

Fig. 6 shows the performance again in a Ricean channel generated by Bello's model, with perfect CSI at the receiver (dashed), with the receiver using the "best-fit" AR(2) and AR(1) models analyzed in Section II-B. Everything is the same as in Fig. 4, only with a smaller Doppler, which is now $f_DT = 0.007$. As a result, the time-correlation of the taps is now approximately three times stronger than before, thus making the tracking task easier. This is demonstrated by the fact that the AR(2) receiver is not significantly better than the receiver employing AR(1) channel modeling, especially in the (dashed) error-free decision feedback simulation, which removes the error propagation effect. In addition, the gap between both the AR(2) and AR(1) receivers from the perfect CSI is less. In this case, with the lower Doppler (stronger correlation), even the first-order receiver model performs hardly worse than the more complicated receiver using the AR(2) model to match the true channel statistics. In addition, observe that error

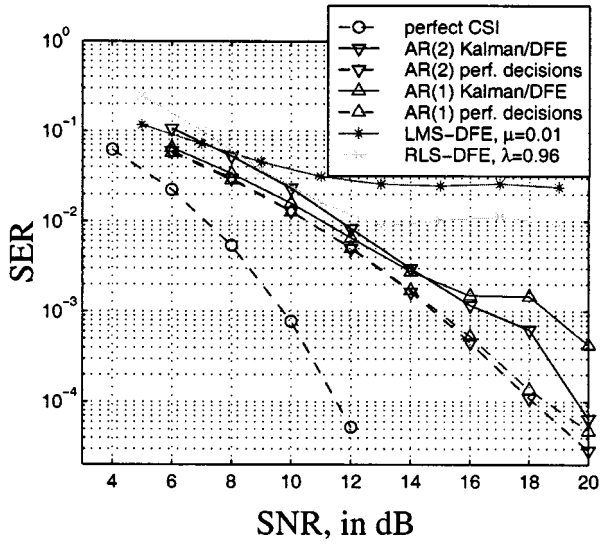

Fig. 6. Performance of (2, 2) system versus SNR. Same as in Fig. 4, except now, the Doppler is lower $f_D T = 0.007$, yielding an easier channel to track.

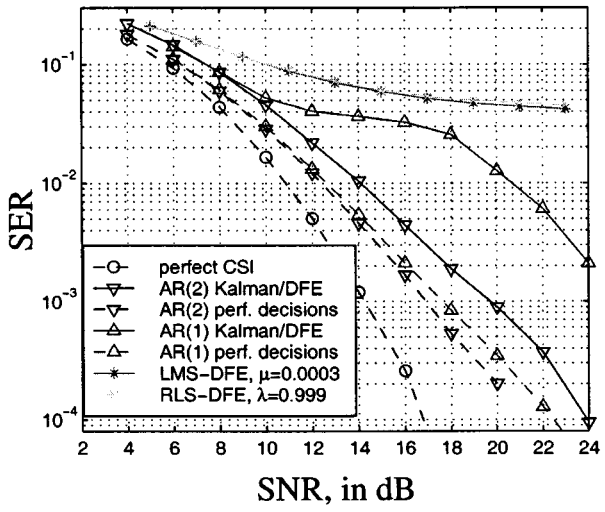

Fig. 7. Performance of (1, 1) system versus SNR in Ricean channel with $f_D T = 0.02$ and $K = 10$ dB. The DFE has $N_f = 3, N_b = 1$ and decision delay $\Delta = 2$.

propagation tends to generate a small anomaly in SER at 18 dB, but eventually, with increasing SNR, the curves continue to approach their "error-free" (dashed) feedback counterparts without exhibiting an error floor. This indicates that higher quality channel tracking in general (either from increased order AR channel modeling or because the channel is slower and easier to track) drastically diminishes the probability of lost blocks and, thus, prevents an error floor.

Fig. 7 shows simulation results for a (1, 1) Ricean channel in the same fashion as above. The mean channel is $(1+j)[2.5\ 1]$ normalized to unit energy, the blocklength is $N = 500$, the Doppler is $f_D T = 0.02$, and $K = 10$ dB. The superiority versus LMS and RLS is maintained, as it was previously. An interesting observation in this case is that although the dashed curves (error-free decisions fed back) with AR(2) and AR(1) approximations for Kalman tracking are not very different, when the real decisions are fed back (solid curves), the better quality of the AR(2) tracking shows, leading to much better performance than the AR(1) tracking (the difference is about 6 dB at SER of

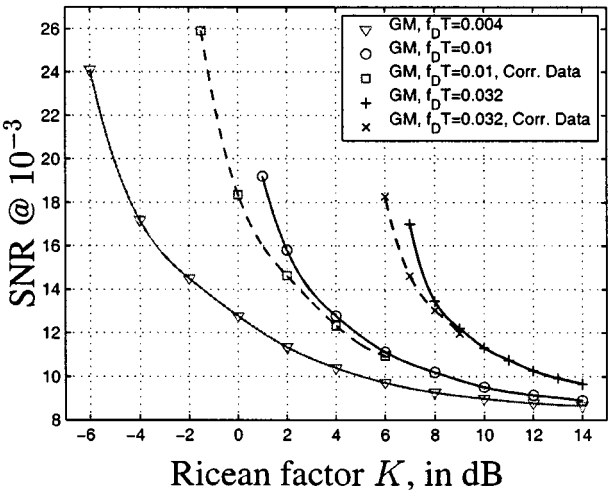

Fig. 8. SNR required for the (2, 2) system in an AR(1) channel to reach SER $= 10^{-3}$ for different channel conditions, i.e., Ricean factor K and Doppler rate [or $a_k(1)$]. The channel is time-varying according to the Gauss–Markov (GM) model of (15).

10^{-2}). Therefore, the better matching of the true channel statistics provided by the AR(2) model manifests itself mainly in the true decision-aided tracking, where the AR(1) receiver is significantly more prone to suffering from error propagation.

The simulation results in Fig. 8 provide some insight on the issue of how the magnitude and speed of the channel variation affect the system performance. At each simulation point, the channel is a pure Gauss–Markov channel, i.e., first-order autoregressive tap variation. We control the rate of variation by the Doppler, namely, by the selection of the AR(1) parameter $a_k(1) = \mathcal{J}_o(2\pi f_D T)$ in the model of (15), and the magnitude of variation with respect to the channel mean by the Ricean factor K of (14). Fig. 8 shows the SNR required for the system to reach an SER of 10^{-3} for given pairs of the abscissa K and the Doppler. We observe the intuitively expected fact that as K decreases and the Doppler rate $f_D T$ increases (less correlated channel), the problem becomes harder, and higher SNR is required for 10^{-3} error performance. In addition, for a given Doppler, sufficiently reducing K causes an error floor, and SER $= 10^{-3}$ is never achieved, no matter how high the SNR. Because the AR(1) coefficient $a_k(1)$ is the same for all channel taps, this simulation pertains to the scenario of colocated transmission through an (n_T, n_R) system, rather than that of n_T separate users, each with different Doppler. However, even in the latter case, the conclusion that increasing Doppler [$a_k(1)$ significantly less than 1 for AR(1) model] worsens tracking performance still holds.

Furthermore, we observe for this AR(1) channel that similar conclusions are true when the Kalman filter is simulated with *correct* decisions fed back (dashed curves), rather than actual decisions from the MIMO DFE (solid curves). This suggests that the problematic performance in channels with small mean, i.e., almost Rayleigh channel taps, relatively small K, coupled with rapid variation [high $f_D T$, small coefficients $a_k(1)$] is only partially due to failure of the DFE because of error propagation; moreover, pilot insertion can mitigate error propagation and increase robustness. The fact that error-free decision feedback also fails at high enough Doppler and low enough K in-

dicates that accurate channel estimation and tracking is a fundamental issue at high Doppler. In fact, idealized simulations in which the DFE is provided with perfect channel knowledge show good equalization performance, regardless of the speed (large f_DT) and magnitude (low K) of the channel variation. The MIMO DFE is an efficient equalization mechanism, as long as correct channel estimates are available, but at high enough Doppler, correct channel estimates may not be possible.

Even with error-free decisions fed back, the Kalman tracking fails under the harsh conditions of low-K/high-Doppler. This Kalman tracking failure for rapid channel variation does not mean that channel estimates diverge. The Kalman filter still follows the true tap trajectories, only with a higher mean-squared error. Thus, the channel estimates it produces are not accurate enough for the MIMO DFE to effectively equalize. It should be noted, however, that it takes quite violent tap fluctuation (low-K/high-f_DT) to cause the Kalman filter to ineffectively track the taps, whereas the baseline LMS/RLS adaptive algorithms exhibit high error floor for much milder channel conditions and never outperformed the Kalman filter tracking in any of our simulations. An additional observation for perspective is that in the limit of very low K and very high f_DT, the channel tap values become effectively i.i.d., which makes any attempt for channel estimation futile. Increasing the order p of the AR model used by the Kalman filter has a positive effect on estimation fidelity at a significant cost in complexity.

Finally, it should be noted that the separation of the performance curves (solid with actual decisions, dashed with correct decisions) for the Kalman/DFE solution from the simulation results with perfect CSI (dashed curves) in Figs. 4, 6, and 7 is not solely due to model mismatch, i.e., not large enough order p of the receiver AR(p) model. In other words, the entire distance to perfect CSI will not be covered if we arbitrarily increase p. The estimation quality is also partly limited because of the filtering essence of the decision-aided Kalman mechanism, which does not allow for a smoothing estimation solution. The Kalman filter is an optimal linear *filtering* estimator and can never be as good as a smoothing estimator taking into account the future, even with matching channel and receiver AR models. However, data in the future are unknown, and therefore, the alternatives are either blind or block-oriented iterative solutions, which are beyond the scope of this paper. However, much of the channel estimation degradation is due to the model mismatch, as suggested by Fig. 5, where exact channel modeling brings the filtering estimation and perfect CSI performance to within 1 dB of each other.

C. Tap Independence Revisited

Here, we explore further the issue of tap independence, which is a simplifying assumption made in Section II. In reality, because the taps of the Ricean channel are not baud-spaced, the combination of the shaping filters at the transmitter and receiver (usually root-raised-cosine with a rolloff factor of $0.2-1$) causes correlation between the effective channel taps, as observed by the receiver. However, this does not change the essence of the proposed Kalman tracking algorithm because the induced correlation is known, and its effect can be taken into account. The formulation and the simulation results do not change significantly. The objective of the tracking is still the independent Ricean taps, although the observations $\mathbf{y}_t$ are produced by the correlated baud-spaced channel after the receiver shaping filter.

Specifically, consider a Ricean channel $\mathbf{h}_t$ with $\nu+1$ independent impulses $h_0,\ldots,h_\nu$ that are arbitrarily spaced in time at instants $\tau_0,\ldots,\tau_\nu$. If the total shaping filter has impulse response $g(\tau)$ (say, a raised cosine), then the effective channel seen by the receiver has impulse response

$$h'(\tau)=\sum_{k=0}^{\nu} h_k g(\tau-\tau_k) \tag{29}$$

and the T-spaced samples of $h'(\tau)$ are now correlated in time. This correlation can be represented by a shaping matrix $\mathbf{S}$ so that the effective T-spaced channel seen by the receiver is $\mathbf{h}'_t$, consisting of samples of $h'(\tau)$ of (29) every baud period T. Hence, $\mathbf{h}'_t=\mathbf{S}\mathbf{h}_t$. The form of the shaping matrix $\mathbf{S}$ is best explained through a specific example, which we also used for the (1, 1) simulation of Fig. 9. The extension to (n_T,n_R) systems is straightforward. Consider two Ricean taps h_1,h_2, at times $\tau_1=2.8T$ and $\tau_2=5.3T$. Then, using a raised cosine pulse shape $g(\tau)$ with 50% rolloff and retaining six bauds of it (because it decays quickly in time), the shaping matrix becomes (the effective channel $\mathbf{h}'_t$ now has nine taps)

$$\mathbf{S}=\begin{bmatrix} g(-2.8T) & 0 \\ g(-1.8T) & 0 \\ g(-0.8T) & 0 \\ g(0.2T) & g(-2.3T) \\ g(1.2T) & g(-1.3T) \\ g(2.2T) & g(-0.3T) \\ 0 & g(0.7T) \\ 0 & g(1.7T) \\ 0 & g(2.7T) \end{bmatrix} \tag{30}$$

and the only thing that changes in the Kalman recursion of (17) is that the matrix $\mathbf{X}$ is replaced by $\mathbf{XS}$ since the observation (16) now becomes

$$\mathbf{y}_t=\mathbf{X}_t\cdot\mathbf{S}\cdot(\bar{\mathbf{c}}+\mathbf{h}_t)+\mathbf{v}_t. \tag{31}$$

Fig. 9 presents the simulation results for this scenario. Observe that the results remain consistent with previous simulations, but the complexity is slightly higher due to the additional matrix operation and the increased effective channel length. The mean channel used is $(1+j)[2\ \ 1]$ normalized to unit energy, the Doppler for the taps varying according to Bello's model is $f_DT=0.02$, the Ricean factor is $K=10$ dB, the blocklength is $N=500$, and the DFE has $N_f=6, N_b=4, \Delta=5$. Note that knowledge of the times $\tau_0,\ldots,\tau_\nu$ of the channel impulses is assumed. In a real-world system, those times are unknown but can be estimated via the algorithm in [11].

V. Conclusion

This paper proposed a receiver structure to track and equalize a MIMO frequency-selective fading channel. A Kalman filter was used for tracking the channel, employing a low-order autoregressive model to best fit the true statistics of the channel variation. An MMSE DFE optimized for decision delay $\Delta\geq 0$

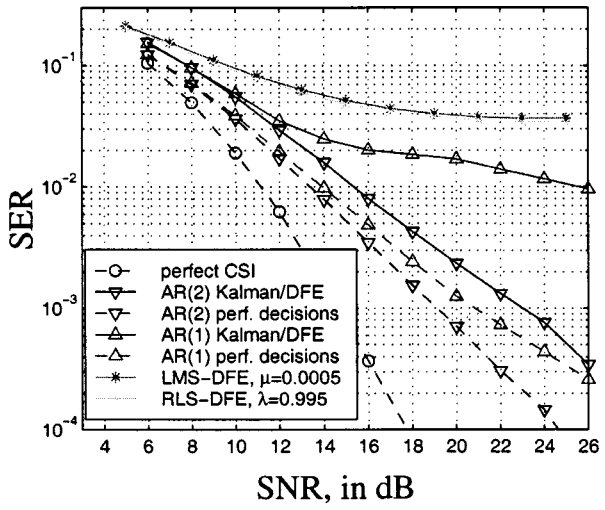

Fig. 9. Performance of (1, 1) system versus SNR in the Ricean channel with $f_D T = 0.02$ and $K = 10$ dB and correlated taps due to shaping. The DFE has $N_f = 6, N_b = 4$, and decision delay $\Delta = 5$.

was used to equalize the channel and suppress CCI. The time gap between channel estimates produced by the Kalman filter and those needed for the MIMO DFE adaptation was bridged by using a simple prediction module. This algorithm, in exchange for larger complexity when compared with simple LMS/RLS updates of the DFE, offers improved performance and good tracking behavior for long unsupervised blocks. This is achieved mainly because of the autoregressive modeling of the channel statistics at the receiver.

REFERENCES

[1] A. Duel-Hallen, "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.

[2] G. J. Foschini and M. J. Gans, "On limits of wireless communications in a fading environment when using multiple antennas," *Wireless Pers. Commun.*, vol. 6, pp. 311–335, 1998.

[3] V. Tarokh, N. Seshadri, and A. R. Calderbank, "Space-time codes for high data rate wireless communication: Performance criterion and code construction," *IEEE Trans. Inform. Theory*, vol. 44, pp. 744–765, Mar. 1998.

[4] D. P. Taylor, G. M. Vitetta, B. D. Hart, and A. Mämmelä, "Wireless channel equalization," *Eur. Trans. Telecom.*, vol. 9, no. 2, pp. 117–143, Mar.–Apr. 1998.

[5] C. Komninakis, C. Fragouli, A. H. Sayed, and R. D. Wesel, "Channel estimation and equalization in fading," in *Proc. 33rd Asilomar Conf. Signals, Syst., Comput.*, Oct. 1999, pp. 1159–1163.

[6] G. Paparisto and K. M. Chugg, "PSP array processing for multipath fading channels," *IEEE Trans. Commun.*, vol. 47, pp. 504–507, Apr. 1999.

[7] S. N. Diggavi, B. C. Ng, and A. Paulraj, "An interference suppression scheme with joint channel-data estimation," *IEEE J. Select. Areas Commun.*, vol. 17, pp. 1924–1939, Nov. 1999.

[8] M. Honig, U. Madhow, and S. Verdú, "Blind adaptive multiuser detection," *IEEE Trans. Inform. Theory*, vol. 41, pp. 944–960, July 1995.

[9] Y. Li and K. J. R. Liu, "Adaptive blind source separation and equalization for multiple-input/multiple-output systems," *IEEE Trans. Inform. Theory*, vol. 44, pp. 2864–2876, Nov. 1998.

[10] N. Al-Dhahir and A. H. Sayed, "The finite-length multi-input multi-output MMSE-DFE," *IEEE Trans. Signal Processing*, vol. 48, pp. 2921–2936, Oct. 2000.

[11] R. A. Iltis, "Joint estimation of PN code delay and multipath using extended Kalman filter," *IEEE Trans. Commun.*, vol. 38, pp. 1677–1685, Oct. 1990.

[12] M. K. Tsatsanis, G. B. Giannakis, and G. Zhou, "Estimation and equalization of fading channels with random coefficients," *Signal Process.*, vol. 53, no. 2–3, pp. 211–229, Sept. 1996.

[13] S. Haykin, A. H. Sayed, J. R. Zeidler, P. Yee, and P. C. Wei, "Adaptive tracking of linear time-variant systems by extended RLS algorithms," *IEEE Trans. Signal Processing*, vol. 45, pp. 1118–1128, May 1997.

[14] P. A. Bello, "Characterization of randomly time-variant linear channels," *IEEE Trans. Commun. Syst.*, vol. CS-11, pp. 360–393, Dec. 1963.

[15] W. C. Jakes Jr., *Microwave Mobile Communications*. New York: Wiley, 1974.

[16] K. E. Baddour and N. C. Beaulieu, "Autoregressive models for fading channel simulation," in *Proc. IEEE Global Telecommun. Conf.*, vol. 2, 2001, pp. 1187–1192.

[17] S. M. Kay, *Modern Spectral Estimation: Theory and Application*. Englewood Cliffs, NJ: Prentice-Hall, 1987.

[18] A. Papoulis, *Probability, Random Variables and Stochastic Processes*, 3rd ed. New York: McGraw-Hill, 1991.

[19] T. Kailath, A. H. Sayed, and B. Hassibi, *Linear Estimation*. Englewood Cliffs, NJ: Prentice-Hall, 2000.

[20] A. H. Sayed and T. Kailath, "Extended Chandrasekhar recursions," *IEEE Trans. Automat. Contr.*, vol. 39, pp. 619–623, Mar. 1994.

[21] T.-A. Chen, M. P. Fitz, W.-Y. Kuo, M. P. Fitz, M. D. Zoltowski, and J. H. Grimm, "A space-time model for frequency nonselective Rayleigh fading channels with applications to space-time modems," *IEEE J. Select. Areas Commun.*, vol. 18, pp. 1175–1190, July 2000.

[22] S. Haykin, *Adaptive Filter Theory*, 3rd ed. Englewood Cliffs, NJ: Prentice-Hall, 1996.

Christos Komninakis (S'97) was born in Athens, Greece, in 1972. He received the Diploma in electrical and computer engineering from the National Technical University of Athens (NTUA), Athens, Greece, in 1996 and the M.S. and Ph.D. degrees in electrical engineering from the University of California, Los Angeles, in 1998 and 2000, respectively.

He is currently with Broadcom Corporation, El Segundo, CA. His research interests are in coding and channel estimation for linear time-varying channels and diversity techniques for increasing the data rate of wireless systems.

Christina Fragouli received the B.S. degree from the National Technical University of Athens (NTUA), Athens, Greece, in 1996 and the M.S. and Ph.D. degrees from the University of California, Los Angeles (UCLA), in 1998 and 2000 respectively.

She was with AT&T Research Labs as a Consultant from September 2000 to September 2001. She is currently a Research Associate at the National Capodistrean University of Athens (NCUA). Her research interests are in the area of communications and channel coding.

Ali H. Sayed (F'00) received the Ph.D. degree in electrical engineering in 1992 from Stanford University, Stanford, CA.

He is Professor of electrical engineering at the University of California, Los Angeles (UCLA). He has over 160 journal and conference publications, is coauthor of the research monograh *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 coeditor 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 of the *International Journal of Adaptive Control and Signal Processing* and has served as coeditor of special issues of the journal *Linear Algebra and Its Applications*. He has contributed several articles to engineering and mathematical encyclopedias and handbooks and has served on the program committees of several international meetings. He has also consulted with industry in the areas of adaptive filtering, adaptive equalization, and echo cancellation. 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. To learn more about his work, visit the website of the UCLA Adaptive Systems Laboratory at http://www.ee.ucla.edu/asl.

Dr. Sayed is a recipient of the 1996 IEEE Donald G. Fink Award. He is Associate Editor of the IEEE TRANSACTIONS ON SIGNAL PROCESSING and is a member the technical committees on Signal Processing Theory and Methods (SPTM) and Signal Processing for Communications (SPCOM), both of the IEEE Signal Processing Society.

Richard D. Wesel (S'91–M'96–SM'01) received both S.B. and S.M. degrees in electrical engineering from the Massachusetts Institute of Technology, Cambridge, in 1989 and the Ph.D. degree in electrical engineering from Stanford University, Stanford, CA, in 1996.

From 1989 to 1991, he was a Member of Technical Staff with AT&T Bell Laboratories. His work at AT&T resulted in two patents. Since 1996, he has been with the University of California, Los Angeles (UCLA) as an Assistant Professor in the Electrical Engineering Department. His research interests are in the area of communication theory with particular interest in the topics of channel coding and distributed communication.

Dr. Wesel has been an Associate Editor for the IEEE TRANSACTIONS ON COMMUNICATIONS in the area of coding and coded modulation since 1999. He received a National Science Foundation CAREER Award in 1998. In 1999, he received an Okawa Foundation Award. He also received the 2000 TRW Excellence in Teaching Award from the UCLA Henry Samueli School of Engineering and Applied Science.