Non-Orthogonal Joint Diagonalization in the Least-Squares Sense With Application in Blind Source Separation

Arie Yeredor, *Member, IEEE*

***Abstract*—Approximate joint diagonalization of a set of matrices is an essential tool in many blind source separation (BSS) algorithms. A common measure of the attained diagonalization of the set is the weighted least-squares (WLS) criterion. However, most well-known algorithms are restricted to finding an *orthogonal* diagonalizing matrix, relying on a whitening phase for the nonorthogonal factor. Often, such an approach implies unbalanced weighting, which can result in degraded performance. In this paper, we propose an iterative alternating-directions algorithm for minimizing the WLS criterion with respect to a *general* (not necessarily orthogonal) diagonalizing matrix. Under some mild assumptions, we prove weak convergence in the sense that the norm of parameters update is guaranteed to fall below any arbitrarily small threshold within a finite number of iterations. We distinguish between Hermitian and symmetrical problems. Using BSS simulations results, we demonstrate the improvement in estimating the mixing matrix, resulting from the relaxation of the orthogonality restriction.**

***Index Terms*—Blind source separation (BSS), independent component analysis (ICA), joint diagonalization.**

I. Introduction

CONSIDER a set $\mathcal{A}$ of K matrices $\boldsymbol{A}_1, \boldsymbol{A}_2, \ldots \boldsymbol{A}_K \in \mathbb{C}^{N\times N}$. The approximate joint diagonalization problem seeks a "diagonalizing matrix" $\boldsymbol{B} \in \mathbb{C}^{N\times N}$ and K associated diagonal matrices $\boldsymbol{\Lambda}_1, \boldsymbol{\Lambda}_2, \ldots \boldsymbol{\Lambda}_K \in \mathbb{C}^{N\times N}$ such that

$$C_{LS}(\boldsymbol{B}, \boldsymbol{\Lambda}_1, \boldsymbol{\Lambda}_2, \ldots \boldsymbol{\Lambda}_K) = \sum_{k=1}^{K} w_k ||\boldsymbol{A}_k - \boldsymbol{B}\boldsymbol{\Lambda}_k \boldsymbol{B}^{\dagger}||_F^2 \quad (1)$$

is minimized, where $w_1, w_2, \ldots w_K \in \mathbb{R}_+^*$ are some positive weights and where $||\cdot||_F^2$ denotes the squared Frobenius norm. The dagger † may denote either the transpose or the conjugate transpose, depending on the application; we will elaborate on this issue in the next section.

The problem of approximate joint diagonalization is instrumental in blind source separation (BSS) and independent component analysis (ICA). Usually, the observed data is used for constructing a "target set" $\mathcal{A}$, consisting of finite-sample estimates of some "true set" that admits exact joint diagonalization. Thus, $\boldsymbol{B}$ serves as an estimate of the true diagonalizer of the "true set," which is usually the mixing matrix (up to scaling and permutation of columns). Intriguingly, an abundant variety of such sets of "target matrices" have been proposed by various authors, all seeking joint diagonalization. To name a few, fourth-order joint-cumulants matrices were proposed by Cardoso [1] (JADE) and have recently been generalized by Moreau [2] to matrices of cumulants of any order greater or equal to three (see also [3]–[5]); van der Veen and Paulraj proposed certain algebraically derived matrices for the case of constant-modulus sources [6] (ACMA); Belouchrani *et al.* proposed second-order statistics matrices for separating sources with different spectra [7] (SOBI) (see also [8]); Yeredor proposed second-derivative matrices of the log characteristic function [9]. In the context of convolutive mixtures, cross-spectral matrices (at various frequencies), as well as higher order cross-lagged cumulant matrices, have been proposed [10]–[13].

Many well-known algorithms using approximate joint diagonalization in the BSS context seek an *orthogonal* diagonalizer $\boldsymbol{B}$. Cardoso [1], [14] and Belouchrani *et al.* [7] (see also Wax [15]) used successive Givens rotations; van der Veen and Paulraj [6] used a "super-generalized Schur decomposition." The motivation for using an orthogonal diagonalizer is twofold.

- The orthogonality constraint on $\boldsymbol{B}$ simplifies the problem by commuting the transformations from $\boldsymbol{\Lambda}_k$ to $\boldsymbol{A}_k$ without changing the Frobenius norm.
- The nonorthogonal factor of the general diagonalizer can often be approximated beforehand, using a preprocessing "whitening" phase (e.g., [1], [2], [7]). The "whitening" operation consists of decomposing a selected matrix: possibly (but not necessarily) one of the matrices in $\mathcal{A}$, say $\boldsymbol{A}_1$, into $\boldsymbol{A}_1 = \boldsymbol{W}\boldsymbol{W}^{\dagger}$ and using $\boldsymbol{W}^{-1}$ to create a new set $\tilde{\mathcal{A}} : \tilde{\boldsymbol{A}}_k = \boldsymbol{W}^{-1}\boldsymbol{A}_k\boldsymbol{W}^{\dagger^{-1}}$. When the orthogonal approximate joint diagonalizer $\tilde{\boldsymbol{B}}$ of the new set is found, the nonorthogonal diagonalizer of the original set is given by $\boldsymbol{B} = \boldsymbol{W}\tilde{\boldsymbol{B}}$. (Usually, the selected matrix is the empirical data correlation matrix, which turns into the identity matrix under the transformation by $\boldsymbol{W}^{-1}$, implying spatially decorrelated data, hence, the term "whitening.")

However, such a whitening phase can practically distort the weighted least squares (LS) criterion. It would attain *exact* diagonalization of the selected matrix $\boldsymbol{A}_1$ at the possible cost of poor diagonalization of the others. Such an operation is equivalent to setting to infinity the corresponding weight w_1 in (1). Since the weights reflect the desired "sharing" of misdiagonalization between all the matrices in $\mathcal{A}$, the attained solution may

Manuscript received June 11, 2001; revised March 1, 2002. Parts of this work were presented at the International Conference on Independent Component Analysis, Helsinki, Finland, June 2000. The associate editor coordinating the review of this paper and approving it for publication was Dr. Hamid Krim.

The author is with the Department of Electrical Engineering—Systems, Tel-Aviv University, Tel-Aviv, Israel (e-mail: arie@eng.tau.ac.il).

Publisher Item Identifier S 1053-587X(02)X11991-0.

deviate significantly from the desired solution. In fact, in [16], it is shown that this "hard whitening" operation imposes a limit on the attainable performance in the context of BSS, especially in the presence of additive noise.

Few algorithms that do not require the "hard whitening" have been proposed: In [17], Pham proposed an iterative algorithm that minimizes an information theoretic (non-LS) criterion and requires that all matrices in $\mathcal{A}$ be positive definite. In [18], van der Veen proposed a subspace-fitting algorithm that uses Gauss–Newton-type iterations (this algorithm will be addressed in here later).

In this paper, we propose a convergent iterative algorithm, where the diagonalizer $\boldsymbol{B} \in \mathbb{C}^{N\times N}$ is a general (not necessarily orthogonal) matrix. The matrices in $\mathcal{A}$ are not required to be positive definite.

Note that generally, C_{LS} does not have a unique minimizer since scaling and permutations of columns in $\boldsymbol{B}$ can be absorbed into the $\boldsymbol{\Lambda}_k$'s. To circumvent these ambiguities, arbitrary scaling and ordering of diagonal values may be imposed. However, since these ambiguities are immaterial to the solutions of the related statistical problems, we refrain from explicitly imposing such artificial constraints.

As mentioned earlier, we distinguish between two different cases, depending on whether the dagger † in (1) denotes the transpose or the conjugate transpose. In Sections II and III, we derive the minimization algorithm for the two cases, respectively. In Section IV, we address convergence and computational issues, compared with the Gauss–Newton algorithm [18]. In Section V, we present some BSS simulations results, demonstrating the potential performance gain that results from the elimination of the "hard whitening" phase. Some concluding remarks appear in Section VI.

II. Minimization Algorithm

Before outlining the proposed algorithm, we address the issue of the dagger † in (1). The joint diagonalization problem arises in applications belonging to two distinct families, which we term "Hermitian" problems and "symmetrical" problems.

In "Hermitian" problems, all the target matrices in $\mathcal{A}$ are Hermitian, and the superscript † in (1) is interpreted as "conjugate transpose." In addition, all the diagonal matrices $\boldsymbol{\Lambda}_1, \boldsymbol{\Lambda}_2, \dots \boldsymbol{\Lambda}_K$ are real valued. "Hermitian" problems usually occur when the target matrices are empirical correlation matrices, such as in [7], or cumulants matrices such as in [1], [3].

In "symmetrical" problems, all the target matrices in $\mathcal{A}$ are symmetric but *not* Hermitian (when complex). Consequently, in that case, the superscript † in (1) is interpreted as "transpose." The diagonal matrices $\boldsymbol{\Lambda}_1, \boldsymbol{\Lambda}_2, \dots \boldsymbol{\Lambda}_K$ may be complex. "Symmetrical" problems occur, e.g., when the target matrices are second derivative matrices, such as in [9]. Of course, when all the target matrices are real valued, the problem is both "symmetrical" and (in a degenerate sense) "Hermitian."

In this section, we will describe the algorithm for the more prevalent "Hermitian" problem. In the next section, we will indicate the differences for the "symmetrical" problem.

Our algorithm is given the acronym $AC-DC$ as it alternates between the two following minimization schemes.

- The AC ("alternating columns") phase minimizes $C_{LS}(\boldsymbol{B}, \boldsymbol{\Lambda}_1, \boldsymbol{\Lambda}_2, \dots \boldsymbol{\Lambda}_K)$ with respect to (w.r.t.) a selected column of $\boldsymbol{B}$ while keeping its other columns, as well as $\boldsymbol{\Lambda}_1, \boldsymbol{\Lambda}_2, \dots \boldsymbol{\Lambda}_K$, fixed.
- The DC ("diagonal centers") phase minimizes $C_{LS}(\boldsymbol{B}, \boldsymbol{\Lambda}_1, \boldsymbol{\Lambda}_2, \dots \boldsymbol{\Lambda}_K)$ w.r.t. all the diagonal matrices $\boldsymbol{\Lambda}_1, \boldsymbol{\Lambda}_2, \dots \boldsymbol{\Lambda}_K$ while keeping $\boldsymbol{B}$ fixed.

A. AC Phase

In this phase, we minimize C_{LS} w.r.t. the lth column of $\boldsymbol{B}$ ($1 \le l \le N$). C_{LS} can be expressed by

$$\begin{aligned} &C_{LS}(\boldsymbol{B}, \boldsymbol{\Lambda}_1, \boldsymbol{\Lambda}_2, \dots \boldsymbol{\Lambda}_K) \\ &\quad = \sum_{k=1}^{K} w_k \|\boldsymbol{A}_k - \boldsymbol{B}\boldsymbol{\Lambda}_k \boldsymbol{B}^H\|_F^2 \\ &\quad = \sum_{k=1}^{K} w_k \left\| \boldsymbol{A}_k - \sum_{n=1}^{N} \lambda_n^{[k]} \boldsymbol{b}_n \boldsymbol{b}_n^H \right\|_F^2 \end{aligned} \tag{2}$$

where $\boldsymbol{b}_n$ is the nth column of $\boldsymbol{B} = [\boldsymbol{b}_1 \boldsymbol{b}_2 \cdots \boldsymbol{b}_N]$ and where $\lambda_n^{[k]}$ is the (n,n)th entry of $\boldsymbol{\Lambda}_k$. The superscript ^{H} denotes the conjugate transpose. Defining, with respect to a specific choice of l

$$\tilde{\boldsymbol{A}}_k \triangleq \boldsymbol{A}_k - \sum_{\substack{n=1 \\ n\neq l}}^{N} \lambda_n^{[k]} \boldsymbol{b}_n \boldsymbol{b}_n^H \tag{3}$$

we have (using the fact that for the "Hermitian" problem all $\lambda_n^{[k]}$ are real valued, as is also evident in the DC phase, which follows)

$$\begin{aligned} &C_{LS}(\boldsymbol{B}, \boldsymbol{\Lambda}_1, \boldsymbol{\Lambda}_2, \dots \boldsymbol{\Lambda}_K) \\ &\quad = \sum_{k=1}^{K} w_k \|\tilde{\boldsymbol{A}}_k - \lambda_l^{[k]} \boldsymbol{b}_l \boldsymbol{b}_l^H\|_F^2 \\ &\quad = \sum_{k=1}^{K} w_k Tr\left\{ \left[\tilde{\boldsymbol{A}}_k - \lambda_l^{[k]} \boldsymbol{b}_l \boldsymbol{b}_l^H\right]^H \left[\tilde{\boldsymbol{A}}_k - \lambda_l^{[k]} \boldsymbol{b}_l \boldsymbol{b}_l^H\right] \right\} \\ &\quad = \tilde{C} - Tr\left\{ \sum_{k=1}^{K} w_k \lambda_l^{[k]} \left[\tilde{\boldsymbol{A}}_k^H \boldsymbol{b}_l \boldsymbol{b}_l^H + \boldsymbol{b}_l \boldsymbol{b}_l^H \tilde{\boldsymbol{A}}_k\right] \right\} \\ &\qquad + Tr\left\{ \sum_{k=1}^{K} w_k \left(\lambda_l^{[k]}\right)^2 \boldsymbol{b}_l \boldsymbol{b}_l^H \boldsymbol{b}_l \boldsymbol{b}_l^H \right\} \\ &\quad = \tilde{C} - \boldsymbol{b}_l^H \left[\sum_{k=1}^{K} w_k \lambda_l^{[k]} \left[\tilde{\boldsymbol{A}}_k^H + \tilde{\boldsymbol{A}}_k\right] \right] \boldsymbol{b}_l \\ &\qquad + \left(\boldsymbol{b}_l^H \boldsymbol{b}_l\right)^2 \sum_{k=1}^{K} w_k \left(\lambda_l^{[k]}\right)^2 \end{aligned} \tag{4}$$

where $\tilde{C}$ is quantity that does not depend on $\boldsymbol{b}_l$ and where $Tr\{\cdot\}$ denotes the trace.

Decomposing $\boldsymbol{b}_l$ into a real-valued scale $b \ge 0$ times a unit-norm vector $\boldsymbol{\beta}$ ($\boldsymbol{b}_l = b\boldsymbol{\beta}$ such that $\boldsymbol{\beta}^H\boldsymbol{\beta} = 1$), (4) can be reduced into

$$C_{LS}(b, \boldsymbol{\beta}) = \tilde{C} - 2b^2 \boldsymbol{\beta}^H \boldsymbol{P} \boldsymbol{\beta} + b^4 p \tag{5}$$

where $\boldsymbol{P}$ is the Hermitian matrix

$$\boldsymbol{P} \triangleq \sum_{k=1}^{K} w_k \lambda_l^{[k]} \tilde{\boldsymbol{A}}_k \tag{6}$$

and

$$p = \sum_{k=1}^{K} w_k \left(\lambda_l^{[k]}\right)^2. \tag{7}$$

Differentiating w.r.t. b and equating zero yields the solutions $b = 0$ or

$$b^2 = \frac{1}{p}\boldsymbol{\beta}^H \boldsymbol{P}\beta. \tag{8}$$

Since $\boldsymbol{P}$ is Hermitian, (8) is real valued. Thus, if the right-hand side of (8) is positive, then the minimizing b is its square root; otherwise, it is zero (because then, $b = 0$ minimizes C_{LS}(5) since p is positive[1]). Consequently, if $\boldsymbol{P}$ is negative-definite, then minimization of C_{LS} w.r.t. $\boldsymbol{b_l}$ is attained by $\boldsymbol{b_l} = \boldsymbol{0}$. However, normally, this is not the case, and substituting b^2 back into (5) reduces the problem into minimization w.r.t. $\boldsymbol{\beta}$ of

$$C_{LS}(\boldsymbol{\beta}) = \tilde{C} - \frac{1}{p}\left(\boldsymbol{\beta}^H \boldsymbol{P}\boldsymbol{\beta}\right)^2 \tag{9}$$

subject to $\boldsymbol{\beta}^H\boldsymbol{\beta} = 1$. The desired solution is attained as the eigenvector of $\boldsymbol{P}$ associated with the largest (*positive*) eigenvalue. Note, however, that the constrained maximization is insensitive to multiplication of $\boldsymbol{\beta}$ by any unit-magnitude (arbitrary phase) complex scalar. This ambiguity is inherent in the "Hermitian" problem and can be resolved, e.g., by imposing that the first nonzero element of $\boldsymbol{\beta}$ be real valued and positive.

Following is a summary of the AC phase for the "Hermitian" problem.

AC phase (Hermitian version):
Minimization of C_{LS} w.r.t. the lth column of $\boldsymbol{B}$:
Inputs:
- Target matrices $\boldsymbol{A}_1, \boldsymbol{A}_2, \ldots \boldsymbol{A}_K$;
- Diagonal matrices $\boldsymbol{\Lambda}_1, \boldsymbol{\Lambda}_2, \ldots \boldsymbol{\Lambda}_K$ (denoting the diagonal elements of $\boldsymbol{\Lambda}_k$ as $\lambda_1^{[k]}, \lambda_2^{[k]}, \ldots \lambda_N^{[k]}$);
- Weights $w_1, w_2, \ldots w_K$;
- Diagonalization matrix $\boldsymbol{B} = [\boldsymbol{b}_1 \boldsymbol{b}_2 \cdots \boldsymbol{b}_N]$;
- Selected column index l.

Algorithm:
1. Calculate

$$\boldsymbol{P} = \sum_{k=1}^{K} w_k \lambda_l^{[k]} \left[\boldsymbol{A}_k - \sum_{\substack{n=1 \\ n\neq l}}^{N} \lambda_n^{[k]} \boldsymbol{b}_n \boldsymbol{b}_n^H\right].$$

2. Find the largest eigenvalue μ and the associated unit-norm eigenvector $\boldsymbol{\beta}$ of $\boldsymbol{P}$, with its arbitrary phase determined such that its first nonzero element is real-valued positive.

[1] p can be zero only if all $\lambda_l^{[k]}$ are zero, in which case, C_{LS} is independent of $\boldsymbol{b_l}$ so that any $\boldsymbol{b_l}$ is a "minimizer."

3. If $\mu < 0$, set $\boldsymbol{b_l} = \boldsymbol{0}$; otherwise, set

$$\boldsymbol{b_l} = \frac{\boldsymbol{\beta}\sqrt{\mu}}{\sqrt{\sum_{k=1}^{K} w_k \left(\lambda_l^{[k]}\right)^2}}.$$

If the largest eigenvalue occurs with multiplicity of more than one, then any of the associated eigenvectors (or any unit-norm linear combination thereof) yields a possible solution.

B. DC Phase

In this phase, we minimize C_{LS} w.r.t. the diagonal matrices $\boldsymbol{\Lambda}_1, \boldsymbol{\Lambda}_2, \ldots \boldsymbol{\Lambda}_K$. Obviously, the minimization can be separated into K distinct minimization problems (for $k = 1, 2, \ldots K$), each minimizing

$$C_k(\boldsymbol{\Lambda}_k) \triangleq \left\|\boldsymbol{A}_k - \boldsymbol{B}\boldsymbol{\Lambda}_k\boldsymbol{B}^H\right\|_F^2 \tag{10}$$

which is a linear LS problem in the parameters vector $\boldsymbol{\lambda}_k = \text{diag}\{\boldsymbol{\Lambda}_k\}$. To formulate this as such, we define $\boldsymbol{a} = \text{vec}\{\boldsymbol{A}_k\}$ (vec$\{\cdot\}$ denoting the matrix-to-vector conversion by concatenation of columns) and rewrite (10) as

$$C_k(\boldsymbol{\lambda}_k) = [\boldsymbol{a} - \boldsymbol{H}\boldsymbol{\lambda}_k]^H [\boldsymbol{a} - \boldsymbol{H}\boldsymbol{\lambda}_k] \tag{11}$$

where it is straightforward to show that the matrix $\boldsymbol{H}$ is given by

$$\boldsymbol{H} = (\boldsymbol{B}^* \otimes \boldsymbol{1}) \odot (\boldsymbol{1} \otimes \boldsymbol{B}) \tag{12}$$

where $\otimes$ denotes Kronecker's product, $\odot$ denotes Hadamard's (element-wise) product, $\boldsymbol{1}$ denotes an $N \times 1$ vector of 1 s, and the superscript * denotes conjugation (note that this expression is sometimes referred to as the Khatri–Rao product of $\boldsymbol{B}^*$ and $\boldsymbol{B}$). The well-known minimizer of the linear LS problem is

$$\boldsymbol{\lambda}_k = \left[\boldsymbol{H}^H\boldsymbol{H}\right]^{-1}\boldsymbol{H}^H\boldsymbol{a} \tag{13}$$

where it is again straightforward to show that

$$\boldsymbol{H}^H\boldsymbol{H} = \left(\boldsymbol{B}^H\boldsymbol{B}\right)^* \odot \left(\boldsymbol{B}^H\boldsymbol{B}\right) \tag{14}$$

and

$$\boldsymbol{H}^H\boldsymbol{a} = \text{diag}\left\{\boldsymbol{B}^H\boldsymbol{A}_k\boldsymbol{B}\right\}. \tag{15}$$

If $\boldsymbol{H}^H\boldsymbol{H}$ is not invertible, then the LS minimizer is not unique. In that case, any $\boldsymbol{\lambda}_k$ that satisfies $\boldsymbol{H}^H\boldsymbol{H}\boldsymbol{\lambda}_k = \boldsymbol{H}^H\boldsymbol{a}$ is a minimizing solution. To eliminate the associated ambiguity, the minimum-norm solution may be chosen. In other words, $\boldsymbol{\lambda}_k$ would always be determined as $\boldsymbol{H}^+\boldsymbol{a}$, where $\boldsymbol{H}^+$ denotes the pseudo-inverse of $\boldsymbol{H}$.

Note that as is evident from (14), $\boldsymbol{H}^H\boldsymbol{H}$ is always real valued. In addition, diag$\{\boldsymbol{B}^H\boldsymbol{A}_k\boldsymbol{B}\}$ is also real valued since $\boldsymbol{A}_k$ is Hermitian. Consequently, the resulting $\boldsymbol{\lambda}_k$ is always real valued, as anticipated for the "Hermitian" problem. The DC phase is summarized in the following.

DC phase (Hermitian version):
Minimization of C_{LS} w.r.t. $\boldsymbol{\Lambda}_1, \boldsymbol{\Lambda}_2, \ldots \boldsymbol{\Lambda}_K$:
Inputs:
- Target matrices $\boldsymbol{A}_1, \boldsymbol{A}_2, \ldots \boldsymbol{A}_K$;
- Diagonalization matrix $\boldsymbol{B}$;

Algorithm:
1. Prepare
$\boldsymbol{G} = [(\boldsymbol{B}^H\boldsymbol{B})^* \odot (\boldsymbol{B}^H\boldsymbol{B})]^{-1}$
2. For $k = 1, 2, \ldots K$,
Set $\boldsymbol{\Lambda}_k = \text{diag}\{\boldsymbol{G}\ \text{diag}\{\boldsymbol{B}^H\boldsymbol{A}_k\boldsymbol{B}\}\}$.

III. "Symmetrical" Problem

In the "symmetrical" problem, both the AC and DC phases are slightly changed, mainly due to the sensitivity of C_{LS} to the complex phases of the columns of $\boldsymbol{B}$, in addition to the fact that $\boldsymbol{\Lambda}_1, \boldsymbol{\Lambda}_2, \ldots \boldsymbol{\Lambda}_K$ are no longer real valued. Equation (4) is rewritten as

$$\begin{aligned} &C_{LS}(\boldsymbol{B}, \boldsymbol{\Lambda}_1, \boldsymbol{\Lambda}_2, \ldots \boldsymbol{\Lambda}_K) \\ &\quad = \sum_{k=1}^{K} w_k Tr\left\{\left[\tilde{\boldsymbol{A}}_k - \lambda_l^{[k]}\boldsymbol{b}_l\boldsymbol{b}_l^T\right]^H\left[\tilde{\boldsymbol{A}}_k - \lambda_l^{[k]}\boldsymbol{b}_l\boldsymbol{b}_l^T\right]\right\} \\ &\quad = \tilde{C} - 2\text{Real}\left\{\boldsymbol{b}_l^T\left[\sum_{k=1}^{K} w_k\lambda_l^{[k]}\tilde{\boldsymbol{A}}_k^H\right]\boldsymbol{b}_l\right\} \\ &\qquad + \left(\boldsymbol{b}_l^H\boldsymbol{b}_l\right)^2\sum_{k=1}^{K} w_k\left|\lambda_l^{[k]}\right|^2 \end{aligned} \tag{16}$$

where now

$$\tilde{\boldsymbol{A}}_k \triangleq \boldsymbol{A}_k - \sum_{\substack{n=1 \\ n\neq l}}^{N} \lambda_n^{[k]}\boldsymbol{b}_n\boldsymbol{b}_n^T. \tag{17}$$

Decomposing $\boldsymbol{b}_l = b\boldsymbol{\beta}$ again, with $b \geq 0$ (real valued) and $\boldsymbol{\beta}^H\boldsymbol{\beta} = 1$, we now obtain

$$C_{LS}(b, \boldsymbol{\beta}) = \tilde{C} - 2b^2\text{Real}\left\{\boldsymbol{\beta}^T\boldsymbol{P}\boldsymbol{\beta}\right\} + b^4 p \tag{18}$$

with

$$\boldsymbol{P} \triangleq \sum_{k=1}^{K} w_k\lambda_l^{[k]}\tilde{\boldsymbol{A}}_k^* \tag{19}$$

and

$$p = \sum_{k=1}^{K} w_k\left|\lambda_l^{[k]}\right|^2. \tag{20}$$

To maximize Real$\{\boldsymbol{\beta}^T\boldsymbol{P}\boldsymbol{\beta}\}$ (w.r.t. $\boldsymbol{\beta}$, subject to $\boldsymbol{\beta}^H\boldsymbol{\beta} = 1$), we decompose $\boldsymbol{\beta} = \boldsymbol{\gamma} + j\boldsymbol{\delta}$ and $\boldsymbol{P} = \boldsymbol{Q} + j\boldsymbol{R}$, where $j = \sqrt{-1}$, $\boldsymbol{\gamma}, \boldsymbol{\delta} \in \mathbb{R}^N$, and $\boldsymbol{Q}, \boldsymbol{R} \in \mathbb{R}^{N\times N}$ and observe that

$$\begin{aligned} \text{Real}\left\{\boldsymbol{\beta}^T\boldsymbol{P}\boldsymbol{\beta}\right\} &= \boldsymbol{\gamma}^T\boldsymbol{Q}\boldsymbol{\gamma} - \boldsymbol{\delta}^T\boldsymbol{Q}\boldsymbol{\delta} - \boldsymbol{\gamma}^T\boldsymbol{R}\boldsymbol{\delta} - \boldsymbol{\delta}^T\boldsymbol{R}\boldsymbol{\gamma} \\ &= [\boldsymbol{\gamma}^T\boldsymbol{\delta}^T]\begin{bmatrix}\boldsymbol{Q} & -\boldsymbol{R} \\ -\boldsymbol{R} & -\boldsymbol{Q}\end{bmatrix}\begin{bmatrix}\boldsymbol{\gamma} \\ \boldsymbol{\delta}\end{bmatrix} \end{aligned} \tag{21}$$

whereas the unity norm constraint translates into $\boldsymbol{\gamma}^T\boldsymbol{\gamma} + \boldsymbol{\delta}^T\boldsymbol{\delta} = 1$. Thus, the solution can be extracted from the eigenvector associated with the largest (positive) eigenvalue of the matrix

$$\tilde{\boldsymbol{P}} \triangleq \begin{bmatrix}\text{Real}\{\boldsymbol{P}\} & -\text{Imag}\{\boldsymbol{P}\} \\ -\text{Imag}\{\boldsymbol{P}\} & -\text{Real}\{\boldsymbol{P}\}\end{bmatrix}. \tag{22}$$

The AC phase therefore assumes the following form.

AC phase (symmetrical version):
Minimization of C_{LS} w.r.t. the lth column of $\boldsymbol{B}$:
Inputs:
- Target matrices $\boldsymbol{A}_1, \boldsymbol{A}_2, \ldots \boldsymbol{A}_K$;
- Diagonal matrices $\boldsymbol{\Lambda}_1, \boldsymbol{\Lambda}_2, \ldots \boldsymbol{\Lambda}_K$ (denoting the diagonal elements of $\boldsymbol{\Lambda}_k$ as $\lambda_1^{[k]}, \lambda_2^{[k]}, \ldots \lambda_N^{[k]}$);
- Weights $w_1, w_2, \ldots w_K$;
- Diagonalization matrix $\boldsymbol{B} = [\boldsymbol{b}_1\boldsymbol{b}_2\cdots\boldsymbol{b}_N]$;
- Selected column index l.

Algorithm:
1. Calculate

$$\boldsymbol{P} = \sum_{k=1}^{K} w_k\lambda_l^{[k]}\left[\boldsymbol{A}_k - \sum_{\substack{n=1 \\ n\neq l}}^{N}\lambda_n^{[k]}\boldsymbol{b}_n\boldsymbol{b}_n^T\right]^*;$$

2. Find the largest eigenvalue μ and the associated unit-norm eigenvector $\boldsymbol{\xi} \in \mathbb{R}^{2N}$ of

$$\tilde{\boldsymbol{P}} = \begin{bmatrix}\text{Real}\{\boldsymbol{P}\} & -\text{Imag}\{\boldsymbol{P}\} \\ -\text{Imag}\{\boldsymbol{P}\} & -\text{Real}\{\boldsymbol{P}\}\end{bmatrix}.$$

3. If $\mu < 0$, set $\boldsymbol{b}_l = \boldsymbol{0}$; otherwise, partition $\boldsymbol{\xi}$ as

$$\boldsymbol{\xi} = \begin{bmatrix}\boldsymbol{\gamma} \\ \boldsymbol{\delta}\end{bmatrix}$$

where $\gamma, \delta \in \mathbb{R}^N$, and set

$$\boldsymbol{b}_l = \frac{(\boldsymbol{\gamma} + j\boldsymbol{\delta})\sqrt{\mu}}{\sqrt{\sum_{k=1}^{K} w_k\left|\lambda_l^{[k]}\right|^2}}.$$

Note that in this case, there's no phase ambiguity in $\boldsymbol{\beta}$ (unless the largest eigenvalue occurs with multiplicity of more than one).

As for the DC phase, we now have (preserving the notations of Section II-B)

$$\boldsymbol{H} = (\boldsymbol{B}\otimes\boldsymbol{1}) \odot (\boldsymbol{1}\otimes\boldsymbol{B}) \tag{23}$$

(or the Khatri–Rao product of $\boldsymbol{B}$ with itself) so that

$$\boldsymbol{H}^H\boldsymbol{H} = (\boldsymbol{B}^H\boldsymbol{B}) \odot (\boldsymbol{B}^H\boldsymbol{B}) \tag{24}$$

and

$$\boldsymbol{H}^H\boldsymbol{a} = \text{diag}\{\boldsymbol{B}^H\boldsymbol{A}_k\boldsymbol{B}^*\}. \tag{25}$$

The resulting DC phase follows.

DC phase (symmetrical version):
Minimization of C_{LS} w.r.t. $\boldsymbol{\Lambda}_1, \boldsymbol{\Lambda}_2, \ldots \boldsymbol{\Lambda}_K$:
Inputs:
- Target matrices $\boldsymbol{A}_1, \boldsymbol{A}_2, \ldots \boldsymbol{A}_K$;
- Diagonalization matrix $\boldsymbol{B}$;

Algorithm:
1. Prepare
$\boldsymbol{G} = \left[(\boldsymbol{B}^H\boldsymbol{B}) \odot (\boldsymbol{B}^H\boldsymbol{B})\right]^{-1}$
2. For $k = 1, 2, \ldots K$,
Set $\boldsymbol{\Lambda}_k = \text{diag}\{\boldsymbol{G}\ \text{diag}\{\boldsymbol{B}^H\boldsymbol{A}_k\boldsymbol{B}^*\}\}$.

As before, if $(\boldsymbol{B}^H\boldsymbol{B}) \odot (\boldsymbol{B}^H\boldsymbol{B})$ is not invertible, the minimum-norm solution may be chosen among all possible minimizing solutions.

IV. Interlacing the AC and DC Phases, Convergence and Computational Issues

The "natural" objective of the $AC-DC$ algorithm is to alternate between minimization w.r.t. $\boldsymbol{\Lambda}_1, \boldsymbol{\Lambda}_2, \ldots \boldsymbol{\Lambda}_K$ and minimization w.r.t. $\boldsymbol{B}$. Although the former is attained via a single DC run, the latter requires consecutive AC "sweeps" over all N columns of $\boldsymbol{B}$ (i.e., for $l = 1, 2, \ldots N$ but not necessarily in that order). Thus, in principle, each DC phase is to be followed by an "infinite" number of AC sweeps before the next DC phase is run.

Alternatively, any fixed number of AC sweeps (or even incomplete sweeps) may interlace DC runs, without attaining "true" minimization w.r.t. $\boldsymbol{B}$ between true minimizations w.r.t. $\boldsymbol{\Lambda}_1, \boldsymbol{\Lambda}_2, \ldots \boldsymbol{\Lambda}_K$.

Nevertheless, with either strategy, C_{LS} is guaranteed not to increase (usually to decrease) with each AC/DC iteration; thus (because it is bounded below), convergence of C_{LS} is guaranteed. Explicitly stated

$$\forall \delta > 0 \; \exists N : C_{LS}^{[n]} - C_{LS}^{[n+1]} < \delta \; \forall n > N \tag{26}$$

where $C_{LS}^{[n]}$ denotes the value of C_{LS} after n iterations. By "iteration" in this context, we refer to any single (isolated) AC or DC iteration). We will use the term "full iteration" to refer to the combination of a DC phase and one or more full AC sweeps.

One possible stopping condition for the algorithm would be to monitor changes in C_{LS} and stop when they are sufficiently small, say, below an arbitrarily fixed value δ. Evidently, the algorithm is guaranteed to meet such a stopping condition after a finite number of iterations. However, in general, convergence of C_{LS} does not necessarily imply convergence of the parameters $\boldsymbol{B}$ and $\boldsymbol{\Lambda}_1, \boldsymbol{\Lambda}_2, \ldots \boldsymbol{\Lambda}_K$. An alternative, more "parameters-oriented," stopping condition would be to monitor the norm of change in the parameters, compared with another arbitrary small threshold ϵ as a stopping condition. We will now show that under some mild assumptions, the algorithm is guaranteed to meet this alternative stopping condition (in a finite number of full iterations) as well.

To this end, we now assume momentarily that in each AC iteration, the largest eigenvalue is unique and that in each DC, iteration $\boldsymbol{H}^H\boldsymbol{H}$ has full rank. In that case, the solution of each AC/DC iteration is the unique global minimizer w.r.t. the respective parameters subset. Denoting this parameter subset (at the nth iteration) by $\boldsymbol{\theta}^{[n+1]}$ and all other parameters by $\boldsymbol{\phi}^{[n]}$, we have

$$\boldsymbol{\theta}^{[n+1]} = \arg\min_{\boldsymbol{\theta} \in \boldsymbol{\Theta}} C_{LS}\left(\boldsymbol{\theta}, \boldsymbol{\phi}^{[n]}\right) \tag{27}$$

where $\boldsymbol{\Theta}$ denotes the set of all "valid" minimizers. For the DC phase, this is simply $\mathbb{R}^N$ (for each $k = 1, 2, \ldots K$); for the AC phase, we eliminate the phase ambiguity in the Hermitian version by defining $\boldsymbol{\Theta}$ to be the set of all vectors in $\mathbb{C}^N$ whose first nonzero element is real valued and positive.

In Appendix A, we show that for each iteration (AC or DC), the above-mentioned uniqueness conditions imply that $\exists \epsilon_0 > 0$ such that

$$\begin{aligned} \forall \epsilon \in (0, \epsilon_0) \exists \delta > 0 : \forall \boldsymbol{\theta} \in \boldsymbol{\Theta} \\ C_{LS}\left(\boldsymbol{\theta}, \phi^{[n]}\right) - C_{LS}\left(\boldsymbol{\theta}^{[n+1]}, \phi^{[n]}\right) < \delta \\ \Rightarrow \left\| \boldsymbol{\theta} - \boldsymbol{\theta}^{[n+1]} \right\| < \epsilon. \end{aligned} \tag{28}$$

Assuming further that there exists a positive infimum $\bar{\epsilon}$ to all the ϵ_0-s (of all iterations), we can conclude by substituting $\boldsymbol{\theta} = \boldsymbol{\theta}^{[n]}$ in (28) and using (26) that

$$\forall \epsilon \in (0, \bar{\epsilon}) \; \exists N : \left\| \boldsymbol{\theta}^{[n]} - \boldsymbol{\theta}^{[n+1]} \right\| < \epsilon \; \forall n > N \tag{29}$$

which implies that a parameter-monitoring stopping condition must be met. It has to be stressed, however, that strictly speaking, this does not imply convergence of the parameters, i.e., it does not imply that the sequence of parameter estimates is a convergent sequence; it merely implies that the change in estimated parameters is guaranteed to fall below any arbitrarily small fixed threshold. However, it has been observed empirically that the parameters indeed converge, but we have not been able to provide a rigorous proof of that property.

Convergence of all the parameters implies in turn (by construction) that the derivatives of C_{LS} w.r.t. all the parameters vanish simultaneously, and hence, a true stationary point is attained. Note, however, that although the stationary point is minimum with respect to each parameters subset individually, this does not necessarily imply that it is minimum w.r.t. all the parameters combined (in other words, the second derivative matrix of C_{LS} w.r.t. all the parameters consists of block-positive definite matrices along its diagonal, but it is not necessarily positive definite).

These arguments are somewhat weakened whenever our assumption (that all AC/DC iterations have unique minimizers) is not satisfied, i.e., when multiple largest eigenvalues occur in an AC iteration or when $\boldsymbol{H}^H\boldsymbol{H}$ is positive semi-definite in a DC iteration. However, in such cases, additional artificial constraints on the parameters (such as minimum norm) may be used so that uniqueness can still be imposed.

A possible intelligent initialization would be to set $\boldsymbol{B}$ to the (exact) joint diagonalizer of any two matrices in $\mathcal{A}$, say, $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$, e.g., via diagonalization of $\boldsymbol{A}_1\boldsymbol{A}_2^{-1}$ (assuming $\boldsymbol{A}_2$ is invertible). The $AC-DC$ algorithm will then start with a DC phase. Alternatively, if an intelligent guess as to the anticipated diagonal values is available, then $\boldsymbol{\Lambda}_1, \boldsymbol{\Lambda}_2, \ldots \boldsymbol{\Lambda}_K$ can be initialized accordingly, and the $AC-DC$ algorithm will start with an AC phase.

In Fig. 1, we demonstrate typical convergence patterns of C_{LS} (in the Hermitian case) for one and for two AC sweeps interlaced between each DC phase. The "true" set $\mathcal{A}_0$ of ten Hermitian 4×4 matrices was generated by drawing at random the elements of the true $\boldsymbol{B}$ (independent, complex normal standard random variables) and of the diagonals of the true $\boldsymbol{\Lambda}_k$ (independent, uniformly distributed in $(0, 1]$). Then, the perturbed target set $\mathcal{A}$ was generated by noising (conjugate symmetrically) all elements of the matrices in $\mathcal{A}_0$ with independent complex normal random variables with zero mean and standard deviation (std)

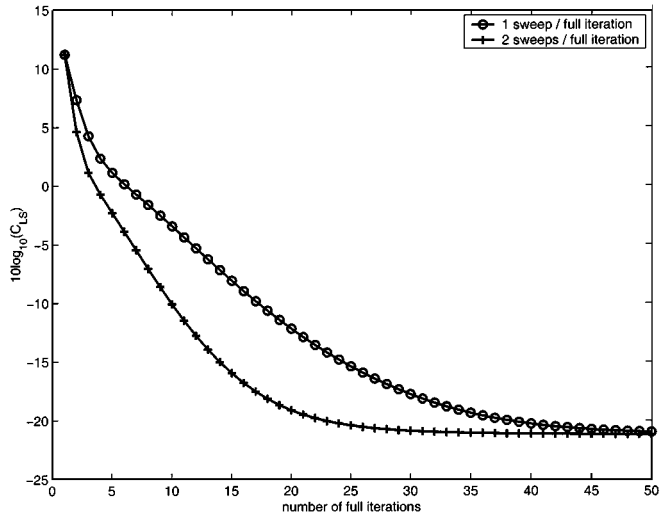

Fig. 1. Typical convergence patterns of C_{LS} for randomly generated matrices with 1 and 2 full AC sweeps per iteration.

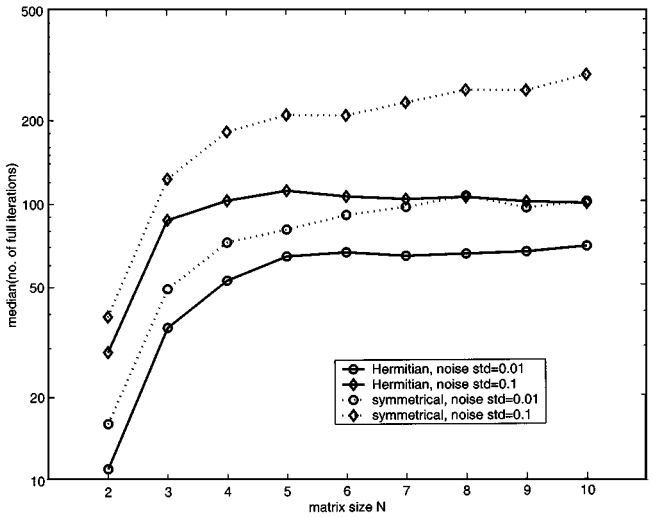

Fig. 2. Median of the required number of full iterations versus the matrices' dimension N for two noise conditions: std. of 0.2, 0.02 for the Hermitian and symmetrical cases. The number of matrices was $K = 10$. Each result reflects the median of 400 trials.

0.02. The patterns show that with two AC sweeps in each iteration, the convergence rate is nearly two times faster. Since the associated computational load per iteration is nearly two times higher, there is no distinct preference for either strategy in this example.

To demonstrate the statistical behavior of the algorithm, we repeated the experiment with matrix orders N ranging from $N = 2$ to $N = 10$, with 400 independent trials per order, running two AC sweeps per full iteration. We maintained $K = 10$ matrices in all experiments. In Fig. 2, we plot the median of the number of full iterations required to reach convergence, where convergence was defined as the state where the absolute change in each estimated λ in the DC phase falls below 0.001. These experiments were also repeated with stronger noise (with std of 0.2) and then (with the same noise levels) for the symmetrical case as well (for which symmetrical noisy matrices were generated). It is seen, as could be expected, that the (median) number of required full iterations is generally higher for the higher noise levels. However, there seem to be some inherent differences between the Hermitian and symmetrical cases in the behavior as a function of N: For the Hermitian case, the required number of full iterations increases with N only up to $N = 4$ and then remains roughly constant, whereas for the symmetrical case, this number seems to increase monotonically for up to $N = 8$ for the low noise level and up to $N = 10$ (or higher) for the higher noise level. In addition, the required number of full iterations for the symmetrical case is generally higher than that of the Hermitian case for fixed N and noise level.

Addressing the issue of computational load per iteration, we will now assume, for simplicity, that each AC phase consists of a single full sweep. The computational load for the DC phase (in terms of the number of complex multiplications) is $\mathcal{O}(KN^3)$. As for the AC phase, the calculation of $\boldsymbol{P}$ requires $\mathcal{O}(KN^3)$ multiplications and formally has to be repeated N times, which would amount to $\mathcal{O}(KN^4)$. Note, however, that the sum $\sum_{\substack{n=1\\ n\neq l}}^{N} \lambda_n^{[k]} \boldsymbol{b}_n \boldsymbol{b}_n^H$ can be expressed as $\boldsymbol{B}\boldsymbol{\Lambda}_k\boldsymbol{B}^H - \lambda_l^{[k]}\boldsymbol{b}_l\boldsymbol{b}_l^H$, and it suffices to compute the K matrices $\boldsymbol{B}\boldsymbol{\Lambda}_k\boldsymbol{B}^H$ just once per sweep so that the load for the entire sweep reduces back to $\mathcal{O}(KN^3)$ (we assume that the load for computing the largest eigenvalue and the associated eigenvector is $\mathcal{O}(N^3)$ and is, hence, negligible). The overall computational load per iteration is therefore $\mathcal{O}(KN^3)$.

As mentioned in the introduction, a subspace-based joint diagonalization algorithm using Gauss–Newton iterations was recently proposed in [18]. When that algorithm converges, its convergence rate is substantially faster than that of $AC - DC$. However, it requires an $\mathcal{O}(N^3)$-fold higher computational cost per iteration, namely, $\mathcal{O}(KN^7)$. Additionally, as is typical for Gauss-Newton type algorithms, it may occasionally diverge, depending on initialization and/or selection of a step-size parameter, whereas $AC - DC$ is guaranteed to meet the stopping condition, regardless of initialization.

V. Simulation Results With JADE

To demonstrate the potential performance improvement in the context of BSS, we present in Fig. 3 some simulations results by applying Cardoso's joint approximate diagonalization of eigen-matrices (JADE) algorithm [19] to a 3×3 noisy BSS problem. We will not go into detail describing the JADE algorithm here. We only mention that it is based on the joint diagonalization of empirical fourth-order cumulant matrices of the observed (mixed) data. The standard application of JADE requires a prewhitening phase, followed by orthogonal joint diagonalization of the transformed matrices using an extended Jacobi algorithm [14]. This algorithm is computationally cheaper (per full iteration) than $AC - DC$ since it has a computational load of $\mathcal{O}(KN^2)$; however, as we will demonstrate, it may be inferior to the $AC - DC$ algorithm in terms of the resulting separation performance.

As an alternative, we applied the $AC - DC$ algorithm to the entire set (which is real valued here so that the "Hermitian" and "symmetrical" algorithms coincide). Conventional JADE was applied to each data set, followed by application of $AC - DC$ to the same data, with three different combinations of weights

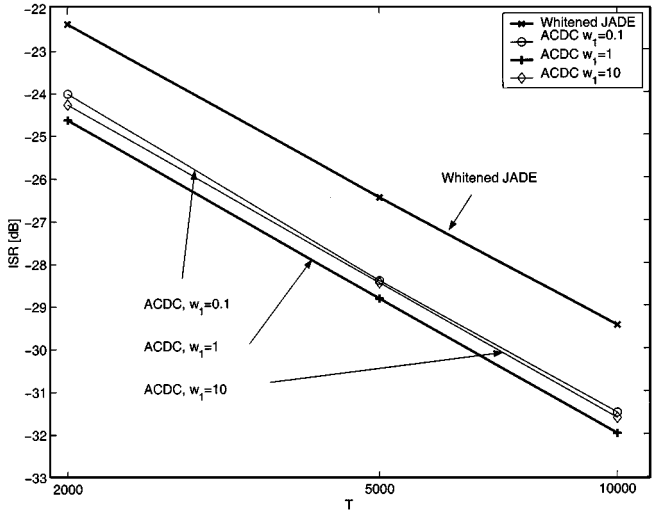

Fig. 3. Average ISR for a 3×3 BSS problem versus the observation length T, using the JADE algorithm. Simulation results are presented for conventional (whitened) JADE as well as for $AC-DC$ with three different weightscombinations. Each point reflects the average of 500 trials. All algorithms operated on the same estimated matrices.

w_k. All the weights associated with the estimated cumulants matrices were set to unity, whereas the weight w_1 associated with the estimated correlation matrix was varied between three different values, namely, 0.1, 1, and 10. Each application of $AC-DC$ used the conventional JADE output as an initial guess.

The simulation setup was the following: The three source signals were generated as independent samples, uniformly distributed between $-\sqrt{3}$ and $\sqrt{3}$ (thus having zero mean and unit variance). The true mixing matrix was

$$\mathbf{B}_0 = \begin{bmatrix} 1.9 & 3 & -0.5 \\ -0.2 & 0.4 & -0.1 \\ -0.2 & -0.3 & 0.9 \end{bmatrix}.$$

T samples from each observed signal were used to obtain unbiased estimates of the correlation and cumulants matrices. The $AC-DC$ algorithm was applied directly to the estimated matrices. Conventional (whitened) JADE was applied as follows. Rather than whiten the entire data set and re-estimate the matrices, we applied the whitening to the already-estimated cumulants matrices (as described in the Introduction) and then used the orthogonal joint diagonalization algorithm of [14].

The results are measured in terms of the average interference to signal ratio (ISR) implied by the estimate of the mixing matrix. The ISR per trial is calculated as follows: First, the composite separation matrix $\boldsymbol{B}^{-1}\boldsymbol{B}_0$ (where $\boldsymbol{B}$ is the estimated diagonalizing matrix) is calculated; then, all its elements are squared; the ISR in each row is the sum of all (squared) elements in the row except for the largest, divided by the largest; the average ISR is the average of all the row ISRs.

In Fig. 3, we demonstrate the resulting average ISR (in decibels) versus the number of samples T for conventional JADE as well as for $AC-DC$ with the three different weight combinations. The most significant improvement over conventional JADE is attained with $w_1 = 1$.

We stress that the optimum with respect to w_1 is attained around $w_1 = 1$ with this specific setup but may be attained by essentially different weighting for different mixing matrices and/or source signals. It is evident, for example, that mere scaling of the signals is equivalent to a relative change of weights since the estimated correlation matrix involves second-order powers of the signals, whereas the estimated cumulants matrices involve the fourth-order powers thereof. The performance of $AC-DC$ may be further improved by proper choice of the entire set of relative weights, possibly considering the statistics of the estimation errors, such as by the approach taken in [8]. However, such considerations are beyond the scope of this paper.

VI. Conclusion

We proposed an iterative algorithm for approximate joint diagonalization of a given set of matrices in the weighted LS sense with arbitrary positive weights. The algorithm is applicable both in the "Hermitian" framework, when all the target matrices are Hermitian, and in the "symmetrical" framework, when all the target matrices are symmetric. Convergence to a stationary point of the LS criterion is guaranteed under mild conditions.

Performance improvement over "hard whitening" followed by orthogonal diagonalization can be attained by proper selection of the weights, which enables a more balanced distribution of the deviations from exact diagonalization among all the matrices.

Finally, note that the proposed algorithm may be easily adapted to find the "reduced" joint diagonalization with smaller diagonal matrices $\boldsymbol{\Lambda}_1, \boldsymbol{\Lambda}_2, \ldots \boldsymbol{\Lambda}_K \in \mathbb{C}^{M\times M}$, where $M < N$ so that the diagonalizing matrix $\boldsymbol{B} \in \mathbb{C}^{N\times M}$ is not square. Such an application occurs, e.g., in blind source separation with more sensors than sources. The only required modification (for both the Hermitian and symmetric versions) is in the AC phase, where each occurrence of N in the algorithm specification should be replaced with M.

A MATLAB® package for the algorithm is available at the ICA-Central web-site:http://tsi.enst.fr/~cardoso/icacentral/Algos/yeredor.

Appendix
Existence of "Unique Neighborhoods" for Unique AC/DC Minima

Certain assumptions (to be restated shortly) imply uniqueness of the minimizers in each AC/DC single iteration. We will now show that under these assumptions, these minimizers are not only unique but also have "unique neighborhoods" in the sense that small changes in the value of C_{LS} must imply small changes in the parameters. Explicitly stated, in each iteration (AC or DC), $\exists \epsilon_0 > 0$ such that

$$\begin{aligned} &\forall \epsilon \in (0,\epsilon_0)\, \exists \delta > 0 : \forall \boldsymbol{\theta} \in \boldsymbol{\Theta} \\ &\quad C_{LS}\left(\boldsymbol{\theta}, \boldsymbol{\phi}^{[n]}\right) - C_{LS}\left(\boldsymbol{\theta}^{[n+1]}, \boldsymbol{\phi}^{[n]}\right) < \delta \\ &\qquad \Rightarrow \left\| \boldsymbol{\theta} - \boldsymbol{\theta}^{[n+1]} \right\| < \epsilon \end{aligned} \tag{30}$$

where $\boldsymbol{\Theta}$ denotes the set of all valid parameters for the minimization, $\boldsymbol{\theta}^{[n+1]}$ denotes the minimizing parameters at the nth iteration, and $\boldsymbol{\phi}^{[n]}$ denotes all the other, irrelevant parameters

$$\boldsymbol{\theta}^{[n+1]} = \arg\min_{\boldsymbol{\theta}\in\boldsymbol{\Theta}} C_{LS}\left(\boldsymbol{\theta}, \boldsymbol{\phi}^{[n]}\right). \tag{31}$$

To simplify the notation, we will drop references to the iteration index n and to the irrelevant parameters $\boldsymbol{\phi}^{[n]}$ and denote the minimizer $\boldsymbol{\theta}^{[n+1]}$ simply as $\boldsymbol{\theta}_0$. With this simplified notation, we intend to show an equivalent statement, namely, that in each iteration, $\exists \epsilon_0 > 0$ such that

$$\forall \epsilon \in (0, \epsilon_0)\ \exists \delta > 0 : \forall \boldsymbol{\theta} \in \boldsymbol{\Theta}, \quad \|\boldsymbol{\theta} - \boldsymbol{\theta}_0\| \geq \epsilon \Rightarrow C(\boldsymbol{\theta}) - C(\boldsymbol{\theta}_0) \geq \delta \tag{32}$$

where $C(\boldsymbol{\theta})$ is shorthand for $C_{LS}(\boldsymbol{\theta}, \boldsymbol{\phi}^{[n]})$.

For the DC phase, $\boldsymbol{\Theta}$ is simply $\mathbb{R}^N$ (for each $k = 1, 2, \ldots K$). The required uniqueness assumption is for $\boldsymbol{H}^H\boldsymbol{H}$ [of either (14) or (24)] to have full rank, in which case, $C(\boldsymbol{\theta})$ is a quadratic function of $\boldsymbol{\theta}$ with a positive-definite constant Hessian. Consequently, denoting $\boldsymbol{\varepsilon} \triangleq \boldsymbol{\theta} - \boldsymbol{\theta}_0$, we have

$$C(\boldsymbol{\theta}) = C(\boldsymbol{\theta}_0 + \boldsymbol{\varepsilon}) = C(\boldsymbol{\theta}_0) + \boldsymbol{\varepsilon}^H \left[\boldsymbol{H}^H\boldsymbol{H}\right] \boldsymbol{\varepsilon} \tag{33}$$

so that (32) is actually satisfied $\forall \epsilon_0 > 0$ with $\delta = \mu_{\min}\epsilon^2$, where $\mu_{\min}$ denotes the minimal (positive) eigenvalue of $\boldsymbol{H}^H\boldsymbol{H}$ since

$$C(\boldsymbol{\theta}) - C(\boldsymbol{\theta}_0) = \boldsymbol{\varepsilon}^H \left[\boldsymbol{H}^H\boldsymbol{H}\right] \boldsymbol{\varepsilon} \geq \mu_{\min}\|\boldsymbol{\varepsilon}\|^2. \tag{34}$$

For the AC phase, the proof is a little more involved. We consider the Hermitian case here, but the proof can also be similarly derived for the symmetrical case. The set $\boldsymbol{\Theta}$ of valid minimization vectors is defined as all vectors in $\mathbb{C}^N$ whose first nonzero element is real valued and positive. The uniqueness condition for the AC phase minimizer is for the largest eigenvalue of $\boldsymbol{P}$ [which is defined in (6)] to be unique. Specifically stated, if we denote $\boldsymbol{P}$'s eigenvalues as $\mu_1 > \mu_2 \geq \cdots \geq \mu_N$ in descending order, there exist some positive κ such that $\mu_1 = \mu_2 + \kappa$.

As was done in Section II-A, we decompose the vector $\boldsymbol{\theta}$ of the minimization parameters into $\boldsymbol{\theta} = b\boldsymbol{\beta}$, where $b \geq 0$ is a non-negative scale, and $\boldsymbol{\beta}$ is a unit-norm vector ($\boldsymbol{\beta}^H\boldsymbol{\beta} = 1$), whose phase is arbitrarily determined such that its first nonzero element is real valued positive. We also denote the minimizing solution as $\boldsymbol{\theta}_0 = b_0\boldsymbol{\beta}_0$. Consequently, we have

$$\|\boldsymbol{\theta} - \boldsymbol{\theta}_0\|^2 = b^2 + b_0^2 - bb_0\left(\boldsymbol{\beta}^H\boldsymbol{\beta}_0 + \boldsymbol{\beta}_0^H\boldsymbol{\beta}\right). \tag{35}$$

We further define vectors $\boldsymbol{\alpha}$ and $\boldsymbol{\alpha}_0$ as follows. Let $\boldsymbol{P} = \boldsymbol{V}\boldsymbol{M}\boldsymbol{V}^H$ denote the eigenvalue decomposition of the Hermitian matrix $\boldsymbol{P}$ such that $\boldsymbol{V}^H\boldsymbol{V} = \boldsymbol{I}_N$ (where $\boldsymbol{I}_N$ denotes the $N \times N$ identity matrix), and $\boldsymbol{M} = \text{diag}\{\mu_1, \mu_2, \ldots \mu_N\}$. Using $\boldsymbol{V}$, we now define

$$\boldsymbol{\alpha} \triangleq \boldsymbol{V}^H\boldsymbol{\beta}, \quad \boldsymbol{\alpha}_0 \triangleq \boldsymbol{V}^H\boldsymbol{\beta}_0 = \boldsymbol{e}_1 \tag{36}$$

where $\boldsymbol{e}_1$ is the first column of $\boldsymbol{I}_N$ (and the right-hand side equality holds since $\boldsymbol{\beta}_0$ is the eigenvector corresponding to μ_1). Thus, $\boldsymbol{\beta}^H\boldsymbol{\beta} = \boldsymbol{\alpha}^H\boldsymbol{\alpha} = \alpha_1$ (the first element of $\boldsymbol{\alpha}$) so that

$$\begin{aligned}\|\boldsymbol{\theta} - \boldsymbol{\theta}_0\|^2 &= b^2 + b_0^2 - 2bb_0 Real\{\alpha_1\} \\ &= (b - b_0)^2 + 2bb_0(1 - Real\{\alpha_1\}).\end{aligned} \tag{37}$$

We can always select the arbitrary phases of $\boldsymbol{V}$s columns such that α_1 is real valued and non-negative. This would enable substitution of the notation $Real\{\alpha_1\}$ simply with α_1. Note further that since $\boldsymbol{\alpha}$ is also a unitary vector, we would always have $0 \leq \alpha_1 \leq 1$.

We now introduce the following Lemma.

Lemma 1: Let $0 < \gamma < 1$ denote an arbitrary constant, and define $\epsilon_0 \triangleq (1 - \gamma)b_0$. Then

$$\|\boldsymbol{\theta} - \boldsymbol{\theta}_0\| < \epsilon_0 \Rightarrow b > \gamma b_0 \tag{38}$$

Proof: Using (37), the Lemma's condition implies

$$\begin{aligned}(1 - \gamma)^2 b_0^2 = \epsilon_0^2 &> (b - b_0)^2 + 2bb_0(1 - \alpha_1) \\ &\geq (b - b_0)^2 = \left(1 - \frac{b}{b_0}\right)^2 b_0^2\end{aligned} \tag{39}$$

from which $b > \gamma b_0$ follows immediately (recall that $b, b_0 \geq 0$).

Observe now from (5) and (9) that $C(\boldsymbol{\theta})$ is given by

$$C(\boldsymbol{\theta}) = \tilde{C} - 2b^2\boldsymbol{\beta}^H\boldsymbol{P}\boldsymbol{\beta} + b^4 p = \tilde{C} - 2b^2\sum_{n=1}^{N}\mu_n|\alpha_n|^2 + b^4 p \tag{40}$$

and the minimized $C(\boldsymbol{\theta}_0)$ is given by

$$C(\boldsymbol{\theta}_0) = \tilde{C} - \frac{\mu_1^2}{p} \tag{41}$$

where $\tilde{C}$ and p are some constants depending on irrelevant parameters. In addition, we have, using (8), $b_0^2 = \mu_1/p$. We therefore obtain

$$\begin{aligned}C(\boldsymbol{\theta}) - C(\boldsymbol{\theta}_0) &= b^4 p - 2b^2\sum_{n=1}^{N}\mu_n|\alpha_n|^2 + \frac{\mu_1^2}{p} \\ &= p\left(b^2 - \frac{\mu_1}{p}\right)^2 + 2b^2\left(\mu_1 - \sum_{n=1}^{N}\mu_n|\alpha_n|^2\right) \\ &= p(b + b_0)^2(b - b_0)^2 \\ &\quad + 2b^2\left(\mu_1 - \sum_{n=1}^{N}\mu_n|\alpha_n|^2\right).\end{aligned} \tag{42}$$

We can now use the following inequality:

$$\begin{aligned}\mu_1 - \sum_{n=1}^{N}\mu_n|\alpha_n|^2 &= \mu_1 - \mu_1\alpha_1^2 - \sum_{n=2}^{N}\mu_n|\alpha_n|^2 \\ &\geq \mu_1 - \mu_1\alpha_1^2 - \mu_2\sum_{n=2}^{N}|\alpha_n|^2 \\ &= \mu_1 - \mu_1\alpha_1^2 + \mu_2\alpha_1^2 - \mu_2\sum_{n=1}^{N}|\alpha_n|^2 \\ &= (\mu_1 - \mu_2)(1 - \alpha_1^2) \\ &\geq \kappa(1 - \alpha_1)\end{aligned} \tag{43}$$

to conclude that

$$C(\boldsymbol{\theta}) - C(\boldsymbol{\theta}_0) \geq pb_0^2(b - b_0)^2 + 2b^2\kappa(1 - \alpha_1). \tag{44}$$

Our ultimate goal is now to prove the following claim.

Lemma 2: $\forall \epsilon < \epsilon_0$, the condition $\|\boldsymbol{\theta} - \boldsymbol{\theta}_0\| \geq \epsilon$, which, from (37), is equivalent to

$$(b - b_0)^2 + 2bb_0(1 - \alpha_1) \geq \epsilon^2 \tag{45}$$

implies that $C(\boldsymbol{\theta}) - C(\boldsymbol{\theta}_0) \geq \delta \triangleq \min\{pb_0\epsilon^2, \kappa\gamma\epsilon^2\}$ by implying

$$(b - b_0)^2 + \frac{2b^2\kappa}{pb_0}(1 - \alpha_1) \geq \min\left\{\epsilon^2, \frac{\kappa\gamma}{pb_0}\epsilon^2\right\} \tag{46}$$

and using (44). ϵ_0 is related to the arbitrary constant γ via Lemma 1.

Proof: Using Lemma 1 and the relation $\epsilon < \epsilon_0$, the condition (45) implies that $b > \gamma b_0$. We will now show that the claim in Lemma 2 holds in each of the two possible cases: $b \geq \mu_1/\kappa$ and $\gamma b_0 < b < \mu_1/\kappa$ (naturally, the latter may sometimes be nonexistent, depending b_0, μ_1, κ and the arbitrary choice of γ).

Assume first that $b \geq \mu_1/\kappa$. We restate the Lemma's condition as

$$(b-b_0)^2 \geq \epsilon^2 - 2bb_0(1-\alpha_1) \tag{47}$$

and use this inequality to obtain

$$\begin{aligned}(b-b_0)^2 &+ \frac{2b^2\kappa}{pb_0}(1-\alpha_1) \\ &\geq \epsilon^2 - 2bb_0(1-\alpha_1) + \frac{2b^2\kappa}{pb_0}(1-\alpha_1) \\ &= \epsilon^2 + 2b(1-\alpha_1)\left[\frac{\kappa b}{pb_0} - b_0\right] \geq \epsilon^2 \end{aligned} \tag{48}$$

where the last inequality results from the assumption $b \geq \mu_1/\kappa = pb_0^2/\kappa$.

Conversely, assuming that $\gamma b_0 < b < \mu_1/\kappa$, we restate the Lemma's condition as

$$2b(1-\alpha_1) \geq \frac{\epsilon^2 - (b-b_0)^2}{b_0} \tag{49}$$

leading to

$$\begin{aligned}(b-b_0)^2 + \frac{2b^2\kappa}{pb_0}(1-\alpha_1) &\geq (b-b_0)^2 + \frac{\kappa b}{pb_0}\frac{\epsilon^2-(b-b_0)^2}{b_0} \\ &= \frac{\kappa b}{pb_0^2}\epsilon^2 + (b-b_0)^2\left(1-\frac{\kappa b}{pb_0^2}\right) \\ &> \frac{\kappa b}{\mu_1}\epsilon^2 > \frac{\kappa\gamma b_0}{\mu_1}\epsilon^2 = \frac{\kappa\gamma}{pb_0}\epsilon^2. \end{aligned} \tag{50}$$

Evidently, in either case, (46) is satisfied.

REFERENCES

[1] J.-F. Cardoso, "Source separation using higher order moments," in *Proc. ICASSP*, 1989, pp. 2109–2112.

[2] E. Moreau, "A generalization of joint-diagonalization criteria for source separation," *IEEE Trans. Signal Processing*, vol. 49, pp. 530–541, Mar. 2001.

[3] L. Tong, Y. Inouye, and L. Ruei-wen, "Waveform-preserving blind estimation of multiple independent sources," *IEEE Trans. Signal Processing*, vol. 41, pp. 2461–2470, July 1993.

[4] L. De Lathauwer, "Signal Processing Based on Multilinear Algebra," Ph.D. dissertation, Katholiecke Univ. Leuven, Leuven, Belgium, 1997.

[5] L. De Lathauwer, B. De Moor, and J. Vandewalle, "Independent component analysis and (simultaneous) third-order tensor diagonalization," *IEEE Trans. Signal Processing*, vol. 49, pp. 2262–2271, Oct. 2001.

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

[7] A. Belouchrani, K. Abed-Meraim, J.-F. Cardoso, and E. Moulines, "A blind source separation technique using second-order statistics," *IEEE Trans. Signal Processing*, vol. 45, pp. 434–444, Feb. 1997.

[8] A. Yeredor, "Blind separation of Gaussian sources via second-order statistics with asymptotically optimal weighting," *IEEE Signal Processing Lett.*, vol. 7, pp. 197–200, July 2000.

[9] ——, "Blind source separation via the second characteristic function," *Signal Process.*, vol. 80, no. 5, pp. 897–902, 2000.

[10] K. Rahbar and J. P. Reilly, "Blind source separation of convolved sources by joint approximate diagonalization of cross-spectral density matrices," in *Proc. ICASSP*, vol. 5, Salt Lake City, UT, 2001.

[11] P. Comon and E. Moreau, "Blind MIMO equalization and joint diagonalization criteria," in *Proc. ICASSP*, vol. 5, Salt Lake City, UT, 2001.

[12] J.-C. Pesquet, B. Chen, and A. P. Petropulu, "Frequency-domain contrast functions for separation of convolutive mixtures," in *Proc. ICASSP*, vol. 5, Salt Lake City, UT, 2001.

[13] B. Chen and A. P. Petropulu, "Frequency domain blind MIMO system identification based on second and higher order statistics," *IEEE Trans. Signal Processing*, vol. 49, pp. 1677–1688, Aug. 2001.

[14] J.-F. Cardoso and A. Souloumiac, "Jacobi angles for simultaneous diagonalization," *SIAM J. Matrix Anal. Appl.*, vol. 17, no. 1, pp. 161–164, 1996.

[15] M. Wax and J. Sheinvald, "A least-squares approach to joint diagonalization," *IEEE Signal Processing Lett.*, vol. 4, pp. 52–53, Feb. 1997.

[16] J.-F. Cardoso, "On the performance of orthogonal source separation algorithms," in *Proc. EUSIPCO*, 1994, pp. 776–779.

[17] D.-T. Pham. (1999) Joint Approximate Diagonalization of Positive-Definite Hermitialn Matrices. [Online]. Available: http://www-lmc.imag.fr/lmc-sms/Dinh-Tuan.Pham/jadiag/jadiag.ps.gz

[18] A.-J van der Veen, "Joint diagonalization via subspace fitting techniques," in *Proc. ICASSP*, vol. 5, Salt Lake City, UT, 2001.

[19] J.-F. Cardoso and A. Souloumiac, "Blind beamforming for non Gaussian signals," *Proc. Inst. Elect. Eng. F*, vol. 140, no. 6, pp. 362–370, 1993.

Arie Yeredor (M'98) was born in Haifa, Israel, in 1963. He received the B.Sc. in electrical engineering (summa cum laude) and Ph.D. degrees from Tel-Aviv University, Tel-Aviv, Israel, in 1984 and 1997, respectively.

From 1984 to 1990, he was with the Israeli Defense Forces (Intelligence Corps), where he was in charge of advanced research and development activities in the fields of statistical and array signal processing. Since 1990, he has been with NICE Systems, Inc., where he holds a consulting position in the fields of speech and audio processing, video processing, and emitter location algorithms. He is also currently a Faculty Member with the Department of Electrical Engineering–Systems at Tel-Aviv University, where he teaches courses in statistical and digital siginal processing. His research interests include estimation theory, statistical processing, and blind source separation.

Dr. Yeredor has been awarded the Best Lecturer of the Faculty of Engineering Award for three consecutive years.