A Simple Geometric Blind Source Separation Method for Bounded Magnitude Sources

Alper T. Erdogan

Abstract—A novel blind source separation approach and the corresponding adaptive algorithm is presented. It is assumed that the observation mixture is obtained through an unknown memoryless linear mapping of independent and bounded magnitude sources. We further assume an initial adaptive prewhitening of the original observation vector which transforms it into a white vector with the same dimension as the original source vector. Our approach is centered around the basic geometric fact that, under a certain boundedness assumption, the unitary mapping which transforms the whitening output vector into an independent vector has the minimum value of maximum (real component) magnitude output over the ensemble of all output components. Therefore, the related criterion is the minimization of the infinity norm of the real component of the unitary separator’s output over all possible output combinations. For the minimization of the corresponding nondifferentiable cost function, we propose the use of subgradient optimization methods to obtain a low complexity iterative adaptive solution. The resulting algorithm is fairly intuitive and simple, and provides a low complexity solution especially to a class of multiuser digital communications problems. We provide examples at the end of this paper to illustrate the performance of our algorithm.

Index Terms—Adaptive filtering, blind source separation, independent component analysis, induced matrix norm, kurtosis, multiple-input multiple-output (MIMO) blind equalization, subgradient.

NOMENCLATURE

\( \mathbf{A} \) (bold capital letters) \quad Matrices.
\( \mathbf{x} \) (bold lowercase letters) \quad Vectors.
\( s \) (normal lowercase letters) \quad Scalars.
\( \mathbf{A}^T \) \quad \( \mathbf{A} \) with conjugated elements.
\( \mathbf{A}^H \) \quad Transpose of \( \mathbf{A} \).
\( \mathbf{A}_{:,n} \) \quad \( n \)th column of \( \mathbf{A} \).
\( \mathbf{A}_{m,:)\) \quad \( m \)th row of \( \mathbf{A} \).
\( \mathcal{R}\mathcal{C} \) \quad Real part operator.
\( \mathcal{I}\mathcal{M} \) \quad Imaginary part operator.
\( \text{vec}(\mathbf{A}) \) \quad Vector generated by stacking the columns of \( \mathbf{A} \).

I. INTRODUCTION

BLIND source separation (BSS) is a subject of interest in many branches of signal processing, with applications ranging from blind multiuser estimation in communications to feature extraction in pattern recognition. It has been an active area of research especially during the last decade, and various methods and approaches have been proposed.

The instantaneous BSS problem can be stated as the recovery of source signals (or components) from their linear combinations, where the linear mapping that creates these mixtures is unknown.

A nice survey of some of the well-known BSS approaches can be found in [1]. These approaches are generally centered around the separation of the statistically independent sources [which is referred to as independent component analysis (ICA)] through the use of the following techniques.

1) Maximum likelihood: The mixture matrix is estimated from the observations through the maximization of the corresponding likelihood function [2], [3].

2) Maximization of non-Gaussianity: The underlying idea is that the distributions of mixtures are closer to a Gaussian distribution than the distributions of the original sources due to the central limit theorem. Therefore, separation can be achieved via maximization of a cost function providing a measure of non-Gaussianity such as kurtosis or negentropy (see [4]–[6] and references therein).

3) Minimization of mutual information: The mutual information among the separator output components is minimized whenever they become mutually independent, and the algorithms under this category exploit this fact [7], [8], [6], [9].

In this paper, we propose a new method for source signals with bounded magnitudes. The assumption about boundedness is realistic in many real-life scenarios. For example, in multiuser digital communications applications, the signals are generated from the constellations which are bounded.

Our approach is motivated by the following two facts.

1) Under certain assumptions on the distribution of the sources, the problem of finding the independent components from the whitened observations is equivalent to the geometrical problem of finding an appropriate rotation and/or reflection which minimizes the maximum magnitude real component of the transformed vector over the ensemble.

2) We can develop efficient subgradient-based algorithms for the minimization of nondifferentiable functions such as...
The $L_\infty$ norm as successfully applied in the single-input single-output (SISO) blind equalization problem [10].

The first fact is a brief geometrical description of the problem of finding the independent components of an uncorrelated vector, where we assume that our initial correlated mixture vector of possibly larger dimension is transformed into an uncorrelated vector by an appropriate whitening transformation. Therefore, our goal is to find a unitary matrix $\Theta$ which is used to transform the uncorrelated vector $\mathbf{x}$ into a vector $\mathbf{z} = \Theta \mathbf{x}$ with independent components. Our claim is that this is achieved with a unitary $\Theta$ matrix for which the magnitude of the maximum magnitude component of $\Re\{\mathbf{z}\}$ over the ensemble is minimized (Theorem 1 in Section III). Therefore, our cost function is

$$J(\Theta) = \sup \|\Re\{\mathbf{z}\}\|_\infty$$

where the supremum is over the ensemble of $\mathbf{z}$. In the adaptive realization, we will make a certain ergodicity assumption in the form of availability of independent (or vanishingly dependent) time samples (or realizations) of $\mathbf{z}$ such that the supremum of $\|\Re\{\mathbf{z}\}\|_\infty$ over these time samples reflects the ensemble behavior.

One apparent property of the proposed cost function, which may be considered as a disadvantage at the first sight, is that it is nondifferentiable. However, the second fact enables the development of low complexity adaptive algorithms, which are based on the subgradient optimization methods, for this cost function. As a matter of fact, in Section IV, we pursue this idea and obtain an adaptive algorithm for the proposed criterion.

The organization of this paper is as follows. Section II provides the notation and the assumed setup for the blind source separation problem. Section III describes the proposed $L_\infty$ norm-based cost function where it is asserted (proven in Appendix I) that the global minimizers of the cost function are the desired unitary matrices that transform the uncorrelated vectors into independent vectors. An adaptive algorithm for the proposed cost function is provided in Section IV. The step-size selection rules, the projection methods to the constraint set (the set of unitary matrices), and the complexity of the proposed algorithm are also discussed in the same section. The simulation examples illustrating the performance of the algorithm are provided in Section V. Section VI is the conclusion.

II. NOTATION AND PRELIMINARIES

A. Blind Source Separation Setup

In the blind source separation setup that we consider throughout this paper, we have the following.

1) $s_1(k), s_2(k), \ldots, s_p(k)$ are the source signals. It is assumed that they are unity variance (without loss of generalization) stationary signals with zero mean and they are mutually independent of each other. It is not required that each $s_i$ is an independent identically distributed (i.i.d.) sequence. We assume that each $s_i$ has sufficient reach variations such that:
   a) the autocorrelation matrix estimate obtained by time averaging reflects the true correlation matrix;
   b) the maximum value of the separator’s output in time reflects the ensemble maximum.

Furthermore, the source signals are considered to be bounded and complex symmetric in the sense that

$$\sup \Re\{s_i\} = \sup \Im\{s_i\} = -\inf \Re\{s_i\} = -\inf \Im\{s_i\} = M$$

over the ensemble of $s_i$. $M$ is the minimum upper bound on the magnitude of real (and imaginary) parts of the sources. We assume the existence of corner points ($\pm M + jM$) in the closure of the domains of the source distributions. The complex sources that satisfy these conditions are quite typical in digital communications applications (e.g., QAM constellations).

In a scenario involving only real sources and mixtures, the assumed symmetry condition can be reduced to

$$\sup \Re\{s_i\} = -\inf \Re\{s_i\} = M$$

which is a valid assumption for a wide range of applications.

2) The source signals are mixed with the memoryless multiple-input multiple-output (MIMO) system with transfer matrix $\mathbf{H}$. $\mathbf{H}$ is a $q \times p$ matrix which implies that this mixing system has $q$ outputs denoted by $y_1(k), y_2(k), \ldots, y_q(k)$, i.e., at any instant $k$, we can write

$$\begin{bmatrix} y_1(k) \\ y_2(k) \\ \vdots \\ y_q(k) \end{bmatrix} = \mathbf{H} \begin{bmatrix} s_1(k) \\ s_2(k) \\ \vdots \\ s_p(k) \end{bmatrix}.$$  \hspace{1cm} (4)

We assume that $q \geq p$, i.e., the number of mixtures is greater than or equal to the number of sources.

3) The purpose of blind source separation is to estimate the source signals (independent components) $s_1(k), s_2(k), \ldots, s_p(k)$ from the observation sequences $y_1(k), y_2(k), \ldots, y_q(k)$ using a linear system with transfer matrix $\mathbf{W}^T$, i.e.,

$$\mathbf{z}(k) = \mathbf{W}^T \mathbf{y}(k)$$  \hspace{1cm} (5)

where $\mathbf{z}(k) = [z_1(k) \ z_2(k) \ \cdots \ z_p(k)]^T$ contains the estimates of the original sources. $\mathbf{W}$ is obtained adaptively from the time samples (realizations) of $y_1(k), y_2(k), \ldots, y_q(k)$. No a priori knowledge of $\mathbf{H}$ and no training sequences are assumed.

We assume that $\mathbf{W}$ is decomposed into two operators

$$\mathbf{W} = \mathbf{W}_{\text{pre}} \Theta$$  \hspace{1cm} (6)

where $\mathbf{W}_{\text{pre}}$ is a $q \times p$ whitening matrix such that

$$\begin{bmatrix} x_1(k) \\ x_2(k) \\ \vdots \\ x_p(k) \end{bmatrix} = \mathbf{W}_{\text{pre}}^T \mathbf{y}(k)$$  \hspace{1cm} (7)

is a white vector, i.e.,

$$E(\mathbf{x}(k)\mathbf{x}^T(k)) = \mathbf{I}.$$  \hspace{1cm} (8)
It is assumed that the whitening matrix \( \mathbf{W}_{\text{pre}} \) is obtained adaptively, possibly through the factorization of the empirical covariance matrix \( \hat{\mathbf{R}}_x \) of \( \mathbf{y}(k) \).

As a result, we can write

\[
\mathbf{z}(k) = \mathbf{W}^T \mathbf{y}(k) = \Theta^T \mathbf{W}_{\text{pre}}^T \mathbf{H} \mathbf{s}(k) = \Theta^T \mathbf{x}(k).
\]

Here \( \mathbf{C} = \mathbf{W}_{\text{pre}}^T \mathbf{H} \) is the combined mixing and whitening transfer functions, and therefore, we can write \( \mathbf{x}(k) = \mathbf{C} \mathbf{s}(k) \).

Note that \( \mathbf{x}(k) \) is a white vector with identity covariance and \( \mathbf{C} \) is a unitary matrix. Although the elements of \( \mathbf{x}(k) \) are uncorrelated, they are not necessarily independent. Therefore, our goal is to obtain a unitary matrix \( \Theta \) as some column permuted version of \( \mathbf{C} \) multiplied by a diagonal matrix with unity magnitude complex entries such that

\[
\mathbf{z}(k) = \Theta^T \mathbf{x}(k) \quad \text{(12)}
\]

\[
= (\Theta \mathbf{E} \mathbf{D})^T \mathbf{C} \mathbf{s}(k) \quad \text{(13)}
\]

\[
= \mathbf{D} \mathbf{E} \mathbf{s}(k) \quad \text{(14)}
\]

where \( \mathbf{E} \) is a permutation matrix and \( \mathbf{D} \) is a diagonal matrix with unit-magnitude complex entries, both of which reflect the unavoidable ambiguities in blind source separation [1].

III. \( \ell_\infty \) NORM-BASED BLIND SOURCE SEPARATION

In the previous section, the blind source separation problem is formulated as finding a unitary \( \Theta \) matrix which converts the whitening output vector \( \mathbf{x}(k) \) to an independent vector \( \mathbf{z}(k) \).

The existing methods to solve this problem generally maximize a cost function of the output of \( \Theta \) under the constraint that \( \Theta \) is a unitary matrix

\[
\text{maximize } J(\mathbf{z}(k)) \quad \text{st. } \Theta^H \Theta = I.
\]

Here the cost function \( J \) is typically chosen as some measure of non-Gaussianity (such as negentropy [1], multiset kurtosis [5], etc.).

In our approach, we introduce the following optimization problem:

\[
\text{minimize } \sup \| \Re \{ \mathbf{z}(k) \} \|_\infty \quad \text{(Problem 1)}
\]

such that \( \Theta^H \Theta = I \)

which is the minimization of the supremum of the infinity norm of the real part of the output vector \( \mathbf{z}(k) \) over the ensemble.

The proposed cost function was inspired by the use of \( \ell_\infty \) norm in the blind equalization problem in [11]. However, we should note that the problem posed in [11] for blind equalization is a convex problem, as the constraint set is also convex. However, the optimization problem proposed for source separation above is not convex, as the constraint set is not a convex set. The following theorem shows that the global minimizers of the above optimization setting are the desired solutions for the source separation problem.

**Theorem 1:** Given the setup in Section II-A and letting \( \mathbf{G} = \Theta^T \mathbf{C} \) be defined as the overall mapping between \( \mathbf{s} \) and \( \mathbf{z} \), then

\[
\Theta_{\text{opt}} \text{ is a global minimizer of Problem 1 if and only if the corresponding } \mathbf{G}_{\text{opt}} \text{ has the form }
\]

\[
\mathbf{G}_{\text{opt}} = \Theta_{\text{opt}}^T \mathbf{C} = \mathbf{DE}
\]

where \( \mathbf{E} \) is a permutation matrix and \( \mathbf{D} \) is a diagonal matrix of the form

\[
\mathbf{D} = \begin{bmatrix}
\mathbf{e}^{j\frac{2\pi}{3}k_1} & 0 & \cdots & 0 \\
0 & \mathbf{e}^{j\frac{2\pi}{3}k_2} & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & \mathbf{e}^{j\frac{2\pi}{3}k_p}
\end{bmatrix}
\]

where \( k_1, k_2, \ldots, k_p \) are integers.

The proof of Theorem 1 is given in Appendix I.

Note that for strictly real sources, under the assumption in (3), it can be easily shown that Theorem 1 is still valid where phase ambiguities in this case would be multiples of \( \pi \).

Based on Theorem 1, we can conclude that the global optimal solutions to the proposed optimization problem (Problem 1) separates the sources perfectly with some unresolvable ambiguities in the permutation and phase factors. The exploitation of the assumed magnitude structure constrains the phase ambiguities to be integer multiples of \( (\pi/2) \) (\( \pi \) in the real case), which simplifies the task of future elimination of these ambiguities.

In order to provide a geometric intuition about the proposed cost function in (15), we provide sample points of two-dimensional real vectors (uncorrelated) \( \mathbf{x} \) and (independent) \( \mathbf{z} \) in Fig. 1(a) and (b), respectively. From the sample points of \( \mathbf{x} \) in Fig. 1(a), it is clear that \( \mathbf{x} \) can be uncorrelated; however, it is not independent, as the rectangular region containing the set of sample points is not equivalent to the Cartesian product of the sample sets of \( x_1 \) and \( x_2 \) components. The independent vector \( \mathbf{z} \), which is obtained through an optimal rotation of \( \mathbf{x} \), has maximum \( \infty \)-norm equal to 1.73 whereas \( \mathbf{x} \) has maximum \( \infty \)-norm equal to 2.41. It is clear from these figures that the \( \infty \)-norm is minimized through a rotation which aligns the borders of the rectangular region with the coordinate axes, which refers to the independent case.
IV. ADAPTIVE ALGORITHM

In the previous section, we introduced the optimization problem (Problem 1) whose global minimizers are the desired separators. In this section, we provide an adaptive algorithm which is based on this criterion.

Before we proceed with the derivation of our algorithm, if we look at the existing stochastic cost function based algorithms (for example, [3]), they generally use a gradient update rule in the form

$$\Theta^{(i+1)} = \mathcal{P}_U\{\Theta^{(i)} - \mu(i) \vec{\nabla} \mathcal{J}_0(\Theta^{(i)})\}$$

(17)

where:

1) $\vec{\nabla} \mathcal{J}_0(\Theta)$ is an empirical estimate of the gradient of the stochastic cost function;

2) $\mathcal{P}_U$ is a projection operator to the set of unitary matrices.

However, this approach is not directly applicable to Problem 1, as the cost function is not differentiable. Fortunately, it is a convex cost function for which we can use subgradient methods. (A discussion on subgradient methods is given in Appendix II.)

In order to obtain a practical algorithm to minimize the cost function in Problem 1, the evaluation of the maximum value of the infinity-norm of $Z(k)$ should be limited to a finite window of time samples of $z$, i.e., $\{z(k) : k \in \{0, 1, \ldots, \Omega - 1\}\}$. Here we make a certain ergodicity assumption that these time samples reflect (or approximate) the ensemble behavior in terms of infinity norm of $Z$. We can rewrite the corresponding optimization problem as

$$\begin{align*}
\text{minimize} & \quad f(\Theta) \quad \text{(Problem 2)} \\
\text{subject to} & \quad \Theta^H \Theta = I
\end{align*}$$

(18)

where

$$f(\Theta) = \max_{k \in \{0, 1, \ldots, \Omega - 1\}} \|\text{Re}[Z(k)]\|_{\infty}.$$  

(19)

If we define

$$X = [x(0) \quad x(1) \quad \ldots \quad x(\Omega - 1)]$$

(20)

as the matrix of input values in the window of interest, then for a given $\Theta$, the corresponding outputs can be placed in a matrix $Z$

$$Z = [Z(0) \quad Z(1) \quad \ldots \quad Z(\Omega - 1)] = \Theta^TX.$$  

(21)

Based on these definitions, the cost function $f$ can be written as

$$
\begin{align*}
f(\Theta) &= \max_{k \in \{0, 1, \ldots, \Omega - 1\}} \|\text{Re}[Z(k)]\|_{\infty} \\
&= \|\text{Re}[\text{vec}(\Theta^TX)]\|_{\infty} \\
&= \|\text{Re}\{[X^T \otimes I]\text{vec}(\Theta^T)\}\|_{\infty}. \\
\end{align*}$$

(22)

(23)

(24)

Therefore, at a point $\Theta$, the subdifferential set (in terms of $\text{vec}(\Theta^T)$) can be written as

$$
\begin{align*}
\partial f(\text{vec}(\Theta^T)) &= \text{Co} \left\{\text{sign}(\text{Re}[Z])_i (X^T \otimes I)_k^H \text{vec}(\text{Re}[Z])_i\right\} \\
&= f(\Theta)
\end{align*}$$

(25)

where Co is the convex hull operation, vec($\text{Re}[Z]$)$_i$ is the $i$th element of vec($\text{Re}[Z]$), and ($X^T \otimes I$)$_{k;i}$ is the $i$th row of ($X^T \otimes I$)$_{k}$. The derivation of the subdifferential set in (25) is given in Appendix II-A.

Based on this subdifferential set, which is in terms of vec($\Theta^T$), we can write a more compact subdifferential set in terms of $\Theta$. For that purpose, we first define the $\mathcal{I}$ as the set of index pairs for which the maximum real magnitude is achieved in matrix $Z$, i.e.,

$$\mathcal{I} = \{(m, n) : \|\text{Re}[Z_{m,n}]\| = f(\Theta)\}.$$  

(26)

Note that each pair $(m, n) \in \mathcal{I}$ corresponds to an index $i$ of vec($\text{Re}[Z]$) in (25). Furthermore, with appropriate indexes $(m, n)$ from $\mathcal{I}$, $\bar{X}_{z,m}e_m^T$ would be the matrix form of $(X^T \otimes I)_{k;i}$ in (25). Therefore, the subdifferential set in matrix notation can be written as

$$\partial f(\Theta) = \text{Co} \left\{\text{sign}(\text{Re}[Z_{m,n}])_{X_{z,m}e_m^T} \right\} \quad (m, n) \in \mathcal{I}.$$  

(27)

where $Z_{m,n}$ is the element of the matrix $Z$ located at the $m$th row and $n$th column (which is $z_m(n)$) and $X_{z,m}$ is the $n$th column of $X$ (which is $x(n)$) with its elements complex conjugated.

Based on the subdifferential set in (27), we can write a possible search direction for $\Theta$ as the negative of the convex combination of subgradients taken from the subdifferential set in (27) (see [10] and references therein)

$$S = -\sum_{(m,n) \in \mathcal{I}} \lambda_{m,n} \text{sign}(\text{Re}[Z_{m,n}])_{X_{z,m}e_m^T}.$$  

(28)

where $\lambda_{m,n}$ are all nonnegative with the property

$$\sum_{(m,n) \in \mathcal{I}} \lambda_{m,n} = 1.$$  

(29)

For convenience, we may choose only one of the $\lambda_{m,n}$s as one and others as zero, in which case the search direction simplifies to

$$S = -\text{sign}(\text{Re}[Z_{m',n'}])_{X_{z,m'}e_m^T}.$$  

(30)

for some $(m', n') \in \mathcal{I}$. As a result, we can write the subgradient-based update rule for our blind source separation algorithm as

$$\Theta^{(i+1)} = \Theta^{(i)} - \mu(i) \text{sign}(\text{Re}[Z_{m',n'}])_{X_{z,m'}e_m^T},$$  

(31)

$$\Theta^{(i+1)} = \mathcal{P}_U\{\Theta^{(i+1)}\}$$  

(32)

where

1. $\Theta^{(i)}$ value of $\Theta$ at the $i$th iteration;
2. $Z^{(i)} = \text{output matrix calculated based on } \Theta^{(i)}$;
3. $(m^{(i)}, n^{(i)})$ index for the maximum real component magnitude entry of $Z^{(i)}$;
4. $\mu(i)$ step size at the $i$th iteration;
5. $\mathcal{P}_U$ projection operator to the unitary matrix set as we defined before.
Inspecting the update rule in (31), we can make the following statements. At each iteration we do the following.

1) Calculate $\Omega$ output vectors.
2) Determine the time point $n^{(i)}$ and the component index $m^{(i)}$ for which the maximum real magnitude is achieved.
3) Update only the corresponding $m^{(i)}$th column of $\Theta$ with the negative of the conjugate of input vector $x(n^{(i)})$ multiplied by the sign of the real part of the output $z^{(i)}_{m^{(i)}}(n^{(i)})$ and the step size $\mu^{(i)}$.
4) Project the resulting matrix to the set of unitary matrices.

A. Step-Size Selection

The choice of the step-size value in (31) is critical, as it affects the convergence speed as well as the accuracy of the final result. Higher step-size values are desirable for faster convergence speeds, whereas for more accurate results, step sizes need to be smaller. The balance between convergence speed and accuracy requirements can be achieved via use of dynamic step sizes with diminishing values. Various dynamic step-size schemes in subgradient optimization and their convergence behaviors have been studied in various references (see, for example, [12]–[16]). Two popular step-size schemes can be listed as follows.

1) $\mu^{(i)} = (\mu_0/(i + 1))$: A dynamic step-size rule that has a clear diminishing behavior with a zero limit. In addition, the cumulative sum of the step size sequence has an unbounded limit. Both of these zero-limit and divergent-sum properties provide guaranteed convergence for convex optimization problems [17], [18]. We should note, however, that in Problem 1, although the cost is a convex function, the constraint set is not convex.

2) $\mu^{(i)} = (f(\Theta^{(i)}) - f^*)/||S^{(i)}||_F^2$ where $f^*$ is the optimal value of the cost function and $||S^{(i)}||_F$ is the Frobenius norm of the search matrix $S^{(i)}$, which is equal to $||x^{(i)}_{m^{(i)}}||_2$ for the update rule in (31). This step-size rule, which is known as the relaxation step-size rule, provides faster convergence than the previous step-size rule for convex optimization problems [19], [12], [13], [20]. However, it requires knowledge of the optimal point $f^*$, which is not known a priori in most problems. Fortunately, in our case, as long as the source distributions are known, $f^*$ is indeed known. This fact follows straight from the result of Theorem 1: since the overall mapping from $s$ to $z$ is given by (15), the optimal value of the cost function is given by

$$f^* = \max |\text{Re}\{s_i\}| = M.$$  

(33)

B. Method for Projection to the Unitary Matrix Set

As we described previously, each subgradient-based update is followed by a projection to the constraint set, which is the set of unitary matrices. Some of the existing blind source separation methods also utilize projection to unitary matrices, and there exists two major approaches [1].

1) Deflation Approach: A Gram–Schmidt procedure is applied to the column vectors of the nonunitary matrix $\Theta^{(i)}$ to obtain a unitary matrix $\Theta^{(i+1)}$. The procedure can be summarized as follows:

Step 1) $\Theta^{(i)}_{n^i} = \frac{\Theta^{(i)}_{n^i}}{||\Theta^{(i)}_{n^i}||_2}$.

Step 2) Set $k = 2$.

Step 3) $\Theta^{(i)}_{n^i k} = \frac{\Theta^{(i)}_{n^i k} - \sum_{l=1}^{k-1} \Theta^{(i)}_{n^i l} \Theta^{(i)}_{n^i k}^H}{||\Theta^{(i)}_{n^i k} - \sum_{l=1}^{k-1} \Theta^{(i)}_{n^i l} \Theta^{(i)}_{n^i k}^H||_2}$.

Step 4) If $k \leq p$ go to Step 3) else Stop.

This scheme does not necessarily choose an orthonormal matrix at a minimum distance to $\Theta^{(0)}$. Therefore, one may consider the alternative projection technique described next.

2) Minimum-Distance Orthogonalization: In order to find the projection operator which projects $\Theta^{(i)}$ to the minimum distance unitary matrix, we use the following theorem (see, for example, [21]).

Theorem 2: Let $\Theta$ be a $C^{p \times p}$ matrix with singular value decomposition (SVD) $\Theta = U \Sigma V^H$ and $Q = \{Q \in C^{p \times p} : Q^H Q = I_p\}$ be the set of unitary matrices. The solution to the optimization problem

$$\min_{Q \in Q} ||\Theta - Q||_F$$

is given by $Q = U \Sigma V^H$. Furthermore

$$\min_{Q \in Q} ||\Theta - Q||_F = \sqrt{\sum_{i=1}^{P} (1 - \sigma_i)^2}$$

(35)

where $\sigma_i$ are the singular values of $\Theta$. Based on this theorem, the minimum-distance projection operator to the set of unitary matrices can be defined as

$$\Theta^{(i+1)} = P_U \{Q^{(i+1)}\} = \{U^{(i+1)} \Sigma^{(i+1)} V^{(i+1)H}\} = U^{(i+1)} V^{(i+1)H}$$

(37)

(38)

where $U^{(i+1)} \Sigma^{(i+1)} V^{(i+1)H}$ is the SVD of $\Theta^{(i+1)}$. These matrices can be conveniently computed using a Gram–Schmidt factorization based algorithm which exploits the fact that $\Theta^{(i+1)}$ is the sum of a unitary matrix and a rank-one matrix. The derivation and the details of this algorithm are provided in Appendix III.

We should note that the minimum distance orthogonalization by (38) is equivalent to the so-called symmetric orthogonalization used in several references (e.g., [1] and [22]). In symmetric orthogonalization

$$\Theta^{(i+1)} = \left(\Theta^{(i+1)} \Theta^{(i+1)H}\right)^{-1/2} \Theta^{(i+1)}.$$

(39)

Here the square root $(\Theta^{(i+1)} \Theta^{(i+1)H})^{-1/2}$ is equal to $U^{(i+1)} D^{(i+1)1/2}$, where $U^{(i+1)} D^{(i+1)} U^{(i+1)H}$ is the eigenvalue decomposition of $\Theta^{(i+1)} \Theta^{(i+1)H}$.
The basic advantage of the proposed algorithm is that it is almost free of any computations as far as the procedure to obtain the search direction from the inputs and outputs is concerned. The computation of the outputs, which is unavoidable, is the source of essential computational burden. Aside from that, there are minor computational requirements, such as determining peak locations via simple comparisons, projection to the set of unitary matrices, and multiplication with step size. With its low computational requirement, the proposed algorithm is less prone to errors due to the fixed-point implementation and therefore more suitable for real-time in-silicon applications. Especially for the embedded applications with a dedicated hardware for computing the outputs, the millions of instruction per second requirement of the algorithm would be very low.

The complexity evaluation of the proposed algorithm can be divided into the following steps.

1) **Covariance Computation:** $q^2 N_c$ complex multiplications and $q^2(N_c - 1)$ complex additions, where $N_c$ is the number of vectors used for the covariance computation.

2) **Updates:**
   a) **Separator output computation:** $Q^2$ complex multiplications $+\Omega(p - 1)$ complex additions.
   b) **Subgradient computation:** $\Omega p$ complex comparisons (additions).
   c) **Subgradient update:** $p$ complex additions, $p$ complex multiplications.
   d) **Projection to unitary set:** $4p^3$ complex multiplications and additions (SVD + Matrix Multiplication).

The update complexity is mostly dominated by the separator output computation, and we can neglect the remaining factors. Therefore, assuming that $\eta$ denotes number of iterations, overall update complexity would be $\eta Q^2$ complex multiplications $+\eta \Omega$ complex additions.

We can compare the complexity of the proposed approach with two of the existing low complexity algorithms.

- **Complex FastICA:** The version of the FastICA algorithm [8] for complex valued signals is introduced in [22]. We assume the use of $G(y) = \log(0.1 + y)$ as the contrast function, as proposed in the same reference. If $\eta_{FT}$ represents the number of complex FastICA iterations, the number of operations required is approximately given by $2\eta_{FT}^2\Omega + 4\eta_{FT}p^3$ complex multiplications, $\eta_{FT}p\Omega$ complex additions, and $2\eta_{FT}\Omega$ real divisions. As illustrated in the Section V, the total number of iterations required for the convergence of the FastICA algorithm is typically less than the proposed algorithm; however, per-iteration complexity is higher.

- **Multiuser Kurtosis (MUK):** The multiuser kurtosis algorithm has been proposed in [5]. Assuming that $\eta_{MUK}$ is the number of MUK iterations, the required number of computations is $\eta_{MUK}(2p^3 + 2p^2)$ complex multiplications and $\eta_{MUK}p^2$ complex additions. Although the per iteration complexity is very low, the total number of iterations is orders of magnitude larger than the proposed algorithm and the FastICA algorithm.

**V. EXAMPLES**

In this section, we will illustrate the proposed algorithm’s performance through some simulation examples. In the first example, we assume the simulation setup used for MUK algorithm in [5], where $H$ is defined as

$$H = \begin{bmatrix} -0.307 + 0.071i & -0.844 + 0.379i \\ -0.616 + 0.691i & 0.1798 - 0.3315i \end{bmatrix}$$

and $s_i$s are the identically distributed sub-Gaussian variables with the probability mass function (pmf)

$$p(s) = \begin{cases} \frac{3}{4}, & s = 0 \\ \frac{1}{36}, & s = 1 + i, 1 - i, -1 + i, -1 - i \end{cases}$$

The output of $H$ is corrupted by additive Gaussian noise and signal-to-noise ratio (SNR) is equal to 30 dB. We simulated the proposed algorithm, together with the complex FastICA [8], [22] (with $\log(0.1 + y)$ as the contrast function) algorithm and MUK algorithm [5]. In these simulations, the number of iterations used for MUK algorithm is set to 25 000 (with a fixed step size of 0.001) where the given window of data is reused several times. In Fig. 2, achieved signal-to-interference [SIR (residual interference from other sources and noise)] power ratio level (for one of the randomly selected separator outputs) as a function of window size (number of samples used) for each algorithm is shown. According to this figure, FastICA algorithm has the best performance where the proposed algorithm has a very close performance.

The typical SIR convergence diagrams and the separator outputs after the convergence for this scenario are shown in Fig. 3 (where we randomly choose one of the sources and display real and imaginary parts of the output), where the FastICA algorithm converges in ten iterations, whereas the proposed algorithm converges in about 15 iterations. We should note that since

---

1SNR is the total ratio of signal power to noise power for the mixture signals.

2This should not be confused with Fig. 1, where we display the values of two different sources on the same graph.
the per-iteration complexity is lower than the FastICA algorithm, the proposed algorithm’s overall computational requirement is lower.

In the second example, we consider a scenario with five sources and five mixtures. Each source is a uniformly distributed 16-QAM digital communication signal. The channel is a random unitary mixing matrix. SNR level is set as 70 dB. In this case, the sources are uncorrelated with each other; however, they are temporally correlated. [A third-order autoregressive infinite impulse response filter with denominator polynomial $0.7z^{-2} - 0.9z^{-1} + 0.9$ with uniformly distributed input having values (0, 1, 2, 3) is used to generate the index of the constellation selector, where the filter output is quantized, to create time-dependent sequences.] According to the SIR versus window-size curves shown in Fig. 4, the proposed algorithm achieves high SIR levels for smaller sample sizes compared to the other algorithms. This behavior can be attributed to the fact that the proposed algorithm is not affected by temporal correlations as long as the data in a given window is sufficiently rich in terms of variations.

The typical convergence curves and the outputs after the convergence (for a long window length) are shown in Fig. 5. Both FastICA and the proposed algorithm’s convergence are affected by the unknown temporal correlation of sources. As in the case of Example 1 (Fig. 3), it is easy to observe that the phase ambiguities for the proposed algorithm are indeed multiples of $(\pi/2)$.

In the third example, we again consider a $5 \times 5$ case. The sources are i.i.d. 64-QAM digital communications signals. The channel is a randomly generated $5 \times 5$ Gaussian i.i.d. matrix. SNR is again selected as 70 dB. At each iteration a new channel is generated, and we averaged over 50 iterations. The channel is first whitened by the whitening transformation obtained from the decomposition of the sample covariance matrix, where the computation of the covariance matrix is restricted to only the assumed number of samples by the algorithms. The resulting SIRs (for one of the separator outputs) as a function of window size are shown in Fig. 6, where it can be seen that the proposed algorithm has better performance than the others.

The typical convergence curves and the outputs after the convergence (for a long window length) are shown in Fig. 7. It is
possible to achieve high SIR levels with about 30 iterations of the proposed algorithm. FastICA algorithm converges in a small number of iterations; however, achieved SIR level is about 7 dB lower.

We should note that, for this example, if the actual available data size is more than what is assumed by the algorithms and if we are allowed to use larger sample sizes for the computation of the sample covariance matrix (this would not bring too much extra computational load, as the major load is due to computation of the unitary separator), then the relative performance of the proposed algorithm becomes much better.

As the final example, we consider a setup involving the speech signal in Fig. 8(a) and the sound signal in Fig. 8(b) as the source signals. We used

\[ \mathbf{H} = \begin{bmatrix} 0.3000 & -0.7000 \\ 0.3000 & 0.4000 \end{bmatrix} \tag{42} \]

as the mixing channel. The resulting mixture signals are shown in Fig. 9.

The subgradient-based blind source separation algorithm is applied to the mixture signals for the time window of samples between 4300 and 4500. In the algorithm, \( \phi^{(i)} = (1/(i+1)) \) is used as the step size. The separator is obtained through 60 iterations and the outputs of the resulting separator matrix are shown in Fig. 10. The combination of the separator and mixing matrices turns out to be equal to

\[ \mathbf{W}^T \mathbf{H} = \begin{bmatrix} 1.0000 & -0.0035 \\ 0.0035 & 1.0000 \end{bmatrix} \tag{43} \]

It is clear from Figs. 8 and 10, and also from (43), that the proposed algorithm successfully separates the original source signals from their mixtures.
VI. CONCLUSION

We introduced a new geometric criterion for instantaneous blind source separation. According to the proposed criterion, the independent sources are recovered from the whitened observations through a unitary transformation which minimizes the supremum of the infinity norm of the (real part of the) output vector over the ensemble. The corresponding adaptive algorithm we proposed makes use of the subgradient search for the minimization of the corresponding nondifferentiable cost function and the minimum distance projection rule for the satisfaction of the unitary norm constraint. The subgradient search boils down to updating of the column of $\Theta$ corresponding to the maximum real-part magnitude output, using the input vector causing this output. Therefore, the resulting adaptive algorithm has a considerably low computational complexity. The simulation examples provided in Section V show that the proposed blind algorithm is successful in recovering the original sources using only a small number of samples and iterations.

Although the proposed algorithm is applicable to only a class of sources satisfying certain boundedness constraint, signals in many applications of interest fit into this class. Probably the most important application involving complex signals within this class is digital communications. Heavily used complex QAM (and real pulse amplitude modulation) constellations have the assumed structure, and the examples in the previous section show that the proposed algorithm offers a desirable performance with a low computational cost for digital communications scenarios fitting to the instantaneous separation setup, such as multiple-antenna multiple-user flat fading channel equalization or the extraction of multiple-user code-division multiple-access signals. Furthermore, there are potentially many applications involving real signals to which the proposed algorithm is applicable. However, we should note that, for the applications where the source distributions have zero tailing property, the proposed algorithm is likely to suffer, which can be considered as one drawback of the algorithm.

APPENDIX I

PROOF OF THEOREM 1

Let $G_{ij}^R$ and $G_{ij}^I$ represent the real and imaginary components of the $(i,j)$th element of the overall mapping $G = \Theta^T C$, respectively. We can write

$$\Re\{z_l(k)\} = \sum_{j=1}^{n} G_{ij}^R x_j^R(k) - \sum_{j=1}^{n} G_{ij}^I x_j^I(k)$$

and therefore, over the ensemble of $z_l(k)$'s, we can write

$$\sup \|\Re\{z_l(k)\}\|_\infty = M \sum_{j=1}^{n} \|G_{ij}^R\|_1 + \|G_{ij}^I\|_1$$

We also define

$$g_l = [G_{11}^R \quad G_{12}^R \quad \cdots \quad G_{1p}^R \quad G_{11}^I \quad G_{12}^I \quad \cdots \quad G_{1p}^I]^T$$

for $l = 1, \ldots, p$. Note that the transpose of $g_l$ is the $l$th row of the matrix

$$[\Re\{G\} \quad \Im\{G\}]$$

and we can write

$$\sup \|\Re\{z_l(k)\}\|_\infty = M \|g_l\|_1$$

As a result, it is easily deduced that

$$\|g_l\|_p = \max_l M \|g_l\|_1$$

or more compactly

$$\sup \|\Re\{z_l(k)\}\|_\infty = M \|\Re\{G\} \quad \Im\{G\}\|_{(\infty,\infty)}$$

where the notation $\|A\|_{(\infty,\infty)}$ is used for the induced matrix norm of matrix $A$, where

$$\|A\|_{(\infty,\infty)} = \sup_{u \neq 0} \| Au \|_\infty / |u|_\infty$$

Since $C$ is unitary and $\Theta$ is constrained to be unitary, $G$ is also unitary. As a result, we can rewrite the equivalent problem in terms of $G$ as

$$\min \|\Re\{G\} \quad \Im\{G\}\|_{(\infty,\infty)}$$

s.t. $G^H G = I$.

Due to the fact that $G$ is unitary, we have

$$\|g_l\|_2 = 1$$

for $l = 1, \ldots, p$. Furthermore, due to the property of p-norms

$$\|g_l\|_1 \geq \|g_l\|_2$$

and the equality is achieved if and only if

$$g_l = \sigma_l e_l$$

where $e_{\eta}$, the standard basis vector which is the $\eta$th column of $I_{2^{\eta} \times 2^{\eta}}$, is either 1 or -1 and $\eta$ is the index corresponding to $l$. It is clear that in the optimal case, $g_l$ for all $l = 1, \ldots, p$ should satisfy the form in (53). Therefore, each row of $G_{\text{opt}}$ that minimizes the cost function has only one nonzero entry with a value from the set $\{1,-1,j,-j\}$. Since $G_{\text{opt}}$ is a unitary matrix, then the rows need to be orthonormal. As a result, $G_{\text{opt}}$ takes the form in (15).

APPENDIX II

SUBGRADIENTS

In this paper, subgradients are used in the optimization of the blind source separation cost function we introduced in Section IV. We provide a short introduction to subgradients below; further details can be found in [10] and the references therein.
Let \( f(x) \) be a convex and possibly nondifferentiable function with domain \( S \). We assume that \( S \) is a convex set and \( \mathcal{P}_S \) is the operator that corresponds to the projection to the set \( S \). The subdifferential of \( f(x) \) at point \( x \) is defined as the set [23, 24]

\[
\partial f(x) = \{ g \mid f(y) \geq f(x) + g^T(y - x) \; \forall y \in S \}.
\] (54)

A vector \( g \) which is a member of \( \partial f(x) \) is called a subgradient of \( f(x) \) at \( x \). Based on the definition provided in (54), a subgradient can be used to construct an affine function which provides a global lower bound to the convex function \( f(x) \). This fact is illustrated in Fig. 11, where two affine functions are shown which are tangent to the function \( f \) at a point \( x \), where \( f \) is not differentiable. It is clear from this figure that both affine functions and their convex combinations provide global lower bounds for the function \( f \). When the function \( f(x) \) is differentiable at point \( x \), then \( \partial f(x) = \{ \nabla f(x) \} \), where \( \nabla f(x) \) is the gradient of function \( f(x) \) at point \( x \), i.e., if a function is differentiable at a point, its subgradient is unique and equivalent to the gradient at that point.

The subgradients can be used to construct algorithms similar to gradient-based ones iterative to minimize convex and possibly nondifferentiable functions [10]. The subgradient search algorithm has the form similar to the gradient descent where the gradient vector is simply replaced by a subgradient vector

\[
\mathbf{w}^{(i+1)} = \mathcal{P}_S \left\{ \mathbf{w}^{(i)} - \mu^{(i)} g^{(i)} \right\}
\] (55)

where \( g^{(i)} \) is a subgradient picked from the subdifferential set \( \partial f(\mathbf{w}^{(i)}) \). Despite the resemblance of the forms of the subgradient and the gradient descent algorithms, in the subgradient iteration it may happen that

\[
f(\mathbf{w}^{(i+1)}) > f(\mathbf{w}^{(i)})
\] (56)

for any \( \mu^{(i)} > 0 \) [23]. However, if the \( \mu^{(i)} \) parameter sequence is properly chosen, then \( \mathbf{w}^{(i)} \) can be made to converge to the optimal point \( \mathbf{w}^* \). In fact, the selection of the step size is a crucial point and has been a research focus for several decades, where various step-size selection schemes have been developed (see, for example, [17]–[19], [13], [12]).

A. Subdifferential Set for the \( l_\infty \) Norm-Based Cost Function

In this section, we provide the details of the derivation of the subdifferential set in (25) for the \( l_\infty \) norm-based cost function in (18), where we can write the cost function as

\[
f(\mathbf{w}) = \| \text{Re}\{\text{vec}(\mathbf{Z})\} \|_\infty = \| \text{Re}\{\text{vec}(\mathbf{Bw})\} \|_\infty
\] (57)

where \( \mathbf{B} = (\mathbf{X}^T \otimes \mathbf{I}) \) and \( \mathbf{w} = \text{vec}(\Theta^T) \).

In order to proceed, we need to use the following two properties of subgradients.

**P1** Given a set of convex functions \( h_i(\mathbf{u}), i = 1 \ldots m \), with the corresponding subdifferential sets \( \partial h_i(\mathbf{u}) \), if

\[
h(\mathbf{u}) = \max_{i=1, m} h_i(\mathbf{u})
\] (58)

then

\[
\partial h(\mathbf{u}) = \text{Co}\{\bigcup_i [\partial h_i(\mathbf{u})] \mid h_i(\mathbf{u}) = h(\mathbf{u})\}.
\] (59)

**P2** Given \( f(\mathbf{u}) = h(\mathbf{A} \mathbf{u} + \mathbf{b}) \), where \( h \) is convex, then

\[
\partial f(\mathbf{u}) = \{ \mathbf{A}^T \mathbf{g} \mid \mathbf{g} \in \partial h(\mathbf{A} \mathbf{u} + \mathbf{b}) \}.
\]

We first note that

\[
f(\mathbf{w}) = \| \text{Re}\{\text{vec}(\mathbf{Bw})\} \|_\infty
\] (60)

\[
= \left\| \left[ \text{Re}\{\mathbf{B}\} - \text{Im}(\mathbf{B}) \right] \begin{bmatrix} \text{Re}(\mathbf{w}) \\ \text{Im}(\mathbf{w}) \end{bmatrix} \right\|_\infty
\] (61)

\[
f(\mathbf{y}) = \| \mathbf{Ay} \|_\infty
\] (62)

where we converted \( f \) to a function of the real vector \( \mathbf{y} \).

If we define \( h(\mathbf{z}) = \| \mathbf{Z}_{\text{sign}(\mathbf{z})} \|_\infty \), then \( f(\mathbf{y}) = h(\mathbf{Ay}) \). By Property P1), the subdifferential set of the function

\[
h(\mathbf{z}) = \| \mathbf{z} \|_\infty = \max_i \{ |z_i| \}
\] (63)

is given by

\[
\partial h(\mathbf{z}) = \text{Co}\{\bigcup_i \{ |z_i| \mathbf{e}_i \} \mid |\mathbf{z}| = \| \mathbf{z} \|_\infty \}.
\] (64)

Therefore, by Property P2), the subdifferential set of \( f(\mathbf{y}) \) can be obtained as

\[
\partial f(\mathbf{y}) = \{ \mathbf{A}^T \mathbf{g} \mid \mathbf{g} \in \partial h(\mathbf{Ay}) \}
\] (65)

\[
= \text{Co}\{\bigcup_i \{ |z_i| \mathbf{A} \mathbf{e}_i \} \mid |\mathbf{z}| = \| \mathbf{z} \|_\infty \}.
\] (66)

In terms of complex vector \( \mathbf{w} \), the subdifferential set can be rewritten as

\[
\partial f(\mathbf{w}) = \text{Co}\{\bigcup_i \{ |z_i| \mathbf{B} \mathbf{e}_i \} \mid |\mathbf{z}| = \| \mathbf{z} \|_\infty \}.
\] (67)

**APPENDIX III**

AN ALGORITHM FOR THE SINGULAR VALUE DECOMPOSITION OF \( \Theta(i+1) \)

A compact expression for the SVD of \( \Theta(i+1) \) can be obtained by exploiting the fact that it is a sum of a unitary matrix and a
rank-one matrix. In fact, based on the update rule in (31), we can conclude that the SVD of $\Theta^{(i+1)}$ will have the form

$$
\Theta^{(i+1)} = \begin{bmatrix}
u_1^{(i+1)} & u_2^{(i+1)} & u_3^{(i+1)} & \ldots & u_p^{(i+1)} \\
\sigma_1^{(i+1)} & 0 & 0 & \ldots & 0 \\
0 & \sigma_2^{(i+1)} & 0 & \ldots & 0 \\
0 & 0 & 1 & \ldots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & 0 & \ldots & 1 
\end{bmatrix}
\begin{bmatrix}
u_1^{(i+1)} \\
\nu_2^{(i+1)} \\
\nu_3^{(i+1)} \\
\vdots \\
\nu_p^{(i+1)} 
\end{bmatrix}
$$

(68)

i.e., only two of the singular values can be different than 1.

We first start by obtaining the right singular vectors which are the eigenvectors of $\Theta^{(i+1)^T} \Theta^{(i+1)}$. Given

$$
\Theta^{(i+1)} = \Theta^{(i)} - \mu^{(i)} \text{sign} \left( \text{Re} \left\{ Z^{(i)}_{m^{(i)}, n^{(i)}} \right\} \right) \tilde{X}_{n^{(i)}} e_{m^{(i)}}^T
$$

from the update (31), we can write

$$
\Theta^{(i+1)^T} \Theta^{(i+1)} = \left( \Theta^{(i)} - \mu^{(i)} \text{sign} \left( \text{Re} \left\{ Z^{(i)}_{m^{(i)}, n^{(i)}} \right\} \right) \tilde{X}_{n^{(i)}} e_{m^{(i)}}^T \right)^T \left( \Theta^{(i)} - \mu^{(i)} \text{sign} \left( \text{Re} \left\{ Z^{(i)}_{m^{(i)}, n^{(i)}} \right\} \right) \tilde{X}_{n^{(i)}} e_{m^{(i)}}^T \right)
$$

from (69), we obtain

$$
\lambda = \mu^{(i)} \text{sign} \left( \text{Re} \left\{ Z^{(i)}_{m^{(i)}, n^{(i)}} \right\} \right) \left| Z^{(i)}_{m^{(i)}, n^{(i)}} \right| + \beta
$$

from this equation, we obtain

$$
\beta^2 + \beta \left( -2i \text{Im} \left\{ Z^{(i)}_{m^{(i)}, n^{(i)}} \right\} \right) + \left| Z^{(i)}_{m^{(i)}, n^{(i)}} \right|^2 = 0.
$$

The roots of the quadratic equation are given by

$$
\lambda_1, 2 = \frac{\mu^{(i)} \text{sign} \left( \text{Re} \left\{ Z^{(i)}_{m^{(i)}, n^{(i)}} \right\} \right) \left| Z^{(i)}_{m^{(i)}, n^{(i)}} \right| - \mu^{(i)} \left| Z^{(i)}_{m^{(i)}, n^{(i)}} \right|^2 + \Delta}{2},
$$

where

$\Delta = \left( 2 \left| \text{Re} \left\{ Z^{(i)}_{m^{(i)}, n^{(i)}} \right\} \right| - \left| Z^{(i)}_{m^{(i)}, n^{(i)}} \right|^2 \right)^2 + 4 \left( \left| Z^{(i)}_{m^{(i)}, n^{(i)}} \right|^2 - \left| Z^{(i)}_{m^{(i)}, n^{(i)}} \right|^2 \right).$

Then the corresponding eigenvalues of $W$ are

$$
\lambda_1, 2 = \frac{\mu^{(i)} \text{sign} \left( \text{Re} \left\{ Z^{(i)}_{m^{(i)}, n^{(i)}} \right\} \right) \left| Z^{(i)}_{m^{(i)}, n^{(i)}} \right| - \mu^{(i)} \left| Z^{(i)}_{m^{(i)}, n^{(i)}} \right|^2 + \Delta}{2}.
$$

As a result, the right singular vectors $v_1^{(i+1)}$ and $v_2^{(i+1)}$ can be written as

$$
v_1^{(i+1)} = \beta_1 e_{m^{(i)}} + Z_{m^{(i)}, n^{(i)}}
$$

and

$$
v_2^{(i+1)} = \beta_2 e_{m^{(i)}} + Z_{m^{(i)}, n^{(i)}}
$$

where

$$
\beta_1 = \frac{2i \text{Im} \left\{ Z^{(i)}_{m^{(i)}, n^{(i)}} \right\} - \text{sign} \left( \text{Re} \left\{ Z^{(i)}_{m^{(i)}, n^{(i)}} \right\} \right) \mu^{(i)} \left| Z^{(i)}_{m^{(i)}, n^{(i)}} \right|^2 + \Delta}{2},
$$

and

$$
\beta_2 = \frac{2i \text{Im} \left\{ Z^{(i)}_{m^{(i)}, n^{(i)}} \right\} - \text{sign} \left( \text{Re} \left\{ Z^{(i)}_{m^{(i)}, n^{(i)}} \right\} \right) \mu^{(i)} \left| Z^{(i)}_{m^{(i)}, n^{(i)}} \right|^2 - \Delta}{2}.
$$
The rest of the right singular vectors \( \{v_3^{(i+1)}, \ldots, v_p^{(i+1)}\} \) can be constructed from \( v_1^{(i+1)}, v_2^{(i+1)} \) and the standard basis vectors through a Gram–Schmidt (GS) process, i.e., we apply GS QR factorization to the matrix as shown in the first equation at the top of the page, where columns 3 to \( p \) of the resulting orthogonal matrix would be the remaining right singular vectors.

A similar process can be followed to obtain \( U^{(i+1)} \), where we perform the eigenvalue decomposition for \( \Theta(i) \Theta(i)^H \) in this case. Skipping some details, we obtain left singular vectors \( u_1^{(i+1)} \) and \( u_2^{(i+1)} \) as

\[
\begin{align*}
\alpha_k^{(i+1)} &= \frac{2i\lambda m \{Z_{m}^{(i),n}^{(i)}\} + \mu^{(i)} \|Z_{m}^{(i),n}^{(i)}\|^2 \text{sign} \{Re\{Z_{m}^{(i),n}^{(i)}\}\} + \sqrt{\Delta}}{2 (1 - \text{sign} \{Re\{Z_{m}^{(i),n}^{(i)}\}\})} \\
\alpha_2^{(i+1)} &= \frac{2i\lambda m \{Z_{m}^{(i),n}^{(i)}\} + \mu^{(i)} \|Z_{m}^{(i),n}^{(i)}\|^2 \text{sign} \{Re\{Z_{m}^{(i),n}^{(i)}\}\} - \sqrt{\Delta}}{2 (1 - \text{sign} \{Re\{Z_{m}^{(i),n}^{(i)}\}\})}
\end{align*}
\]

where \( k = 1, 2 \).

The remaining left singular vectors can be obtained by GS-QR factorization to the matrix

\[
\begin{bmatrix}
v_1^{(i+1)} & v_2^{(i+1)} & e_1 & \cdots & e_{m-1} & e_{m+1} & \cdots & e_p
\end{bmatrix}
\]

Acknowledgment

The author would like to thank the anonymous reviewers for their valuable comments, which contributed to the quality of this paper.

References


Alper T. Erdogan was born in Ankara, Turkey, in 1971. He received the B.S. degree from Middle East Technical University, Ankara, in 1993 and the M.S. and Ph.D. degrees from Stanford University, Stanford, CA, in 1995 and 1999, respectively, all in electrical engineering. He was a Principal Research Engineer with Globespan-Virata Corporation (formerly Excess Bandwidth and Virata Corporations) from September 1999 to November 2001. He has been an Assistant Professor in the Electrical and Electronics Engineering Department, Koc University, Istanbul, Turkey, since January 2002. His research interests include wireless and wireline communications, adaptive signal processing, optimization, system theory and control, and information theory.