SAM 2002, Aug. 4-6, Rosslyn, VA, USA, pp. 273-277.

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

Marcel Joho[1], *Heinz Mathis*[2]

[1]Phonak Inc., 1901 South First Street, Champaign, IL, USA
[2]University of Applied Sciences, Rapperswil, Switzerland
joho@ieee.org, *heinz.mathis@hsr.ch*

ABSTRACT

Joint diagonalization of several correlation matrices is a powerful tool for blind signal separation. This paper addresses the blind signal separation problem for the case where the source signals are non-stationary and / or non-white, and the sensors are possibly noisy. We present cost functions for jointly diagonalizing several correlation matrices. The corresponding gradients are derived and used in a gradient-based joint-diagonalization algorithms. Several variations are given, depending on desired properties of the separation matrix, e.g., unitary separation matrix. These constraints are either imposed by adding a penalty term to the cost function or by projecting the gradient onto the desired manifold. The performance of the proposed joint-diagonalization algorithm is verified by simulating a blind signal separation application.

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 $\mathbf{0}$. $E\{.\}$ denotes the expectation operator. Vector or matrix dimensions are given in superscript. The Frobenius norm and the trace of a matrix are denoted by $\|.\|_F$ and $\mathrm{tr}(.)$, respectively ($\|\mathbf{A}\|_F^2 \triangleq \mathrm{tr}(\mathbf{A}\mathbf{A}^H)$). $\mathbf{a} = \mathrm{diag}(\mathbf{A})$ is 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 $\mathbf{a}$. $\mathrm{ddiag}(\mathbf{A})$ zeros all off-diagonal elements of $\mathbf{A}$ and

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

zeros the diagonal elements of $\mathbf{A}$. For a square matrix $\mathbf{A}$ we have $\mathrm{ddiag}(\mathbf{A}) \triangleq \mathrm{diag}(\mathrm{diag}(\mathbf{A}))$.

1.2. Outline

In Section 1.3 we first describe two joint-diagonalization problems in a pure mathematical way. Gradient-based algorithms are derived then in Section 2 and Section 3. In Section 4 we describe the blind signal separation (BSS) problem and show the relationship to the joint-diagonalization problems defined in Section 1.3. Finally, in Section 5 we give a simulation example to demonstrate the behavior of different gradient-based algorithms.

1.3. Problem definition

We define the following two problems:

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:

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

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

$$\sum_{p=1}^{P} \beta_p \| \mathbf{R}_p \|_F^2 = 1 . \tag{3}$$

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 cost function

$$\mathcal{J}_2 \triangleq \sum_{p=1}^{P} \beta_p \left\| \mathrm{off}\left(\mathbf{W}(\mathbf{R}_p - \mathbf{N})\mathbf{W}^H \right) \right\|_F^2 . \tag{4}$$

As in Problem 1, we require again that the weights $\{\beta_p\}$ are normalized such that (3) is fulfilled.

1.4. Comments

The cost function (2) is minimized if $\{\mathbf{W}\mathbf{R}_p\mathbf{W}^H\}_{p=1}^P$ becomes a set of diagonal matrices. Perfect joint diagonalization is normally not possible for an arbitrary set of positive definite Hermitian 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, (2) 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 semi-definite diagonal matrix.

The purpose of choosing the normalization in (3) is to make the cost functions (2) and (4) *independent* of the absolute norms $\{\| \mathbf{R}_p \|_F\}$, which is a very nice property, especially when real-world signals are applied. Note, if $\{\{\mathbf{R}_p\}, \{\beta_p\}'\}$ do not fulfill (3), we can always find a scaled set of weighting factors $\{\beta_p\} = \{\beta_p'/\gamma\}$ with $\gamma = \sum_{p=1}^P \beta_p' \| \mathbf{R}_p \|_F^2$ such that (3) is fulfilled.

Note, $\mathbf{W} = \mathbf{0}$ minimizes (2) and (4). Therefore, to make the two defined problems meaningful, we require some additional properties of $\mathbf{W}$ to prevent the trivial solution, e.g., $\mathbf{W}$ should be unitary. Basically, there are two ways to incorporate such requirements into an optimization: (i) by imposing a hard constraint into

the optimization, or, (ii) by adding an additional *penalty term* to the main cost function [1, 2, 3]. Possible cost functions for a penalty term are

$$\mathcal{J}_3 \triangleq \left\| \mathbf{W}\mathbf{W}^H - \mathbf{I} \right\|_F^2 \tag{5}$$

$$\mathcal{J}_4 \triangleq \| \operatorname{ddiag}(\mathbf{W} - \mathbf{I}) \|_F^2 \ . \tag{6}$$

$\mathcal{J}_3$ penalizes the deviation of $\mathbf{W}$ of being a unitary matrix and $\mathcal{J}_4$ is minimal if the diagonal elements of $\mathbf{W}$ are one. Other choices of penalty terms are listed in [4].

2. JOINT DIAGONALIZATION BY USING GRADIENT METHODS

First we want to tackle Problem 1 and search for a gradient method. To this end we need a well defined overall cost function $\mathcal{J}(\mathbf{W})$ which describes the final objective in a mathematical manner. We then compute the corresponding gradient with respect to the unknown variable, $\nabla_{\mathbf{W}}\mathcal{J}(\mathbf{W})$, and use it for the following negative-gradient search (steepest descent) algorithm

$$\mathbf{W}_{k+1} = \mathbf{W}_k + \Delta\mathbf{W}_k \tag{7}$$

$$\Delta\mathbf{W}_k = -\mu \nabla_{\mathbf{W}}\mathcal{J}(\mathbf{W}_k) \tag{8}$$

where μ is a step size to control the adaptation rate and $\Delta\mathbf{W}_k$ is the incremental update of $\mathbf{W}_k$.

2.1. Minimizing $\mathcal{J}_1$ with a unitary matrix by using a penalty term

As previously mentioned, using the cost function $\mathcal{J}_1$ does not sufficiently describe the objective of the joint diagonalization. We therefore have to incorporate an additional term that prevents the trivial solution $\mathbf{W} = \mathbf{0}$. If we aim at finding a unitary matrix that jointly diagonalizes a set of matrix, we can use the following cost function:

$$\mathcal{J}_{\{1,3\}} \triangleq \mathcal{J}_1 + \alpha_3 \mathcal{J}_3 \tag{9}$$

where $\mathcal{J}_1$ and $\mathcal{J}_3$ are defined in (2) and (5), respectively. With α_3 we can weight the penalty term $\mathcal{J}_3$. Therefore, α_3 has an influence on the adaptation trajectory, and also on the final solution in the case where there does not exist a true unitary matrix that jointly diagonalizes $\{\mathbf{R}_p\}$.

To find the minima of $\mathcal{J}_{\{1,3\}}$ with the algorithm (7) and (8), we compute the gradient

$$\nabla_{\mathbf{W}}\mathcal{J}_{\{1,3\}} = \nabla_{\mathbf{W}}\mathcal{J}_1 + \alpha_3 \nabla_{\mathbf{W}}\mathcal{J}_3 \tag{10}$$

where the individual gradients are [4]

$$\nabla_{\mathbf{W}}\mathcal{J}_1 = 2\sum_{p=1}^{P} \beta_p \operatorname{off}\left(\mathbf{W}_k \mathbf{R}_p \mathbf{W}_k^H \right) \mathbf{W}_k \mathbf{R}_p^H + 2\sum_{p=1}^{P} \beta_p \operatorname{off}\left(\mathbf{W}_k \mathbf{R}_p^H \mathbf{W}_k^H \right) \mathbf{W}_k \mathbf{R}_p \tag{11}$$

$$\nabla_{\mathbf{W}}\mathcal{J}_3 = 4(\mathbf{W}_k \mathbf{W}_k^H - \mathbf{I})\mathbf{W}_k \, . \tag{12}$$

As we see from (10), the weight α_3 has also an influence on the gradient, and therefore the search direction of the algorithm. Choosing a large α_3 prevents $\mathbf{W}_k$ from deviating too much from a unitary matrix during the adaptation process.

2.2. Minimizing $\mathcal{J}_1$ with a unitary matrix by applying projections

Alternatively to adding a penalty term to the cost function $\mathcal{J}_1$, we can apply a projection operation of the gradient $\nabla_{\mathbf{W}}\mathcal{J}_1$ such that the unitary property of $\mathbf{W}_k$ is approximately preserved within an update step. To this end, the gradient $\nabla_{\mathbf{W}}\mathcal{J}_1$ in (11) is modified as [5, 6]

$$\widetilde{\nabla}_{\mathbf{W}}\mathcal{J}_1 \triangleq \nabla_{\mathbf{W}}\mathcal{J}_1 - \mathbf{W}_k (\nabla_{\mathbf{W}}\mathcal{J}_1)^H \mathbf{W}_k \tag{13}$$

where $\widetilde{\nabla}_{\mathbf{W}}\mathcal{J}_1$ is the tangent gradient of $\mathcal{J}_1$ with respect to $\mathbf{W}_k$ on the *complex Stiefel manifold*, the space of unitary matrices. Hence, the use of $\Delta\mathbf{W}_k = -\mu \widetilde{\nabla}_{\mathbf{W}}\mathcal{J}(\mathbf{W}_k)$ instead of (8) as the incremental update in (7) maintains approximately the unitary property of $\mathbf{W}_{k+1}$ if $\mathbf{W}_k$ was already unitary. Thus, we start with a unitary matrix, e.g., $\mathbf{W}_0 = \mathbf{I}$. However, simulations have shown that it still is advantageous to incorporate a penalty term $\alpha_3\mathcal{J}_3$ with a small α_3 into the overall cost function $\mathcal{J}$, to prevent the algorithm from becoming unstable.

We can also constrain $\mathbf{W}_{k+1}$ to be unitary after every update step. We therefore define $\mathbf{W}'_{k+1} = \mathbf{W}_k + \Delta\mathbf{W}_k$ where $\Delta\mathbf{W}_k$ is the unconstraint incremental update defined in (8). Afterwards we compute the SVD (singular value decomposition) $\mathbf{W}'_{k+1} = \mathbf{U}_{k+1}\mathbf{\Sigma}_{k+1}\mathbf{V}_{k+1}^H$. Then by choosing $\mathbf{W}_{k+1} = \mathbf{U}_{k+1}\mathbf{V}_{k+1}^H = \pi(\mathbf{W}'_{k+1})$ we get the closest unitary matrix to $\mathbf{W}'_{k+1}$ in the sense that $\| \mathbf{W}'_{k+1} - \mathbf{W}_{k+1} \|_F^2$ is minimal among all unitary matrices [6], [7, p.429-30]. We denote the projection of a matrix $\mathbf{W}$ on the Stiefel manifold as $\pi(\mathbf{W}) = \pi(\mathbf{U}\mathbf{\Sigma}\mathbf{V}^H) \triangleq \mathbf{U}\mathbf{V}^H$ where $\mathbf{W} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^H$ is the SVD of $\mathbf{W}$.

The technique of updating a matrix within the Stiefel manifold was applied in blind signal separation by Douglas [8], Rahbar and Reilly [9], and Choi *et al.* [10] for the real case, and recently by Rahbar and Reilly [11] for the complex case. A very elegant method for solving Problem 1 under the unitary constraint was presented by Cardoso and Souloumiac in [12, 13]. Further insight on the joint-diagonalization problem under the unitary constraint was given by Wax and Sheinvald in [14].

2.3. Joint diagonalization using non-unitary matrices

Sometimes it is desirable to search for a non-unitary matrix for the joint-diagonalization task, as we might wish to demand other specific properties of $\mathbf{W}$, see [15, 16, 17, 18, 19]. For example, if we wish to constrain the diagonal elements of $\mathbf{W}$ to be one, we can use the penalty term $\mathcal{J}_4$ defined in (6) in addition to $\mathcal{J}_1$ in the overall cost function, i.e., $\mathcal{J}_{\{1,4\}} \triangleq \mathcal{J}_1 + \alpha_4\mathcal{J}_4$, where α_4 is a weighting factor. Furthermore, the corresponding gradient

$$\nabla_{\mathbf{W}}\mathcal{J}_4 = 2 \operatorname{ddiag}(\mathbf{W} - \mathbf{I}) \tag{14}$$

is used for the computation of $\nabla_{\mathbf{W}}\mathcal{J}_{\{1,4\}} = \nabla_{\mathbf{W}}\mathcal{J}_1 + \alpha_4 \nabla_{\mathbf{W}}\mathcal{J}_4$, where $\nabla_{\mathbf{W}}\mathcal{J}_1$ is defined in (11). A modified version of $\nabla_{\mathbf{W}}\mathcal{J}_4$ is

$$\breve{\nabla}_{\mathbf{W}}\mathcal{J}_4 = 2 \operatorname{ddiag}(\mathbf{W} - \mathbf{I})\, \mathbf{W} \tag{15}$$

which increases / decreases the elements of $\mathbf{W}$ in a more uniform way if the diagonal elements of $\mathbf{W}_k$ deviate from unity, as not only the diagonal elements of $\mathbf{W}_k$ are adapted. However, in this particular case it is easier to impose a hard constraint on the update of $\mathbf{W}_{k+1}$. Using

$$\bar{\nabla}_{\mathbf{W}}\mathcal{J}_1 \triangleq \operatorname{off}(\nabla_{\mathbf{W}}\mathcal{J}_1) \tag{16}$$

instead of $\nabla_{\mathbf{W}}\mathcal{J}_1$ for updating $\mathbf{W}_k$ leaves the diagonal elements unchanged and updates only the off-diagonal elements of $\mathbf{W}_k$. We therefore initialize $\mathbf{W}_0 = \mathbf{I}$. The choice between (14), (15), and (16) affects the adaptation trajectory of the algorithm and therefore reveals a different convergence behavior.

3. EXTENSION TO PROBLEM 2

We now want to investigate Problem 2. Since the cost function (4) incorporates two unknowns, $\mathbf{W}$ and $\mathbf{N}$, we not only update $\mathbf{W}$ with (7) and (8), but also $\mathbf{N}$ with

$$\mathbf{N}_{k+1} = \mathbf{N}_k + \Delta\mathbf{N}_k \tag{17}$$

$$\Delta\mathbf{N}_k = -\eta\ \mathrm{ddiag}(\ \nabla_{\mathbf{N}}\mathcal{J}(\mathbf{N}_k)\ )\ . \tag{18}$$

Since we constrain $\mathbf{N}$ to be diagonal by definition, in fact, we only have to update the diagonal elements. Thus, we applied $\mathrm{ddiag}(\ .\ )$ in (18) to set all off-diagonal elements of $\nabla_{\mathbf{N}}\mathcal{J}(\mathbf{N}_k)$ to zero. Consequently, we have to choose the initial matrix $\mathbf{N}_0$ to be diagonal.

The gradients of $\mathcal{J}_2$ with respect to both unknowns are [19, 4]

$$\nabla_{\mathbf{W}}\mathcal{J}_2 = 4\sum_{p=1}^{P}\beta_p\ \mathrm{off}\Big(\ \mathbf{W}_k\left(\mathbf{R}_p - \mathbf{N}_k\right)\mathbf{W}_k^H\ \Big)\ \mathbf{W}_k\left(\mathbf{R}_p - \mathbf{N}_k\right) \tag{19}$$

$$\nabla_{\mathbf{N}}\mathcal{J}_2 = -2\sum_{p=1}^{P}\beta_p\mathbf{W}_k^H\ \mathrm{off}\Big(\ \mathbf{W}_k\left(\mathbf{R}_p - \mathbf{N}_k\right)\mathbf{W}_k^H\ \Big)\ \mathbf{W}_k\,. \tag{20}$$

For (19) we used $\mathbf{R}_p^H = \mathbf{R}_p$ and $\mathbf{N}_k^H = \mathbf{N}_k$, since $\mathbf{R}_p$ and $\mathbf{N}$ are assumed to be Hermitian in Problem 2. Likewise to Problem 1, we can proceed the same way and define

$$\mathcal{J}_{\{2,3\}} \triangleq \mathcal{J}_2 + \alpha_3\mathcal{J}_3 \tag{21}$$

$$\mathcal{J}_{\{2,4\}} \triangleq \mathcal{J}_2 + \alpha_4\mathcal{J}_4 \tag{22}$$

or use

$$\widetilde{\nabla}_{\mathbf{W}}\mathcal{J}_2 \triangleq \nabla_{\mathbf{W}}\mathcal{J}_2 - \mathbf{W}_k(\nabla_{\mathbf{W}}\mathcal{J}_2)^H\mathbf{W}_k \tag{23}$$

in the case where we want to maintain approximately the unitary property of $\mathbf{W}_k$ during the adaptation.

Note, the update of $\mathbf{N}_k$ is influenced neither by the penalty term $\mathcal{J}_3$ nor by $\mathcal{J}_4$ directly, as they are independent of $\mathbf{N}$.

4. JOINT DIAGONALIZATION TECHNIQUES FOR BLIND SIGNAL SEPARATION

4.1. Description of the mixing model

In a blind signal separation setup we assume the following mixing model:

$$\mathbf{x}_t = \mathbf{A}\mathbf{s}_t + \mathbf{n}_t \tag{24}$$

where $\mathbf{s}_t \triangleq (s_1, \ldots, s_M)_t^T$, $\mathbf{x}_t \triangleq (x_1, \ldots, x_M)_t^T$, and $\mathbf{n}_t \triangleq (n_1, \ldots, n_M)_t^T$ contain the samples of the unknown source signals, the sensor signals, and the sensor noise at time t, respectively. $\mathbf{A}^{M\times M}$ is the unknown mixing matrix. In (24) only $\{\mathbf{x}_t\}$ is known. The separation process is described as

$$\mathbf{u}_t = \mathbf{W}\mathbf{x}_t = \mathbf{W}(\mathbf{A}\mathbf{s}_t + \mathbf{n}_t) = \mathbf{G}\mathbf{s}_t + \mathbf{W}\mathbf{n}_t \tag{25}$$

where $\mathbf{W}^{M\times M}$ is a separation matrix such that $\mathbf{u}_t$ becomes a waveform-preserving estimate of $\mathbf{s}_t$, up to scaling and permutation of the elements. In the blind signal separation problem, we aim at estimating $\{\mathbf{s}_t\}$ by knowing only $\{\mathbf{x}_t\}$ and some statistics of $\{\mathbf{s}_t\}$, e.g., non-stationarity, non-whiteness or non-Gaussianity. $\mathbf{G} \triangleq \mathbf{W}\mathbf{A}$ is the total transfer matrix of the global system.

In the following, we assume non-stationary and / or non-white source signals [20, 21, 22, 23], stationary white noise signals, and for all source and noise signals mutual independence. Therefore

$$\mathbf{R}_{\mathbf{ss}}(t,\tau) \triangleq E\{\mathbf{s}_t\mathbf{s}_{t-\tau}^H\} \tag{26}$$

$$\mathbf{R}_{\mathbf{nn}}(t,\tau) \triangleq E\{\mathbf{n}_t\mathbf{n}_{t-\tau}^H\} = \delta(\tau)\mathbf{R}_{\mathbf{nn}} \tag{27}$$

$$\mathbf{R}_{\mathbf{sn}}(t,\tau) \triangleq E\{\mathbf{s}_t\mathbf{n}_{t-\tau}^H\} = \mathbf{0} \quad \forall t,\tau \tag{28}$$

where $\mathbf{R}_{\mathbf{ss}}(t,\tau)$ are diagonal matrices $\forall t,\tau$, and $\mathbf{R}_{\mathbf{ss}}(t,0)$ and $\mathbf{R}_{\mathbf{nn}}$ are real positive semi-definite diagonal matrices. Furthermore, from (24) and (25), we have

$$\mathbf{R}_{\mathbf{xx}}(t,\tau) \triangleq E\{\mathbf{x}_t\mathbf{x}_{t-\tau}^H\} = \mathbf{A}\mathbf{R}_{\mathbf{ss}}(t,\tau)\mathbf{A}^H + \delta(\tau)\mathbf{R}_{\mathbf{nn}} \tag{29}$$

$$\mathbf{R}_{\mathbf{uu}}(t,\tau) \triangleq E\{\mathbf{u}_t\mathbf{u}_{t-\tau}^H\} = \mathbf{W}\mathbf{R}_{\mathbf{xx}}(t,\tau)\mathbf{W}^H\,. \tag{30}$$

4.2. Joint diagonalization of correlation matrices

In the following blind signal separation problem, we will focus on finding a so-called *zero-forcing* solution for $\mathbf{W}$ such that $\mathbf{G}$ becomes close to a scaled permutation matrix. This is equivalent to minimizing the output *interchannel interference* (ICI), regardless of a possible noise amplification by $\mathbf{W}$.

Let $\{\mathbf{R}_{\mathbf{xx}\,p}(\tau)\} \triangleq \{\mathbf{R}_{\mathbf{xx}}(t_p,\tau)\}$ be a given set of correlation matrices which stem from a noisy mixture of non-stationary and non-white source signals, as defined in (24). Let t_p denote the pth time sample. We then propose the following cost function for finding a zero-forcing separation matrix $\mathbf{W}$:

$$\mathcal{J}_5 \triangleq \sum_{\tau}\sum_{p}\beta_{p,\tau}\ \Big\|\ \mathrm{off}\Big(\ \mathbf{W}\ \Big(\mathbf{R}_{\mathbf{xx}\,p}(\tau) - \delta(\tau)\hat{\mathbf{R}}_{\mathbf{nn}}\Big)\ \mathbf{W}^H\ \Big)\ \Big\|_F^2 \tag{31}$$

$$= \sum_{p}\beta_{p,0}\ \Big\|\ \mathrm{off}\Big(\ \mathbf{W}\ \Big(\mathbf{R}_{\mathbf{xx}\,p}(0) - \hat{\mathbf{R}}_{\mathbf{nn}}\Big)\ \mathbf{W}^H\ \Big)\ \Big\|_F^2 + \sum_{\tau\neq 0}\sum_{p}\beta_{p,\tau}\ \Big\|\ \mathrm{off}\Big(\ \mathbf{W}\mathbf{R}_{\mathbf{xx}\,p}(\tau)\mathbf{W}^H\ \Big)\ \Big\|_F^2\,. \tag{32}$$

To make the cost function independent of the absolute signal powers, we normalize the weights $\beta_{p,\tau}$ such that

$$\sum_{\tau}\sum_{p}\beta_{p,\tau}\ \ \|\ \mathbf{R}_{\mathbf{xx}\,p}(\tau)\ \|_F^2 = 1\,. \tag{33}$$

Note that the left and right sum of (32) have *the same form* as $\mathcal{J}_1$ and $\mathcal{J}_2$, respectively. The corresponding relationship is $\mathbf{R}_{\mathbf{xx}\,p} \sim \mathbf{R}_p$ and $\mathbf{R}_{\mathbf{nn}} \sim \mathbf{N}$. Hence, by analogy, the derivation of the gradients $\nabla_{\mathbf{W}}\mathcal{J}_5$ and $\nabla_{\hat{\mathbf{R}}_{\mathbf{nn}}}\mathcal{J}_5$ is straightforward. Note, in the complex case we usually have $\mathbf{R}_{\mathbf{xx}\,p}^H(\tau) \neq \mathbf{R}_{\mathbf{xx}\,p}(\tau)$ for $\tau \neq 0$. Thus, $\mathbf{R}_{\mathbf{xx}\,p}(\tau)$ *is not Hermitian* in general for $\tau \neq 0$. Following the same steps as in [4] and using $\mathbf{R}_{\mathbf{xx}\,p}^H(0) = \mathbf{R}_{\mathbf{xx}\,p}(0)$ and

$\hat{\mathbf{R}}^H_{\mathbf{nn}_k} = \hat{\mathbf{R}}_{\mathbf{nn}_k}$ we get

$$\begin{aligned}\nabla_{\mathbf{W}}\mathcal{J}_5 = 4\sum_p \beta_{p,0}\,\mathrm{off}\Big(\mathbf{W}_k\left(\mathbf{R}_{\mathbf{xx}\,p}(0) - \hat{\mathbf{R}}_{\mathbf{nn}_k}\right)\mathbf{W}_k^H\Big) \\ \cdot \mathbf{W}_k\left(\mathbf{R}_{\mathbf{xx}\,p}(0) - \hat{\mathbf{R}}_{\mathbf{nn}_k}\right) \\ + 2\sum_{\tau\neq 0}\sum_p \beta_{p,\tau}\,\mathrm{off}\Big(\mathbf{W}_k\mathbf{R}_{\mathbf{xx}\,p}(\tau)\mathbf{W}_k^H\Big)\mathbf{W}_k\mathbf{R}^H_{\mathbf{xx}\,p}(\tau) \\ + 2\sum_{\tau\neq 0}\sum_p \beta_{p,\tau}\,\mathrm{off}\Big(\mathbf{W}_k\mathbf{R}^H_{\mathbf{xx}\,p}(\tau)\mathbf{W}_k^H\Big)\mathbf{W}_k\mathbf{R}_{\mathbf{xx}\,p}(\tau)\end{aligned} \tag{34}$$

$$\begin{aligned}\nabla_{\hat{\mathbf{R}}_{\mathbf{nn}}}\mathcal{J}_5 = -2\sum_p \beta_{p,0}\;\mathbf{W}_k^H \\ \cdot \mathrm{off}\Big(\mathbf{W}_k\left(\mathbf{R}_{\mathbf{xx}\,p}(0) - \hat{\mathbf{R}}_{\mathbf{nn}_k}\right)\mathbf{W}_k^H\Big)\mathbf{W}_k\,.\end{aligned} \tag{35}$$

Since the trivial solution $\mathbf{W} = \mathbf{0}$ also minimizes $\mathcal{J}_5$, again, we can add a penalty term to $\mathcal{J}_5$, e.g., $\mathcal{J}_{\{5,3\}} \triangleq \mathcal{J}_5 + \alpha_3\mathcal{J}_3$ or $\mathcal{J}_{\{5,4\}} \triangleq \mathcal{J}_5 + \alpha_4\mathcal{J}_4$. After computing the corresponding gradients, e.g., $\nabla_{\mathbf{W}}\mathcal{J}_{\{5,3\}}$ and $\nabla_{\mathbf{R}_{\mathbf{nn}}}\mathcal{J}_{\{5,3\}}$, we can use a gradient-based method to update $\{\mathbf{W}_k\}$ and $\{\hat{\mathbf{R}}_{\mathbf{nn}_k}\}$. A useful method for finding an initial value $\mathbf{W}_0$ is described in Appendix A.

In case we constrain $\mathbf{W}$ to be unitary, we can also use a projected gradient similar to (23)

$$\widetilde{\nabla}_{\mathbf{W}}\mathcal{J}_5 \triangleq \nabla_{\mathbf{W}}\mathcal{J}_5 - \mathbf{W}_k(\nabla_{\mathbf{W}}\mathcal{J}_5)^H\mathbf{W}_k\,. \tag{36}$$

Joint diagonalization of correlation matrices under the unitary constraint is usually applied after a PCA stage (principal component analysis) [24, 25].

With (31) we have the following two special cases: if the source signals are (i) non-stationary but white, then we can set $\beta_{p,\tau} = 0 \;\; \forall\tau \neq 0$, (ii) non-white but stationary , then we can set $\beta_{p,\tau} = 0 \;\; \forall p > 1$.

5. SIMULATION

In the following, we give a simulation example to analyze the behavior of a proposed algorithm. We generate a set of $P = 15$ correlation matrices where $\{\mathbf{R}_p\} = \{\mathbf{A}\mathbf{\Lambda}_p\mathbf{A}^H + \mathbf{\Sigma}\}$, $\mathbf{A}^{5\times 5}$ is a random unitary matrix, $\mathbf{\Sigma} = \mathrm{diag}(\,0.1, 0.2, 0.3, 0.4, 0.5\,)$, and $\{\mathbf{\Lambda}_p\}$ are randomly chosen diagonal matrices whose elements are in the range $[0, 1]$. Our objective is to find a unitary matrix $\mathbf{W}$ and a diagonal matrix $\mathbf{N}$ that minimize $\mathcal{J}_2$ defined in (4).

We compare three different algorithms: (a) This algorithm uses $\mathcal{J}_{\{2,3\}}$ from (21) as the overall cost function with $\alpha_3 = 1$ and the gradients (19) and (12). (b) This algorithm uses $\mathcal{J}_2$ but with the gradient (23). (c) This algorithm uses $\mathcal{J}_{\{2,3\}}$ as the overall cost function with $\alpha_3 = 0.1$ and the gradients (23) and (12). Algorithm (c) is a combination of (a) and (b). For all three algorithms we used $\mu = 0.2$, $\eta = 0.2$, $\mathbf{W}_0 = \mathbf{I}$, and $\mathbf{N}_k$ was updated with (17), (18), and (20).

Fig. 1 shows the performance curves of $\mathcal{J}_2$ and $\mathcal{J}_3$ of a single run of each algorithm with the same set of correlation matrices. We make the following observations: (i) The initial convergence rate of the algorithm (b) and (c) are almost identical. However, algorithm (b) becomes unstable after a while, whereas (c) shows a stable behavior, due to the additional penalty term $\mathcal{J}_3$, see curves of $\mathcal{J}_3$. (ii) Algorithm (c) is slightly faster than (a) due to the improved initial behavior. (iii) At the beginning there is a slow convergence behavior. This is because $\mathbf{N}_k$ has not yet converged (not shown in the plot). Interesting to note is that $\mathbf{N}_k$ converged to $\mathbf{\Sigma} - 0.1\cdot\mathbf{I}$ and not to $\mathbf{\Sigma}$. This is because we constrain $\mathbf{W}$ to be unitary and the residual estimation error $\mathbf{W}(0.1\cdot\mathbf{I})\mathbf{W}^H$ does not contribute to the error $\mathcal{J}_2$, as it has no off-diagonal elements.

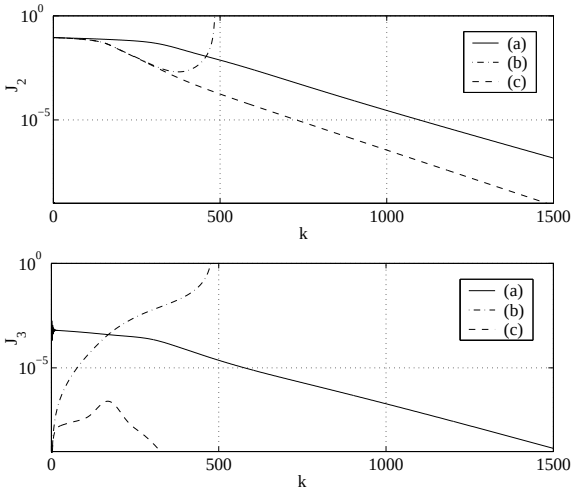

Fig. 1. Learning curves of $\mathcal{J}_2$ (top) and $\mathcal{J}_3$ (bottom) for three different algorithms: (a) $\mathcal{J}_{\{2,3\}}$ with $\alpha_3 = 1$, (b) $\mathcal{J}_{\{2\}}$ using (23), and (c) $\mathcal{J}_{\{2,3\}}$ using (23) and $\alpha_3 = 0.1$.

6. SUMMARY

We have analyzed the joint-diagonalization problem and have derived several gradient-based methods to find either a unitary or a non-unitary joint-diagonalizing matrix. In the case of a unitary-matrix constraint, we have proposed three possibilities to incorporate this constraint into the update equation: (i) adding a penalty term to the main objective function, (ii) modifying the gradient-based update such that the unitary property is approximately fulfilled after every update, and (iii) projecting the update on the desired manifold such that the unitary property is exactly fulfilled after every update.

Furthermore, we have also described how these methods can be applied to blind signal separation. We have proposed in (31) a cost function that considers the case where the source signals are non-stationary and / or non-white and additive white sensor noise was involved in the mixing process. Many of the ideas presented in this paper pave the way to investigate faster-converging Newton-type algorithms, see [26].

In (3) we have proposed a useful way to normalize the joint-diagonalization cost function such that it becomes independent of the absolute norms $\{\|\mathbf{R}_p\|_F\}$. This is very convenient for gradient methods and is similar to using a NLMS versus a LMS in adaptive signal processing.

A nice feature of the gradient-based algorithms is that they need only additions and multiplications (no divisions), which is well suited for an implementation on a DSP or on dedicated hardware.

7. ACKNOWLEDGMENT

The authors would like to thank Kamran Rahbar, Michael Kramer, and Nail Çadallı for helpful discussions.

A. INITIALIZATION OF THE BSS ALGORITHM

For non-stationary source signals, a reasonable initial value $\mathbf{W}_0$ for minimizing the cost function (31) can be found by defining the matrix

$$\mathbf{Q} \triangleq (\mathbf{R_{xx}}(t_1, 0) - \hat{\mathbf{R}}_{\mathbf{nn}})^{-1}(\mathbf{R_{xx}}(t_2, 0) - \hat{\mathbf{R}}_{\mathbf{nn}}) \qquad (37)$$

for $t_1 \neq t_2$, where $\hat{\mathbf{R}}_{\mathbf{nn}}$ is the initial estimate of $\mathbf{R_{nn}}$. Then we compute the corresponding eigenvalue decomposition (EVD), i.e., $\mathbf{Q} \triangleq \mathbf{T}\mathbf{\Lambda}\mathbf{T}^{-1}$, where $\mathbf{T}$ contains the normalized eigenvectors of $\mathbf{Q}$ in its columns. If all λ_m are distinct, $\mathbf{W}_0 = \mathbf{T}^H$ is a reasonable choice for initializing the algorithm, see [27, 18, 4].

For the case of non-white source signals, we define $\mathbf{Q} \triangleq (\mathbf{R_{xx}}(t_1, \tau_1))^{-1}(\mathbf{R_{xx}}(t_2, \tau_2))$ for $\tau_1 \neq \tau_2$ and apply the same technique [28].

B. REFERENCES

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

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

[3] D. P. Bertsekas, *Nonlinear Programming*, Athena Scientific, 2nd edition, 1999.

[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] A. Edelman, T. A. Arias, and S. T. Smith, "The geometry of algorithms with orthogonality constraints," *SIAM Review*, vol. 20, no. 2, pp. 303–353, Apr. 1998.

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

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

[8] S. C. Douglas, "Combined subspace tracking, prewhitening, and contrast optimization for noisy blind signal separation," in *Proc. ICA*, Helsinki, Finland, June 19–22, 2000, pp. 579–584.

[9] K. Rahbar and J. P. Reilly, "Geometric optimization methods for blind source separation of source signals," in *Proc. ICA*, Helsinki, Finland, June 19–22, 2000, pp. 375–380.

[10] S. Choi and A. Cichocki, "Correlation matching approach to source separation in the presence of spatially correlated noise," in *Proc. ISSPA*, Singapore, Aug. 6–8, 2001.

[11] 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.

[12] 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.

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

[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. 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.

[16] D. T. Pham and J. F. Cardoso, "Blind separation of instantaneous mixtures of nonstationary sources," *IEEE Trans. Signal Processing*, vol. 49, no. 9, pp. 1837–1848, Sept. 2001.

[17] 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.

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

[19] 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.

[20] E. Weinstein, M. Feder, and A. V. Oppenheim, "Multi-channel signal separation by decorrelation," *IEEE Trans. Speech and Audio Processing*, vol. 1, no. 4, pp. 405–413, Oct. 1993.

[21] C. Chang, Z. Ding, S. F. Yau, and F. H. Y. Chan, "A matrix–pencil approach to blind separation of colored nonstationary signals," *IEEE Trans. Signal Processing*, vol. 48, no. 3, pp. 900–907, Mar. 2000.

[22] S. Choi and A. Cichocki, "Blind separation of nonstationary and temporally correlated sources from noisy mixtures," in *Proc. NNSP*, Sydney, Australia, Dec. 11–13, 2000, vol. 2, pp. 405–414.

[23] 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.

[24] 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.

[25] M. Joho, H. Mathis, and R. H. Lambert, "Overdetermined blind source separation: Using more sensors than source signals in a noisy mixture," in *Proc. ICA*, Helsinki, Finland, June 19–22, 2000, pp. 81–86.

[26] M. Joho and K. Rahbar, "Joint diagonalization of correlation matrices by using Newton methods with application to blind signal separation," in *Proc. SAM*, Rosslyn, VA, Aug. 4–6, 2002, pp. 403–407.

[27] M. K. Tsatsanis and C. Kweon, "Blind source separation of non-stationary sources using second-order statistics," in *Proc. Asilomar Conf. Signals, Syst., Comput.*, Pacific Grove, CA, Nov. 1–4, 1998, vol. II, pp. 1574–1578.

[28] L. Molgedey and H. Schuster, "Separation of independent signals using time-delayed correlations," *Phys. Rev. Lett.*, vol. 72, no. 23, pp. 3634–3637, Oct. 1994.