REFERENCES


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 multiple 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 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 μs in duration. Within each bit period, there is roughly a 10 μs 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 μs 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 maxw [SINR = (wH Aw/ wH Bw)],(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 μs 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(n3) computation, this computation is quite intensive.

In an attempt to simplify this problem, an alternative method [8] constructs a lower dimensional m × m matrix pencil (A, B) for an m-antenna problem. We next compute the first p (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-

1 Superscript H denotes Hermitian transpose matrix operation.
For the $p$ generalized eigenvectors, we have $p$ such outputs.

Let $w_i \in \mathbb{C}^n$ ($i = 1, \ldots, n$) be the weight vectors for the input layer, and let $v_i \in \mathbb{C}^m$ ($i = 1, \ldots, m$) 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 $w_j$ 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 $d_i$ as

$$d = d - \sum_{j=1}^{i-1} \gamma_j v_j w_j^H x$$

where $\gamma_j$ for $j = 1, \ldots, i-1$ 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 $j = 1, \ldots, i-1$. Note that this process is equivalent to the deflation [1] of the desired output. The constrained MSE criterion at the network output is

$$J(w_i, v_i) = E \left[ \left| d_i - \sum_{j=1}^{i-1} \gamma_j v_j w_j^H x - v_i w_i^H x \right|^2 \right]$$

$$+ \mu (w_i^H B w_i - 1)$$

where $\mu$ is a Lagrange multiplier. Applying the gradient descent approach on (3), we obtain the update equation for $w_i$ as

$$w_{i+1} = w_i + \eta \left( Mv_i - Bw_i(w_i^H M v_i) - B \sum_{j=1}^{i-1} \gamma_j w_j w_j^H v_i \right)$$

where $\eta$ is a gain constant. Differentiating (3) with respect to $v_i$ and equating it to zero, we obtain the optimum value of $v_i$ as $M^H w_i$. Substituting this $v_i$ in (4), we obtain

$$w_{i+1} = w_i + \eta \left( Mv_i - Bw_i(w_i^H M v_i) - B \sum_{j=1}^{i-1} \gamma_j w_j w_j^H v_i \right)$$

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.

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 $x \in \mathbb{C}^n$ be an input to the network to be classified into one out of $m$ classes $\omega_1, \ldots, \omega_m$. If $x \in \omega_i$, then the desired output $d = e_i$ ($i$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 $x$ and $d$, which are the inputs and outputs, respectively, of the network as follows.

$$M = E[xd^H] \in \mathbb{C}^{n \times n}, \quad A = MM^H \in \mathbb{C}^{n \times n},$$

and

$$B = E[xd^H] \in \mathbb{C}^{n \times n}.$$  

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 $w_i \in \mathbb{C}^n$ ($i = 1, \ldots, n$) be the weight vectors for the input layer, and let $v_i \in \mathbb{C}^m$ ($i = 1, \ldots, m$) 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 $w_i$ 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 $d_i$, as

$$d = d - \sum_{j=1}^{i-1} \gamma_j v_j w_j^H x$$

where $\gamma_j$ for $j = 1, \ldots, i-1$ 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 $j = 1, \ldots, i-1$. Note that this process is equivalent to the deflation [1] of the desired output. The constrained MSE criterion at the network output is

$$J(w_i, v_i) = E \left[ \left| d_i - \sum_{j=1}^{i-1} \gamma_j v_j w_j^H x - v_i w_i^H x \right|^2 \right]$$

$$+ \mu (w_i^H B w_i - 1)$$

where $\mu$ is a Lagrange multiplier. Applying the gradient descent approach on (3), we obtain the update equation for $w_i$ as

$$w_{i+1} = w_i + \eta \left( Mv_i - Bw_i(w_i^H M v_i) - B \sum_{j=1}^{i-1} \gamma_j w_j w_j^H v_i \right)$$

where $\eta$ is a gain constant. Differentiating (3) with respect to $v_i$, and equating it to zero, we obtain the optimum value of $v_i$ as $M^H w_i$. Substituting this $v_i$ in (4), we obtain

$$w_{i+1} = w_i + \eta \left( Mv_i - Bw_i(w_i^H M v_i) - B \sum_{j=1}^{i-1} \gamma_j w_j w_j^H v_i \right)$$

$$w_{i+1} = w_i + \eta \left( Mv_i - Bw_i(w_i^H M v_i) - B \sum_{j=1}^{i-1} \gamma_j w_j w_j^H v_i \right)$$

$$w_{i+1} = w_i + \eta \left( Mv_i - Bw_i(w_i^H M v_i) - B \sum_{j=1}^{i-1} \gamma_j w_j w_j^H v_i \right)$$
Let $W_k$ be the matrix whose $i$th column is $w_i$. Then, we can write (5) in matrix form as

$$W_{k+1} = W_k + \eta(\mathcal{A}W_k - BW_k)^\text{T} U_i [W_i^T \mathcal{A}W_k]$$  

where $U_i[\cdot]$ sets all elements below the diagonal of its matrix argument to zero and multiplies the $j$th row above the diagonal with $(\geq 1)$, 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 $w_i$ and $v_i$, respectively. It is, therefore, worthwhile to explore the gradient descent procedure on the error function below instead of (3)

$$\hat{J}(W) = E[[\mathcal{A} - M^H WWW^H]x_i]^2].$$  

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).

$$W_{k+1} = W_k + \eta(2\mathcal{A}W_k - BW_k)^\text{T} U_i [W_i^T \mathcal{A}W_k] - \mathcal{A}W_k^\text{T} U_i [W_i^T B W_k].$$  

**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 \(\{A_k\} \subset C_{n \times n}\) and \(\{B_k\} \subset C_{n \times n}\) with $\lim_{k \to \infty} E[A_k] = A$ and $\lim_{k \to \infty} E[B_k] = B$. 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

$$W_{k+1} = W_k + \eta_k (2A_kW_k - B_k W_k)^\text{T} U_i [W_i^T A_kW_k] - A_k W_k^\text{T} U_i [W_i^T B_k W_k].$$  

Here, $W_k \in C_{n \times n}$, 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 matrices $\{\mathcal{A}\}$ and $\{\mathcal{B}\}$, respectively, whereby $\lim_{k \to \infty} E[\mathcal{A}] = A$ and $\lim_{k \to \infty} E[\mathcal{B}] = B$. The following algorithm gives an example for generating $\{A_k\}$ and $\{B_k\}$.

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

$$B_k = B_{k-1} + \tau_k (y_k y_k^H - B_{k-1})$$  

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

$$W_{k+1} = W_k + \eta_k (A_k W_k - B_k W_k)^\text{T} U_i [W_i^T A_k W_k].$$  

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 $A_k = x_k x_k^H$, where $x_k$ is the input vector. Let $w_i$ be the weight vector and $o_i$ be the output of the $i$th unit; then, from (11)

$$\Delta w_i = o_i (x_i - o_i B w_i - \sum_{j=1}^i \gamma_j o_j B w_j)$$  

where $o_i = w_i^H x_k$.

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)

\[
\frac{d}{dt} W(t) = \lim_{k \to \infty} E[2A_k W(t) - B_k W(t) UT, [W(t)^{it} A_k W(t)]] - A_k W(t) UT, [W(t)^{it} B_k W(t)]
\]

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 \( \eta_k \);

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 \( x_k \) and \( y_k \) is bounded with probability one, and \( \lim_{k \to \infty} E[x_k x_k^T] = A \) and \( \lim_{k \to \infty} E[y_k y_k^T] = B \), where \( A \) and \( B \) are positive definite.

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

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

**Lemma 1:** Let \( A_1 \) and \( A_2 \) hold. Let \( W^* \) be a locally asymptotically stable (in the sense of Liapunov) solution to the ODE

\[
\frac{d}{dt} W(t) = 2A W(t) - B W(t) UT, [W(t)^{it} AW(t)] - A W(t) UT, [W(t)^{it} BW(t)]
\]

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 \( W_k \to W^* \) with probability one as \( k \to \infty \).

We denote \( \lambda_1 > \lambda_2 > \cdots > \lambda_p \geq \cdots \geq \lambda_n \geq 0 \) as the generalized eigenvalues of \( A \) with respect to \( B \) and \( \phi_i \) as the generalized eigenvector corresponding to \( \lambda_i \) such that \( \phi_1, \cdots, \phi_n \) are orthonormal with respect to \( B \). Let \( \Phi = [\phi_1, \cdots, \phi_n] \) and \( \Lambda = \text{diag}(\lambda_1, \cdots, \lambda_n) \) denote the matrix of generalized eigenvectors and eigenvalues of \( A \) with respect to \( B \). Note that if \( \phi_i \) 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 \( d_i \phi_i \cdots d_i \phi_i \), where \( \sum d_i = 1 \) for \( i = 1, \cdots, p \).

**Lemma 2:** For the ODE (12), let A1 and A3 hold. Then, \( W = \Phi D P \) are equilibrium points of (12), where \( D = [D_i[0]^H] \) is an \( n \times p \) matrix with \( D_i \) 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 \( D = [D_i[0]^H] \), \( D_i = \text{diag}(d_i, \cdots, d_i) \), \( |d_i| = 1 \)) are stable equilibrium points of the ODE (12). In addition, \( W = \Phi D P (d_i = 0 \text{ for } i \leq p \text{ or } P \neq I) \) 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 \( D = [D_i[0]^H] \), \( D_i = \text{diag}(d_i, \cdots, d_i) \), \( |d_i| = 1 \) for \( i = 1, \cdots, p \) are asymptotically stable.

**Lemma 5:** Let A1–A3 hold. Then, there exists a uniform upper bound for \( \eta_k \) 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 \( \Phi D \). Then, with probability one, \( \lim_{k \to \infty} W_k = \Phi D \).

**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 \( \mu s \) 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 \( \mu s \), during which we estimate the signal plus interference correlation matrix \( A \). This leaves us 117 \( \mu s \) to estimate the cochannel interference correlation matrix \( B \). We sample the signal and interference at 0.5 \( \mu s \) 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 \( \mu s \). 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 \( x_k \) and 234 interference vectors \( y_k \). 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 \( \{x_k\} \) and \( \{y_k\} \).

**B. Numerical Results**

We first compute the signal and interference correlation matrices \( A \) and \( B \), respectively, by averaging all \( x_k x_k^H \) and \( y_k y_k^H \) 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 \( \{x_k\} \)
We use (9) with $\gamma_j = \gamma$ for $j = 1, \cdots, p$. We first choose different values of $\eta_k$, 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 $\eta_k = 1/(400 + k)$, and $\{\eta_k\}$, 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

$\left[ \frac{\phi^T_k \hat{\phi} \alpha_k^T \phi_k}{\|\phi_k\| \|\hat{\phi}\|} \right]$, where $\phi_k$ is the estimated generalized eigenvector at $k$th update, and $\hat{\phi}$ is the actual generalized eigenvector computed from all collected samples by the conventional (numerical analysis) method.

Fig. 4. Convergence of the first two principal generalized eigenvectors by algorithm (9) with different $\eta_k$ and $\gamma = 1$. (a) $\phi_1$ with $\eta_k = 1/(400 + k)$. (b) $\phi_2$ with $\eta_k = 1/(400 + k)$. (c) $\phi_1$ with $\eta_k = 1/(600 + k)$. (d) $\phi_2$ with $\eta_k = 1/(600 + k)$. (e) $\phi_1$ with $\eta_k = 1/(800 + k)$. (f) $\phi_2$ with $\eta_k = 1/(800 + k)$.
Fig. 5. Convergence of the first two principal generalized eigenvectors by algorithm (9) with different \( \eta \) and \( \gamma = 1 \). (a) \( \gamma_1 \) with \( \eta = 1/(400 + k) \). (b) \( \gamma_2 \) with \( \eta = 1/(400 + k) \). (c) \( \gamma_1 \) with \( \eta = 1/(600 + k) \). (d) \( \gamma_2 \) with \( \eta = 1/(600 + k) \). (e) \( \gamma_1 \) with \( \eta = 1/(800 + k) \). (f) \( \gamma_2 \) with \( \eta = 1/(800 + k) \).

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 $\gamma$ and $\eta_k = 1/(400+k)$. (a) $\phi_2$. (b) $\gamma_2$.

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) $\phi_1$. (b) $\phi_2$. (c) $\gamma_1$. (d) $\gamma_2$.

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 80162 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