A NONLINEAR RECURSIVE LEAST-SQUARES ALGORITHM FOR THE BLIND SEPARATION OF FINITE-ALPHABET SOURCES

Scott C. Douglas[1] *and S.-Y. Kung*[2]

[1]Department of Electrical Engineering, Southern Methodist University, Dallas, Texas 75275 USA

[2]Department of Electrical Engineering, Princeton University, Princeton, New Jersey 08544 USA

ABSTRACT

In this paper, we present an adaptive algorithm that blindly separates mixtures of finite-alphabet sources given knowledge of the source alphabet and distribution. The algorithm is a nonlinear recursive least-squares procedure that employs a simple and numerically-robust square root Householder update. Simulations verify that the algorithm can separate large-scale noisy mixtures of finite-alphabet sources without any knowledge of the number of sources in the mixture.

1. INTRODUCTION

Blind source separation (BSS) is a signal processing task in which one or more desired signal components are extracted from linear mixtures of the signals. The problem can be stated as follows: Given a sequence of p-dimensional real-valued measured vectors $\mathbf{x}(n) = [x_1(n) \; \cdots \; x_p(n)]^T$ as

$$\mathbf{x}(n) \quad = \quad \mathbf{A}\mathbf{s}(n) + \eta(n), \tag{1}$$

where $\mathbf{A}$ is an unknown $(p \times m)$ matrix, $\mathbf{s}(n) = [s_1(n) \; \cdots \; s_m(n)]^T$ contains m independent source signals $s_i(n)$, and $\eta(n)$ contains p jointly-Gaussian noise signals $\eta_i(n)$, find an $(m \times p)$ linear transformation $\mathbf{B}$ such that

$$\mathbf{y}(n) \quad = \quad \mathbf{B}\mathbf{x}(n) \tag{2}$$

contains estimates of the sources in $\mathbf{s}(n)$, possibly with an arbitrary shuffling of their order and with arbitrary nonzero scaling factors. When $\eta(n) = \mathbf{0}$, this solution becomes

$$\mathbf{B}\mathbf{A} \quad = \quad \mathbf{\Phi}\mathbf{D} \tag{3}$$

where $\mathbf{\Phi}$ is an $(m \times m)$ permutation matrix and $\mathbf{D}$ is a diagonal nonsingular scaling matrix. Blind source separation has received much recent research attention, and several practical applications have been identified in diverse fields including medicine, communications, data mining, and acoustics.

All block-based and on-line BSS procedures assume that the signals to be separated have particular statistical or structural properties. The accuracy of these source signal assumptions determines how well a given separation method will work for a given source mixture. In this paper, we shall focus on methods which assume that the samples of each source signal are independently-drawn from a finite alphabet $\mathcal{S} = \{S_1, S_2, \ldots, S_\nu\}$, such that the probability density function (p.d.f.) of any $s_i(n)$ is $p_i(s) = \sum_{i=1}^{\nu} \rho_i \delta(s - S_i)$, where $\rho_i > 0$ and $\sum_i \rho_i = 1$. Such signals appear frequently in digital communication tasks.

Source separation algorithms that exploit the finite-alphabet properties of the sources are described in [1, 2]. These procedures attempt to model the measured sequence $\mathbf{x}(n)$ as

$$\mathbf{x}(n) \quad = \quad \widehat{\mathbf{A}}\widehat{\mathbf{s}}(n), \tag{4}$$

where $\widehat{\mathbf{A}}$ and $\widehat{\mathbf{s}}(n)$ are estimates of $\mathbf{A}$ and $\mathbf{s}(n)$, respectively. The procedures alternate between the estimates $\widehat{\mathbf{A}}$ and $\{\widehat{\mathbf{s}}(n)\}$, where the entries of $\widehat{\mathbf{s}}(n)$ are constrained to values within the alphabet. Both block-based and sequential (snapshot by snapshot) methods are described. These methods have the following drawbacks:

- The complexity of the source enumeration step within the iterative least squares with enumeration (ILSE) algorithm is exponential in the number of source values being estimated. Even in the on-line version of this algorithm [1] where a single snapshot $\widehat{\mathbf{s}}(n)$ is being estimated, one has to enumerate all ν^m possibilities, a prohibitive task even if ν and m are only somewhat large.
- The methods in [1] do not prevent identical sources from being extracted several times within the procedure. Special tricks are required to guarantee that $\widehat{\mathbf{A}}$ remains full-rank.
- The block methods in [1, 2] incur a significant computational burden in updating the matrix $\widehat{\mathbf{A}}$, despite the fact that its value is ultimately discarded.

In this paper, we develop an adaptive BSS procedure that extracts multiple spatially-uncorrelated uncorrelated finite-alphabet sources from noisy mixtures of these sources. The algorithm is closely related to recently-derived recursive maximum likelihood procedures for estimating finite-alphabet sources [3]. The procedure consists of m parallel nonlinear recursive least-squares estimators, in which the uniqueness of the m source estimates are ensured through a coefficient orthogonality condition. Simulations indicate the robust and accurate separation capabilities of the procedure on a large-scale (20 source) noisy mixture without any knowledge of the number of sources in the mixture.

2. BLOCK-BASED SEPARATION OF FINITE-ALPHABET SOURCES

The adaptive method to be derived is closely related to the block-based BSS techniques described in [3]. As such, we review the relevant ideas in this section. Suppose $p = m$, and let

$$\begin{aligned} \mathbf{S} \quad &= \quad [\mathbf{s}(1) \; \cdots \; \mathbf{s}(N)] \; = \; [\underline{\mathbf{s}}_1^T \; \cdots \; \underline{\mathbf{s}}_m^T]^T & (5) \\ \mathbf{X} \quad &= \quad [\mathbf{x}(1) \; \cdots \; \mathbf{x}(N)] & (6) \end{aligned}$$

denote the $(m \times N)$-dimensional source and mixed signal matrices, respectively, where

$$\underline{\mathbf{s}}_i \quad = \quad [s_i(1) \; \cdots \; s_i(N)]. \tag{7}$$

In later sections, we shall assume that each source signal vector $\underline{\mathbf{s}}_i$ contains spatially-independent signals, such that $E\{s_i(n)s_j(n)\} = 0$, although such an assumption is not required for the remainder of this section.

Given the mixture model in (1) assuming no noise ($\eta(n) = \mathbf{0}$), we can write

$$\mathbf{X} \quad = \quad \mathbf{AS}. \tag{8}$$

The separation method in [3] uses the following recursive procedure for estimating a single row of $\mathbf{S}$ from $\mathbf{X}$: For an initial guess $\widehat{\underline{\mathbf{s}}}_0$ whose elements satisfy $\widehat{s}_0(n) \in \{S_1, S_2, \ldots, S_\nu\}$,

$$\widehat{\underline{\mathbf{s}}}_{l+1} \quad = \quad \underline{\mathbf{g}}(\widehat{\underline{\mathbf{s}}}_l \mathbf{X}^T(\mathbf{X}\mathbf{X}^T)^{-1}\mathbf{X}), \quad l = 0, 1, 2, \ldots \tag{9}$$

where the vector nonlinearity $\underline{\mathbf{g}}(\cdot)$ independently maps each argument to the nearest value in the set $\{S_1, S_2, \ldots, S_\nu\}$.

It can be easily shown that

$$\mathbf{X}^T(\mathbf{X}\mathbf{X}^T)^{-1}\mathbf{X} \quad = \quad \mathbf{S}^T(\mathbf{S}\mathbf{S}^T)^{-1}\mathbf{S} \tag{10}$$

such that the algorithm in (9) can be written as

$$\widehat{\underline{\mathbf{s}}}_{l+1} \quad = \quad \underline{\mathbf{g}}(\widehat{\underline{\mathbf{s}}}_l \mathbf{S}^T(\mathbf{S}\mathbf{S}^T)^{-1}\mathbf{S}), \quad l = 0, 1, 2, \ldots \tag{11}$$

This update does not depend on the mixing matrix $\mathbf{A}$. The convergence behavior of the algorithm depends only on the initial source sequence estimate $\widehat{\underline{\mathbf{s}}}_0$ and the source matrix $\mathbf{S}$. This property is analogous to the equivariant property shared by several recently-developed BSS procedures that rely on statistical independence of the sources [4, 5].

Kung *et al* have proven that the iterative procedure in (9) converges to a solution satisfying the fixed-point condition

$$\widehat{\underline{\mathbf{s}}} \quad = \quad \underline{\mathbf{g}}(\widehat{\underline{\mathbf{s}}}). \tag{12}$$

Moreover, this solution corresponds to $\widehat{\underline{\mathbf{s}}} = d\underline{\mathbf{s}}_j$ for some $1 \leq j \leq m$ and a value of d satisfying $|d| = 1$. In other words, the resulting fixed point is a valid BSS solution. For more details on these results, see [3].

3. BSS USING NONLINEAR LEAST-SQUARES ESTIMATION

The technique in (9) calculates an entire block of source signal samples in $\widehat{\underline{\mathbf{s}}}_l$ using a block of N received signal measurements in $\mathbf{X}$. In this section, we consider a slightly different formulation to this task. In this new formulation, the following variations are made:

a) The input signal block size is allowed to grow, such that the $(m \times n)$ data matrix $\mathbf{X}(n)$ is defined as

$$\mathbf{X}(n) \quad = \quad [\mathbf{X}(n-1) \;\; \mathbf{x}(n)]. \tag{13}$$

The source signal estimate vector therefore also grows with the size of the data matrix $\mathbf{X}(n)$.

b) Only a single iteration of the fixed-point update is allowed for each signal sample. Thus, once a source signal sample is estimated, its estimate is not changed via additional updates. The $(1 \times n)$ estimated source signal vector is therefore defined as

$$\widehat{\underline{\mathbf{s}}}(n) \quad = \quad [\widehat{\underline{\mathbf{s}}}(n-1) \;\; \widehat{s}(n)]. \tag{14}$$

c) The vector quantizer nonlinearity $\underline{\mathbf{g}}(\cdot)$ is allowed to be different for each element of $\widehat{\underline{\mathbf{s}}}(n)$ so that its accuracy can be tuned for each extracted source signal sample estimate.

With these variations, the goal of the iterative algorithm can be stated as follows: Given $\mathbf{X}(n)$ and $\widehat{\underline{\mathbf{s}}}(n-1)$, calculate the estimate $\widehat{s}(n)$.

To derive the proposed algorithm, consider the algorithm in (9) where $N = n - 1$, and define the weight vector

$$\mathbf{b}(n-1) = (\mathbf{X}(n-1)\mathbf{X}^T(n-1))^{-1}\mathbf{X}(n-1)\widehat{\underline{\mathbf{s}}}^T(n-1). \tag{15}$$

If $\mathbf{b}(n-1)$ is accurate enough, then (12) holds for $n-1 = N$; *i.e.*

$$\begin{aligned} \widehat{\underline{\mathbf{s}}}(n-1) \quad &= \quad \underline{\mathbf{g}}(\mathbf{b}^T(n-1)\mathbf{X}(n-1)) && (16) \\ &= \quad \mathbf{b}^T(n-1)\mathbf{X}(n-1). && (17) \end{aligned}$$

Therefore, we can use $\mathbf{b}(n-1)$ to produce an accurate estimate of the current source sample as

$$\begin{aligned} \widehat{s}(n) \quad &= \quad g_n(y(n)) && (18) \\ y(n) \quad &= \quad \mathbf{b}^T(n-1)\mathbf{x}(n). && (19) \end{aligned}$$

Here, $g_n(y)$ is a quantizer nonlinearity whose input gain is specifically tuned for the value of $y(n)$. The new weight vector is then computed in accordance to (15) as

$$\mathbf{b}(n) \quad = \quad (\mathbf{X}(n)\mathbf{X}^T(n))^{-1}\mathbf{X}(n)\widehat{\underline{\mathbf{s}}}(n), \tag{20}$$

where $\mathbf{X}(n)$ and $\widehat{\underline{\mathbf{s}}}(n)$ are as defined in (13)–(14). Eqns. (18)–(20) define the iterative single-source estimation procedure.

The structure of the fixed-point recursive estimation procedure in (18)–(20) is classic in algorithmic form. It is a nonlinear recursive least-squares procedure that minimizes

$$\mathcal{J}_n(\mathbf{b}(n)) \quad = \quad \sum_{k=1}^{n} |\widehat{s}(k) - \mathbf{b}^T(n)\mathbf{x}(k)|^2 \tag{21}$$

where $\widehat{s}(k) = g_k(\mathbf{b}^T(k-1)\mathbf{x}(k))$. Thus, it can be efficiently implemented in a sample-by-sample update procedure using standard recursive least-squares methods. The connection with existing least-squares methods also allows a simple modification that improves the convergence behavior of the iterative procedure. Since the procedure is intended to produce more-accurate source signal estimates $\widehat{s}(n)$ as n is increased, it is desirable to discount less-recent errors in favor of more-recent ones. For this reason, we introduce the exponentially-weighted cost function

$$\mathcal{J}_n(\mathbf{b}(n)) \quad = \quad \sum_{k=1}^{n} \lambda^{n-k}|\widehat{s}(k) - \mathbf{b}^T(n)\mathbf{x}(k)|^2, \tag{22}$$

where $0 \ll \lambda < 1$. With this change, the iterative algorithm can be implemented using any number of recursive least-squares methods. The most well-known is the $\mathcal{O}(m^2)$ procedure that iteratively updates the Kalman gain vector

$$\mathbf{k}(n) \quad = \quad \mathbf{R}_{\mathbf{xx}}^{-1}(n)\mathbf{x}(n) \tag{23}$$

where $r_0 > 0$ and

$$\mathbf{R}_{\mathbf{xx}}(n) \quad = \quad r_0\lambda^n\mathbf{I} + \sum_{k=1}^{n} \lambda^{n-k}\mathbf{x}(k)\mathbf{x}^T(k). \tag{24}$$

The corresponding equations are

$$\begin{aligned} y(n) &= \mathbf{b}^T(n-1)\mathbf{x}(n) && (25)\\ e(n) &= g_n(y(n)) - y(n) && (26)\\ \mathbf{k}(n) &= \frac{\mathbf{R}_{\mathbf{xx}}^{-1}(n-1)\mathbf{x}(n)}{\lambda + \mathbf{x}^T(n)\mathbf{R}_{\mathbf{xx}}^{-1}(n-1)\mathbf{x}(n)} && (27)\\ \mathbf{b}(n) &= \mathbf{b}(n-1) + e(n)\mathbf{k}(n) && (28)\\ \mathbf{R}_{\mathbf{xx}}^{-1}(n) &= \frac{1}{\lambda}\left(\mathbf{R}_{\mathbf{xx}}^{-1}(n-1) - \mathbf{k}(n)\mathbf{x}^T(n)\mathbf{R}_{\mathbf{xx}}^{-1}(n-1)\right) && (29)\end{aligned}$$

In this update, the design of the scalar quantizer $g_n(y)$ is critical to the algorithm's success. In particular, the scaling of the input to the quantizer has to be accurate for the system's separation performance to improve with increasing data memory. For these reasons, we suggest the following design. Specify a fixed quantizer $g(y)$ for unit-variance input, so that if $E\{s_i^2(k)\} = 1$,

$$g(s_i(k)) = s_i(k). \quad (30)$$

Then, estimate the power of the output signal as

$$\gamma(n) \;=\; \lambda\gamma(n-1) + (1-\lambda)y^2(n). \quad (31)$$

Finally, set

$$g_n(y(n)) \;=\; g\left(\frac{y(n)}{\sqrt{\gamma(n)}}\right). \quad (32)$$

4. BSS FOR PARALLEL EXTRACTION OF FINITE ALPHABET SOURCES

We now shall consider m BSS systems that blindly extract m sources in parallel. The simplest such implementation would run m identical systems to that above, each with a different weight vector $\mathbf{b}_i(n)$, $1 \le i \le m$. Define the matrix

$$\mathbf{B}(n) \;=\; [\mathbf{b}_1(n) \;\cdots\; \mathbf{b}_m(n)]^T. \quad (33)$$

Then, the coefficient updates for the simplest form of this parallel update is

$$\begin{aligned} \mathbf{y}(n) &= \mathbf{B}(n-1)\mathbf{x}(n) && (34)\\ \mathbf{e}(n) &= \mathbf{g}_n(\mathbf{y}(n)) - \mathbf{y}(n) && (35)\\ \mathbf{B}(n) &= \mathbf{B}(n-1) + \mathbf{e}(n)\mathbf{k}^T(n), && (36)\end{aligned}$$

where $\mathbf{g}_n(\mathbf{y}(n)) = [g_{1,n}(y_1(n)) \;\cdots\; g_{m,n}(y_m(n))]^T$,

$$\begin{aligned} g_{i,n}(y_i(n)) &= g\left(\frac{y_i(n)}{\sqrt{\gamma_i(n)}}\right) && (37)\\ \gamma_i(n) &= \lambda\gamma_i(n-1) + (1-\lambda)y_i^2(n), && (38)\end{aligned}$$

and $\mathbf{k}(n)$ is computed as before. The primary drawback of this procedure is that $\mathbf{B}(n)$ is not guaranteed to be nonsingular; in other words, two or more rows of $\mathbf{B}(n)$ could converge to be co-linear and extract the same source $s_i(n)$ at multiple outputs.

We now present an algorithm modification to avoid such difficulties. The algorithm employs the Householder RLS algorithm described in [6] (see also [7]). This algorithm calculates two $(m \times m)$ matrices $\mathbf{P}(n)$ and $\mathbf{W}(n)$ such that

$$\begin{aligned} \mathbf{P}(n)\mathbf{R}_{\mathbf{xx}}(n)\mathbf{P}^T(n) &= \mathbf{I} && (39)\\ \mathbf{W}(n)\mathbf{P}(n) &= \mathbf{B}(n). && (40)\end{aligned}$$

The coefficient updates for $\mathbf{P}(n)$ and $\mathbf{W}(n)$ can be determined directly from the results in [6] and are listed below.

$$\begin{aligned} \mathbf{v}(n) &= \mathbf{P}(n-1)\mathbf{x}(n) && (41)\\ \mathbf{y}(n) &= \mathbf{W}(n-1)\mathbf{v}(n) && (42)\\ \mathbf{u}(n) &= \mathbf{P}^T(n-1)\mathbf{v}(n) && (43)\\ \zeta(n) &= \frac{1}{\lambda + \|\mathbf{v}(n)\|^2 + \sqrt{\lambda(\lambda + \|\mathbf{v}(n)\|^2)}} && (44)\\ \mathbf{P}(n) &= \frac{1}{\sqrt{\lambda}}\left(\mathbf{P}(n) - \zeta(n)\mathbf{y}(n)\mathbf{u}^T(n)\right) && (45)\\ \mathbf{W}(n) &= \sqrt{\lambda}\mathbf{W}(n-1)\\ &\quad + \left[\frac{\mathbf{g}_n(\mathbf{y}(n))}{\sqrt{\lambda + \|\mathbf{v}(n)\|^2}} - \sqrt{\lambda}\zeta(n)\mathbf{y}(n)\right]\mathbf{v}^T(n) && (46)\end{aligned}$$

The algorithm in (41)–(46) is mathematically-equivalent to that in (34)–(36). Thus, this new algorithm form provides no advantages as given other than the robust numerical behavior that it provides, because it is a square-root RLS algorithm. The key idea is to use the structure of $\mathbf{W}(n)$ to define a new *vector* nonlinearity $\widetilde{\mathbf{g}}_n(\mathbf{y}(n))$ that guarantees the uniqueness of the source signal estimates $\widehat{\mathbf{s}}(n)$ over time. The new nonlinearity uses the following facts. Because of the whitening properties provided by $\mathbf{P}(n)$ as exemplified in (39) and the uncorrelatedness of the elements of $\mathbf{s}(n)$, it can be shown when $\eta(n) = \mathbf{0}$ that $\mathbf{W}(n)$ will extract unique sources in $\mathbf{y}(n)$ if and only if the rows of $\mathbf{W}(n)$ are orthogonal. If this is the case, $\mathbf{B}(n)$ will be full-rank. Therefore, we propose to define an orthogonalized output signal vector $\widetilde{\mathbf{y}}(n)$ as

$$\widetilde{\mathbf{y}}(n) \;=\; \widetilde{\mathbf{W}}(n-1)\mathbf{v}(n), \quad (47)$$

where $\widetilde{\mathbf{W}}(n-1)$ is created by orthogonalizing the rows of $\mathbf{W}(n-1)$ using a Gram-Schmidt procedure. In this procedure,

$$\begin{aligned} \widetilde{\mathbf{w}}_1 &= \mathbf{w}_1 && (48)\\ \widetilde{\mathbf{w}}_i &= \mathbf{w}_i - \sum_{j=1}^{i-1}\widetilde{\mathbf{w}}_j\frac{\widetilde{\mathbf{w}}_j^T\mathbf{w}_i}{\|\widetilde{\mathbf{w}}_i\|^2} && (49)\end{aligned}$$

where we have suppressed the $(n-1)$ time indices for notational simplicity. Then, $\mathbf{g}_n(\mathbf{y}(n))$ is replaced by $\widetilde{\mathbf{g}}_n(\widetilde{\mathbf{y}}(n))$, where

$$\begin{aligned} \widetilde{g}_{i,n}(\widetilde{y}_i(n)) &= g\left(\frac{\widetilde{y}_i(n)}{\sqrt{\widetilde{\gamma}_i(n)}}\right) && (50)\\ \widetilde{\gamma}_i(n) &= \lambda\widetilde{\gamma}_i(n-1) + (1-\lambda)\widetilde{y}_i^2(n) && (51)\end{aligned}$$

All other aspects of the algorithm remain the same.

In this modified algorithm, the orthogonality conditions on $\mathbf{W}(n)$ is used to specify nonlinear source estimates that are orthogonal within the least-squares cost function, although the orthogonality conditions are not imposed on $\mathbf{W}(n)$ directly at each iteration. This choice is motivated by the fact that, in practice, the measured mixtures $\mathbf{x}(n)$ will be noisy, and hence the rows of $\mathbf{W}(n)$ will not be exactly orthogonal. The vector quantizer nonlinearity $\widetilde{\mathbf{g}}_n(\widetilde{\mathbf{y}}(n))$, however, will be accurate enough to adapt the value of $\mathbf{W}(n)$ towards an accurate source separation solution with unique source estimates at each output. Simulations in the next section illustrate this claim.

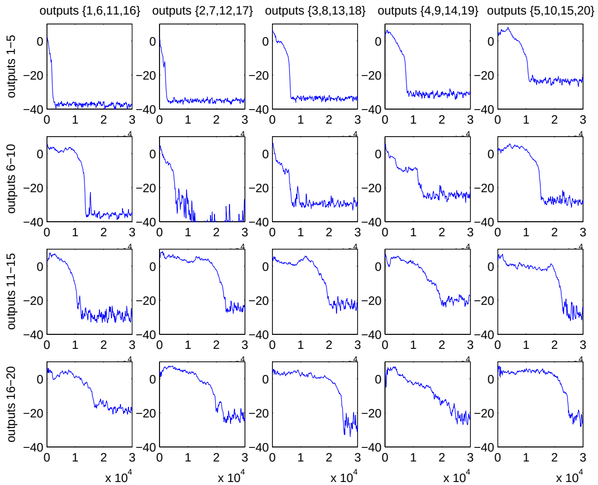

Fig. 1: Normalized inter-channel interferences (ICIs) for the first twenty outputs of the separation system in the simulation example. (For each subplot, the x-axis is the number of iterations, and the y-axis is the normalized ICI in dB.)

5. SIMULATIONS

We now explore the behavior of the proposed algorithm via simulations. In these simulations, each $s_i(k)$ of $\mathbf{s}(k)$ is generated as an independent sequence of unit-variance 4−PAM finite-alphabet symbols, where $\mathcal{S} = \{-3/\sqrt{5}, -1/\sqrt{5}, 1/\sqrt{5}, 3/\sqrt{5}\}$. Twenty sources are mixed using a (30 × 20)-dimensional mixing matrix with random left and right orthonormal singular vector matrices and with logarithmically-spaced singular values given by $\{0.001, 0.002, 0.005, 0.01, 0.02, 0.05, \ldots, 500, 1000, 2000\}$. Uncorrelated Gaussian noise with standard deviation $\sigma_\eta = 0.001$ was added to each mixture to obtain the measured signals in $\mathbf{x}(n)$. These signals were then processed by the proposed algorithm with $\lambda = 0.995$ and $\mathbf{P}(0) = \mathbf{W}(0) = \mathbf{I}$, where the unit-variance quantizer for 4-PAM signals is

$$g(y) \quad = \quad \frac{1}{\sqrt{5}}\text{sgn}(y) \cdot \left[2 + \text{sgn}\left(|y| - \frac{2}{\sqrt{5}}\right)\right], \quad (52)$$

and sgn(y) is the sign function. Note that a 30-by-30 separation system was applied to the measurements; no knowledge of the number of sources in the mixture was assumed.

Fifty simulations were run to verify the performance of the proposed algorithm in this scenario. Shown in Fig. 1 are the convergence trajectories of the measured normalized inter-channel interferences (ICIs) for the first twenty outputs of the system from a typical simulation run, defined as

$$ICI_i(n) \quad = \quad \sum_{j=1}^{m}\left(\frac{c_{ij}^2(n)}{\max\limits_{1\leq k\leq m} c_{ik}^2(n)} - 1\right), \quad (53)$$

where $c_{ij}(n)$ is the (i,j)th element of the combined system matrix $\mathbf{C}(n) = \mathbf{W}(n)\mathbf{P}(n)\mathbf{A}$. As can be seen, the normalized ICI for each source is reduced to about -20 dB or less after about 25000 iterations. Fig. 2 shows the squared elements $\{c_{ij}^2(n)/c_{ref}^2\}$ of every row of the combined system matrix at the end of the simulation run, where $c_{ref} = 1$. Several features of these plots are noteworthy, namely:

- Every source is represented at one and only one system output.

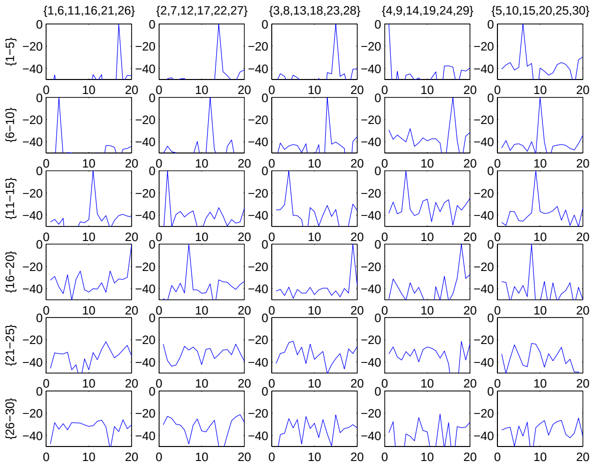

Fig. 2: The thirty rows of the squared elements of the combined system matrix at the end of the simulation example. (For each subplot, the x-axis is the column number j, and the y-axis is $c_{ij}^2(n)/c_{ref}^2$ in dB.)

- The power of each output source is unity.
- The first twenty outputs contain the twenty sources in random permuted order.
- The last ten outputs are *nearly zero*–the residual components are at a -20dB level.
- The residual interference from any unwanted source is at most -20 dB.

In short, the algorithm achieves a practical solution, with m perfectly-scaled sources in the first m outputs and zero outputs otherwise. Out of the 50 runs, 49 attained this result after 30000 iterations; the one remaining simulation run had extracted 19 of the 20 sources after 30000 iterations.

6. CONCLUSIONS

A nonlinear recursive least-squares algorithm for blindly-separating mixtures of spatially-uncorrelated finite-alphabet source signals has been described. The algorithm is simple, numerically-robust, and works without any knowledge of the number of sources in noisy mixtures. Simulations verify its capabilities in a large-scale separation task.

REFERENCES

[1] S. Talwar, M. Viberg, and A. Paulraj, "Blind separation of synchronous co-channel digital signals using an antenna array—Part 1: Algorithms," *IEEE Trans. Signal Processing,* vol. 44, pp. 1184-1197, May 1996.

[2] T. Li and N.D. Sidiropoulos, "Blind digital signal separation using successive interference cancellation iterative least squares," *IEEE Trans. Signal Processing,* vol. 48, pp. 3146-3152, Nov. 2000.

[3] S.-Y. Kung and X. Zhang, "An associative memory approach to blind signal recovery for SIMO/MIMO systems *(invited paper)*," *Proc. IEEE Workshop Neural Networks Signal Processing,* North Falmouth, MA, pp. 343-362, Sept. 2001.

[4] S. Amari, A. Cichocki, and H.H. Yang, "A new learning algorithm for blind signal separation," *Adv. Neural Inform. Proc. Sys. 8* (Cambridge, MA: MIT Press, 1996), pp. 757-763.

[5] J.-F. Cardoso, "Blind signal separation: Statistical principles," *Proc. IEEE,* vol. 86, pp. 2009-2025, Oct. 1998.

[6] S.C. Douglas, "Numerically-robust $\mathcal{O}(N^2)$ RLS algorithms using least-squares prewhitening," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing,* Istanbul, Turkey, vol. I, pp. 412-415, June 2000.

[7] A.A. Rontogiannis and S. Theodoridis, "Inverse factorization adaptive least-squares algorithms," *Signal Processing,* vol. 52, no. 1, pp. 35-47, July 1996.