

REFERENCES

- [1] J. C. M. Bermudez and N. J. Bershad, "A nonlinear analytical model for the quantized LMS algorithm—The arbitrary step size case," *IEEE Trans. Signal Processing*, vol. 44, pp. 1175–1183, May 1996.
- [2] N. J. Bershad and J. C. M. Bermudez, "A nonlinear analytical model for the quantized LMS algorithm—The powers-of-two case," *IEEE Trans. Signal Processing*, vol. 44, pp. 2895–2900, Nov. 1996.
- [3] C. Caraiscos and B. Liu, "A roundoff error analysis of the LMS adaptive algorithm," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-32, pp. 34–41, Feb. 1984.
- [4] A. Chandrakasan *et al.*, "Minimizing power using transformations," *IEEE Trans. Comput.-Aided Design*, vol. 14, pp. 12–31, Jan. 1995.
- [5] R. D. Gitlin, J. F. Hayes, and S. B. Weinstein, *Data Communications Principles*, New York: Plenum, 1992.
- [6] G. H. Im and J. J. Werner, "Bandwidth-efficient digital transmission up to 155 Mb/s over unshielded twisted-pair wiring," *IEEE J. Select. Areas Commun.*, vol. 13, pp. 1643–1655, Dec. 1995.
- [7] N. R. Shanbhag and M. Goel, "Low-power adaptive filter architectures and their application to 51.84 Mb/s ATM-LAN," *IEEE Trans. Signal Processing*, vol. 45, pp. 1276–1290, May 1997.
- [8] N. R. Shanbhag and K. K. Parhi, "Relaxed look-ahead pipelined LMS adaptive filters and their application to ADPCM coder," *IEEE Trans. Circuits Syst.*, vol. 40, pp. 753–766, Dec. 1993.

On Hetero-Associative Neural Networks and Adaptive Interference Cancellation

Chanchal Chatterjee and Vwani P. Roychowdhury

Abstract—We discuss two novel adaptive algorithms for generalized eigendecomposition that are derived from a two-layer linear feedforward hetero-associative neural network. In addition, we provide a rigorous convergence analysis of the adaptive algorithms by using stochastic approximation theory. Finally, we use these algorithms for on-line multiuser access interference cancellation in code-division-multiple-access-based cellular communications. Numerical simulations are reported to demonstrate their rapid convergence.

Index Terms—Adaptive generalized eigen-decomposition, on-line interference cancellation.

I. INTRODUCTION

We study two novel adaptive algorithms for generalized eigendecomposition that are derived from a two-layer linear hetero-associative neural network. We discuss applications of these algorithms in an adaptive beamforming example to solve the near-far problem in code-division-multiple-access (CDMA) based cellular communications. Note that the well-studied topic of principal component analysis [1] provides adaptive algorithms for eigendecomposition of a correlation matrix A , which is the limit matrix of a single sequence of random matrices. We, on the other hand, provide adaptive algorithms for generalized eigendecomposition of a matrix

Manuscript received July 24, 1997; revised October 13, 1997. This work was supported in part by NSF Grants ECS-9308814 and ECS-9523423. The associate editor coordinating the review of this paper and approving it for publication was Prof. Yu-Hen Hu.

C. Chatterjee is with GDE Systems Inc., San Diego, CA 92127 USA.

V. P. Roychowdhury is with the Electrical Engineering Department, University of California, Los Angeles, CA 90095 USA.

Publisher Item Identifier S 1053-587X(98)03937-3.

pair (A, B) , which are the limit matrices of two sequences of random matrices.

A. Adaptive Beamforming for CDMA Based Cellular: A Case Study

As an example of an application that requires adaptive generalized eigendecomposition, we study the problem of on-line cochannel interference cancellation to solve the near-far problem in CDMA-based cellular communications. A number of nonadaptive methods have been proposed [3], [5]–[7] to solve this problem. A common scheme uses multiple (say, m) antennas to receive the signal at the base. The output of each antenna is put through a matched filter corresponding to the code of the desired user [7]–[9] (see Fig. 1). Although there are many methods to extract the desired signal at the base, we next consider a particular method that has been studied by several researchers [8], [9]. In the IS-95 standard, the bit period of the signal is on the order of 100 μ s in duration. Within each bit period, there is roughly a 10 μ s or so interval during which the desired filtered signal occurs. During this period of time, the signal plus interference correlation matrix A is estimated. In the remaining 90 μ s or so, we estimate the interference correlation matrix B .

Given the correlation matrices A and B of signal plus interference and interference, respectively, we compute the weight vector w of a transversal filter such that we maximize the signal-to-interference plus noise ratio (SINR) expressed as $\max_w \{ \text{SINR} = (w^H A w / w^H B w) \}$.¹ The solution to this problem is the generalized eigenvector of the matrix pencil (A, B) corresponding to the largest generalized eigenvalue. Although this computation appears to be relatively uncomplicated, for typical urban multipath time delay spreads, the correlation matrices A and B are of rather large dimension, even when the number of receiving antennas are relatively small. For example, if we sample two times per microsecond for a 10 μ s time delay spread of the desired signal, and if we have eight receiving antennas, the resulting space-time correlation matrices A and B are of dimension 160×160 . Since generalized eigendecomposition requires $O(n^3)$ computation, this computation is quite intensive.

In an attempt to simplify this problem, an alternative method [8] constructs a lower dimensional $m \times m$ matrix pencil (A, B) for an m -antenna problem. We next compute the first p ($< m$) generalized eigenvectors of the matrix pencil (A, B) corresponding to the p largest generalized eigenvalues. The p weight vectors transform the initial m -dimensional antenna space to a p -dimensional beam space. The first principal generalized eigenvector is now computed in the reduced dimensional beam space with lower dimensional spatial correlation matrices (A, B) .

In all of the above-mentioned schemes, interference cancellation can be achieved by first computing the matrix pencil (A, B) after collecting all of the samples and then the application of a numerical procedure [2], i.e., by working in a batch fashion. If the principal generalized eigenvectors are computed in a batch mode, the time delay needed to make a decision would not only include bit times needed to average the spatial correlation matrices but the subsequent time required to compute the generalized eigenvectors as well. In addition, the batch mode operation will not, in general, exploit the fact that there is a gradual time variation of the weight vector w in a urban mobile environment and that we need to recompute w after every few (say 4) bits. In order to reduce this computation and obtain effective interference cancellation, an adaptive (i.e., on-

¹Superscript H denotes Hermitian transpose matrix operation.

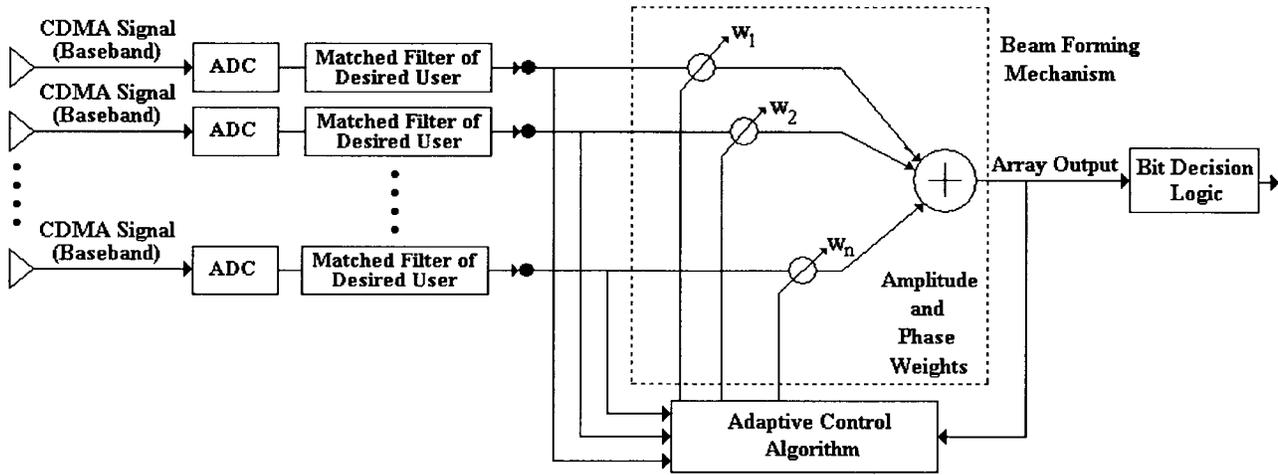


Fig. 1. Adaptive antenna array for interference cancellation with the first principal generalized eigenvector of signal correlation matrix with respect to interference correlation matrix. For p generalized eigenvectors, we have p such outputs.

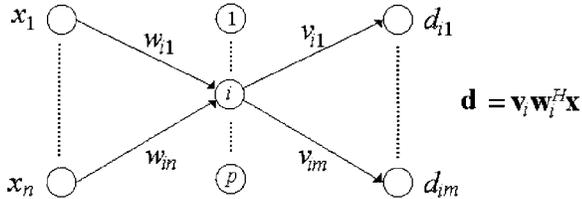


Fig. 2. Hetero-associative network for extracting the generalized eigenvectors.

line) algorithm is required. In the following sections, we derive two such algorithms and provide a proof of convergence. Fig. 1 shows an example of an adaptive antenna array to maximize SINR.

In Section II, we describe two novel iterative algorithms for generalized eigendecomposition from a two-layer hetero-associative network. In Section III, we describe two adaptive algorithms for generalized eigendecomposition from two sequences of random matrices with a convergence proof. In Section IV, we discuss an application of these algorithms in an interference cancellation problem. Section V has the concluding remarks.

II. HETERO-ASSOCIATIVE NETWORKS AND ITERATIVE ALGORITHMS FOR GENERALIZED EIGENDECOMPOSITION

We consider a two-layer linear network performing a one-from- m classification (see Fig. 2). Let $\mathbf{x} \in C^n$ be an input to the network to be classified into one out of m classes $\omega_1, \dots, \omega_m$. If $\mathbf{x} \in \omega_i$, then the desired output $\mathbf{d} = \mathbf{e}_i$ (i th std. basis vector). Without loss of generality, we assume the inputs to be a zero-mean stationary process with a nonsingular covariance matrix. Since our computations require the matrices (A, B) , we define them in terms of the random vectors \mathbf{x} and \mathbf{d} , which are the inputs and outputs, respectively, of the network as follows.

$$M = E[\mathbf{d}\mathbf{d}^H] \in C^{m \times m}, \quad A = MM^H \in C^{n \times n},$$

and

$$B = E[\mathbf{x}\mathbf{x}^H] \in C^{n \times n}. \quad (1)$$

A. Extracting the Principal Generalized Eigenvectors

In the two-layer linear hetero-associative network, let there be p neurons in the hidden layer and m output units (see Fig. 2). The aim is to develop an algorithm so that individual weight vectors for the first layer converge to the first $p \leq m$ generalized eigenvectors

corresponding to the p significant generalized eigenvalues arranged in decreasing order. Let $\mathbf{w}_i \in C^n$ ($i = 1, \dots, n$) be the weight vectors for the input layer, and let $\mathbf{v}_i \in C^m$ ($i = 1, \dots, m$) be the weight vectors for the output layer (see Fig. 2).

The neurons are trained sequentially, i.e., the training of the i th neuron is started only after the weight vector of the $(i-1)$ th neuron has converged. Assume that all the $i-1$ previous neurons have already been trained and that their weights have converged to the optimal weight vectors \mathbf{w}_j for $j \in [1, i-1]$. To extract the i th generalized eigenvector in the output of the i th neuron, the updating model for this neuron should be constructed by subtracting the results from all previously computed $i-1$ generalized eigenvectors from the desired output \mathbf{d}_i as

$$\tilde{\mathbf{d}} = \mathbf{d}_i - \sum_{j=1}^{i-1} \gamma_j \mathbf{v}_j \mathbf{w}_j^H \mathbf{x} \quad (2)$$

where γ_j for $j = 1, \dots, i-1$ are a set of real-valued weights for the $i-1$ previous neurons. A sufficient condition for the convergence of our algorithms is $\gamma_j \geq 1$ for $j = 1, \dots, i-1$. Note that this process is equivalent to the deflation [1] of the desired output. The constrained MSE criterion at the network output is

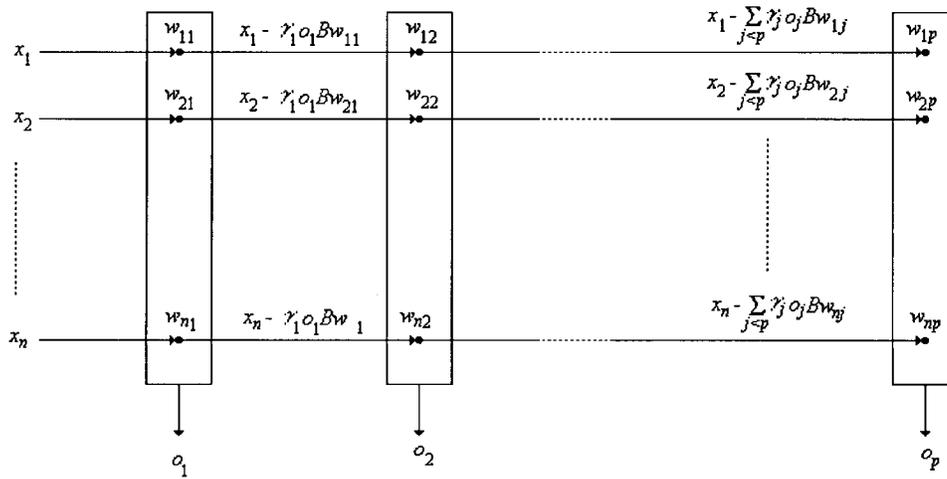
$$J(\mathbf{w}_i, \mathbf{v}_i) = E \left[\left\| \mathbf{d}_i - \sum_{j=1}^{i-1} \gamma_j \mathbf{v}_j \mathbf{w}_j^H \mathbf{x} - \mathbf{v}_i \mathbf{w}_i^H \mathbf{x} \right\|^2 \right] + \mu (\mathbf{w}_i^H B \mathbf{w}_i - 1) \quad (3)$$

where μ is a Lagrange multiplier. Applying the gradient descent approach on (3), we obtain the update equation for \mathbf{w}_i as

$$\mathbf{w}_{k+1}^i = \mathbf{w}_k^i + \eta \left(M \mathbf{v}_k^i - B \mathbf{w}_k^i (\mathbf{w}_k^i)^H M \mathbf{v}_k^i - B \sum_{j=1}^{i-1} \gamma_j \mathbf{w}_k^j \mathbf{v}_k^{jH} \mathbf{v}_k^i \right) \quad (4)$$

where η is a gain constant. Differentiating (3) with respect to \mathbf{v}_i and equating it to zero, we obtain the optimum value of \mathbf{v}_i as $M^H \mathbf{w}_i$. Substituting this \mathbf{v}_i in (4), we obtain

$$\mathbf{w}_{k+1}^i = \mathbf{w}_k^i + \eta \left(A \mathbf{w}_k^i - B \mathbf{w}_k^i (\mathbf{w}_k^i)^H A \mathbf{w}_k^i - B \sum_{j=1}^{i-1} \gamma_j \mathbf{w}_k^j \mathbf{v}_k^{jH} A \mathbf{w}_k^i \right). \quad (5)$$


 Fig. 3. Implementation of algorithm (11). The time index k is not shown.

Let W_k be the matrix whose i th column is \mathbf{w}_k^i . Then, we can write (5) in matrix form as

$$W_{k+1} = W_k + \eta(AW_k - BW_k \text{UT}_\gamma[W_k^H AW_k]) \quad (6)$$

where $\text{UT}_\gamma[\cdot]$ sets all elements below the diagonal of its matrix argument to zero and multiplies the j th row above the diagonal with (≥ 1) , thereby making the matrix upper triangular.

B. A Different Iterative Algorithm for Generalized Eigendecomposition

In the previous analysis for a two-layer linear hetero-associative network, we observed the optimum value for $V = W^H M$, where the i th column of W and row of V are formed by \mathbf{w}_i and \mathbf{v}_i , respectively. It is, therefore, worthwhile to explore the gradient descent procedure on the error function below instead of (3)

$$\tilde{J}(W) = E[\|\mathbf{d} - M^H W W^H \mathbf{x}\|^2]. \quad (7)$$

By differentiating this error function with respect to W and including the deflation process, we obtain the following update procedure for W instead of (6).

$$W_{k+1} = W_k + \eta(2AW_k - BW_k \text{UT}_\gamma[W_k^H AW_k] - AW_k \text{UT}_\gamma[W_k^H BW_k]). \quad (8)$$

III. ADAPTIVE ALGORITHMS FOR GENERALIZED EIGENDECOMPOSITION AND PROOF OF CONVERGENCE

Since in many applications we do not have the matrices A and B [which are required in algorithms (6) and (8)], we need to obtain an adaptive algorithm for generalized eigendecomposition. Here, instead of A and B , we have two sequences of random matrices $\{A_k \in C^{n \times n}\}$ and $\{B_k \in C^{n \times n}\}$ with $\lim_{k \rightarrow \infty} E[A_k] = A$ and $\lim_{k \rightarrow \infty} E[B_k] = B$. We assume that A and B are Hermitian and positive definite. We suggest the following generalization of (8) to obtain an adaptive algorithm for generalized eigendecomposition

$$W_{k+1} = W_k + \eta_k(2A_k W_k - B_k W_k \text{UT}_\gamma[W_k^H A_k W_k] - A_k W_k \text{UT}_\gamma[W_k^H B_k W_k]). \quad (9)$$

Here, $W_k \in C^{n \times p}$ ($p \leq n$), and $\{\eta_k\}$ is a sequence of scalar gains (see Section III-A). The sequences $\{A_k\}$ and $\{B_k\}$ are instantaneous values of the matrices A and B , respectively. One commonly used model for generating $\{A_k\}$ and $\{B_k\}$ is from sequences of random vectors $\{\mathbf{x}_k\}$ and $\{\mathbf{y}_k\}$, respectively, whereby $\lim_{k \rightarrow \infty} E[\mathbf{x}_k \mathbf{x}_k^H] = A$ and $\lim_{k \rightarrow \infty} E[\mathbf{y}_k \mathbf{y}_k^H] = B$. The following algorithm gives an example for generating $\{A_k\}$ and $\{B_k\}$.

$$A_k = A_{k-1} + \tau_k(\mathbf{x}_k \mathbf{x}_k^H - A_{k-1})$$

and

$$B_k = B_{k-1} + \tau_k(\mathbf{y}_k \mathbf{y}_k^H - B_{k-1}) \quad (10)$$

where A_0 and B_0 are Hermitian, and $\{\tau_k\}$ is a scalar gain sequence. Similar to (9), we obtain an adaptive algorithm for generalized eigendecomposition from (6) as

$$W_{k+1} = W_k + \eta_k(A_k W_k - B_k W_k \text{UT}_\gamma[W_k^H A_k W_k]). \quad (11)$$

Although we derive algorithms (9) and (11) from the network MSE, this derivation does not guarantee their convergence. Hence, we provide a proof of convergence by using *stochastic approximation theory*. Since the proofs for the two algorithms are similar, we give the convergence results only for algorithm (9).

Algorithms (9) and (11) are suitable for neural network implementations. One such implementation of (11) with a one-layer network of p linear parallel units can be obtained by considering $A_k = \mathbf{x}_k \mathbf{x}_k^H$, where \mathbf{x}_k is the input vector. Let \mathbf{w}_k^i be the weight vector and o_k^i be the output of the i th unit; then, from (11)

$$\Delta \mathbf{w}_k^i = \eta_k o_k^i \left(\mathbf{x}_k^i - o_k^i B_k \mathbf{w}_k^i - \sum_{j=1}^{i-1} \gamma_j o_k^j B_k \mathbf{w}_k^j \right),$$

where

$$o_k^j = \mathbf{w}_k^j H \mathbf{x}_k.$$

The network implementation of the above equation is given in Fig. 3.

A. Stochastic Approximation Convergence Proof for Algorithm (9)

In order to prove the convergence of (9), we use stochastic approximation theory due to Ljung [4]. In stochastic approximation theory, we study the asymptotic properties of (9) in terms of the

ordinary differential equation (ODE)

$$\begin{aligned} \frac{d}{dt}W(t) &= \lim_{k \rightarrow \infty} E[2A_k W(t) - B_k W(t) \text{UT}_\gamma[W(t)^H A_k W(t)] \\ &\quad - A_k W(t) \text{UT}_\gamma[W(t)^H B_k W(t)]] \end{aligned}$$

where $W(t)$ is the continuous time counterpart of W_k with t denoting continuous time. The method of proof requires

- i) establishing a set of conditions to be imposed on $A, B, A_k, B_k,$ and η_k ;
- ii) finding the stable stationary points of the ODE;
- iii) demonstrating that W_k visits a compact subset of the domain of attraction of a stable stationary point infinitely often.

We use Theorem 1 of Ljung [4] for the convergence proof. The following is a general set of assumptions for the convergence proof of (9):

Assumption A1: Each \mathbf{x}_k and \mathbf{y}_k is bounded with probability one, and $\lim_{k \rightarrow \infty} E[\mathbf{x}_k \mathbf{x}_k^H] = A$ and $\lim_{k \rightarrow \infty} E[\mathbf{y}_k \mathbf{y}_k^H] = B$, where A and B are positive definite.

Assumption A2: $\{\eta_k \in \mathbb{R}^+\}$ satisfies $\eta_k \downarrow 0, \sum_{k=0}^{\infty} \eta_k = \infty, \sum_{k=0}^{\infty} \eta_k^r < \infty (r > 1)$, and $\lim_{k \rightarrow \infty} \sup(\eta_k^{-1} - \eta_{k-1}^{-1}) < \infty$.

Assumption A3: The p largest generalized eigenvalues of A with respect to B are each of unit multiplicity.

Lemma 1: Let A1 and A2 hold. Let W^* be a locally asymptotically stable (in the sense of Liapunov) solution to the ODE

$$\begin{aligned} \frac{d}{dt}W(t) &= 2AW(t) - BW(t) \text{UT}_\gamma[W(t)^H AW(t)] \\ &\quad - AW(t) \text{UT}_\gamma[W(t)^H BW(t)] \end{aligned} \quad (12)$$

with domain of attraction $D(W^*)$. Then, if there is a compact subset S of $D(W^*)$ such that $W_k \in S$ infinitely often, then we have $W_k \rightarrow W^*$ with probability one as $k \rightarrow \infty$. ■

We denote $\lambda_1 > \lambda_2 > \dots > \lambda_p > \lambda_{p+1} \geq \dots \geq \lambda_n \geq 0$ as the generalized eigenvalues of A with respect to B and ϕ_i as the generalized eigenvector corresponding to λ_i such that ϕ_1, \dots, ϕ_n are orthonormal with respect to B . Let $\Phi = [\phi_1 \dots \phi_n]$ and $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_n)$ denote the matrix of generalized eigenvectors and eigenvalues of A with respect to B . Note that if ϕ_i is a generalized eigenvector, then $d_i \phi_i$ ($|d_i| = 1$) is also a generalized eigenvector.

In the next two lemmas, we first prove that all the possible equilibrium points of the ODE (12) are up to an arbitrary permutation of the p generalized eigenvectors of A with respect to B corresponding to the p largest generalized eigenvalues. We next prove that all these equilibrium points of the ODE (12) are unstable equilibrium points, except for $[d_1 \phi_1 \dots d_n \phi_n]$, where $|d_i| = 1$ for $i = 1, \dots, p$.

Lemma 2: For the ODE (12); let A1 and A3 hold. Then, $W = \Phi D P$ are equilibrium points of (12), where $D = [D_1 | 0]^H$ is a $n \times p$ matrix with D_1 being a $p \times p$ diagonal matrix with diagonal elements d_i such that $|d_i| = 1$ or $d_i = 0$, and P is a $n \times n$ arbitrary permutation matrix. ■

Lemma 3: Let A1 and A3 hold. Then, $W = \Phi D$ (where $D = [D_1 | 0]^H, D_1 = \text{diag}(d_1, \dots, d_p), |d_i| = 1$) are stable equilibrium points of the ODE (12). In addition, $W = \Phi D P$ ($d_i = 0$ for $i \leq p$ or $P \neq I$) are unstable equilibrium points of the ODE (12). ■

Lemma 4: For the ODE (12); let A1 and A3 hold. Then, the points $W = \Phi D$ (where $D = [D_1 | 0]^H, D_1 = \text{diag}(d_1, \dots, d_p), |d_i| = 1$ for $i = 1, \dots, p$) are asymptotically stable. ■

Lemma 5: Let A1–A3 hold. Then, there exists a uniform upper bound for η_k such that W_k is uniformly bounded with probability one. ■

The convergence of algorithm (9) can now be established by referring to Theorem 1 of Ljung.

Theorem 1: Let A1–A3 hold. Assume that with probability one the process $\{W_k\}$ visits infinitely often a compact subset of the domain of attraction of one of the asymptotically stable points ΦD . Then, with probability one, $\lim_{k \rightarrow \infty} W_k = \Phi D$. ■

IV. EXPERIMENTAL RESULTS

We describe the performance of the adaptive generalized eigen-decomposition algorithms (9) and (11) with an example of on-line cochannel interference cancellation to solve the near–far problem in digital mobile communications. The problem occurs when a desired user transmits a signal from a far distance to the base, whereas another cochannel user simultaneously transmits very near to the base, providing significant interference. The solution involves the design of a system that efficiently and accurately cancels the interference and receiver noise and detects the desired signal.

A. Data Model

The details of the data model can be found in [8] and [9]. Here, we present the information needed to set up the sequences $\{A_k\}$ and $\{B_k\}$ necessary for our algorithms. In this application, each bit period is of 127 μs duration. We receive the signal at the base with $m = 8$ uniformly spaced linear array of antennas. Both the desired signal and interference are CDMA signals with 127 chips per bit. The output of each antenna is put through a matched filter corresponding to the code of the desired user. Experimental results in an urban cellular environment reveal the worst-case time delay spread due to multipath is 10 μs , during which we estimate the signal plus interference correlation matrix A . This leaves us 117 μs to estimate the cochannel interference correlation matrix B . We sample the signal and interference at 0.5 μs interval.

For each antenna, we therefore obtain 20 samples of the desired signal plus interference and 234 samples of the interference for the bit period of 127 μs . Due to eight antennas, for each time sample, we obtain a complex data vector of dimension $n = 8$. Thus, for each bit period, we obtain 20 desired signal plus interference vectors \mathbf{x}_k and 234 interference vectors \mathbf{y}_k . We start the algorithm after the first interference vector is received.

Our goal is to generate the optimum weight matrix W that maximizes the signal power and consists of the $p < 8$ principal generalized eigenvectors of A with respect to B . For a practical system, our objective is to cancel the interference (i.e., determine W) in as few bits as possible. For the state of the art, accurate interference cancellation within 2–5 bits is considered acceptable [8], [9]. We use our adaptive procedures to estimate W from a sequence of samples $\{\mathbf{x}_k\}$ and $\{\mathbf{y}_k\}$.

B. Numerical Results

We first compute the signal and interference correlation matrices A and B , respectively, by averaging all $\mathbf{x}_k \mathbf{x}_k^H$ and $\mathbf{y}_k \mathbf{y}_k^H$ collected over 5-bit periods. We shall refer to the generalized eigenvectors and eigenvalues computed from these A and B matrices by a standard numerical analysis method [2] as the *actual values*. The four largest generalized eigenvalues of A with respect to B are 25.88, 14.76, 1.80, and 1.60. Clearly, the first two generalized eigenvalues and the corresponding generalized eigenvectors are important. We use adaptive algorithm (9) to compute the first and second generalized eigenvectors and eigenvalues of A with respect to B . We use (10) to generate A_k and B_k sequences from $\{\mathbf{x}_k\}$

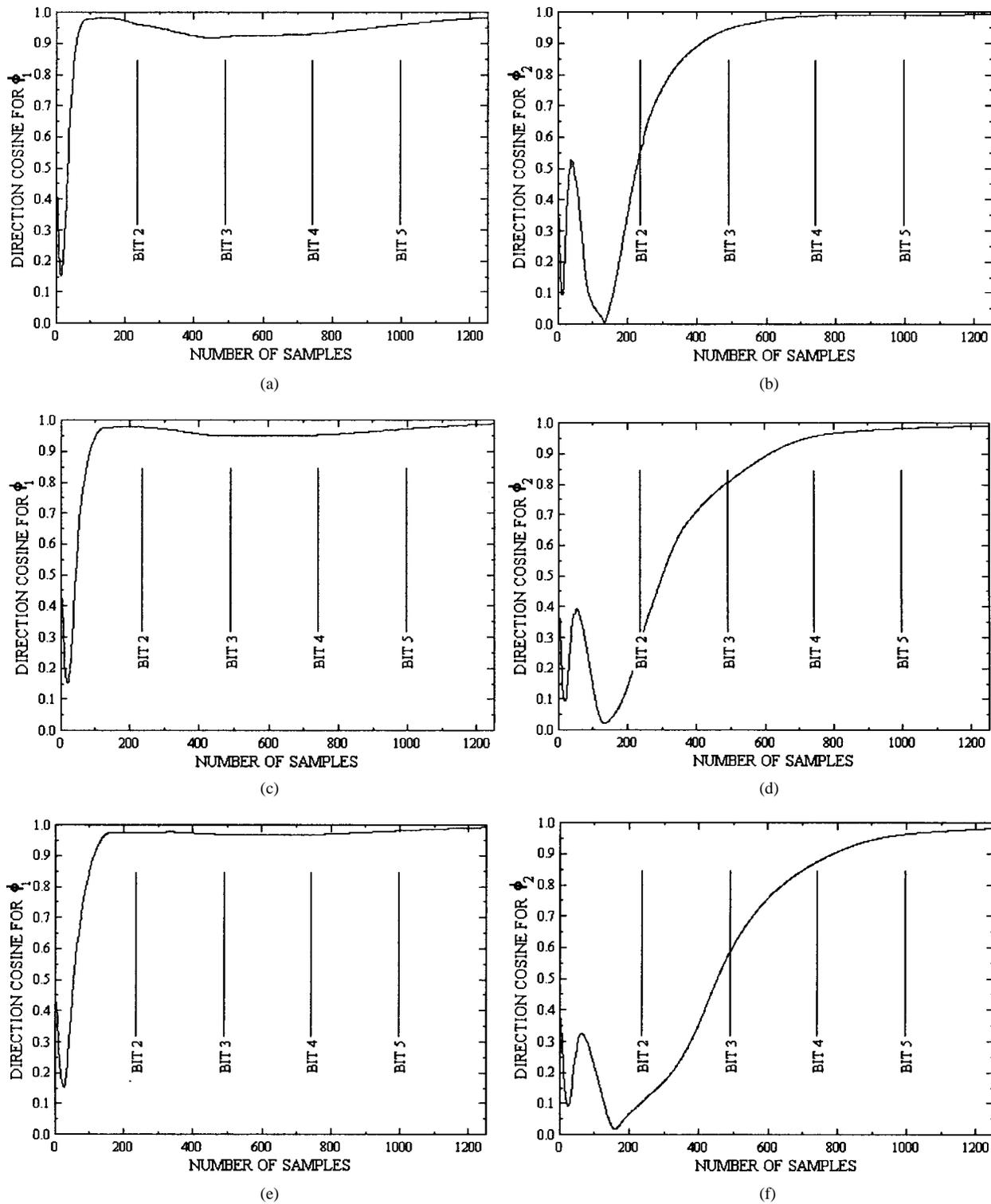


Fig. 4. Convergence of the first two principal generalized eigenvectors by algorithm (9) with different η_k and $\gamma = 1$. (a) ϕ_1 with $\eta_k = 1/(400 + k)$. (b) ϕ_2 with $\eta_k = 1/(400 + k)$. (c) ϕ_1 with $\eta_k = 1/(600 + k)$. (d) ϕ_2 with $\eta_k = 1/(600 + k)$. (e) ϕ_1 with $\eta_k = 1/(800 + k)$. (f) ϕ_2 with $\eta_k = 1/(800 + k)$.

and $\{y_k\}$, respectively, with $\tau_k = 1/k$. In order to measure the accuracy of the estimated generalized eigenvectors, we compute the direction cosine at k th update of the adaptive algorithm as $|\mathbf{w}_k^H \phi| / \|\mathbf{w}_k\| \|\phi\|$, where \mathbf{w}_k is the estimated generalized eigenvector at k th update, and ϕ is the actual generalized eigenvector computed from all collected samples by the conventional (numerical analysis) method.

We use (9) with $\gamma_j = \gamma$ for $j = 1, \dots, p$. We first choose different values of η_k , and $\gamma = 1$ and show the convergence results in Figs. 4 and 5. We observe the following.

- i) The first principal generalized eigenvector and eigenvalue converges within just two bits of signal.
- ii) The convergence of the second principal generalized eigenvector and eigenvalue is slower. However, for $\eta_k = 1/(400 + k)$,

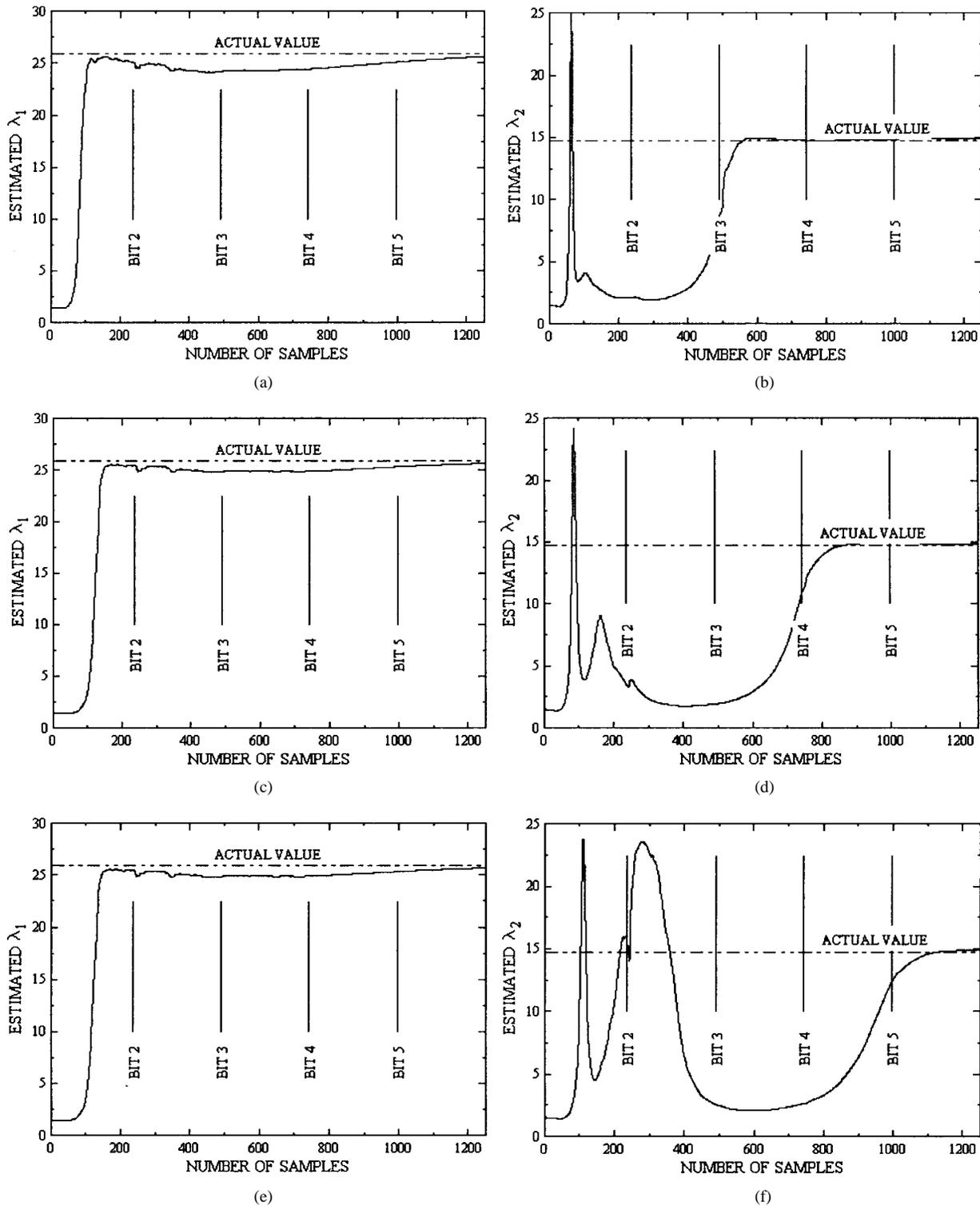


Fig. 5. Convergence of the first two principal generalized eigenvectors by algorithm (9) with different η_k and $\gamma = 1$. (a) γ_1 with $\eta_k = 1/(400 + k)$. (b) γ_2 with $\eta_k = 1/(400 + k)$. (c) γ_1 with $\eta_k = 1/(600 + k)$. (d) γ_2 with $\eta_k = 1/(600 + k)$. (e) γ_1 with $\eta_k = 1/(800 + k)$. (f) γ_2 with $\eta_k = 1/(800 + k)$.

we estimate the second principal generalized eigenvector to 94.5% accuracy in two bits of signal and to 99.2% accuracy in five bits of signal. For $\eta_k = 1/(800 + k)$, we estimate it to 98.2% accuracy in five bits of signal.

- iii) As $\{\eta_k\}$ increases up to an upper limit given by Lemma 5, we improve the convergence of algorithm (9).
- iv) Similar results are observed for increasing γ as shown in Fig. 6.

We now compare the convergence of Algorithms (9) and (11). The experiments in Fig. 7 suggest that Algorithm (9) estimates the generalized eigenvectors faster than Algorithm (11). The difference in convergence rate appears to be more for the minor generalized eigenvectors and eigenvalues than for the first principal generalized eigenvector and eigenvalue. An analytical study on this experimental observation will appear elsewhere.

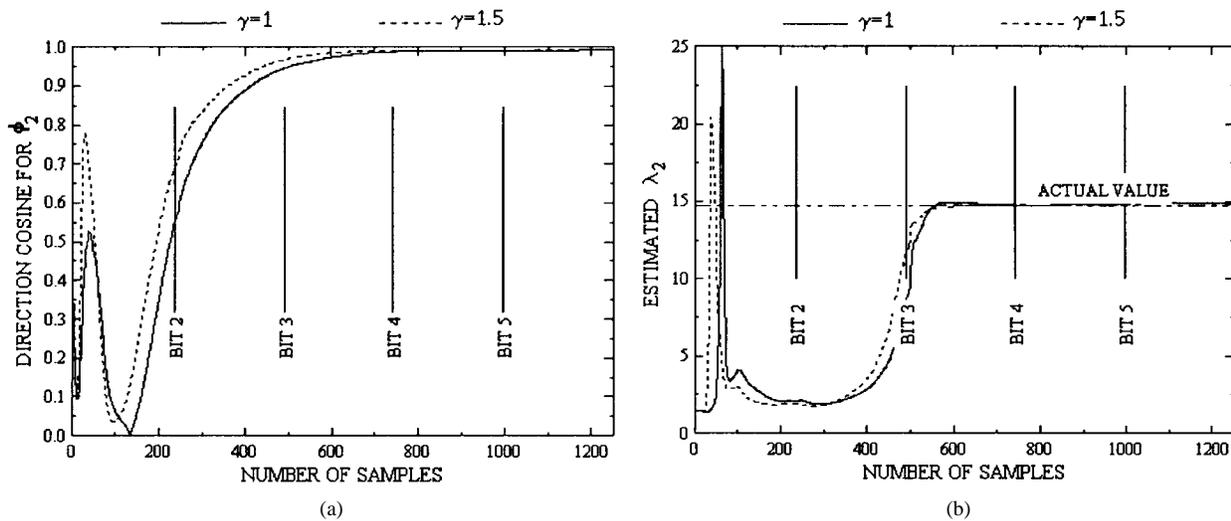


Fig. 6. Convergence of the second principal generalized eigenvector and eigenvalue by algorithm (9) with different γ and $\eta_k = 1/(400 + k)$. (a) ϕ_2 . (b) λ_2 .

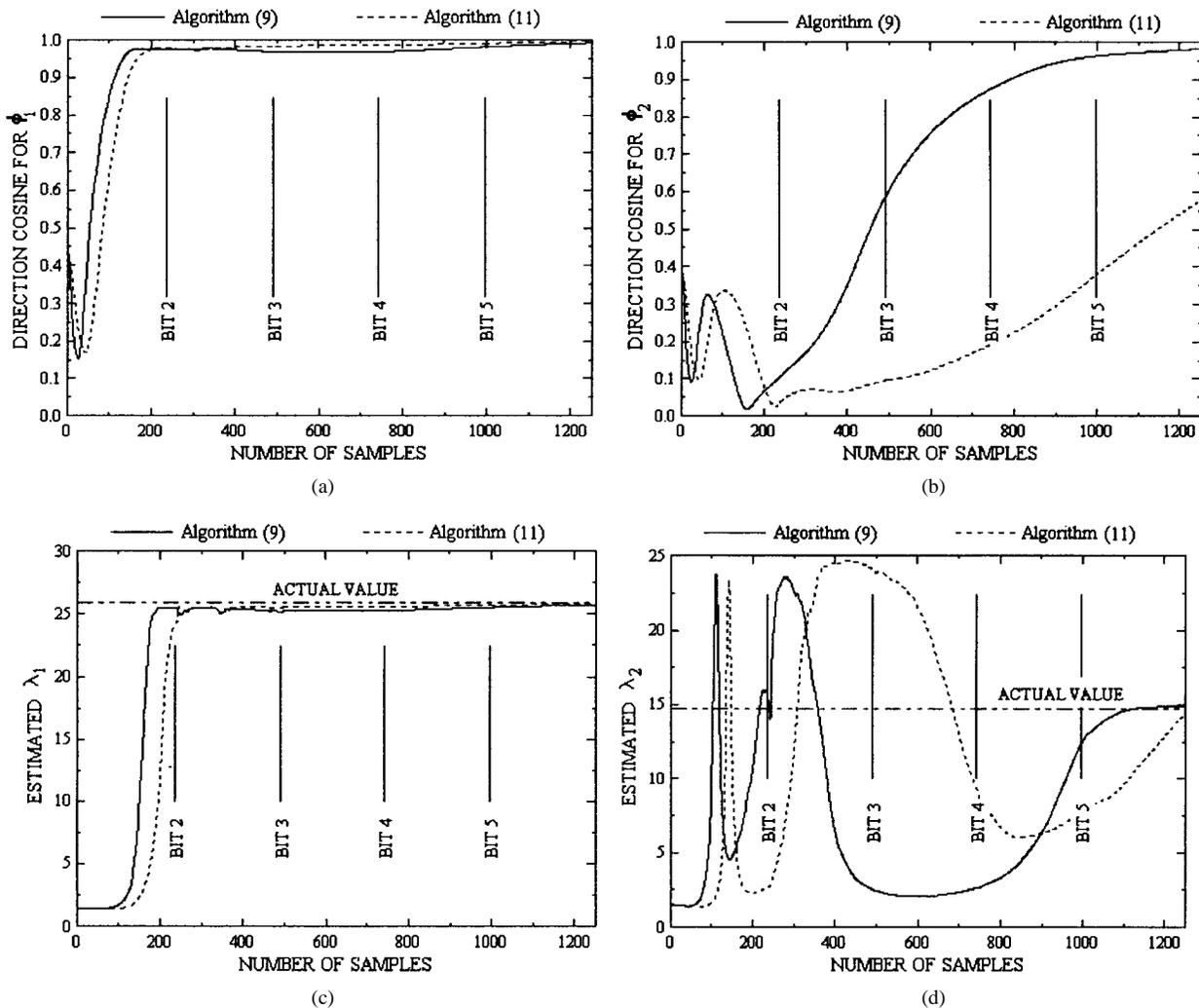


Fig. 7. Relative convergence of the first two principal generalized eigenvectors and eigenvalues for algorithms (9) and (11) with $\eta_k = 1/(800 + k)$ and $\gamma = 1$. (a) ϕ_1 . (b) ϕ_2 . (c) λ_1 . (d) λ_2 .

We also compare computational time for the conventional method with each recursion of the adaptive algorithm (9). The computational time is measured in MATLAB floating point operations (*flops*), where additions, subtractions, multiplications and divisions count one flop

each for real arguments. The MATLAB algorithm required 80 162 flops for generalized eigen-decomposition, and sorting by decreasing eigenvalue. In comparison, one recursion of the adaptive algorithm (9) needed 6722 flops. Thus, when the estimates are required after each

sample, the adaptive method has a definite computational advantage, and for this application, it converges to within 2% of the actual value in just 5 bits of transmitted signal.

V. CONCLUDING REMARKS

We present two adaptive algorithms for generalized eigendecomposition from a sequence of random matrices and derive them from a two-layer linear feedforward network. We provide a convergence proof for the algorithms and demonstrate their effectiveness in an interference cancellation problem with on-line data.

ACKNOWLEDGMENT

The authors acknowledge Prof. M. D. Zoltowski for providing the data set for the CDMA example, which is used to validate our algorithms.

REFERENCES

- [1] K. I. Diamantaras and S.-Y. Kung, "Multilayer neural networks for reduced-rank approximation," *IEEE Trans. Neural Networks*, vol. 5, pp. 684–697, 1994.
- [2] G. H. Golub and C. F. VanLoan, *Matrix Computations*. Baltimore, MD: Johns Hopkins Univ. Press, 1983.
- [3] M. Honig, U. Madhow, and S. Verdu, "Blind adaptive multiuser detection," *IEEE Trans. Inform. Theory*, vol. 41, pp. 944–960, July 1995.
- [4] L. Ljung, "Analysis of recursive stochastic algorithms," *IEEE Trans. Automat. Contr.*, vol. AC-22, pp. 551–575, Aug. 1977.
- [5] R. Lupas and S. Verdu, "Near-far resistance of multi-user detectors in asynchronous channels," *IEEE Trans. Commun.*, vol. 38, pp. 496–508, Apr. 1990.
- [6] U. Madhow and M. L. Honig, "MMSE interference suppression for direct-sequence spread-spectrum CDMA," *IEEE Trans. Commun.*, vol. 42, pp. 3178–3188, Dec. 1994.
- [7] S. Verdu, "Multiuser detection," in *Advances in Detection and Estimation*. New York: JAI, 1993.
- [8] M. D. Zoltowski and J. Ramos, "Blind adaptive beamforming for CDMA based PCS/cellular," in *Proc. 29th Asilomar IEEE Conf. Signals, Syst., Comput.*, 1995.
- [9] M. D. Zoltowski and J. Ramos, "Blind multi-user access interference cancellation for CDMA based PCS/cellular using antenna arrays," in *Proc. ICASSP*, Atlanta, GA, May 1996, pp. 2730–2733.