Blind Digital Signal Separation Using Successive Interference Cancellation Iterative Least Squares

Tao Li and Nicholas D. Sidiropoulos, *Senior Member, IEEE*

***Abstract*—Blind separation of instantaneous linear mixtures of digital signals is a basic problem in communications. When little or nothing can be assumed about the mixing matrix, signal separation may be achieved by exploiting structural properties of the transmitted signals, e.g., finite alphabet or coding constraints. We propose a monotonically convergent and computationally efficient iterative least squares (ILS) blind separation algorithm based on an optimal scaling lemma. The signal estimation step of the proposed algorithm is reminiscent of successive interference cancellation (SIC) ideas. For well-conditioned data and moderate SNR, the proposed SIC-ILS algorithm provides a better performance/complexity tradeoff than competing ILS algorithms. Coupled with blind algebraic digital signal separation methods, SIC-ILS offers a computationally inexpensive true least squares refinement option. We also point out that a widely used ILS finite alphabet blind separation algorithm can exhibit limit cycle behavior.**

***Index Terms*—Array signal processing, decision feedback equalizers, digital communication, iterative methods, least squares methods.**

I. Introduction

CONSIDER the instantaneous multiple-input multiple-output (I-MIMO) observation model $\mathbf{X} = \mathbf{AS} + \mathbf{V}$, where

- $\mathbf{X}$ $m \times N$ data matrix;
- $\mathbf{A}$ $m \times d$ mixing matrix;
- $\mathbf{S}$ $d \times N$ signal matrix;
- $\mathbf{V}$ $m \times N$ matrix of i.i.d. Gaussian random variables;

and it is assumed that $m \geq d$, $N \geq d$, and that $\mathbf{A}$ and $\mathbf{S}$ are full rank (d). This model arises, e.g., in the context of antenna array reception of d narrowband sources impinging on an array of m antennas, wherein the ith row of $\mathbf{S}$ contains the symbol sequence corresponding to the ith source. In digital communications applications, the elements of $\mathbf{S}$ are drawn from a finite alphabet, and they are also usually coded for error protection.

If $\mathbf{A}$ is known or can be estimated via training symbols, then recovering $\mathbf{S}$ from $\mathbf{X}$ can be done in a variety of standard ways, including simple linear solutions like zero-forcing (ZF) or minimum mean squared error (MMSE) equalization, followed by quantization, up to computationally more demanding maximum likelihood (ML) methods. In mobile communications, propagation parameters can be rapidly varying. This necessitates frequent retraining, which wastes bandwidth and is a prime motivation behind the pursuit of so-called *blind* methods. In the absence of noise, the objective of blind source separation is to factor $\mathbf{X}$ into $\mathbf{A}$ and $\mathbf{S}$ by exploiting known properties of either (or both) of $\mathbf{A}$, $\mathbf{S}$. One approach is to constrain $\mathbf{S}$ to satisfy known structural properties, e.g., finite alphabet (FA) or constant modulus (CM) [1], [4], [6]–[11]; let us denote this by $\mathbf{S} \in \boldsymbol{\Phi}$. Given $\mathbf{X}$, d, and $\boldsymbol{\Phi}$, a key issue is whether or not the factors are unique (modulo the inherent permutation and scale ambiguity) in the noiseless case; this has been addressed in [8] for the FA property. In the presence of noise, an *optimal* factorization is sought, e.g., in the least squares (LS) sense

$$\min_{\mathbf{A}, \mathbf{S} \in \boldsymbol{\Phi}} \|\mathbf{X} - \mathbf{AS}\|_F^2 \tag{1}$$

where $\|\cdot\|_F$ is the Frobenius norm. Criterion (1) coincides with conditional maximum likelihood (treating $\mathbf{A}$ and $\mathbf{S}$ as deterministic unknowns).

Even though the I-MIMO model appears to be quite restrictive due to its memoryless nature, more complicated convolutive models can be reduced to it by means of blind equalization techniques [5], [9], [13], in which case one has to rely on residual signal structure (e.g., FA or forward error correcting codes) to resolve the resulting I-MIMO mixture and recover $\mathbf{S}$. This is, in fact, a strong motivation behind the study of blind source separation methods that rely on FA structure alone.

Existing FA factorization algorithms can be classified into noniterative analytical methods [1], [4], [9]–[11] and iterative methods [6]–[8]. Analytical algorithms are typically derived for the noiseless model $\mathbf{X} = \mathbf{AS}$ and subsequently augmented to deal with modest amounts of noise via SVD, clustering, and/or iterative refinement. Iterative algorithms, on the other hand, are typically (but not always) least squares-oriented: Starting from given (possibly random) initial estimates of $\mathbf{A}$ and $\mathbf{S}$, they attempt to solve (1) in an iterative fashion. Analytical methods can serve to provide good initial estimates for subsequent iterative least squares (ILS) refinement in the fully blind case. ILS algorithms are also useful in the semi-blind case, wherein limited training (and hence a coarse estimate of $\mathbf{A}$) is available. Existing iterative methods [6]–[][8] will be reviewed in detail in Section II. Analytical methods are briefly reviewed next.

Factoring $\mathbf{X} = \mathbf{AS}$ subject to FA constraints on $\mathbf{S}$ can be viewed as a clustering problem [1], e.g., for BPSK modulation $\mathbf{X}$ can only contain up to 2^d distinct m-dimensional vectors.

Manuscript received June 23, 1999; revised July 31, 2000. This work was supported by NSF/CAREER Grant CCR-9733540. A preliminary version of part of the material contained in this paper was presented at the IEEE International Conference on Acoustics, Speech, and Signal Processing, Phoenix, AZ, March 15–19, 1999. The associate editor coordinating the review of this paper and approving it for publication was Dr. Vikram Krishnamurthy.

T. Li was with the Department of Electrical Engineering, University of Virginia, Charlottesville, VA 22903 USA. He is now with Philips Semiconductor Co., San Jose, CA USA.

N. D. Sidiropoulos is with the Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN 55455 USA (e-mail: nikos@ece.umn.edu).

Publisher Item Identifier S 1053-587X(00)09293-X.

1053–0587X/00$10.00 © 2000 IEEE

Based on this viewpoint, a two-step algorithm has been proposed in [1]. A drawback is that complexity is exponential in d. A geometric approach has been pursued in [4], viewing the columns of $\mathbf{S}$ as vertices of a hypercube in $\mathbb{R}^d$ and seeking a linear transformation from the hypercube to a parallelotope in $\mathbb{R}^m$ corresponding to the columns of $\mathbf{X}$. The key step of this algorithm is to find a separating hyperplane parallel to one of the hyperplanes that define the received signal vectors. The complexity of this algorithm is $\mathcal{O}(d^3N + Kd^2N)$, where K is a constant. Note that for both [1] and [4] N has to be big enough for $\mathbf{S}$ to contain all 2^d distinct d-tuples as columns.

A different approach was taken in [10] and [11] based on transforming the factorization of $\mathbf{X}$ subject to FA or CM constraints into an appropriate joint diagonalization problem, which can be solved in a variety of ways. The resulting algorithm is widely known as the analytical constant modulus algorithm (ACMA), but an FA version is also available.

The rest of this paper is organized as follows. Existing ILS FA separation algorithms are reviewed in Section II. A key lemma and the ensuing novel ILS FA separation algorithm are presented in Section III, including proof of monotone convergence and complexity analysis. Simulation results are presented in Section IV, and conclusions are drawn in Section V. In the Appendix, it is shown that in contrast to the algorithm proposed herein, a widely used ILS FA separation algorithm known as ILSP can exhibit limit cycle behavior.

II. Iterative Least Squares Algorithms for Digital Signal Separation

The basic idea behind ILS solutions of (1) is simple. Each time, compute an LS update for *one* of the unknown matrices conditioned on a previously obtained estimate for the other matrix, proceed to update the other matrix, and repeat until convergence of the LS cost function is reached. Convergence of the LS cost is guaranteed because each (conditional LS) update may either improve or maintain, but cannot worsen, the fit. The final output is generally dependent on the initialization.

Two ILS FA separation algorithms have been proposed in [8]: iterative least squares with enumeration (ILSE), which is a true ILS algorithm, and iterative least squares with projection (ILSP), which is a much simpler pseudo-ILS algorithm. These are reviewed next.

Algorithm 1: **ILSE** and **ILSP:** Given $\mathbf{A}_0, k = 0$

1) $k = k + 1$
   - either (**ILSE**)

$$\mathbf{S}_k \leftarrow \min_{\mathbf{S}\in\boldsymbol{\Omega}} \|\mathbf{X} - \mathbf{A}_{k-1}\mathbf{S}\|_F^2 \Leftrightarrow$$
$$\Leftrightarrow \sum_{n=1}^{N} \min_{\mathbf{s}(n)\in\boldsymbol{\Omega}} \|\mathbf{x}(n) - \mathbf{A}_{k-1}\mathbf{s}(n)\|_F^2$$

   - or **(ILSP)** $\hat{\mathbf{S}}_k = \mathbf{A}_{k-1}^{\dagger}\mathbf{X}; \mathbf{S}_k = proj_{\boldsymbol{\Omega}}[\hat{\mathbf{S}}_k]$.
   - $\mathbf{A}_k = \mathbf{X}\mathbf{S}_k^{\dagger}$.

2) Go to step 1 until $(\mathbf{A}_k, \mathbf{S}_k) = (\mathbf{A}_{k-1}, \mathbf{S}_{k-1})$

In the above, $\dagger$ stands for pseudo-inverse, and each of the N column-wise minimizations is carried out by enumeration over all possible finite-alphabet d-tuples,[1] and hence, complexity is exponential in d. ILSE is a true ILS algorithm (guaranteed to converge, actually in a finite number of steps in this case [8]) but prohibitively complex even for moderate d. In ILSP, the computationally demanding enumeration step is replaced by a simple finite-alphabet projection of the unconstrained LS update. Unfortunately, this two-step update is not necessarily LS-optimal, and it may actually worsen the fit. This means that ILSP is not guaranteed to converge in general. Although ILSP usually does converge (albeit nonmonotonically) in practice, we show in the Appendix that it can exhibit limit cycle behavior, which is clearly undesirable. ILSP is more prone to spurious minima than ILSE, and it tends to provide measurably worse results. On the other hand, the complexity of ILSP is $\mathcal{O}(Nmd)$ per iteration, whereas ILSE requires $\mathcal{O}(NmdL^d)$ per iteration (recall that both m, N are $\geq d$), where L is the size of the finite alphabet.

A related recent algorithm is decoupled weighted ILSP (DW-ILSP) [6]. The basic idea behind DW-ILSP is as follows. For sufficiently big N, one may accurately estimate the data correlation matrix and subsequently whiten the data. This effectively decouples the blind estimation problem across sources. Each one of the d signals may then be iteratively estimated irrespective of the other $d-1$ signals. This is algorithmically implemented as follows. Fix $r \in \{1, \cdots, d\}$. Let $\mathbf{Z} := \hat{\mathbf{R}}_{\mathbf{X}}^{-(1/2)}\mathbf{X}$, $\mathbf{b}_r := \hat{\mathbf{R}}_{\mathbf{X}}^{-(1/2)}\mathbf{a}_r$, where $\hat{\mathbf{R}}_{\mathbf{X}} := (1/N)\mathbf{X}\mathbf{X}^H$, and ^{H} stands for Hermitian transpose. Let $\mathbf{s}_r^H$ stand for the rth signal (row of $\mathbf{S}$).

Algorithm 2: **DW-ILSP:** Repeat until no change in $\hat{\mathbf{s}}_r^H$

$$\left\{\mathbf{s}_r^H = proj_{\boldsymbol{\Omega}}\left(\frac{1}{\|\mathbf{b}_r\|_2^2}\mathbf{b}_r^H\mathbf{Z}\right); \mathbf{b}_r = \mathbf{Z}\mathbf{s}_r\frac{1}{\|\mathbf{s}_r\|_2^2}\right\}$$

The primary claim behind DW-ILSP is that it attains roughly the same BER versus SNR performance as ILSP at a smaller complexity cost. At first sight, it may appear that DW-ILSP is closely related to the algorithm that will be proposed herein; this is not the case; therefore, it is worthwhile to clarify this issue. Notice that DW-ILSP aims to *decouple* the problem via the whitening transformation and subsequently estimate one signal at a time, irrespective of all other signals. In fact, this makes complete sense for spatially uncorrelated symbol streams and large N. However, the model itself may not be valid for large N, e.g., the implicitly assumed time invariance of $\mathbf{A}$ may be violated due to mobility-induced fast fading and/or variation in the directions of arrival, in which case, increasing N comes at the price of unmodeled dynamics. The approach proposed herein specifically aims toward *coupling* symbol decisions for all d signal streams, letting all users benefit from correct decisions made for any given user. We will show that this leads to an algorithm that can attain performance close to ILSE (thus, *a fortiori* outperforming DW-ILSP) at the complexity cost of ILSP (higher than DW-ILSP).

All ILS algorithms (including ILSP, ILSE, DW-ILSP, and the one to be proposed herein) are suboptimal in the sense that there are no guarantees that the global minimum of (1) will be

[1]We use $\boldsymbol{\Omega}$ to denote the FA restriction on vectors and matrices alike.

reached. The only truly optimal "algorithm" is pure enumeration over all possible $\mathbf{S}$, which is exponential in the product dN and, hence, impractical. Concluding this section, note that a decision-directed MMSE FA separation approach has been pursued in [7]; it requires side information for initialization purposes, and its convergence analysis is asymptotic in N.

III. Proposed Algorithm

The core idea behind the novel algorithm proposed herein can be summarized as follows. Instead of updating $\mathbf{S}$ suboptimally as a whole (like ILSP) or optimally one column at a time (this being very complex, as in ILSE), update one row of $\mathbf{S}$ (user symbol stream) at a time conditioned on $\mathbf{A}$ and the remaining rows of $\mathbf{S}$. As soon as a new row update is made available for any given user, it affects all subsequent row updates for all users. This is reminiscent of successive interference cancellation (SIC) and decision feedback (DF) ideas [12] since it uses previously obtained estimates of other users to "cancel" the multiuser interference and obtain an improved estimate for a user of "current interest." The term successive interference cancellation is more appropriate in our block-oriented context since DF is usually associated primarily with temporal (rather than spatial) decision feedback. Our approach differs from other SIC/DF approaches in that

i) our problem is blind (the mixing is unknown);
ii) the process shifts back and forth between estimating the mixing matrix and updating the estimated user symbol streams until convergence;
iii) the updating is least-squares driven.

Interestingly, it turns out that the optimal update of one row of $\mathbf{S}$ conditioned on all other rows is easy to compute—in fact, it is equivalent to projecting the unconstrained LS row update to the finite alphabet. Contrast this with the ILSE update of one column of $\mathbf{S}$ at a time—which is optimal but requires enumeration over all possible finite-alphabet d-tuples. The optimality of projecting unconstrained LS row updates is a ramification of the following *optimal scaling lemma*, whose proof can be found in Bro and Sidiropoulos [2].

Lemma 1: Let $\mathbf{X}$ be a given $m \times N$ matrix, and let $\mathbf{a} \neq \mathbf{0}$ be a given $m \times 1$ vector. The problem $\min_{\mathbf{s}\in\boldsymbol{\Phi}} \|\mathbf{X} - \mathbf{a}\mathbf{s}^H\|_F^2$ is equivalent to $\min_{\mathbf{s}\in\boldsymbol{\Phi}} \|\mathbf{b} - \mathbf{s}\|_2^2$, where $\mathbf{b}$ stands for the *unconstrained* minimizer of $\|\mathbf{X} - \mathbf{a}\mathbf{s}^H\|_F^2$ with respect to $\mathbf{s}$, i.e., $\mathbf{b} := (1/\|\mathbf{a}\|_2^2)\mathbf{X}^H\mathbf{a}$. The above holds for general $\boldsymbol{\Phi}$ (not necessarily finite-alphabet constraints). Note that depending on $\boldsymbol{\Phi}$, the constrained solution may or may not be unique; we denote $proj_{\boldsymbol{\Phi}}(\mathbf{b}) := \arg\min_{\mathbf{s}\in\boldsymbol{\Phi}} \|\mathbf{b} - \mathbf{s}\|_2^2$ with the understanding that it stands for "*an* argument that minimizes ...." ■

To see how the above lemma applies to the problem at hand, isolate one row of $\mathbf{S}$, say, row r, and denote it by $\mathbf{s}_r^H$. Let $\mathbf{a}_r$ be the corresponding rth column of $\mathbf{A}$, and consider the LS update for $\mathbf{s}_r^H$ conditioned on everything else.

$$\min_{\mathbf{s}_r\in\boldsymbol{\Omega}} \|\mathbf{X} - \mathbf{A}\mathbf{S}\|_F^2 = \min_{\mathbf{s}_r\in\boldsymbol{\Omega}} \|\mathbf{X} - \mathbf{A}^{(r)}\mathbf{S}^{(r)} - \mathbf{a}_r\mathbf{s}_r^H\|_F^2 = \min_{\mathbf{s}_r\in\boldsymbol{\Omega}} \|\tilde{\mathbf{X}}^{(r)} - \mathbf{a}_r\mathbf{s}_r^H\|_F^2$$

where $\mathbf{A}^{(r)}$ is $m \times (d-1)$ consisting of all but the rth column of $\mathbf{A}$, and $\mathbf{S}^{(r)}$ is $(d-1) \times N$ consisting of all but the rth row of $\mathbf{S}$. From Lemma 1, it follows that the LS update for $\mathbf{s}_r$ is given by $proj_{\boldsymbol{\Omega}}((1/\|\mathbf{a}_r\|_2^2)(\tilde{\mathbf{X}}^{(r)})^H\mathbf{a}_r)$. Starting from the first row, we update the rows of $\mathbf{S}$ one by one until all rows have been updated. Following a complete $\mathbf{S}$ update, we update $\mathbf{A}$ using the pseudo-inverse of $\mathbf{S}$. The block diagram in Fig. 1 illustrates the key step utilized to update one row of $\mathbf{S}$. The detailed algorithm can be found in Table I. Notice that the update of a given row depends on all previously obtained updates of all rows. An interesting twist is that different row update orders may give rise to different trajectories in the search space, potentially exhibiting different convergence rates and performance characteristics. Although we do not pursue this thread herein, we would like to point out that one way of resolving this issue is to *rank* possible row updates according to the resulting improvement in fit, pick the one that provides the best improvement, and repeat. This is well motivated from an optimization viewpoint (it results in a stepwise *steepest* descent), and it also makes sense in a near-far situation since row updates corresponding to more powerful users are likely to lead to more significant improvements in fit.

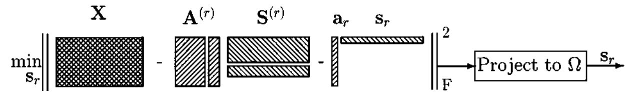


Fig. 1. Key step of SIC-ILS algorithm.

TABLE I
SIC-ILS Algorithm

Algorithm 1 SIC-ILS:

1. $k = 0$; $\mathbf{S}_0$ = *random FA*; $\mathbf{A}_0 = \mathbf{X}\mathbf{S}_0^\dagger$.
2. $k = k + 1$;
3. $\mathbf{S}$-*Update:*
   - $\mathbf{S}_k = \mathbf{S}_{k-1}$;
   - *for* $r = 1$ *to* d,
     - $\mathbf{a}_r$ = r-*th column of* $\mathbf{A}_{k-1}$;
     - $\mathbf{s}_r^H$ = r-*th row of* $\mathbf{S}_k$;
     - $\tilde{\mathbf{X}}^{(r)} = \mathbf{X} - (\mathbf{A}_{k-1}\mathbf{S}_k - \mathbf{a}_r\mathbf{s}_r^H)$;
     - r-*th row of* $\mathbf{S}_k = \left(proj_{\boldsymbol{\Omega}}(\frac{1}{\|\mathbf{a}_r\|_2^2}(\tilde{\mathbf{X}}^{(r)})^H\mathbf{a}_r)\right)^H$;
   - *end*
4. $\mathbf{A}$-*Update:*
   - $\mathbf{A}_k = \mathbf{X}\mathbf{S}_k^\dagger$;
5. *Goto step 2 until convergence.*

A. Complexity

The complexity of SIC-ILS is similar to ILSP. It requires $5Nmd - 2Nm - Nd$ flops per $\mathbf{S}$ update, whereas ILSP needs $Ndm + 2d^2(m - (d/3)) + Nd^2$ flops per $\mathbf{S}$ update, and ILSE requires $NmL^d(d+1)$ flops to enumerate [8]. The SIC-ILS $\mathbf{S}$ update complexity claim may not be obvious from the pseudo-code listing. It requires updating the $\mathbf{A}_{k-1}\mathbf{S}_k$ matrix by subtracting the rank-1 contribution of row r before its update and adding the rank-1 contribution of row r after its update (instead of actually computing the product as listed in the pseudo-code for clarity of exposition). All algorithms require $Ndm + 2d^2(N - (d/3)) + md^2$ flops to update $\mathbf{A}$ [3].

It follows that the complexity per iteration costs of ILSP, ILSE, and SIC-ILS are $\mathcal{O}(Nmd)$, $\mathcal{O}(NmdL^d)$ [9], and $\mathcal{O}(Nmd)$, respectively (note that m, N are $\geq d$). The runtime complexity of all algorithms depends on the actual number of iterations, which depends on the specific dataset and termination criterion. In practice, ten iterations are usually sufficient for all algorithms; hence, per-iteration complexity comparisons are meaningful. At any rate, monotone convergence is important even if one is required to abruptly stop the update process after a fixed number of iterations since monotone convergence assures that the resulting estimates are no worse than all previously obtained estimates.

- ACMA + SIC-ILS: If limited training is available (semi-blind case), then we have seen that SIC-ILS is capable of achieving performance close to ILSE at the complexity cost of ILSP. When no training is available, then algebraic methods offer an attractive way of initializing SIC-ILS, whereas SIC-ILS offers an attractive way to refine algebraic estimates. Algebraic methods bring identifiability to the table, whereas ILS methods offer joint LS (deterministic ML) optimality, provided the algebraic estimates are close enough to the true parameters. The FA version of the ACMA [10], [11] is ideally suited for our purposes. ACMA estimates are often refined using ILSE, but this can be very complex even for moderate d. The idea is to replace ILSE refinement of ACMA estimates with SIC-ILS refinement. Fig. 10 presents a comparison of BER (average for all users) versus number of users (d) performance results for i) binary ACMA followed by two complete iterations ($k = 1, 2$) of ILSE and ii) binary ACMA followed by SIC-ILS (allowed to iterate till convergence). Fig. 11 presents a comparison of the corresponding CPU run times. Here, $m = 10$ (so $\mathbf{A}$ is tall for all d considered), angular separation between the users is set to $13°$, and $E_b/N_o = -3$ dB for all users and all d. From Fig. 10, it is clear that BER is roughly the same, regardless of whether one refines ACMA estimates using ILSE or SIC-ILS, but as shown in Fig. 11, SIC-ILS runtime is significantly lower, especially for higher d, even though it is allowed to iterate till convergence.

B. Convergence

Theorem 1: SIC-ILS is monotonically convergent in a finite number of steps.

Proof: The proof is essentially a consequence of Lemma 1. Each row update may either improve or maintain, but cannot worsen, the fit—thus, convergence of the (bounded, non-negative) cost function is established. Convergence of the cost function and the parameter matrices in a finite number of steps follows because there is only a finite number of distinct possibilities for $\mathbf{S}$ (due to the finite-alphabet constraint), each of which is paired with a unique LS update for $\mathbf{A}$ using the pseudo inverse. In the worst case, the iteration will cycle over all the distinct possibilities once. Note that the same result applies (as shown in [8]) to ILSE but not to ILSP. ■

C. Incorporating Coding Constraints

Notice that Lemma 1 may also be used to incorporate user-wise coding constraints, such as forward error correction coding (FEC) or CDMA spreading into SIC-ILS *without sacri-*

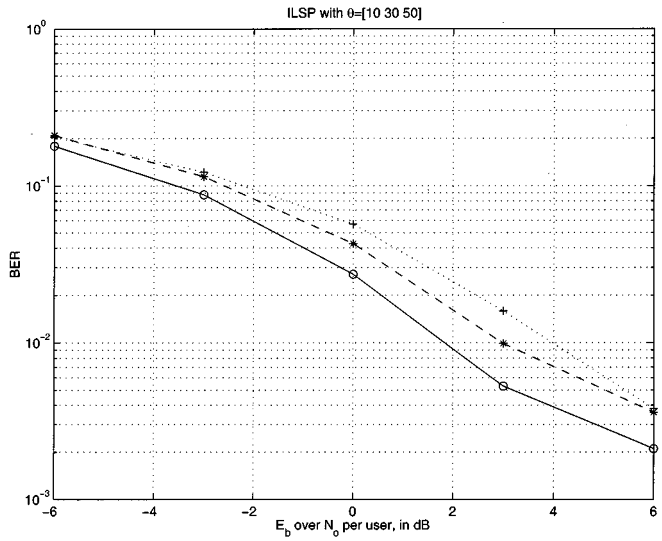


Fig. 2. BER versus E_b/N_o curves for ILSP—blind case.

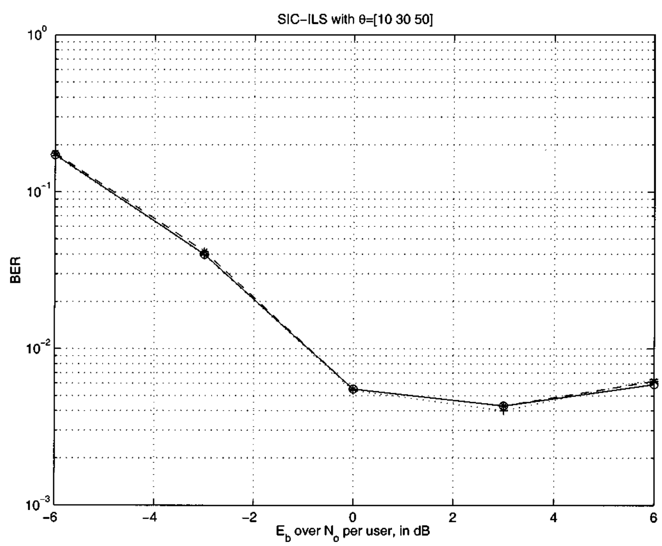


Fig. 3. BER versus E_b/N_o curves for SIC-ILS—blind case.

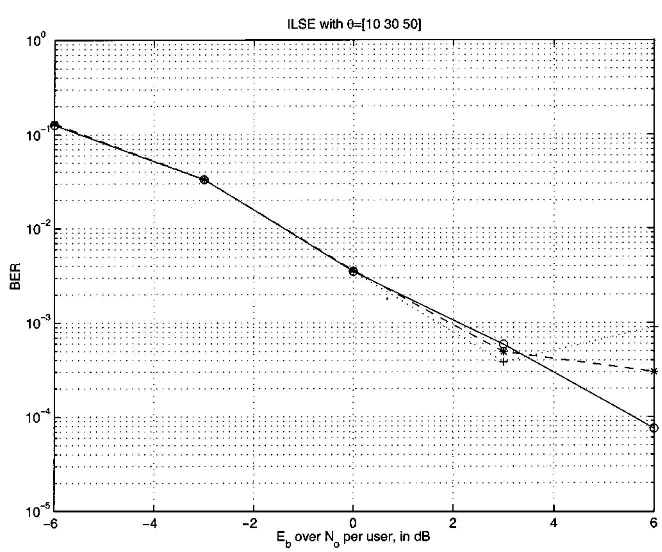


Fig. 4. BER versus E_b/N_o curves for ILSE—blind case.

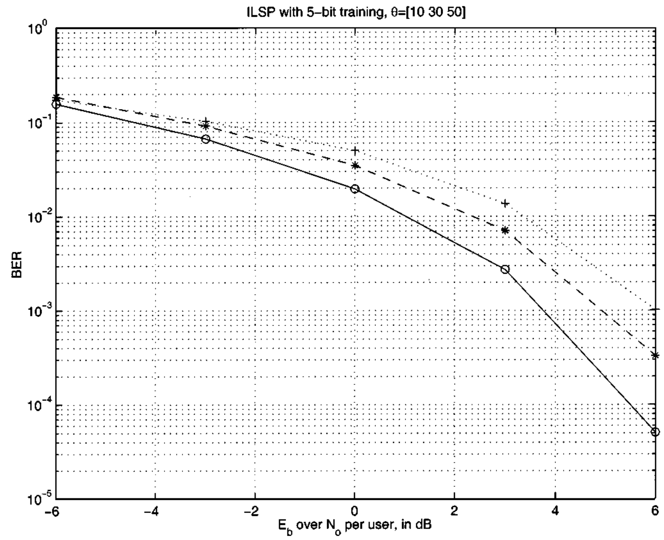


Fig. 5. BER versus E_b/N_o curves for ILSP—semi-blind 5% training.

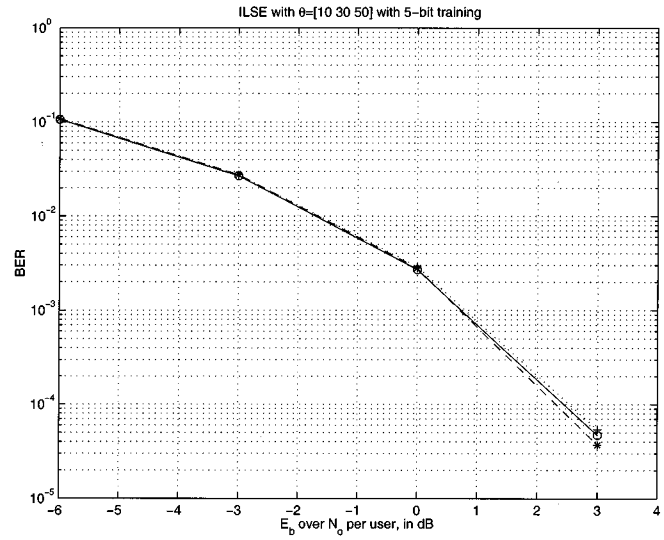


Fig. 7. BER versus E_b/N_o curves for ILSE—semi-blind 5% training.

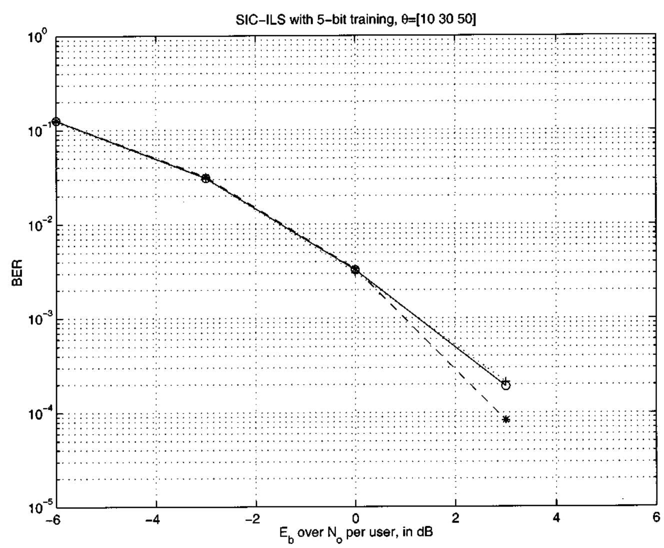


Fig. 6. BER versus E_b/N_o curves for SIC-ILS—semi-blind 5% training.

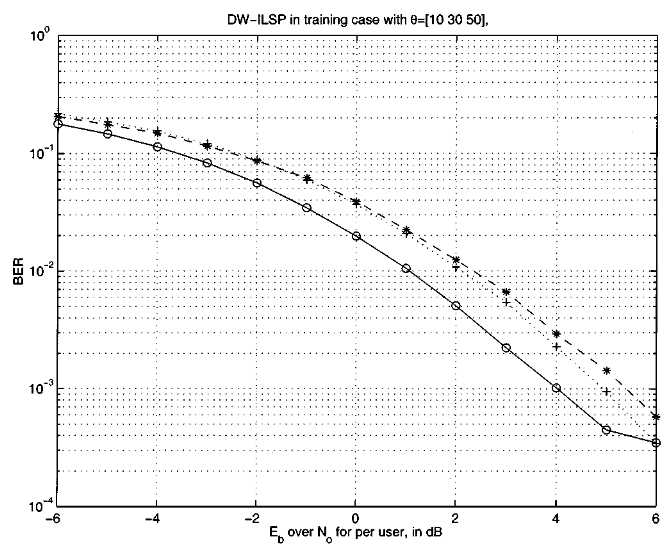


Fig. 8. BER versus E_b/N_o curves for DW-ILSP—semi-blind 5% training.

ficing monotone convergence. This is not possible with ILSP or ILSE. Toward this end, one has to have efficient means of computing $\min_{\mathbf{s}\in\mathcal{C}} \|\mathbf{b}-\mathbf{s}\|_2^2$, where $\mathcal{C}$ is the FEC codebook. In other words, one has to have an efficient algorithm for computing the projection of a hypothetical "received data" sequence onto the codebook. For example, if the signals are convolutionally coded, then the sought algorithm is the well-known "soft" Viterbi decoder for the additive white Gaussian noise channel. Similar algorithms are available for many block codes as well. Note that least squares optimality of the decoding algorithm is crucial for maintaining monotone convergence of the overall blind source separation iteration—suboptimal pseudo-projections onto the codebook will not do. Thus, Lemma 1 allows us to easily take advantage of FEC coding constraints (meant to guard against noise) to remove structured multiple access interference.

IV. Monte-Carlo Performance Results

In all our simulations, $\mathbf{A}$ corresponds to a ULA of $m = 4$ sensors ($\lambda/2$ sensor spacing) receiving $d = 3$ signals arriving from $[10°, 30°, 50°]$ relative to the array broadside. Unless otherwise noted, $N = 100$ snapshots, modulation is ± 1 BPSK, and 10 000 Monte Carlo trials were conducted for each datum reported.

• Blind FA Source Separation: In the fully blind case, a maximum of two random reinitializations per trial were allowed for each algorithm. Figs. 2–4 present BER versus E_b/N_o results for ILSP, SIC-ILS, and ILSE, respectively. Each figure depicts three separate curves: one per user. For low to moderate SNR, SIC-ILS attains the performance of ILSE at the complexity cost of ILSP. In contrast to ILSP, SIC-ILS demodulates all three users at the same BER. A reasonably accurate initialization is important for DW-ILSP,

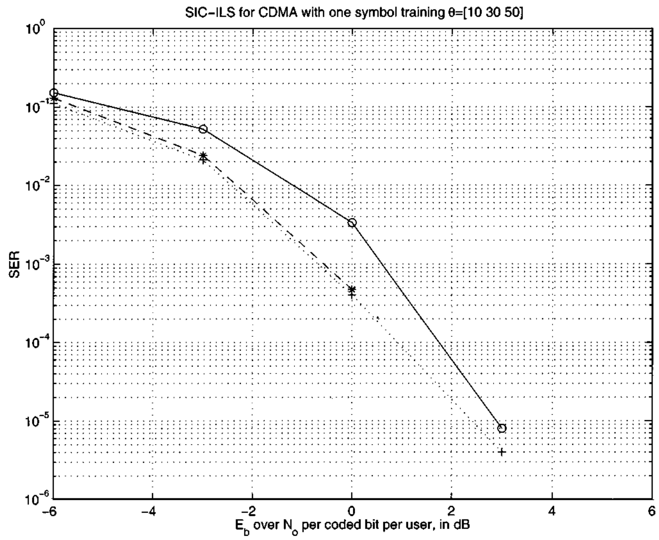


Fig. 9. SER versus E_b/N_o curves for SIC-ILS—semi-blind non-⊥ CDMA 4% training.

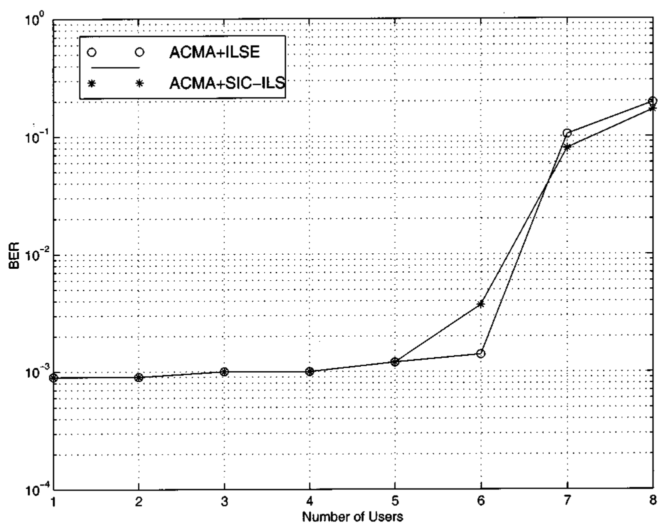


Fig. 10. BER versus number of users for ACMA/ILSE versus ACMA/SIC-ILS.

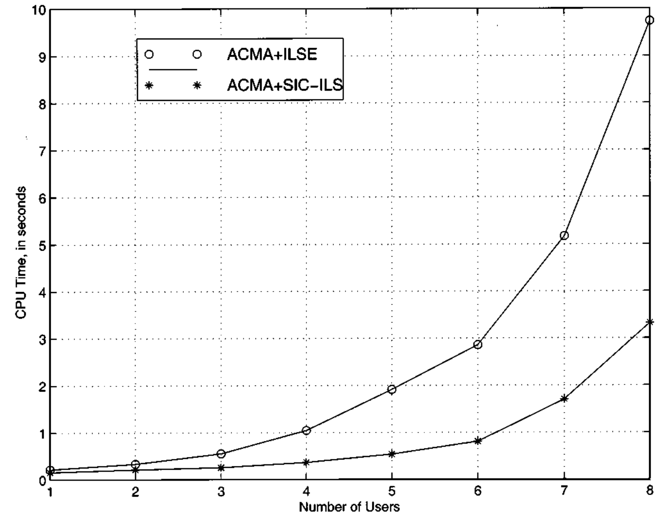


Fig. 11. CPU time versus number of users for ACMA/ILSE versus ACMA/SIC-ILS.

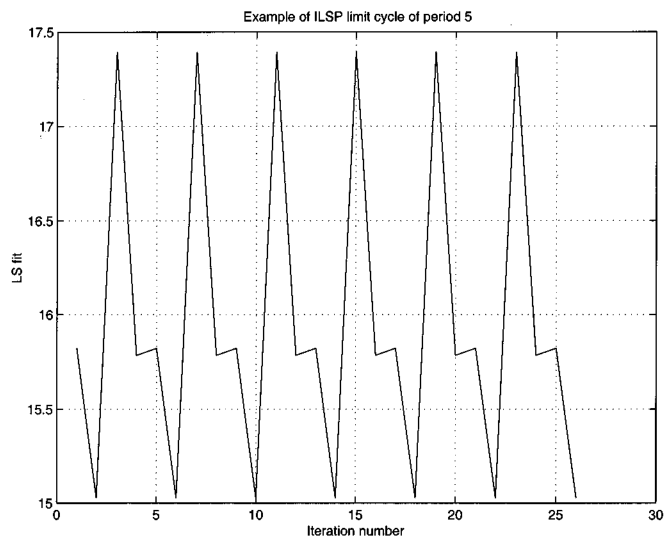


Fig. 12. Example of ILSP limit cycle of period four. Least squares fit versus iteration number.

and in this completely blind case, DW-ILSP gives error rates above 10^{-1} for two of the three users (DW-ILSP result not shown for brevity). Note that fully blind ILS algorithms (including ILSP and ILSE) exhibit a BER flare-up effect at higher SNR due to spurious minima [11].

- Semi-Blind FA Source Separation: Figs. 5–8 present ILSP, SIC-ILS, ILSE, and DW-ILSP performance results in the semi-blind case, utilizing a 5-bit training sequence to obtain an initial estimate of the mixing matrix $\mathbf{A}$. SIC-ILS attains roughly the same performance as ILSE and is one order of magnitude better than ILSP and DW-ILSP in terms of BER.
- Coding and Spreading Constraints: Fig. 9 presents SIC-ILS results (one curve for each user) for CDMA spread signals. Given that spatial diversity is also available, the spreading codes were purposefully chosen to be short (four chips) and nonorthogonal (random, normalized to unit norm), to yield measurable BER for our Monte Carlo simulation. With one symbol (four chips) of training out of a total of 25 symbols (100 chips), SIC-ILS (modified to regress over ± the signature of a given user every four chips) delivers very good symbol error rate (SER) results even at low SNR.

V. Discussion and Conclusions

We have proposed a new algorithm for blind separation of linear mixtures of digital communication signals. The algorithm features moderate complexity, monotone convergence, and performance close to ILSE in all cases considered with the exception of the high SNR fully blind regime. A bonus feature is that the optimal scaling lemma allows easy incorporation of coding constraints.

APPENDIX
ILSP LIMIT CYCLES

One example of a $(\mathbf{X}, \mathbf{A}_0)$ pair that leads to ILSP limit cycle behavior is

$$\mathbf{X}^T = \begin{bmatrix} -0.0043 & -0.4456 & 0.8021 & -0.4141 \\ 0.5070 & 0.6667 & -1.2023 & -0.1669 \\ 1.6372 & 0.3560 & 0.7875 & 0.3559 \\ 0.1186 & -0.5222 & -0.7979 & -0.1650 \\ 2.7683 & 0.5776 & -1.0536 & 0.4642 \\ -0.4317 & 0.9933 & -0.2428 & -0.4142 \\ 0.1822 & 1.9417 & 0.1702 & 0.0814 \\ 0.7631 & 0.6297 & 0.0554 & 0.5359 \end{bmatrix}$$

$$\mathbf{A}_0 = \begin{bmatrix} -0.5666 & -0.1800 \\ 1.1609 & 0.1331 \\ -0.7191 & -0.8503 \\ 0.7401 & 2.0833 \end{bmatrix}.$$

In this case, ILSP actually oscillates between the following two signal matrix estimates:

$$\begin{bmatrix} -1 & 1 & -1 & -1 & 1 & 1 & 1 & 1 \\ -1 & 1 & 1 & 1 & 1 & -1 & 1 & 1 \end{bmatrix}$$

$$\begin{bmatrix} -1 & 1 & -1 & -1 & -1 & 1 & 1 & 1 \\ -1 & 1 & 1 & 1 & 1 & -1 & -1 & 1 \end{bmatrix}.$$

Although high-frequency limit cycles can potentially be detected and ILSP can be randomly reinitialized when this happens, reinitializations are likely to be avoided in practice due to complexity considerations; hence, oscillatory behavior can be problematic. In addition, lower frequency limit cycles are possible. Fig. 12 depicts the evolution of ILSP least squares fit versus iteration number for another choice of $(\mathbf{X}, \mathbf{A}_0)$, leading to a cycle of period four. Low-frequency limit cycles are harder to detect than high-frequency limit cycles.

ACKNOWLEDGMENT

The authors would like to acknowledge insightful suggestions (and a Matlab implementation of ACMA) provided to them by Prof. A.-J. van der Veen. They would also like to thank Dr. A. Ranheim for providing a Matlab implementation of DW-ILSP.

REFERENCES

[1] K. Anand, G. Mathew, and V. U. Reddy, “Blind separation of multiple co-channel BPSK signals arriving at an antenna array,” *IEEE Signal Processing Lett.*, vol. 2, pp. 176–178, Sept. 1995.

[2] R. Bro and N. D. Sidropoulos, “Least squares algorithms under unimodality and nonnegativity constraints,” *J. Chemometr.*, vol. 12, pp. 223–247, 1998.

[3] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 3rd ed. Baltimore, MD: Johns Hopkins Univ. Press, 1996.

[4] L. K. Hansen and G. Xu, “A hyperplane-based algorithm for the digirtal co-channel communications problem,” *IEEE Trans. Inform. Theory*, vol. 43, pp. 1536–1548, Sept. 1997.

[5] H. Liu and G. Xu, “A deterministic approach to blind symbol estimation,” *IEEE Signal Processing Lett.*, vol. 1, pp. 205–207, Dec. 1994.

[6] A. Ranheim, “A decoupled approach to adaptive signal separation using an antenna array,” *IEEE Trans. Veh. Technol.*, vol. 48, pp. 676–682, May 1999.

[7] A. Swindlehurst, S. Daas, and J. Yang, “Analysis of a decision directed beamformer,” *IEEE Trans. Signal Processing*, vol. 43, pp. 2920–2927, Dec. 1995.

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

[9] A.-J. van der Veen, S. Talwar, and A. Paulraj, “Blind estimation of multiple digital signals transmitted over FIR channels,” *IEEE Signal Processing Lett.*, vol. 2, pp. 99–102, May 1995.

[10] A.-J. van der Veen and A. Paulraj, “An analytical constant-modulus algorithm,” *IEEE Trans. Signal Processing*, vol. 44, pp. 1136–1155, May 1996.

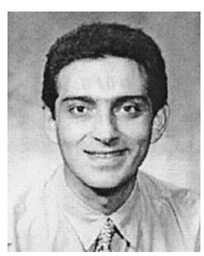
[11] A.-J. van der Veen, “Analytical method for blind binary signal separation,” *IEEE Trans. Signal Processing*, vol. 45, pp. 1078–1082, Apr. 1997.

[12] S. Verdu, *Multiuser Detection*. Cambridge, U.K.: Cambridge Univ. Press, 1998.

[13] D. Yellin and B. Porat, “Blind identification of FIR systems excited by discrete-alphabet inputs,” *IEEE Trans. Signal Processing*, vol. 41, pp. 1331–1339, Mar. 1993.

Tao Li received the B.S. degree in automotive engineering from Tsinghua University, Beijing, China, the M.S. degree in electrical engineering from the State University of New York, Stony Brook, and the Ph.D degree in electrical engineering from the University of Virginia, Charlottesville, in 1993, 1997 and 2000, respectively.

He is currently working for Philips Semiconductor Co., San Jose, CA. His research interests are in signal processing for communications and communication system design.



Nicholas D. Sidiropoulos (M’92–SM’99) received the Diploma in electrical engineering from the Aristotelian University of Thessaloniki, Thessaloniki, Greece, and the M.S. and Ph.D. degrees in electrical engineering from the University of Maryland, College Park (UMCP), in 1988, 1990, and 1992, respectively.

From 1988 to 1992, he was a Fulbright Fellow and a Research Assistant at the Institute for Systems Research (ISR), UMCP. From September 1992 to June 1994, he served his military service as a Lecturer in the Hellenic Air Force Academy. From October 1993 to June 1994, he also was Member of Technical Staff, Systems Integration Division, G-Systems Ltd., Athens, Greece. He held Postdoctoral (1994 to 1995) and Research Scientist (1996 to 1997) positions at ISR-UMCP before joining the Department of Electrical Engineering, University of Virginia, Charlottesville, in July 1997 as an Assistant Professor. He is currently an Associate Professor with the Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis. His current research interests are primarily in multi-way analysis and its applications in signal processing for communications and networking.

Dr. Sidiropoulos is a member of the Signal Processing for Communications Technical Committee (SPCOM-TC) of the IEEE SP Society and currently serves as Associate Editor for IEEE TRANSACTIONS ON SIGNAL PROCESSING and IEEE SIGNAL PROCESSING LETTERS. He received the NSF/CAREER award (Signal Processing Systems Program) in June 1998.