SAM 2002, Aug. 4-6, Rosslyn, VA, USA, pp. 403-407.

# JOINT DIAGONALIZATION OF CORRELATION MATRICES BY USING NEWTON METHODS WITH APPLICATION TO BLIND SIGNAL SEPARATION

*Marcel Joho*[1], *Kamran Rahbar* [2]

[1]Phonak Inc., 1901 South First Street, Champaign, IL, USA
[2]ECE Department, McMaster University, Hamilton, Ont., Canada
*joho@ieee.org*, *kamran@reverb.crl.mcmaster.ca*

## ABSTRACT

This paper addresses the blind signal separation problem in the presence of sensor noise for the case where the source signals are non-stationary and / or non-white. This problem can be formulated as a joint-diagonalization problem where the objective is to jointly diagonalize a set of correlation matrices $\{\mathbf{R}_p\}$, using a single matrix $\mathbf{W}$. We derive a Newton-type algorithm for two joint-diagonalization cost functions, which are related to the aforementioned blind signal separation problem. To this end, we derive the gradient and also the Hessian of the joint diagonalization cost function in closed form. The most general case is considered, in which the source signals and the unknown mixing matrix are assumed to be complex.

## 1. INTRODUCTION

### 1.1. Notation

The notation used throughout this paper is the following: Vectors are written in lower case, matrices in upper case. Matrix and vector transpose, complex conjugation and Hermitian transpose are denoted by $(.)^T$, $(.)^*$, and $(.)^H \triangleq ((.)^*)^T$, respectively. The identity matrix is denoted by $\mathbf{I}$, a vector or a matrix containing only zeros by $\mathbf{0}$, and a vector or a matrix containing only ones by $\mathbf{1}$. $E\{.\}$ denotes the expectation operator. Vector or matrix dimensions are given in superscript, e.g., $\mathbf{I}^M$. The Frobenius norm and the trace of a matrix are denoted by $\|\,.\,\|_F$ and $\mathrm{tr}\,(.)$, respectively ( $\|\mathbf{A}\|_F^2 \triangleq \mathrm{tr}(\mathbf{A}\mathbf{A}^H)$ ). With $\mathbf{a} = \mathrm{diag}(\,\mathbf{A}\,)$ we obtain a vector whose elements are the diagonal elements of $\mathbf{A}$ and $\mathrm{diag}(\,\mathbf{a}\,)$ is a square diagonal matrix which contains the elements of $\mathbf{a}$. $\mathrm{ddiag}(\,\mathbf{A}\,)$ is a diagonal matrix where its diagonal elements are the same as the diagonal elements of $\mathbf{A}$ and

$$\mathrm{off}(\,\mathbf{A}\,) \triangleq \mathbf{A} - \mathrm{ddiag}(\,\mathbf{A}\,)\;. \tag{1}$$

For a square matrix $\mathbf{A}$ we have $\mathrm{ddiag}(\,\mathbf{A}\,) \triangleq \mathrm{diag}(\,\mathrm{diag}(\,\mathbf{A}\,)\,)$. $\mathrm{vec}(\,\mathbf{W}\,)$ forms a column vector by stacking the columns of $\mathbf{W}$, and $\otimes$ denotes the Kronecker product [1]. $\mathbf{P}_{\text{vec}}$ is the involutary ($\mathbf{P}_{\text{vec}}^{-1} = \mathbf{P}_{\text{vec}}$) permutation matrix which is uniquely defined with

$$\mathbf{P}_{\text{vec}}\,\mathrm{vec}(\,\mathbf{W}^T\,) = \mathrm{vec}(\,\mathbf{W}\,)\,. \tag{2}$$

Furthermore, we define the two following $M^2 \times M^2$ diagonal projection matrices

$$\mathbf{P}_{\text{diag}} \triangleq \mathrm{diag}(\,\mathrm{vec}(\,\mathbf{I}^M\,)\,) = \mathrm{diag}(\,(\mathbf{e}_1^T, \cdots, \mathbf{e}_M^T)^T\,) \tag{3}$$

$$\mathbf{P}_{\text{off}} \triangleq \mathrm{diag}(\,\mathrm{vec}(\,\mathrm{off}(\,\mathbf{1}^{M\times M}\,)\,)\,) = \mathbf{I}^{M^2} - \mathbf{P}_{\text{diag}} \tag{4}$$

where $\mathbf{e}_m$ denotes the $m$th unit vector and $\mathbf{I}^M = \mathrm{ddiag}\big(\,\mathbf{1}^{M\times M}\,\big)$.

### 1.2. Problem definition

We define the following two problems [2]:

**Problem 1:** Let $\{\mathbf{R}_p\}_{p=1}^P$ be a set of $P$ given *correlation matrices*. We aim at finding a matrix $\mathbf{W}$ that minimizes the following cost function:

$$\mathcal{J}_1 \triangleq \sum_{p=1}^{P} \beta_p\, \mathcal{J}_1^{(p)} \triangleq \sum_{p=1}^{P} \beta_p\, \left\| \mathrm{off}(\,\mathbf{W}\mathbf{R}_p\mathbf{W}^H\,) \right\|_F^2 \tag{5}$$

where $\{\beta_p\}$ are positive weighting factors, *normalized* such that

$$\sum_{p=1}^{P} \beta_p\, \|\,\mathbf{R}_p\,\|_F^2 = 1\,. \tag{6}$$

**Problem 2:** Let $\{\mathbf{R}_p\}_{p=1}^P$ be a set of $P$ given *positive definite Hermitian matrices*. We aim at finding a matrix $\mathbf{W}$ and a real diagonal matrix $\mathbf{N}$ with diagonal elements $n_{i,i} \geq 0$ such that $\{\mathbf{W}, \mathbf{N}\}$ minimize the following cost function

$$\mathcal{J}_2 \triangleq \sum_{p=1}^{P} \beta_p\, \mathcal{J}_2^{(p)} \triangleq \sum_{p=1}^{P} \beta_p\, \left\| \mathrm{off}(\,\mathbf{W}(\mathbf{R}_p - \mathbf{N})\mathbf{W}^H\,) \right\|_F^2\;. \tag{7}$$

As in Problem 1, we require again that the weights $\{\beta_p\}$ are normalized such that (6) is fulfilled.

### 1.3. Comments

Perfect joint diagonalization is normally not possible for an arbitrary set of correlation matrices $\{\mathbf{R}_p\}$. However, if $\{\mathbf{R}_p\} = \{\mathbf{A}\mathbf{\Lambda}_p\mathbf{A}^H\}$ with $\{\mathbf{\Lambda}_p\}$ being diagonal matrices, full diagonalization is possible and, therefore, the cost function (5) is zero at its global minimum. For Problem 2, perfect joint diagonalization is possible, when $\{\mathbf{R}_p\} = \{\mathbf{A}\mathbf{\Lambda}_p\mathbf{A}^H + \mathbf{D}\}$ and $\mathbf{D}$ is a positive semidefinite diagonal matrix.

The purpose of choosing the normalization in (6) is to make the cost functions (5) and (7) *independent* of the absolute norms $\{\|\,\mathbf{R}_p\,\|_F\}$ [2]. Even though Newton-type algorithms are insensitive to scaling of the cost function, we employ this normalization, because in the initial stage of the algorithm we might use a few gradient-based iterations.

Note, since $\mathbf{W} = \mathbf{0}$ minimizes (5) and (7), we require also some additional properties of $\mathbf{W}$ to prevent the trivial solution. Possible constraints are that $\mathbf{W}$ should be unitary, or the diagonal elements of $\mathbf{W}$ are constraint to be one. The associated cost functions of these two constraints are

$$\mathcal{J}_3 \triangleq \left\| \mathbf{W}\mathbf{W}^H - \mathbf{I} \right\|_F^2 \tag{8}$$

$$\mathcal{J}_4 \triangleq \| \operatorname{ddiag}( \mathbf{W} - \mathbf{I} ) \|_F^2 \tag{9}$$

which can be used either as a hard constraint or as a penalty term [3] in the optimization of (5) or (7). Other possible choices of penalty terms are listed in [4].

In Table 3 the gradient $\mathbf{D}_{\mathbf{W}}$ and Hessian $\{\mathbf{H}_{\mathbf{W}}, \mathbf{C}_{\mathbf{W}}\}$ of the cost functions $\mathcal{J}_1$ to $\mathcal{J}_4$ are summarized. Due to space limitation, we can not present all derivations. However, one example is given in Appendix A which shows the principal steps, and in Appendix B we summarize all relations which were helpful for the derivation.

## 2. SECOND-ORDER APPROXIMATION

In this paper we follow the notation of Manton in [5] and express the second-order Taylor series approximation of a cost function $\mathcal{J} : \mathbb{C}^{M\times M} \to \mathbb{R}$ in the *matrix form*

$$\begin{aligned}\mathcal{J}(\mathbf{W} + \delta\mathbf{Z}) &= \mathcal{J}(\mathbf{W}) + \delta\, \mathfrak{Re}\{\operatorname{tr}\left(\mathbf{Z}^H\mathbf{D}_{\mathbf{W}}\right)\} \\ &+ \frac{\delta^2}{2} \operatorname{vec}(\mathbf{Z})^H \mathbf{H}_{\mathbf{W}} \operatorname{vec}(\mathbf{Z}) \\ &+ \frac{\delta^2}{2} \mathfrak{Re}\{\operatorname{vec}(\mathbf{Z})^T \mathbf{C}_{\mathbf{W}} \operatorname{vec}(\mathbf{Z})\} + O(\delta^3)\end{aligned} \tag{10}$$

where $\mathbf{D}_{\mathbf{W}} \in \mathbb{C}^{M\times M}$ is the derivative of $\mathcal{J}$ evaluated at $\mathbf{W}$, and $\mathbf{H}_{\mathbf{W}}, \mathbf{C}_{\mathbf{W}} \in \mathbb{C}^{M^2\times M^2}$ are the *Hessian* of $\mathcal{J}$ evaluated at $\mathbf{W}$. To ensure uniqueness, we require $\mathbf{H}_{\mathbf{W}}^H = \mathbf{H}_{\mathbf{W}}$ and $\mathbf{C}_{\mathbf{W}}^T = \mathbf{C}_{\mathbf{W}}$.

As pointed out in [5], the matrix form (10) is an alternative way of formulating the second-order Taylor series approximation in *vector form* with real-valued elements, i.e., $\mathcal{J} : \mathbb{R}^{2M^2} \to \mathbb{R}$ in the form

$$\mathcal{J}(\mathbf{w} + \delta\mathbf{z}) = \mathcal{J}(\mathbf{w}) + \delta\, \mathbf{z}^T\mathbf{d}_{\mathbf{w}} + \frac{\delta^2}{2}\mathbf{z}^T\mathbf{H}_{\mathbf{w}}\mathbf{z} + O(\delta^3) \tag{11}$$

where $\mathbf{w} \in \mathbb{R}^{2M^2}$, $\mathbf{d}_{\mathbf{w}} \in \mathbb{R}^{2M^2}$ is the derivative of $\mathcal{J}$ evaluated at $\mathbf{w}$, and $\mathbf{H}_{\mathbf{w}} \in \mathbb{R}^{2M^2\times 2M^2}$ is the Hessian of $\mathcal{J}$, evaluated at $\mathbf{w}$, if $\mathbf{H}_{\mathbf{w}}$ is symmetric, i.e., $\mathbf{H}_{\mathbf{w}} = \mathbf{H}_{\mathbf{w}}^T$. A possible transformation between (10) and (11) is given in Appendix C for $\mathbf{w} \triangleq (\mathbf{w}_{\text{re}}^T\ \mathbf{w}_{\text{im}}^T)^T \in \mathbb{R}^{2M^2}$ where $\mathbf{w}_{\text{re}} \triangleq \mathfrak{Re}\{\operatorname{vec}(\mathbf{W})\}$ and $\mathbf{w}_{\text{im}} \triangleq \mathfrak{Im}\{\operatorname{vec}(\mathbf{W})\}$.

## 3. JOINT DIAGONALIZATION BY USING NEWTON METHODS

For an unconstraint optimization problem formulated in the vector form (11), the Newton step $\Delta\mathbf{w}_k$ at iteration $k$ is obtained from solving the equation $\mathbf{H}_{\mathbf{w}k}\Delta\mathbf{w}_k = -\mathbf{d}_{\mathbf{w}k}$. Hence, the Newton update is $\mathbf{w}_{k+1} = \mathbf{w}_k + \Delta\mathbf{w}_k = \mathbf{w}_k - \mathbf{H}_{\mathbf{w}k}^{-1}\mathbf{d}_{\mathbf{w}k}$. Problems arise when $\mathbf{H}_{\mathbf{w}k}$ is not positive definite because $-\mathbf{H}_{\mathbf{w}k}^{-1}\mathbf{d}_{\mathbf{w}k}$ no longer is guaranteed to be a descent direction. By adding a multiple of the identity matrix to $\mathbf{H}_{\mathbf{w}k}$, we can always make $\mathbf{H}_{\mathbf{w}k} + \sigma_k\mathbf{I}$ positive definite by choosing $\sigma_k > -\lambda_{min}(\mathbf{H}_{\mathbf{w}k})$. Thus, the modified Newton update becomes [6]

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \left[\mathbf{H}_{\mathbf{w}k} + \sigma_k\mathbf{I}\right]^{-1}\mathbf{d}_{\mathbf{w}k}\,. \tag{20}$$

Properties of (20) are that in the initial stage its behavior is similar to a steepest-descent algorithm. However, once $\mathbf{H}_{\mathbf{w}k}$ becomes positive definite, we can set $\sigma_k$ to zero and hence the algorithm switches to the pure Newton algorithm with a *quadratic* convergence rate. We choose the update (20) as the basis for our Newton algorithm for the joint-diagonalization task.

**Table 1**. Newton algorithm for the joint-diagonalization task with a unitary matrix.

**Newton-JD**

Initialization ($k=0$):

$$\mathbf{W}_0 = \mathbf{I} \qquad \text{(or any other unitary matrix)} \tag{12}$$

For $k=1,2,\ldots$

$$\mathbf{D}_{\mathbf{W}k} = \mathbf{D}_{\mathbf{W}}(\mathbf{W}_k) \tag{13}$$

$$\mathbf{H}_{\mathbf{W}k} = \mathbf{H}_{\mathbf{W}}(\mathbf{W}_k) \tag{14}$$

$$\mathbf{C}_{\mathbf{W}k} = \mathbf{C}_{\mathbf{W}}(\mathbf{W}_k) \tag{15}$$

$$\{\mathbf{H}_{\mathbf{W}k},\mathbf{C}_{\mathbf{W}k}\} \xrightarrow{(68)} \mathbf{H}_{\mathbf{w}k} \tag{16}$$

$$\sigma_k \geq \max(0, -\lambda_{min}(\mathbf{H}_{\mathbf{w}k})) \tag{17}$$

$$\Delta\mathbf{W}_k = \texttt{cpoint}(\mathbf{W}_k, \mathbf{D}_{\mathbf{W}k}, \mathbf{H}_{\mathbf{W}k} + \sigma_k\mathbf{I}, \mathbf{C}_{\mathbf{W}k}) \tag{18}$$

$$\mathbf{W}_{k+1} = \pi(\mathbf{W}_k + \Delta\mathbf{W}_k)\,. \tag{19}$$

In principle, there are two possible update strategies: either update the matrix $\mathbf{W}_{k+1}$ directly or update first the vector $\mathbf{w}_{k+1}$ (e.g., with (20)) and then use (69) to obtain $\mathbf{W}_{k+1}$. Each form can be transformed into the other form as described in Appendix C. Recently, Manton presented in [5] the `cpoint` routine, which can be used for computing the Newton step on the *complex Stiefel manifold* (manifold of unitary matrices). We will use the `cpoint` routine for directly updating a *unitary* matrix $\mathbf{W}_{k+1}$.

In Table 1 we summarize the proposed Newton algorithm for the joint-diagonalization task using $\mathcal{J} = \mathcal{J}_1$ from (5) as our objective function. We restrict ourselves to the case where we constrain $\mathbf{W}$ to be unitary.

In (12) we initialize $\mathbf{W}_0$. The gradient of the $k$th iteration is computed in (13). Since $\mathcal{J} = \sum \beta_p \mathcal{J}_1^{(p)}$ with the elementary cost functions $\mathcal{J}_1^{(p)}$, the overall gradient of $\mathcal{J}$ becomes $\mathbf{D}_{\mathbf{W}} = \sum \beta_p \mathbf{D}_{\mathbf{W}1}^{(p)}$ where $\mathbf{D}_{\mathbf{W}1}^{(p)}$ is the gradient of $\mathcal{J}_1^{(p)}$. The gradients $\mathbf{D}_{\mathbf{W}1}^{(p)}$ are obtained from Table 3 with $\mathbf{D}_{\mathbf{W}1}(\mathbf{R}{=}\mathbf{R}_p)$. The Hessian of the $k$th iteration are computed in (14) and (15). Here we can also write $\mathbf{H}_{\mathbf{W}} = \sum \beta_p \mathbf{H}_{\mathbf{W}1}^{(p)}$ and $\mathbf{C}_{\mathbf{W}} = \sum \beta_p \mathbf{C}_{\mathbf{W}1}^{(p)}$. Thereafter, in (16) the Hessian $\{\mathbf{H}_{\mathbf{W}k}, \mathbf{C}_{\mathbf{W}k}\}$ of the matrix form is transformed with (68) into the equivalent Hessian $\mathbf{H}_{\mathbf{w}k}$ of the vector form. The smallest eigenvalue $\lambda_{min}$ of $\mathbf{H}_{\mathbf{w}k}$ is used in (17) to determine $\sigma_k$ such that $\mathbf{H}_{\mathbf{w}k} + \sigma_k\mathbf{I}$ becomes positive semidefinite. It is easy to show, that this regularization is equal to the regularization $\mathbf{H}_{\mathbf{W}k} + \sigma_k\mathbf{I}$ used in (18) as

$$\{\mathbf{H}_{\mathbf{W}k} + \sigma_k\mathbf{I}, \mathbf{C}_{\mathbf{W}k}\} \xrightarrow{(68)} \mathbf{H}_{\mathbf{w}k} + \sigma_k\mathbf{I}\,. \tag{21}$$

In (18) the Newton step $\Delta\mathbf{W}_k$ is computed with `cpoint` and used in the update equation (19). Finally, with the projection operator $\pi(.)$, where

$$\pi(\mathbf{W}) = \pi(\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^H) \triangleq \mathbf{U}\mathbf{V}^H \tag{22}$$

**Table 2**. Steepest-descent algorithm for the joint-diagonalization task with a unitary matrix [8, 2].

**Steepest-descent JD**

Initialization ($k=0$):

$$\mathbf{W}_0 = \mathbf{I} \qquad \text{(or any other unitary matrix)} \tag{23}$$

For $k=1,2,\ldots$

$$\mathbf{D}_{\mathbf{W}k} = \mathbf{D}_{\mathbf{W}}(\mathbf{W}_k) \tag{24}$$

$$\Delta\mathbf{W}_k = -\mu(\mathbf{D}_{\mathbf{W}k} - \mathbf{W}_k \mathbf{D}_{\mathbf{W}k}^H \mathbf{W}_k) \tag{25}$$

$$\mathbf{W}_{k+1} = \pi(\mathbf{W}_k + \Delta\mathbf{W}_k)\,. \tag{26}$$

is the closest projection of a matrix $\mathbf{W}$ onto the Stiefel manifold [5, 7], we obtain $\mathbf{W}_{k+1}$ which is unitary again. $\mathbf{W} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^H$ is the singular value decomposition (SVD) of $\mathbf{W}$. The proposed Newton algorithm also works in the case where we choose $\mathcal{J} = \mathcal{J}_2$ from (7) and assume that $\mathbf{N}$ is known.

## 4. JOINT-DIAGONALIZATION TECHNIQUES FOR BLIND SIGNAL SEPARATION

In the blind signal separation task (BSS), joint-diagonalization techniques were first applied in [9] for the case where the source signals were non-Gaussian. Afterwards, in [10] it has been shown that for the case where the source signals are non-white, the blind signal separation problem can also be reduced to solving a joint-diagonalization problem. The case was extended in [11, 12] for source signals that are also non-stationary. The corresponding cost function can be written as

$$\mathcal{J}_5 \triangleq \sum_\tau \sum_p \beta_{p,\tau} \left\| \mathrm{off}\left( \mathbf{W} \left( \mathbf{R}_{\mathbf{xx}_p}(\tau) - \delta(\tau)\hat{\mathbf{R}}_{\mathbf{nn}} \right) \mathbf{W}^H \right) \right\|_F^2 \tag{27}$$

where $\{\mathbf{R}_{\mathbf{xx}_p}(\tau)\} \triangleq \{\mathbf{R}_{\mathbf{xx}}(t_p,\tau)\}$ is a given set of correlation matrices from different time points $t_p$ and with different time lags $\tau$. $\hat{\mathbf{R}}_{\mathbf{nn}}$ is the estimate of the sensor noise correlation matrix. It is shown in [2], that the cost function $\mathcal{J}_5$ can be subdivided into the two cost functions $\mathcal{J}_1$ and $\mathcal{J}_2$.

## 5. SIMULATION

In the following we analyze the behavior of the proposed Newton algorithm via numerical simulations. We generate a set of $P=15$ correlation matrices where $\{\mathbf{R}_p\} = \{\mathbf{A}\boldsymbol{\Lambda}_p\mathbf{A}^H\}$, $\mathbf{A} \in C^{5\times5}$ is a randomly chosen unitary matrix, and $\{\boldsymbol{\Lambda}_p\}$ are randomly chosen complex diagonal matrices. The real and imaginary part of the diagonal elements of $\boldsymbol{\Lambda}_p$ are in the range $[-1,1]$. We choose $\beta_p = 1/\sum \|\mathbf{R}_p\|_F^2$ which fulfills (6).

In this simulation, our objective is to find a unitary matrix $\mathbf{W}$ that minimizes the cost function $\mathcal{J}_1$ defined in (5). We compare two different algorithms: (a) proposed Newton-based algorithm (Table 1) and (b) steepest-descent algorithm (Table 2). Fig. 1 shows the performance curves of ten independent runs of each algorithm with ten different sets of correlation matrices. For the gradient-based algorithm, the step-size $\mu = 0.5$ was chosen to achieve the highest convergence speed without becoming unstable.

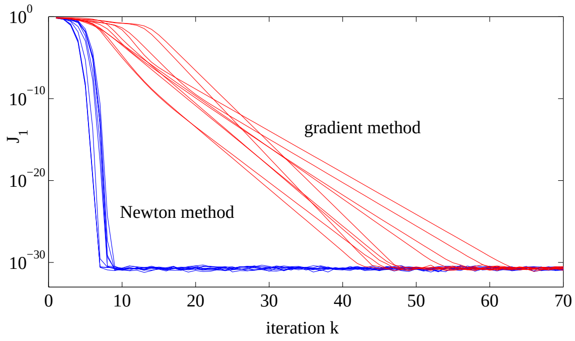

**Fig. 1**. Learning curves of $\mathcal{J}_1$ for ten independent simulations with $M=5$ and $P=15$ using (a) proposed Newton algorithm (Table 1) and (b) steepest-descent algorithm (Table 2).

We clearly see that the proposed Newton-algorithm dramatically outperforms the gradient-based algorithm. The Newton algorithm needs only a few steps to fully converge. A few regularized steps with $\sigma_k > 0$ are usually carried out at the beginning. Once the Newton algorithm comes to the vicinity of a minimum, $\mathbf{H}_{\mathbf{w}k}$ becomes positive definite and the algorithm converges within a few iterations to a global minimum. As can be seen in Fig. 1, for a given step size $\mu$ the slope of the learning curve for the gradient algorithm depends on the set of correlation matrices.

## 6. CONCLUSIONS

We have presented a Newton algorithm for the problem of joint (approximate) diagonalization of a given set of correlation matrices with a unitary matrix. To this end, we have derived the gradient, and also the Hessian, of the corresponding joint-diagonalization cost function in closed form. The proposed algorithm enjoys the property of self-adjusting its step size: far away from a minimum, the algorithm behaves like gradient based algorithm, and in the vicinity of a minimum it softly switches to a pure Newton algorithm with a quadratic convergence rate.

Under the unitary constraint we can also maximize $\mathcal{J}_6 \triangleq \sum \beta_p \| \,\mathrm{ddiag}\big(\mathbf{W}\mathbf{R}_p\mathbf{W}^H\big)\|_F^2$ instead of minimizing $\mathcal{J}_1$, see [13, 14]. The gradient and Hessian of $\mathcal{J}_6$ are easily obtained by taking $\mathbf{D}_{\mathbf{W}1}$, $\mathbf{H}_{\mathbf{W}1}$, and $\mathbf{C}_{\mathbf{W}1}$ from Table 3 and replacing $\mathrm{off}(\,.\,)$ with $\mathrm{ddiag}(\,.\,)$, and $\mathbf{P}_{\text{off}}$ with $\mathbf{P}_{\text{diag}}$.

Recently, a slightly different Newton algorithm with similar performance was proposed by Nikpour *et al.* in [8]. They compared their algorithm with the well known joint-diagonalization algorithm of [13]. In [15] a Gauss-Newton algorithm was proposed, which uses an approximation of the Hessian. Recently, Newton methods were also derived and applied in blind signal separation for the convolutive mixing problem [16, 17]. Also, there is a strong interest in finding non-unitary matrices for solving the joint-diagonalization problem, see [18, 19].

In contrast to [8], the gradients and Hessians in Table 3 also hold for a non-unitary matrix $\mathbf{W}$, hence, they can be used as well for Newton algorithms which update a non-unitary matrix $\mathbf{W}$.

## A. DERIVATION OF GRADIENT AND HESSIAN

Due to space limitations, we cannot derive all gradient and Hessians given in Table 3. Since most of the derivations look similar, we choose

$$\mathcal{J}(\mathbf{W}) \triangleq \|\,\mathrm{off}(\mathbf{W}\mathbf{R}\mathbf{W}^H)\,\|_F^2 \tag{28}$$
$$= \|\mathbf{W}\mathbf{R}\mathbf{W}^H\,\|_F^2 - \|\,\mathrm{ddiag}(\mathbf{W}\mathbf{R}\mathbf{W}^H)\,\|_F^2 \tag{29}$$
$$= \mathrm{tr}(\mathbf{W}\mathbf{R}\mathbf{W}^H\mathbf{W}\mathbf{R}\mathbf{W}^H) - \mathrm{tr}(\mathbf{W}\mathbf{R}\mathbf{W}^H\mathrm{ddiag}(\mathbf{W}\mathbf{R}\mathbf{W}^H)\,) \tag{30}$$

as an example and assume that $\mathbf{R}=\mathbf{R}^H$. Eq. (28) is a special case of $\mathcal{J}_2$ with $\mathbf{N}=\mathbf{0}$, $P=1$, and $\beta_1=1$. We expand $\mathcal{J}(\mathbf{W}+\delta\mathbf{Z})$ and then compare terms with the Taylor series expansion (10). The relations used for the derivation are summarized in Appendix B.

$$\begin{aligned}\mathcal{J}(\mathbf{W}+\delta\mathbf{Z}) = \mathcal{J}(\mathbf{W}) &+ 4\delta\,\mathfrak{Re}\{\mathrm{tr}(\mathbf{Z}^H\mathrm{off}(\mathbf{W}\mathbf{R}\mathbf{W}^H)\mathbf{W}\mathbf{R})\}\\ &+ 2\delta^2\,\mathrm{tr}(\mathbf{Z}^H\mathrm{off}(\mathbf{W}\mathbf{R}\mathbf{W}^H)\mathbf{Z}\mathbf{R} + \mathbf{Z}^H\mathrm{off}(\mathbf{Z}\mathbf{R}\mathbf{W}^H)\mathbf{W}\mathbf{R})\\ &+ 2\delta^2\,\mathfrak{Re}\{\mathrm{tr}(\mathbf{Z}\mathbf{R}\mathbf{W}^H\mathrm{off}(\mathbf{Z}\mathbf{R}\mathbf{W}^H))\} + O(\delta^3)\,.\end{aligned} \tag{31}$$

The latter three terms of (31) can be rewritten as

$$\mathrm{tr}(\mathbf{Z}^H\mathrm{off}(\mathbf{W}\mathbf{R}\mathbf{W}^H)\mathbf{Z}\mathbf{R})$$
$$= \mathrm{vec}(\,\mathbf{Z}\,)^H\,(\mathbf{R}^T\otimes\mathrm{off}(\mathbf{W}\mathbf{R}\mathbf{W}^H))\,\mathrm{vec}(\,\mathbf{Z}\,) \tag{32}$$
$$\mathrm{tr}(\mathbf{Z}^H\mathrm{off}(\mathbf{Z}\mathbf{R}\mathbf{W}^H)\mathbf{W}\mathbf{R})$$
$$= \mathrm{vec}(\,\mathbf{Z}\,)^H\,\mathrm{vec}(\,\mathrm{off}(\mathbf{Z}\mathbf{R}\mathbf{W}^H)\mathbf{W}\mathbf{R}\,) \tag{33}$$
$$= \mathrm{vec}(\,\mathbf{Z}\,)^H\,(\mathbf{R}^T\mathbf{W}^T\otimes\mathbf{I})\,\mathrm{vec}(\,\mathrm{off}(\mathbf{Z}\mathbf{R}\mathbf{W}^H)\,) \tag{34}$$
$$= \mathrm{vec}(\,\mathbf{Z}\,)^H\,(\mathbf{R}^T\mathbf{W}^T\otimes\mathbf{I})\mathbf{P}_{\text{off}}\,\mathrm{vec}(\,\mathbf{Z}\mathbf{R}\mathbf{W}^H\,) \tag{35}$$
$$= \mathrm{vec}(\,\mathbf{Z}\,)^H\,(\mathbf{R}^T\mathbf{W}^T\otimes\mathbf{I})\mathbf{P}_{\text{off}}(\mathbf{W}^*\mathbf{R}^*\otimes\mathbf{I})\,\mathrm{vec}(\,\mathbf{Z}\,) \tag{36}$$
$$\mathrm{tr}(\mathbf{Z}\mathbf{R}\mathbf{W}^H\mathrm{off}(\mathbf{Z}\mathbf{R}\mathbf{W}^H))$$
$$= \mathrm{vec}(\,\mathbf{Z}\,)^T\mathbf{P}_{\text{vec}}\,\mathrm{vec}(\,\mathbf{R}\mathbf{W}^H\,\mathrm{off}(\mathbf{Z}\mathbf{R}\mathbf{W}^H)\,) \tag{37}$$
$$= \mathrm{vec}(\,\mathbf{Z}\,)^T\mathbf{P}_{\text{vec}}(\mathbf{I}\otimes\mathbf{R}\mathbf{W}^H)\,\mathrm{vec}(\,\mathrm{off}(\mathbf{Z}\mathbf{R}\mathbf{W}^H)\,) \tag{38}$$
$$= \mathrm{vec}(\,\mathbf{Z}\,)^T\mathbf{P}_{\text{vec}}(\mathbf{I}\otimes\mathbf{R}\mathbf{W}^H)\mathbf{P}_{\text{off}}\,\mathrm{vec}(\,\mathbf{Z}\mathbf{R}\mathbf{W}^H\,) \tag{39}$$
$$= \mathrm{vec}(\,\mathbf{Z}\,)^T\mathbf{P}_{\text{vec}}(\mathbf{I}\otimes\mathbf{R}\mathbf{W}^H)\mathbf{P}_{\text{off}}(\mathbf{W}^*\mathbf{R}^*\otimes\mathbf{I})\,\mathrm{vec}(\,\mathbf{Z}\,) \tag{40}$$
$$= \mathrm{vec}(\,\mathbf{Z}\,)^T\,(\mathbf{R}\mathbf{W}^H\otimes\mathbf{I})\mathbf{P}_{\text{vec}}\mathbf{P}_{\text{off}}(\mathbf{W}^*\mathbf{R}^*\otimes\mathbf{I})\,\mathrm{vec}(\,\mathbf{Z}\,)\,. \tag{41}$$

After inserting (32), (36), and (41) into (31) and comparing terms with (10) we get

$$\mathbf{D}_{\mathbf{W}} = 4\,\mathrm{off}(\mathbf{W}\mathbf{R}\mathbf{W}^H)\mathbf{W}\mathbf{R} \tag{42}$$
$$\mathbf{H}_{\mathbf{W}} = 4(\mathbf{R}^T\otimes\mathrm{off}(\mathbf{W}\mathbf{R}\mathbf{W}^H)) + 4(\mathbf{R}^T\mathbf{W}^T\otimes\mathbf{I})\mathbf{P}_{\text{off}}(\mathbf{W}^*\mathbf{R}^*\otimes\mathbf{I}) \tag{43}$$
$$\mathbf{C}_{\mathbf{W}} = 4(\mathbf{R}\mathbf{W}^H\otimes\mathbf{I})\mathbf{P}_{\text{vec}}\mathbf{P}_{\text{off}}(\mathbf{W}^*\mathbf{R}^*\otimes\mathbf{I})\,, \tag{44}$$

which are equal to $\mathbf{D}_{\mathbf{W}2}$, $\mathbf{H}_{\mathbf{W}2}$, and $\mathbf{C}_{\mathbf{W}2}$ in Table 3 for $\mathbf{N}=\mathbf{0}$.

## B. USEFUL EQUALITIES FOR THE DERIVATION OF THE GRADIENT AND HESSIAN

The following equalities were very useful for the derivation of the gradient and Hessian. Equalities with the Frobenius norm and the trace function are [4]

$$\|\,\mathbf{A}\,\|_F^2 = \mathrm{tr}(\mathbf{A}\mathbf{A}^H) \tag{45}$$
$$\mathrm{off}(\,\mathbf{A}\,) = \mathbf{A} - \mathrm{ddiag}(\,\mathbf{A}\,) \tag{46}$$
$$\|\,\mathrm{off}(\,\mathbf{A}\,)\,\|_F^2 = \|\,\mathbf{A}\,\|_F^2 - \|\,\mathrm{ddiag}(\,\mathbf{A}\,)\,\|_F^2 \tag{47}$$
$$\mathrm{tr}\,(\mathbf{A}\mathbf{B}) = \mathrm{tr}\,(\mathbf{B}\mathbf{A}) \tag{48}$$
$$\mathrm{tr}\,(\mathbf{A}\,\mathrm{ddiag}(\,\mathbf{B}\,)) = \mathrm{tr}\,(\mathrm{ddiag}(\,\mathbf{A}\,)\,\mathbf{B}) \tag{49}$$
$$= \mathrm{tr}\,(\mathrm{ddiag}(\,\mathbf{A}\,)\,\mathrm{ddiag}(\,\mathbf{B}\,)) \tag{50}$$
$$\mathrm{tr}\,(\mathbf{A}\,\mathrm{off}(\,\mathbf{B}\,)) = \mathrm{tr}\,(\mathrm{off}(\,\mathbf{A}\,)\,\mathbf{B}) \tag{51}$$
$$= \mathrm{tr}\,(\mathrm{off}(\,\mathbf{A}\,)\,\mathrm{off}(\,\mathbf{B}\,)) \tag{52}$$
$$\mathrm{tr}\,(\mathrm{off}(\,\mathbf{A}\,)\,\mathrm{ddiag}(\,\mathbf{B}\,)) = 0\,. \tag{53}$$

Furthermore, we have some useful equalities with the $\mathrm{vec}(\,.\,)$ operation and Kronecker product [1]:

$$\mathbf{A}\mathbf{B}\otimes\mathbf{C}\mathbf{D} = (\mathbf{A}\otimes\mathbf{C})(\mathbf{B}\otimes\mathbf{D}) \tag{54}$$
$$\mathbf{A}\otimes\mathbf{B} = (\mathbf{A}\otimes\mathbf{I})(\mathbf{I}\otimes\mathbf{B}) \tag{55}$$
$$= (\mathbf{I}\otimes\mathbf{B})(\mathbf{A}\otimes\mathbf{I}) \tag{56}$$
$$= \mathbf{P}_{\text{vec}}(\mathbf{B}\otimes\mathbf{A})\mathbf{P}_{\text{vec}} \tag{57}$$
$$(\mathbf{A}\otimes\mathbf{B})^H = \mathbf{A}^H\otimes\mathbf{B}^H \tag{58}$$
$$\mathrm{vec}(\,\mathbf{A}\mathbf{D}\mathbf{B}\,) = (\mathbf{B}^T\otimes\mathbf{A})\,\mathrm{vec}(\,\mathbf{D}\,) \tag{59}$$
$$\mathrm{vec}(\,\mathbf{A}^T\,) = \mathbf{P}_{\text{vec}}\,\mathrm{vec}(\,\mathbf{A}\,) \tag{60}$$
$$\mathrm{tr}(\mathbf{A}^H\mathbf{B}) = \mathrm{vec}(\,\mathbf{A}\,)^H\,\mathrm{vec}(\,\mathbf{B}\,) \tag{61}$$
$$\mathrm{tr}\,(\mathbf{A}\mathbf{B}) = \mathrm{vec}(\,\mathbf{A}^T\,)^T\,\mathrm{vec}(\,\mathbf{B}\,) \tag{62}$$
$$= \mathrm{vec}(\,\mathbf{A}\,)^T\mathbf{P}_{\text{vec}}\,\mathrm{vec}(\,\mathbf{B}\,) \tag{63}$$
$$\mathrm{vec}(\,\mathrm{ddiag}(\,\mathbf{A}\,)\,) = \mathbf{P}_{\text{diag}}\,\mathrm{vec}(\,\mathbf{A}\,) \tag{64}$$
$$\mathrm{vec}(\,\mathrm{off}(\,\mathbf{A}\,)\,) = \mathbf{P}_{\text{off}}\,\mathrm{vec}(\,\mathbf{A}\,) \tag{65}$$

where $\mathbf{P}_{\text{vec}}$, $\mathbf{P}_{\text{diag}}$, and $\mathbf{P}_{\text{off}}$ are defined in Section 1.1.

## C. TRANSFORMATIONS BETWEEN MATRIX AND VECTOR FORM OF SECOND-ORDER TAYLOR APPROXIMATION

The *matrix form* of the second-order Taylor approximation (10) can be transformed into the *vector form* (11) with

$$\mathbf{w} = \begin{pmatrix}\mathbf{w}^{\text{re}}\\ \mathbf{w}^{\text{im}}\end{pmatrix} \triangleq \begin{pmatrix}\mathfrak{Re}\{\mathrm{vec}(\mathbf{W})\}\\ \mathfrak{Im}\{\mathrm{vec}(\mathbf{W})\}\end{pmatrix} \tag{66}$$
$$\mathbf{d}_{\mathbf{w}} = \begin{pmatrix}\mathbf{d}_{\mathbf{w}}^{\text{re}}\\ \mathbf{d}_{\mathbf{w}}^{\text{im}}\end{pmatrix} \triangleq \begin{pmatrix}\mathfrak{Re}\{\mathrm{vec}(\mathbf{D}_{\mathbf{W}})\}\\ \mathfrak{Im}\{\mathrm{vec}(\mathbf{D}_{\mathbf{W}})\}\end{pmatrix} \tag{67}$$
$$\mathbf{H}_{\mathbf{w}} = \begin{bmatrix}\mathfrak{Re}\{\mathbf{H}_{\mathbf{W}}+\mathbf{C}_{\mathbf{W}}\} & -\mathfrak{Im}\{\mathbf{H}_{\mathbf{W}}+\mathbf{C}_{\mathbf{W}}\}\\ \mathfrak{Im}\{\mathbf{H}_{\mathbf{W}}-\mathbf{C}_{\mathbf{W}}\} & \mathfrak{Re}\{\mathbf{H}_{\mathbf{W}}-\mathbf{C}_{\mathbf{W}}\}\end{bmatrix}. \tag{68}$$

The *vector form* of the second-order Taylor approximation (11) can be transformed into the *matrix form* (10) with

$$\mathbf{W} = [\mathbf{w}^{\text{re}} + j\mathbf{w}^{\text{im}}]^{\{M\times M\}} \tag{69}$$
$$\mathbf{D}_{\mathbf{W}} = [\mathbf{d}_{\mathbf{w}}^{\text{re}} + j\mathbf{d}_{\mathbf{w}}^{\text{im}}]^{\{M\times M\}} \tag{70}$$
$$\mathbf{H}_{\mathbf{W}} = \frac{1}{2}(\mathbf{H}_{\mathbf{w}11} + \mathbf{H}_{\mathbf{w}22} + j(\mathbf{H}_{\mathbf{w}21} - \mathbf{H}_{\mathbf{w}12})) \tag{71}$$
$$\mathbf{C}_{\mathbf{W}} = \frac{1}{2}(\mathbf{H}_{\mathbf{w}11} - \mathbf{H}_{\mathbf{w}22} - j(\mathbf{H}_{\mathbf{w}21} + \mathbf{H}_{\mathbf{w}12}))\,. \tag{72}$$

Here we used $\mathbf{W}^{M\times M} = [\mathbf{w}]^{\{M\times M\}}$ to denote the inverse operation of $\mathbf{w} = \mathrm{vec}(\,\mathbf{W}^{M\times M}\,)$. Furthermore, $\mathbf{H}_{\mathbf{w}} \triangleq [\mathbf{H}_{\mathbf{w}mn}]$.

## D. ACKNOWLEDGMENT

The authors would like to thank James Reilly, Jonathan Manton, and Phil Schniter for helpful discussions, and Jonathan Manton for providing his `cpoint` routine [5].

## E. REFERENCES

[1] J. W. Brewer, "Kronecker products and matrix calculus in system theory," *IEEE Trans. Circuits Syst.*, vol. 25, no. 9, pp. 772–781, Sept. 1978.

**Table 3**. Elementary cost functions with corresponding gradients and Hessian

| Objective | Elementary cost function $\mathcal{J}_i\,(\mathbf{W},\mathbf{N})$ | Gradient $\mathbf{D_W}$ and Hessian $\{\mathbf{H_W},\mathbf{C_W}\}$ |
|---|---|---|
| Diagonalization (Problem 1) | $\mathcal{J}_1 \triangleq \left\| \mathrm{off}(\mathbf{WRW}^H) \right\|_F^2$ | $\mathbf{D}_{\mathbf{W}\,1} = 2\,\mathrm{off}(\,\mathbf{WRW}^H\,)\mathbf{WR}^H + 2\,\mathrm{off}(\,\mathbf{WR}^H\mathbf{W}^H\,)\mathbf{WR}$<br>$\mathbf{H}_{\mathbf{W}\,1} = 2(\mathbf{R}^* \otimes \mathrm{off}(\,\mathbf{WRW}^H\,)) + 2(\mathbf{R}^T \otimes \mathrm{off}(\,\mathbf{WR}^H\mathbf{W}^H\,))$<br>$\quad + 2\Big(\mathbf{R}^T\mathbf{W}^T \otimes \mathbf{I}\Big)\mathbf{P}_{\mathrm{off}}\Big(\mathbf{W}^*\mathbf{R}^* \otimes \mathbf{I}\Big)$<br>$\quad + 2\Big(\mathbf{R}^*\mathbf{W}^T \otimes \mathbf{I}\Big)\mathbf{P}_{\mathrm{off}}\Big(\mathbf{W}^*\mathbf{R}^T \otimes \mathbf{I}\Big)$<br>$\mathbf{C}_{\mathbf{W}\,1} = 2\Big(\mathbf{RW}^H \otimes \mathbf{I}\Big)\mathbf{P}_{\mathrm{vec}}\,\mathbf{P}_{\mathrm{off}}\Big(\mathbf{W}^*\mathbf{R}^* \otimes \mathbf{I}\Big)$<br>$\quad + 2\Big(\mathbf{R}^H\mathbf{W}^H \otimes \mathbf{I}\Big)\mathbf{P}_{\mathrm{off}}\,\mathbf{P}_{\mathrm{vec}}\Big(\mathbf{W}^*\mathbf{R}^T \otimes \mathbf{I}\Big)$ |
| Diagonalization (Problem 2)<br>(note: $\mathbf{R}^H = \mathbf{R}$) | $\mathcal{J}_2 \triangleq \left\| \mathrm{off}\,(\mathbf{W}(\mathbf{R}-\mathbf{N})\mathbf{W}^H) \right\|_F^2$ | $\mathbf{D}_{\mathbf{W}\,2} = 4\,\mathrm{off}\,(\mathbf{W}(\mathbf{R}-\mathbf{N})\mathbf{W}^H)\mathbf{W}(\mathbf{R}-\mathbf{N})$<br>$\mathbf{H}_{\mathbf{W}\,2} = 4\,\Big((\mathbf{R}-\mathbf{N})^* \otimes \mathrm{off}\,(\mathbf{W}(\mathbf{R}-\mathbf{N})\mathbf{W}^H)\Big)$<br>$\quad + 4\,\Big((\mathbf{R}-\mathbf{N})^T\mathbf{W}^T \otimes \mathbf{I}\Big)\,\mathbf{P}_{\mathrm{off}}\Big(\mathbf{W}^*(\mathbf{R}-\mathbf{N})^* \otimes \mathbf{I}\Big)$<br>$\mathbf{C}_{\mathbf{W}\,2} = 4\,\Big((\mathbf{R}-\mathbf{N})\mathbf{W}^H \otimes \mathbf{I}\Big)\,\mathbf{P}_{\mathrm{vec}}\,\mathbf{P}_{\mathrm{off}}\Big(\mathbf{W}^*(\mathbf{R}-\mathbf{N})^* \otimes \mathbf{I}\Big)$<br>$\mathbf{D}_{\mathbf{N}\,2} = -2\mathbf{W}^H\,\mathrm{off}\,(\mathbf{W}\,(\mathbf{R}-\mathbf{N})\,\mathbf{W}^H)\mathbf{W}$<br>$\mathbf{H}_{\mathbf{N}\,2} = 2(\mathbf{W}^T \otimes \mathbf{W}^H)\mathbf{P}_{\mathrm{off}}(\mathbf{W}^* \otimes \mathbf{W})$<br>$\mathbf{C}_{\mathbf{N}\,2} = \mathbf{0}$ |
| unitary $\mathbf{W}$ | $\mathcal{J}_3 \triangleq \left\| \mathbf{WW}^H - \mathbf{I} \right\|_F^2$ | $\mathbf{D}_{\mathbf{W}\,3} = 4(\mathbf{WW}^H - \mathbf{I})\mathbf{W} = 4\,\mathbf{W}(\mathbf{W}^H\mathbf{W} - \mathbf{I})$<br>$\mathbf{H}_{\mathbf{W}\,3} = 4\Big(\mathbf{I} \otimes \mathbf{WW}^H + \mathbf{W}^T\mathbf{W}^* \otimes \mathbf{I} - \mathbf{I} \otimes \mathbf{I}\Big)$<br>$\mathbf{C}_{\mathbf{W}\,3} = 2\mathbf{P}_{\mathrm{vec}}(\mathbf{W}^* \otimes \mathbf{W}^H) + 2(\mathbf{W}^H \otimes \mathbf{W}^*)\mathbf{P}_{\mathrm{vec}}$ |
| $\mathrm{ddiag}(\,\mathbf{W}\,) = \mathbf{I}$ | $\mathcal{J}_4 \triangleq \left\| \mathrm{ddiag}\,(\mathbf{W}-\mathbf{I}) \right\|_F^2$ | $\mathbf{D}_{\mathbf{W}\,4} = 2\,\mathrm{ddiag}(\,\mathbf{W}-\mathbf{I}\,)$<br>$\mathbf{H}_{\mathbf{W}\,4} = 2\,\mathbf{P}_{\mathrm{diag}}$<br>$\mathbf{C}_{\mathbf{W}\,4} = \mathbf{0}$ |

[2] M. Joho and H. Mathis, "Joint diagonalization of correlation matrices by using gradient methods with application to blind signal separation," in *Proc. SAM*, Rosslyn, VA, Aug. 4–6, 2002, pp. 273–277.

[3] D. G. Luenberger, *Linear and Nonlinear Programming*, Addison-Wesley, 2nd edition, 1989.

[4] M. Joho, R. H. Lambert, and H. Mathis, "Elementary cost functions for blind separation of non-stationary source signals," in *Proc. ICASSP*, Salt Lake City, UT, May 7–11, 2001, vol. 5, pp. 2793–2796.

[5] J. H. Manton, "Optimisation algorithms exploiting unitary constraints," *IEEE Trans. Signal Processing*, vol. 50, no. 3, pp. 635–650, Mar. 2002.

[6] R. Fletcher, *Practical Methods of Optimization*, John Wiley & Sons, 2nd edition, 1987.

[7] R. A. Horn and C. R. Johnson, *Matrix Analysis*, Cambridge University Press, 1990.

[8] M. Nikpour, J. H. Manton, and G. Hori, "Algorithms on the Stiefel manifold for joint diagonalisation," in *Proc. ICASSP*, Orlando, FL, May 13–17, 2002, vol. 2, pp. 1481–1484.

[9] J.-F. Cardoso and A. Souloumiac, "Blind beamforming for non Gaussian signals," *IEE Proceedings-F*, vol. 140, no. 6, pp. 362–370, Dec. 1993.

[10] 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, no. 2, pp. 434–444, Feb. 1997.

[11] L. Parra and C. Spence, "Convolutive blind separation of non-stationary sources," *IEEE Trans. Speech and Audio Processing*, vol. 8, no. 3, pp. 320–327, 2000.

[12] S. Choi, A. Cichocki, and A. Belouchrani, "Blind separation of second-order nonstationary and temporally colored sources," in *Proc. SSP*, Singapore, Aug. 6–8, 2001.

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

[14] M. Wax and J. Sheinvald, "A least-squares approach to joint diagonalization," *IEEE Signal Processing Lett.*, vol. 4, no. 2, pp. 52–53, Feb. 1997.

[15] A.-J. van der Veen, "Joint diagonalization via subspace fitting techniques," in *Proc. ICASSP*, Salt Lake City, UT, May 7–11, 2001.

[16] K. Rahbar and J. P. Reilly, "Blind source separation algorithm for MIMO convolutive mixtures," in *Proc. ICA*, San Diego, CA, Dec. 9–12, 2001, pp. 224–229.

[17] K. Rahbar, J. P. Reilly, and J. H. Manton, "A frequency domain approach to blind identification of mimo fir systems driven by quasi-stationary signals," in *Proc. ICASSP*, Orlando, FL, May 13–17, 2002, vol. 2, pp. 1717–1720.

[18] D. T. Pham, "Joint approximate diagonalization of positive definite hermitian matrices," *SIAM J. Matrix Anal. and Appl.*, vol. 22, no. 4, pp. 1136–1152, 2001.

[19] A. Yeredor, "Non-orthogonal joint diagonalization in the least-squares sense with application in blind source separation," *IEEE Trans. Signal Processing*, vol. 50, no. 7, pp. 1545–1553, July 2002.