A BLOCK ADAPTIVE FREQUENCY DOMAIN MIMO DFE FOR WIDEBAND CHANNELS

Vassilis Kekatos[1], Kostas Berberidis[1], Athanasios A. Rontogiannis[2],
[1]Dept. of Computer Engineering & Informatics / C.T.I.-R&D, University of Patras, Greece
[2]Institute for Space Applications and Remote Sensing, N.O.A., Athens, Greece
Emails: {kekatos,berberid}@ceid.upatras.gr, tronto@space.noa.gr

Abstract— **The equalization of MIMO frequency selective channels is a challenging task due to the increased filter lengths required and the presence of interstream interference. In wideband systems where high delay spread channels arise and long transmission bursts are employed, adaptive equalization is necessary. In this work, we develop an adaptive MIMO DFE for wideband time varying systems. To reduce computational complexity and accelerate convergence rate, the equalizer is fully implemented in the frequency domain, while an inherent causality problem is efficiently solved through an iterative scheme. As verified by simulations, the proposed method offers a good trade off in terms of complexity and convergence, compared to other related adaptive algorithms.**

Index Terms— **MIMO systems, Adaptive equalizers.**

I. INTRODUCTION

In high data rate spatial multiplexing multi-input multi-output (MIMO) systems, the introduced intersymbol and inter-stream interference cause a severe performance degradation. In order to deal with this situation, sophisticated receivers should be designed [1]. When the channel does not change significantly within a burst of data, then single carrier cyclic prefixed (SC-CP) techniques may be employed [2], [3]. In this case, the insertion of the CP makes the channel matrix circulant thus reducing the receiver's complexity. The price paid is that the overall throughput is affected due to the CP overhead.

However, when the channel impulse response changes within a burst (a case arising in relatively long bursts and/or fast varying conditions), the above techniques fail to equalize the channel and efficient adaptive methods are required. Two linear adaptive equalizers have recently been presented in [4], [5]. To our knowledge, the only adaptive MIMO decision feedback equalizer (DFE) designs are those proposed in [6] and [7]. Both these equalizers are updated using the recursive least squares (RLS) algorithm, although their underlying structures are different.

Two are the main problems in adaptive MIMO equalization compared to classical SISO equalization. First, the size of the equalizer is naturally increased, and second, due to interstream interference, the noise turns out to be colored. These characteristics affect severely the convergence rate of a least mean square (LMS) based equalizer. On the other hand, despite the fast convergence of RLS equalizers, their high computational complexity may be prohibitive. Motivated by the work in [8], we develop an LMS MIMO DFE fully implemented in the frequency domain. The gain of such an approach is

twofold. First, computational complexity is drastically reduced by employing efficient fast Fourier transform (FFT) operations. Furthermore, convergence speed is improved by appropriately weighting the different modes of the adaptive MIMO process. The proposed algorithm operates on a block-by-block basis, and the causality problem, which is inherent in a block adaptive DFE, is resolved using an efficient iterative scheme.

II. SYSTEM MODEL

Let us consider a MIMO communication system operating over a frequency selective wireless channel. The channel is considered quasi-static, i.e., it remains fixed for a short duration of Q symbol periods. The system employs M transmit and N receive antennas, with $M \leq N$, while spatial multiplexing is assumed. If the transmitted signal at time k is

$$\mathbf{s}(k) = \begin{bmatrix} s_1(k) & s_2(k) & \ldots & s_M(k) \end{bmatrix}^T \quad (1)$$

where $s_m(k)$, $m = 1, \ldots, M$, are i.i.d. symbols, then the symbol spaced signal at receive antenna n is

$$x_n(k) = \frac{1}{\sqrt{M}} \sum_{m=1}^{M} \sum_{l=0}^{L} h_{nm}(l) s_m(k-l) + \eta_n(k) \quad (2)$$

where $h_{nm}(l)$, for $l = 0, \ldots, L$, is the sampled impulse response between transmitter m and receiver n, and $\eta_n(k)$, $n = 1, \ldots, N$, are white Gaussian complex noise samples of variance $\sigma^2/2 = N_0/2$ per dimension.

III. EQUALIZER DESIGN

To mitigate the intersymbol and interstream interference involved in the system described by (2), we propose a MIMO DFE completely implemented in the frequency domain. The algorithm originates from a proper block formulation of the MIMO DFE in the time domain as briefly discussed next.

A. Block MIMO DFE in the Time Domain

The proposed MIMO DFE is a set of M MISO DFEs operating in parallel to extract the M transmitted streams [1]. Each MISO DFE consists of N feedforward filters with a temporal span of K_f symbol periods, each one adjusted to a receive antenna, and M feedback filters that subtract the contribution of K_b postcursor symbols from all streams. Thus, the feedforward part of the MIMO DFE can be described by the $K_f N \times M$ matrix filter

$$\mathbf{W}(k) = \begin{bmatrix} \mathbf{w}_1(k) & \cdots & \mathbf{w}_M(k) \end{bmatrix} \quad (3)$$

where its m-th column processes the samples from all receive antennas to extract the m-th stream, and can be written as

$$\mathbf{w}_m(k) = \begin{bmatrix} \mathbf{w}_{m,1}^T(k) \cdots \mathbf{w}_{m,N}^T(k) \end{bmatrix}^T, \ m = 1, \ldots, M. \quad (4)$$

Likewise, the feedback part of the equalizer is described by the $K_b M \times M$ matrix

$$\mathbf{B}(k) = \begin{bmatrix} \mathbf{b}_1(k) & \cdots & \mathbf{b}_M(k) \end{bmatrix}. \quad (5)$$

In a block adaptive operation of the equalizer, its filters are updated on a block-by-block basis, i.e., the filters are kept fixed for a block of Q symbol periods, and are finally updated at the end of the block. During the b-th block interval, the received data are collected and accumulated together with the data of the previous block in the $Q \times K_f N$ matrix

$$\mathbf{X}(k) = \begin{bmatrix} \mathbf{X}_1(k) & \cdots & \mathbf{X}_N(k) \end{bmatrix} \quad (6)$$

where $k = bQ$, and the $Q \times K_f$ Toeplitz matrix $\mathbf{X}_n(k)$, $n = 1, \ldots, N$, is given by

$$\mathbf{X}_n(k) = \begin{bmatrix} x_n(k) & \cdots & x_n(k - K_f + 1) \\ \vdots & \ddots & \vdots \\ x_n(k + Q - 1) & \cdots & x_n(k + Q - K_f) \end{bmatrix}. \quad (7)$$

Similarly to the received data, decisions from all streams are stored in the $Q \times K_b M$ matrix

$$\mathbf{D}(k) = \begin{bmatrix} \mathbf{D}_1(k) & \cdots & \mathbf{D}_M(k) \end{bmatrix} \quad (8)$$

where the Toeplitz matrix $\mathbf{D}_m(k)$, is defined as

$$\mathbf{D}_m(k) = \begin{bmatrix} d_m(k-1) & \cdots & d_m(k - K_b) \\ \vdots & \ddots & \vdots \\ d_m(k + Q - 2) & \cdots & d_m(k + Q - K_b - 1) \end{bmatrix}, \quad (9)$$

for $m = 1, \ldots, M$, and $d_m(k)$ is the decision for symbol $s_m(k)$. Notice that the decisions in the upper triangular part of $\mathbf{D}_m(k)$ are related to symbols of previous blocks, i.e., $d_m(k - K_b), \ldots, d_m(k-1)$, whereas its strictly lower triangular part contains decisions of the current block.

After having defined the input data and the filters of the MIMO DFE, its output can be compactly expressed by

$$\mathbf{Y}(k) = \mathbf{X}(k)\mathbf{W}(k) + \mathbf{D}(k)\mathbf{B}(k) \quad (10)$$

where the $Q \times M$ output matrix is defined as

$$\mathbf{Y}(k) = \begin{bmatrix} y_1(k) & \cdots & y_M(k) \\ \vdots & \ddots & \vdots \\ y_1(k + Q - 1) & \cdots & y_M(k + Q - 1) \end{bmatrix}. \quad (11)$$

Each output $y_m(k)$ is fed into the slicer to yield the decision $d_m(k)$, i.e., $d_m(k) = f[y_m(k)]$, where $f[\cdot]$ stands for the decision device function. In (10), a 'causality' problem arises, that is, decisions of the current block are needed to extract symbols of the same block. This problem is solved by the iterative scheme presented later. Until then, let us assume that all required decisions are available. Such an assumption is valid during the training mode operation of the equalizer.

During the training mode, known symbol sequences are utilized to form the error at the equalizer output, i.e.,

$$\mathbf{E}(k) = \mathbf{S}(k) - \mathbf{Y}(k) \quad (12)$$

where the $Q \times M$ symbol matrix $\mathbf{S}(k)$ is expressed by using (1) as $\mathbf{S}(k) = \begin{bmatrix} \mathbf{s}(k) & \cdots & \mathbf{s}(k + Q - 1) \end{bmatrix}^T$. In decision directed mode, matrix $\mathbf{S}(k)$ is replaced by the hard decisions on $\mathbf{Y}(k)$, i.e., $f[\mathbf{Y}(k)]$. In both cases, an appropriate delay Δ should be inserted between the received samples and the desired symbol, which is typically set to $\Delta = K_f - 1$.

The update of the equalizer can now be performed as

$$\begin{bmatrix} \mathbf{W}(k + Q) \\ \mathbf{B}(k + Q) \end{bmatrix} = \begin{bmatrix} \mathbf{W}(k) \\ \mathbf{B}(k) \end{bmatrix} + \mu \begin{bmatrix} \mathbf{X}^H(k) \\ \mathbf{D}^H(k) \end{bmatrix} \mathbf{E}(k) \quad (13)$$

where $(\cdot)^H$ stands for Hermitian transposition, and μ is the step size of the LMS algorithm. A normalized version of the LMS (NLMS) can be obtained by substituting the constant μ by $\mu(k) = \frac{\mu}{p(k)+\epsilon}$, where $p(k)$ is a power estimate of the equalizer input, and ϵ a small positive constant.

Equations (10), (12), and (13) constitute the time domain block DFE algorithm. We now proceed with the derivation of the algorithm in the frequency domain.

B. Block MIMO DFE in the Frequency Domain

The time domain block LMS algorithm presented in the previous subsection suffers from two problems: 1) the correlation and convolution quantities in (10), (13) are of high complexity, and 2) the block implementation limits the allowable range of the step size and consequently reduces the convergence rate of the algorithm [9], [8]. A frequency domain implementation can alleviate these problems.

To proceed with the derivation of the algorithm in the frequency domain, we define $S = 2 \max\{K_f, K_b, Q\}$, and $\mathbf{F}$ as the $S \times S$ FFT matrix operator. To describe zero padding and sample removal operations required by the overlap and save method, we define the $S \times T$ constraint matrices

$$\mathbf{G}_{1,T} = \begin{bmatrix} \mathbf{I}_T \\ \mathbf{0}_{(S-T) \times T} \end{bmatrix}, \quad \mathbf{G}_{2,T} = \begin{bmatrix} \mathbf{0}_{(S-T) \times T} \\ \mathbf{I}_T \end{bmatrix}. \quad (14)$$

Moreover, to sum up outputs from different receive antennas or streams, we define the $S \times SM$ matrix $\mathbf{G}_M$ as

$$\mathbf{G}_M = \begin{bmatrix} \mathbf{I}_S & \cdots & \mathbf{I}_S \end{bmatrix}. \quad (15)$$

Then, input data are transformed in the frequency domain. At the n-th receive antenna, S samples are collected, FFT transformed, and stored in the diagonal matrix

$$\mathcal{X}_n(k) = diag \left(\mathbf{F} \begin{bmatrix} x_n(k + Q - S) \\ \vdots \\ x_n(k + Q - 1) \end{bmatrix} \right) \quad (16)$$

where $diag(\mathbf{x})$ stands for a diagonal matrix having the elements of vector $\mathbf{x}$ on its diagonal. In a similar way, frequency domain expressions for decision sequences can be stored in diagonal matrices

$$\mathcal{D}_m(k) = diag \left(\mathbf{F} \begin{bmatrix} d_m(k + Q - S - 1) \\ \vdots \\ d_m(k + Q - 2) \end{bmatrix} \right) \quad (17)$$

for $m = 1, \ldots, M$. The frequency domain quantities of (16)-(17) are accumulated in the diagonal matrix

$$\mathcal{L}(k) = \begin{bmatrix} \mathcal{X}_1(k) & \cdots & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathcal{X}_N(k) & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \mathcal{D}_1(k) & \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \cdots & \mathcal{D}_M(k) \end{bmatrix}. \quad (18)$$

The feedforward filters can be transformed in the frequency domain after zero padding, yielding the $SN \times M$ matrix

$$\mathcal{W}(k) = \left(\mathbf{I}_N \otimes \mathbf{F} \mathbf{G}_{1,K_f} \right) \mathbf{W}(k), \quad (19)$$

where $\otimes$ denotes the Kronecker matrix product. The feedback part of the MIMO DFE is represented by the $SM \times M$ matrix

$$\mathcal{B}(k) = \left(\mathbf{I}_M \otimes \mathbf{F} \mathbf{G}_{1,K_b} \right) \mathbf{B}(k). \quad (20)$$

By using the definitions of (18), (19), (20), and (15), the convolutions of (10) are replaced by multiplications in the frequency domain as

$$\mathcal{Y}(k) = \mathbf{G}_{N+M} \mathcal{L}(k) \begin{bmatrix} \mathcal{W}(k) \\ \mathcal{B}(k) \end{bmatrix}. \quad (21)$$

The output of the MIMO equalizer in the time domain can be calculated after IFFT transforming (21), and keeping the last Q elements for each stream, i.e.,

$$\mathbf{Y}(k) = \mathbf{G}_{2,Q}^T \mathbf{F}^H \mathcal{Y}(k) \quad (22)$$

The time domain output is fed into the slicer, the corresponding error signal is formed through (12), and subsequently transformed in the frequency domain in the $S \times M$ matrix

$$\mathcal{E}(k) = \mathbf{F} \mathbf{G}_{2,Q} \mathbf{E}(k). \quad (23)$$

The correction term needed to update the MIMO DFE is given by $\mathcal{L}^H(k) \mathbf{G}_{N+M}^T \mathcal{E}(k)$. To have an exact correspondence between the time and frequency domain implementations of the algorithm, each frequency domain filter should be premultiplied by the constraint matrix $\mathbf{F} \mathbf{G}_{1,K} \mathbf{G}_{1,K}^T \mathbf{F}^H$, where K is the filter length. The application of this constraint over each filter is described by the block diagonal matrix

$$\mathbf{C} = \begin{bmatrix} \mathbf{I}_N \otimes \mathbf{F} \mathbf{G}_{1,K_f} \mathbf{G}_{1,K_f}^T \mathbf{F}^H & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_M \otimes \mathbf{F} \mathbf{G}_{1,K_b} \mathbf{G}_{1,K_b}^T \mathbf{F}^H \end{bmatrix} \quad (24)$$

which premultiplies the correction term defined above.

Up to this point, a new mathematically equivalent implementation of the block adaptive MIMO DFE of subsection III-A has been derived, while a complexity reduction advantage has been gained through FFT operations. To improve convergence rate as well, one can exploit the approximate decoupling of the frequency domain filters [8]. It can be proved[1] that the autocorrelation matrix of the problem has a structure of near to diagonal blocks, and can be approximated as

$$\hat{\mathcal{R}}(k) = \lambda \hat{\mathcal{R}}(k - Q) \quad (25)$$
$$+ (1 - \lambda) \frac{Q}{S} \begin{bmatrix} \mathcal{X}_1^H(k) \mathcal{X}_1(k) & \cdots & \mathcal{X}_1^H(k) \mathcal{D}_M(k) \\ \vdots & \ddots & \vdots \\ \mathcal{D}_M^H(k) \mathcal{X}_1(k) & \cdots & \mathcal{D}_M^H(k) \mathcal{D}_M(k) \end{bmatrix}$$

[1]The proof is omitted here due to space limitations.

TABLE I
SKETCH OF THE PROPOSED ALGORITHM

1) Transform equalizer inputs in the frequency domain, (16)-(18).
2) Calculate equalizer output in the frequency domain (21).
3) Transform output in the time domain (22).
4) Calculate error in the time domain (12).
5) Transform error in the frequency domain (23).
6) Update the autocorrelation matrix from (25), and compute its inverse.
7) Update frequency domain equalizers (26).

where $0 < \lambda \leqslant 1$ is an exponential forgetting factor. Matrix $\hat{\mathcal{R}}(k)$ consists of $(N + M)^2$ diagonal blocks of dimension S. The inverse of $\hat{\mathcal{R}}(k)$, $\mathcal{P}(k)$, has the same structure as $\hat{\mathcal{R}}(k)$, and hence, inversion can be performed by simply inverting S smaller matrices of dimension $(N + M)$ in $O\left(S(N+M)^3\right)$ operations. Alternatively, $\mathcal{P}(k)$ can be calculated even more efficiently, by applying the matrix inversion lemma [9].

By incorporating the constraint (24), as well as the matrix step size $\mathcal{P}(k)$, we end up with the update equation

$$\begin{bmatrix} \mathcal{W}(k+Q) \\ \mathcal{B}(k+Q) \end{bmatrix} = \begin{bmatrix} \mathcal{W}(k) \\ \mathcal{B}(k) \end{bmatrix} + \mu \mathbf{C} \mathcal{P}(k) \mathcal{L}^H(k) \mathbf{G}_{N+M}^T \mathcal{E}(k). \quad (26)$$

The main steps of the proposed algorithm are summarized in Table I. The unknown decisions for the current block appearing in step 1 (Eq.(17)) are obtained through the iterative scheme presented next.

C. Iterative Scheme

To solve the 'causality' problem in (10), the iterative method of [8] is properly modified and generalized to the MIMO case. At each block step, the filtering equation of the MIMO-DFE is iterated I times, while keeping fixed the DFE filters. At the end of the i-th iteration new hard estimates are extracted for the current symbols, denoted by $d_m^i(p)$, $p = k, \ldots, k + Q - 1$, $m = 1, \ldots, M$.

Let us first separate the decisions of the previous and the current block. Thus, the decision vector appearing in the right-hand side of (17) at the end of the i-th iteration can be divided into two parts:

$$\mathcal{D}_m^i(k) = \mathcal{D}_m^p(k) + \mathcal{D}_m^{c,i}(k) \quad (27)$$

where matrix $\mathcal{D}_m^p(k)$ is the frequency domain representation of previous block symbols, i.e.,

$$\mathcal{D}_m^p(k) = diag \left(\mathbf{F} \mathbf{G}_{1,S-Q+1} \begin{bmatrix} d_m(k+Q-S-1) \\ \vdots \\ d_m(k-1) \end{bmatrix} \right), \quad (28)$$

and matrix $\mathcal{D}_m^{c,i}(k)$ is the frequency domain representation of the current decisions, i.e.,

$$\mathcal{D}_m^{c,i}(k) = diag \left(\mathbf{F} \mathbf{G}_{2,Q-1} \begin{bmatrix} d_m^i(k) \\ \vdots \\ d_m^i(k+Q-2) \end{bmatrix} \right). \quad (29)$$

To get the final estimate of the current block, the equalizer iterates through Steps 1-3 of the algorithm in Table I. However, at the i-th iteration the decisions made at the $(i-1)$-th iteration
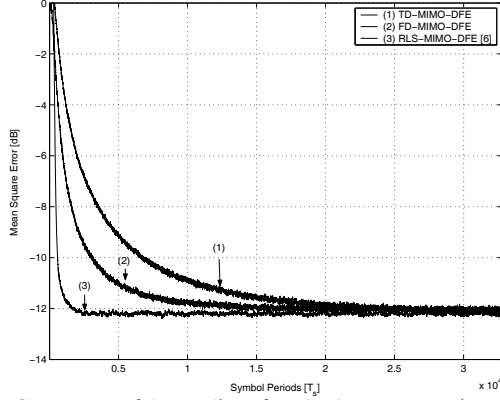
Fig. 1. Convergence of the equalizers for a 2×2 system operating constantly in training mode over a static channel at SNR=16dB.

Fig. 2. Convergence and tracking performance of equalizers for a 2×2 system trained for $4096T_s$ over a time varying channel at SNR=25dB.

are used, and thus matrices $\mathcal{D}_m^{i-1}(k)$ substitute $\mathcal{D}_m(k)$ in (17). Each iteration ends by making hard decisions for the desired symbols based on the soft equalizer output of this iteration.

During the first iteration, unknown decisions of the current block are initialized to zeroes. Furthermore, by using the rationale of [8], it can be proved that the number of iterations required for the method to converge to the decisions provided by the MMSE solution is Q. After the iterative procedure has been completed, the remaining steps of Table I are executed.

Finally, we consider the computational complexity of the proposed algorithm in terms of complex multiplications per symbol period. To simplify the analysis, we assume $K_f = K_b = Q$, and $M = N$. During the training mode, the complexity of the proposed equalizer (FD-MIMO-DFE) is $C_{FD}^T = (8M^2 + 10M)\log_2 Q + 16M^3 + 69M^2$. During the decision directed mode and assuming I iterations, its complexity increases to $C_{FD}^{DD} = C_{FD}^T + (4M\log_2 Q + 4M^2 + 4M)I$. Since M is typically much smaller than Q, the proposed algorithm is $O(Q\log_2 Q)$. The symbol-by-symbol NLMS implementation of the algorithm is $O(Q)$, while the RLS equalizer of [6] is $O(Q^2)$. Notice that the above complexity of FD-MIMO-DFE appears only when $I \simeq Q$, but usually fewer iterations are adequate. Hence, the proposed algorithm offers a good trade off in terms of complexity and convergence, as will be shown in the simulations.

IV. PERFORMANCE EVALUATION & CONCLUSION

The proposed equalizer was evaluated through computer simulations. We considered a 2×2 system transmitting uncoded QPSK symbols of T_s=0.2μsec over a wireless channel simulated according to the SUI-5 channel model [10]. The proposed equalizer (FD-MIMO-DFE) was compared with its symbol-by-symbol NLMS counterpart (TD-MIMO-DFE), as well as with the RLS equalizer (RLS-MIMO-DFE) of [6], whereas K_f=K_b=Q=64 was selected for all the equalizers.

To study the convergence of the equalizers, the channel was kept static for a period of $32768T_s$, while the system was operating constantly in training mode at SNR=16dB. As can be seen by the MSE plots of Fig. 1, RLS-MIMO-DFE has the fastest convergence rate, while FD-MIMO-DFE curve lies between RLS and NLMS curves. The tracking performance together with the error propagation effects were studied by
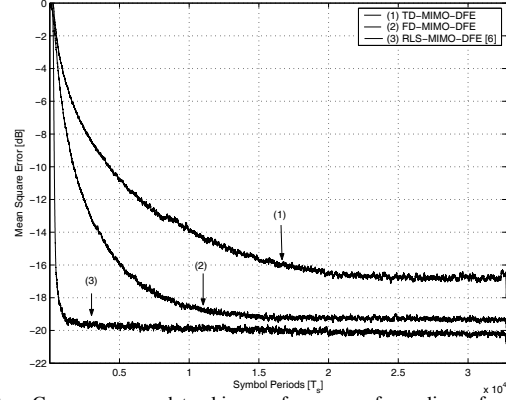
simulating the system over a time-varying channel, whereas the training period was restricted to $8192T_s$ and SNR=25dB. A normalized Doppler frequency of $f_D T_s = 1.8 \cdot 10^{-6}$ was simulated by using the Jakes method. As illustrated in Fig. 2, all the algorithms managed to track channel variation. The differences in the achieved steady-state errors are due to the different convergence and tracking properties and the related error propagation effects.

To conclude, in this paper we developed a block adaptive frequency domain MIMO DFE. To combat the causality problem inherent in the block DFE formulation, we generalized the iterative scheme of [8], while a more accurate approximation of the inverse autocorrelation matrix of the problem was employed. The resultant equalizer exhibits favorable convergence properties at reasonable computational complexity, making it a feasible solution for wideband MIMO systems.

REFERENCES

[1] N. Al-Dhahir and A. H. Sayed, "The finite length multi-input multi-output MMSE-DFE," *IEEE Trans. Signal Processing*, vol. 48, no. 10, pp. 2921–2936, Oct. 2000.
[2] X. Zhu and R. Murch, "Layered space-frequency equalization in a single-carrier MIMO system for frequency-selective channels," *IEEE Trans. Wireless Commun.*, vol. 3, no. 3, pp. 701–708, May 2004.
[3] R. Kalbasi, R. Dinis, D. Falconer, and A. H. Banihashemi, "Hybrid time-frequency layered space-time receivers for severe time-dispersive channels," in *Proc. IEEE Workshop on Signal Processing Advances on Wireless Communications*, Lisbon, July 2004.
[4] J. Coon, S. Armour, M. Beach, and J. McGeehan, "Adaptive frequency-domain equalization for single-carrier multiple-input multiple-output wireless transmissions," *IEEE Trans. Signal Processing*, vol. 53, no. 8, pp. 3247–3256, Aug. 2005.
[5] H. H. Dam, S. Nordholm, and H. J. Zepernick, "Frequency domain adaptive equalization for MIMO systems," in *Proc. IEEE Vehicular Technology Conf. VTC Fall*, Orlando, FL, Oct. 2003.
[6] A. Maleki-Tehrani, B. Hassibi, and J. M. Cioffi, "Adaptive equalization of multiple-input multiple-output (MIMO) channels," in *Proc. IEEE Int. Conf. Communications*, New Orleans, LA, June 2000, pp. 1670–1674.
[7] A. A. Rontogiannis, V. Kekatos, and K. Berberidis, "An adaptive decision feedback equalizer for time-varying frequency selective MIMO channels," in *Proc. IEEE Workshop on Signal Processing Advances in Wireless Communications*, Cannes, July 2006.
[8] K. Berberidis and P. Karaivazoglou, "An efficient block adaptive decision feedback equalizer implemented in the frequency domain," *IEEE Trans. Signal Processing*, vol. 50, no. 9, pp. 2273–2285, Sept. 2002.
[9] H. Buchner, J. Benesty, and W. Kellermann, "Generalized multichannel frequency-domain adaptive filtering: Efficient realization and application to hands-free speech communication," *Signal Processing (Elsevier)*, vol. 85, no. 3, pp. 549–570, Mar. 2005.
[10] V. Erceg et al , "Channel models for fixed wireless applications," *IEEE 802.16a cont. 802.16.3c-01/29r1*, Feb. 2001.