Available online at www.sciencedirect.com
ScienceDirect
Progress in Natural Science 18 (2008) 735–739
Progress in Natural Science
www.elsevier.com/locate/pnsc

Non-orthogonal joint diagonalization with diagonal constraints

Guoxu Zhou *, Zuyuan Yang, Zongze Wu, Jinlong Zhang

School of Electrics and Information Engineering, South China University of Technology, Guangzhou 510641, China

Received 20 November 2007; received in revised form 28 December 2007; accepted 2 January 2008

Abstract

Joint diagonalization has attracted much attention and many algorithms have been presented so far. However, some ambiguities still exist in the objective functions for joint diagonalization. In this paper, some criterions are proposed to eliminate the above ambiguities at first, and then a new objective function which satisfies these criterions is presented. The new objective function introduces diagonal constraints for joint diagonalization, thus the trivial and unbalance solutions are excluded easily. Exactly jointly diagonalizable theorem is built to interpret the reasonableness of the new objective function and the conjugate gradient method is used to provide fast and reliable convergence. Finally, non-orthogonal joint diagonalization algorithm with diagonal constraints (DDiag) is developed. Simulations show that DDiag is efficient and robust.

Keywords: Joint diagonalization; Blind source separation; Conjugate gradient method

1. Introduction

Given a matrix pencil $\{\boldsymbol{R}_k:k=1,2,\cdots,K\}$, joint diagonalization (JD) tries to find a matrix $\boldsymbol{W}$, which makes $\boldsymbol{W}\boldsymbol{R}_k\boldsymbol{W}^H$ as diagonal as possible for all k. When $K=2$, this problem can be simply solved via the generalized eigenvalue decomposition (GED), but when $K>2$, this problem is rather complicated. JD has attracted much attention, because of its wide application in blind source separation (BSS) [1,2], beam forming [3] and so on. In this paper, superscript H means the conjugate transpose; $\|\cdot\|_F$ denotes Frobenius norm, and the matrix $\boldsymbol{W}$ is simply called a diagonalizer for $\{\boldsymbol{R}_k:k=1,2,\cdots,K\}$. $\boldsymbol{R}_k$ is an $N\times N$ Hermitian matrix and $\boldsymbol{I}$ is the $N\times N$ identical matrix. $\boldsymbol{U}=\mathrm{eig}(\boldsymbol{A},\boldsymbol{B})$ is the GED of $\boldsymbol{A}$ and $\boldsymbol{B}$, $\boldsymbol{U}$ is their eigenvector matrix.

Up to now, there have been many algorithms to tackle JD. These algorithms mainly use the following two objective functions:

(i) Minimization of diagonalization error. They use the following objective function:

$$\min_{W} F(\boldsymbol{W})=\sum\nolimits_k \lambda_k off(\boldsymbol{W}\boldsymbol{R}_k\boldsymbol{W}^H), \tag{1}$$

where $off(\boldsymbol{M})=\sum_{i\neq j}\boldsymbol{M}_{ij}^2$, λ_k are weight factors. Because (1) does not need to consider the diagonal elements, (1) is relatively easy to optimize. However, there are trivial solutions in (1), for instance, $\boldsymbol{W}=\boldsymbol{0}$. So, additional constraints on $\boldsymbol{W}$ are necessary. According to the background of JD, Cardoso developed the Given rotations-based algorithms which restricted $\boldsymbol{W}$ to be orthogonal [4]. Unfortunately sometimes the orthogonality restriction is too tight. Later Ziehe proposed non-orthogonal FFDiag algorithm and required $\boldsymbol{W}$ to be invertible [5]. Vollgraf imposed row norm constraints on $\boldsymbol{W}$ [6].

(ii) Minimization of subspace fitting error. These algorithms use the following objective function:

$$\min_{V,\Lambda} E(V,\Lambda)=\sum_k \lambda_k\|\boldsymbol{R}_k-\boldsymbol{V}\Lambda_k\boldsymbol{V}^H\|_F^2 \tag{2}$$

* Corresponding author. Tel./fax: +86 20 87114709.
E-mail address: zhou.guoxu@mail.scut.edu.cn (G. Zhou).

1002-0071/$ - see front matter
doi:10.1016/j.pnsc.2008.01.019

Here, $\boldsymbol{V}$ is actually the estimation of the inverse of $\boldsymbol{W}$. Model (2) excludes trivial solutions and need no additional constraints on $\boldsymbol{V}$, but, it has to estimate the diagonal matrix $\boldsymbol{\Lambda}_k$ corresponding to $\boldsymbol{R}_k$. In fact, the optimizing of (2) is very difficult, and currently the method used is optimizing the columns of $\boldsymbol{V}$ and diagonal matrix $\boldsymbol{\Lambda}_k$ alternately, ACDC is a representative of this kind of algorithms [2].

An objective function which need no additional constraints on $\boldsymbol{W}$ was proposed by Pham, but restriction on $\{\boldsymbol{R}_k\}$ arose: each matrix for diagonalization must be positively definite. This restriction hindered its application in practice. Because the optimization of (2) is too difficult, currently most work is mainly concentrated on the improvements of (1). FFDiag updates $\boldsymbol{W}$ by a series of strictly diagonally dominant matrices. Vollgraf imposed column norm constraints on $\boldsymbol{W}$ to avoid trivial and unbalance solutions, but it was proved that it is unable to exclude degeneration solutions [7].

Lastly Li proposed the following objective function [7]:

$$\min_{W} J(W) = \sum_{k} \lambda_k off(\boldsymbol{W}\boldsymbol{R}_k\boldsymbol{W}^H) - \beta \log|\det \boldsymbol{W}|. \tag{3}$$

Notice that $J(\boldsymbol{W}) \to +\infty$ as $\boldsymbol{W} \to$ some singular matrix, so this model can avoid trivial and degenerate solutions strictly. However, there are flaws in (3): (i) if $\{\boldsymbol{R}_k\}$ is exactly jointly diagonalizable and $\sum_k \lambda_k off(\boldsymbol{W}\boldsymbol{R}_k\boldsymbol{W}^H) = 0$, then log|det $\boldsymbol{W}| \to +\infty$, thus $\|W\|_F \to +\infty$, the algorithm cannot converge at all. This point is also confirmed by simulations. (ii) Similar to [2,6], the optimization method seems unreliable, since optimal solution with respect to each column of $\boldsymbol{W}$ cannot guarantee an optimal solution with respect to $\boldsymbol{W}$.

There are also many other objective functions, but they often have the following common limitations. That $\boldsymbol{W}$ is invertible is a necessary condition, but is it sufficient? To (1), (3), even if $\boldsymbol{W}$ is not degenerate, and off-diagonal elements are vanished, may these go with nearly vanishing diagonal elements? If these do, the solution is not the desired. We think of a well-defined model for JD which should satisfy the following criterions: (1) the objective function should be explicit, but not ambiguous. Such as (1), it does minimize the off-diagonal elements, but a direct optimization leads to trivial solutions; (2) diagonal elements should be taken into consideration, that is, it should be guaranteed that the diagonal elements will not vanish with vanishing off-diagonal elements; (3) the objective function is easy to optimize. Model (2) is difficult to optimize, and the optimization in Refs. [2,6,7] seems unreliable; (4) the solution space is not redundant, that is, the objective function has the ability of merging the essentially equivalent solutions. This ability will make the solution space as simple as possible. Concretely, for model (1), let invertible matrix $\boldsymbol{W}_0$ be the ideal solution of (1) with $F(\boldsymbol{W}_0) = 0$, then for any invertible diagonal matrix $\boldsymbol{D}$, $\boldsymbol{D}\boldsymbol{W}_0$ is also the solution of (1), so the row norm indeterminacy exists. The column norm indeterminacy exists in (2). As to (3), if $\sum_k \lambda_k off(\boldsymbol{W}\boldsymbol{R}_k\boldsymbol{W}^H) = 0$ then $\|W\|_F \to +\infty$. Even if these indeterminacies may not affect the quality of diagonalization occasionally, they will hamper the efficiency of optimization. In view of these points, a new objective function and its optimization scheme will be proposed by this paper, and hope to realize the above purposes.

The remainder of this paper is organized as follows: in Section 2, exactly jointly diagonalizable theorem is built, and thus the relationship between exact JD and GED is revealed. In Section 3, non-orthogonal JD algorithm with diagonal constraints (DDiag) is developed. Comparisons between existing popular algorithms are provided in Section 4. Finally, Section 5 is devoted to the conclusions.

2. Exactly jointly diagonalizable theorem

For a given matrix pencil, usually we cannot achieve an exact JD, the main reason is that the algorithms are not good enough. Does the characteristic of the matrix pencil determine itself? The following exactly jointly diagonalizable theorem will answer this question.

Theorem 1. *Given a matrix pencil* $\{\boldsymbol{R}_k\}$ *and* $\boldsymbol{A}\boldsymbol{\Lambda}_k\boldsymbol{A}^H = \boldsymbol{R}_k$ *- for all* $k \in \{1,2,\cdots,K\}$, $\boldsymbol{A}$ *is some invertible matrix and* $\boldsymbol{\Lambda}_k$ *is diagonal. If there exist m, n, such that* $\boldsymbol{\Gamma} = \boldsymbol{\Gamma}_n\boldsymbol{\Gamma}_m^{-1}$ *has distinct diagonal elements, then* $\boldsymbol{W}\boldsymbol{A} = \boldsymbol{P}\boldsymbol{D}$. *Here* $\boldsymbol{W}$ *is a diagonalizer for* $\boldsymbol{R}_m$ *and* $\boldsymbol{R}_n$, *i.e.* $\boldsymbol{W}\boldsymbol{R}_m\boldsymbol{W}^H = \boldsymbol{\Gamma}_m$, $\boldsymbol{W}\boldsymbol{R}_n\boldsymbol{W}^H = \boldsymbol{\Gamma}_n$, $\boldsymbol{W}$ *and* R_m *are invertible matrices,* $\boldsymbol{\Gamma}_m$ *and* $\boldsymbol{\Gamma}_n$ *are diagonal matrices.* $\boldsymbol{P}$ *is a permutation matrix and* $\boldsymbol{D}$ *is a diagonal matrix.*

Proof. Suppose $\boldsymbol{C} = \boldsymbol{W}\boldsymbol{A}$, and $\boldsymbol{\Lambda} = \boldsymbol{\Lambda}_n\boldsymbol{\Lambda}_m^{-1}$, then $\boldsymbol{C}\boldsymbol{\Lambda}\boldsymbol{C}^{-1} = \boldsymbol{\Gamma}$, that is, $\boldsymbol{C}\boldsymbol{\Lambda} = \boldsymbol{\Gamma}\boldsymbol{C}$, where $\boldsymbol{\Lambda} = diag(\lambda_i)$, $\boldsymbol{\Gamma} = diag(\lambda_i)$. For any i, j, $\boldsymbol{C}_{ij}\lambda_j = \gamma_i\boldsymbol{C}_{ij}$, thus $\lambda_j = \gamma_i$ if $\boldsymbol{C}_{ij} \neq 0$.

Since $\boldsymbol{C} = \boldsymbol{W}\boldsymbol{A}$ is invertible, there is at least one nonzero element in each column of $\boldsymbol{C}$. For any j, without loss of generality, suppose $\boldsymbol{C}_{ij} \neq 0$, thus $\lambda_j = \gamma_i$. If $\boldsymbol{C}_{kj} \neq 0$ for some $k \neq i$, $\lambda_j = \gamma_k$ yields from $\boldsymbol{C}_{kj}\lambda_j = \boldsymbol{C}_{kj}\gamma_k$. Consequently $\gamma_i = \gamma_k$ holds, which conflicts with the supposition that $\boldsymbol{\Gamma}$ has distinct diagonal elements. So there must be $\boldsymbol{C}_{kj} = 0$. As a result, there is one and only one nonzero element in each column of $\boldsymbol{C}$. Since $\boldsymbol{C}$ is invertible, $\boldsymbol{C} = \boldsymbol{P}\boldsymbol{D}$ yields. Q.E.D.

In fact a simpler result has been presented as Theorem 4.1 in Ref. [8], where $\boldsymbol{R}_m$ should be positively definite, thus Theorem 1 can be regarded as a generalization of that theorem. According to Theorem 1, if two matrices in $\{\boldsymbol{R}_k\}$ can be jointly diagonalized by $\boldsymbol{W}$ (suppose the conditions in Theorem 1 are satisfied), then the whole matrix pencil $\{\boldsymbol{R}_k\}$ can either be exactly jointly diagonalized by $\boldsymbol{W}$, or cannot be exactly jointly diagonalized by any invertible matrix. Since JD of two matrices can be solved via the GED, Theorem 1 provides us a brief way to judge whether the matrix pencil can be exactly jointly diagonalized. Thus the following definition is useful.

Definition 1. Matrix pencil $\{\boldsymbol{R}_k\}$ is a well-matrix-pencil (WMP), if it contains at least one matrix pair $\{\boldsymbol{R}_m, \boldsymbol{R}_n\}$

which satisfies that $\{R_m, \boldsymbol{R}_n\}$ has mutual distinct generalized eigenvalues, and $\{\boldsymbol{R}_m, \boldsymbol{R}_n\}$ is called a representative matrix pair (RMP) of the WMP.

One of the most important classes of WMP is the regular matrix pencils (a regular matrix pencil contains at least one positively definite matrix), which have actually been studied in Refs. [6,9]. A WMP at least contains a matrix pair which can be exactly jointly diagonalized. If not, the matrix pencil is thought of as an ill-matrix-pencil (IMP). Generally JD for an IMP loses its sense for only very poor JD quality can be achieved. So this paper concentrates on the JD for WMP.

Thus the GED of an RMP can provide a nice approximation of the diagonalizer, especially for exactly jointly diagonalizable matrix pencil. Note that the covariance rate defined in Ref. [10] is essentially based on GED and thus the theorem in which is consistent to Theorem 1. Usually, if more matrices are used for joint diagonalization, robustness will be improved. In the next section, a solution for the general WMP is proposed.

3. Non-orthogonal JD with diagonal constraints

For a general WMP, the following model is considered:

$$\begin{aligned}\min_{W} H(\boldsymbol{W}) = (1-q)\sum_{k=m,n}\|\boldsymbol{W}\boldsymbol{R}_k\boldsymbol{W}^H - \boldsymbol{D}_k\|_F^2 \\ + q\sum_{k\neq m,n}\lambda_k off(\boldsymbol{W}\boldsymbol{R}_k\boldsymbol{W}^H),\end{aligned} \tag{4}$$

where $q \in (0,1)$, generally is 0.01 or $1/K$, $\boldsymbol{D}_k$ is diagonal. Model (4) means that the desired diagonalizer of the whole matrix pencil is at least the diagonalizer of $\{\boldsymbol{R}_m, \boldsymbol{R}_n\}$. According to Theorem 1, the two terms in $H(\boldsymbol{W})$ are consistent. Suppose $\boldsymbol{R}_m$ is invertible, this model can be further simplified:

$$\begin{aligned}\min_{W} H(\boldsymbol{W}) = (1-q)\|\boldsymbol{W}\boldsymbol{R}_m\boldsymbol{W}^H - \boldsymbol{D}_m\|_F^2 \\ + q\sum_{k\neq m}\lambda_k off(\boldsymbol{W}\boldsymbol{R}_k\boldsymbol{W}^H).\end{aligned} \tag{5}$$

Comparing (5) with (1) and (2), it can be final that the first term is just from (2) and the second term is from (1), and (4) integrates their advantages successfully,

(1) Since $\boldsymbol{W}\boldsymbol{R}_m\boldsymbol{W}^H \approx \boldsymbol{D}_m$ when $H(\boldsymbol{W})$ is minimized, degenerate and trivial solutions are excluded.
(2) The norm of $\boldsymbol{W}$ is approximately fixed (especially if $\boldsymbol{R}_m$ is positive definite).
(3) Diagonal constraints are provided to avoid unbalance solutions.

Currently all the models try to minimize the off-diagonal error, but few care the diagonal elements. If the diagonal elements are minimized along with the minimization of off-diagonal elements, the JD loses its sense. However, model (4) provides a natural constraint for the diagonal elements.

Now turn to the optimization of (5). As mentioned above, many JD algorithms utilize quasi optimization methods. They do this to make their algorithms converge faster, but the result is that they do not guarantee a true extremum. Gradient descent method provides reliable extremum, but converges slowly. Joho considered Newton methods [11], but the Hessian matrix is with the dimension of $N^2 \times N^2$. When N increases, the required storage of computation increases rapidly, so these methods are not good choices. This paper adopts conjugate gradient method, for it converges much faster than the gradient descent method and uses the first order gradient only.

The gradient of (5) is

$$\begin{aligned}\nabla H(\boldsymbol{W}) = 4(1-q)(\boldsymbol{S}_m\boldsymbol{W}\boldsymbol{R}_m - \boldsymbol{D}_m\boldsymbol{W}\boldsymbol{R}_m) \\ + 4q\sum_{k\neq m}\lambda_k(\boldsymbol{S}_k - ddiag(\boldsymbol{S}_k))\boldsymbol{W}\boldsymbol{R}_k\end{aligned} \tag{6}$$

Here, $\boldsymbol{S}_k = \boldsymbol{W}\boldsymbol{R}_k\boldsymbol{W}^H$. The outline of non-orthogonal JD algorithm with diagonal constraints (DDiag) is displayed below

DDiag Algorithm

Input: WMP $\{\boldsymbol{R}_k\}$, and $\{\boldsymbol{R}_m, \boldsymbol{R}_n\}$ is an RPM.
Output: diagonalizer $\boldsymbol{W}$.
Step 1: Let $\boldsymbol{U}_0 = \mathrm{eig}(R_m, R_n)$, then $\boldsymbol{W}_0 = U_0^H$ and $\boldsymbol{D}_m = ddiag(W_0 R_m W_0^H)$. $\boldsymbol{g}_0 = \nabla H(W_0)$, $\boldsymbol{d}_0 = -\boldsymbol{g}_0$.
Step 2: While termination criterion is not satisfied
Search step factor;

$\boldsymbol{W}_{k+1}\boldsymbol{W}_k + \boldsymbol{d}_k, g_{k+1} = \nabla F(\boldsymbol{W}_{k+1})$;
$\beta = trace(g_{k+1}g_{k+1}^H)/trace(\boldsymbol{g}_k\boldsymbol{g}_k^H)$;
$\boldsymbol{d}_k = -\boldsymbol{g}_{k+1} + \beta\boldsymbol{d}_k$
End While
Step 3: $\boldsymbol{W}_k$ is the required diagonalizer.

In practice, restart procedure of conjugate gradient method is utilized to improve the robustness of the algorithm. From steps 1 and 2, if $\{\boldsymbol{R}_k\}$ is exactly jointly diagonalizable, the exact diagonalizer is obtained immediately from $\boldsymbol{W}_0$.

4. Simulations

Here, we only compare DDiag with the three latest algorithms: FFDiag [5], QDiag [6] and Fajd [7]. In all the experiments, DDiag uses the following parameters: $q = 0.01$, the number of maximum iteration is 200, updating step is initialized to be 0.1, and once an update fails, the step is halved.

According to the background of JD, $\{\boldsymbol{R}_k\}$ is generated as

$$\boldsymbol{R}_k = \boldsymbol{A}\boldsymbol{\Lambda}_k\boldsymbol{A}^H + \sigma\boldsymbol{E}_k\boldsymbol{E}_k^H, \tag{7}$$

where the elements of $\boldsymbol{A}$, $\boldsymbol{\Lambda}_k$ and $\boldsymbol{E}_k$ are drawn from a standard normal distribution, respectively. To evaluate the diagonalization quality, the following diagonalization error index is adopted [5]:

$$error(G) = \frac{1}{2}\left[\sum_i \left(\sum_j \frac{|\boldsymbol{G}_{ij}|^2}{\max_l |\boldsymbol{G}_{il}|^2} - 1\right) + \sum_j \left(\sum_i \frac{|\boldsymbol{G}_{ij}|^2}{\max_l |\boldsymbol{G}_{lj}|^2} - 1\right)\right] \quad (8)$$

Li proposed the following Generalized Log Likelihood Criterion (GLLC) to evaluate the diagonalization quality [7]:

$$J_{GLLC}(W) = \frac{1}{2}\sum_{k=1}^{K} \lambda_k \left|\log\left|\frac{\det diag(WR_k W^H)}{\det(WR_k W^H)}\right|\right|. \quad (9)$$

But it seems that it cannot evaluate diagonalization error accurately. For example, $\boldsymbol{G}_1 = \boldsymbol{W}_1 \boldsymbol{R} \boldsymbol{W}_1^H = \begin{bmatrix} 1 & 0.5 & 0 \\ 0.5 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix}, \boldsymbol{G}_2 = \boldsymbol{W}_2 \boldsymbol{R} \boldsymbol{W}_2^H = \begin{bmatrix} 1 & 0 & 0.5 \\ 0 & 1 & 0 \\ 0.5 & 0 & -1 \end{bmatrix}$, then $\boldsymbol{G}_1$ and $\boldsymbol{G}_2$ should have the same diagonalization error, but $J_{GLLC}(G_1) = 0.1438$, $J_{GLLC}(G_2) = 0.1116$. In addition, $\boldsymbol{W}\boldsymbol{R}_k\boldsymbol{W}^H$may be singular for some k even if $\boldsymbol{W}$ is non-singular. Since (8) cannot identify degenerate solutions this paper proposes the following degeneration measure (DM) to identify generative solutions:

$$DM(W) = \left|\log\left|\det\left(\sqrt{N}\frac{\boldsymbol{W}}{\|\boldsymbol{W}\|_F}\right)\right|\right|. \quad (10)$$

$DM(\boldsymbol{W})$ is non-negative, and $DM(\boldsymbol{W}) = +\infty$, if $\boldsymbol{W}$ is singular. And it is easy to prove that $DM(\boldsymbol{W}) = 0$ if and only if $\boldsymbol{W}$ is orthogonal, which means that the best condition number of $\boldsymbol{W}$ is achieved (see Appendix). As a result, a smaller value of $DM(\boldsymbol{W})$ corresponds to a well-conditioned diagonalizer.

Experiment 1: Experiments on given matrix pencils. $N = 10$, $K = 10$, $=10^{-6}$–10^{-1}. Each algorithm runs 100 times and the average performances of diagonalization error and degeneration measure are shown in Fig. 1a and b, respectively.

As Fig. 1 shows, DDiag provides the smallest diagonalization error. The degeneration measures of the solutions obtained by FFDiag and Fajd are the smallest; this fact can be conjectured, for Fajd and FFDiag utilize very strong invertible criterions. However, all the degeneration measures are acceptable.

We also found that for exactly jointly diagonalizable matrix pencils, FFDiag, QDiag and DDiag can obtain an exact diagonalizer, but Fajd cannot converge. The main reason is that Fajd has to use the inverse matrix of $\boldsymbol{Q}_i$ (see Eq. (25) in Ref. [7] and $\boldsymbol{Q}_i$ is defined by Eq. (15) in the same reference). In fact, when the matrix pencil is exactly or nearly exactly jointly diagonalizable, i.e. there exists a non-singular matrix which satisfies $\sum_{k=1}^{K} off(\boldsymbol{W}\boldsymbol{R}_k\boldsymbol{W}^T) \approx 0$, the matrix $\boldsymbol{Q}_i$ is either a singular matrix or is even equal to the zero matrix after iterations for Fajd.

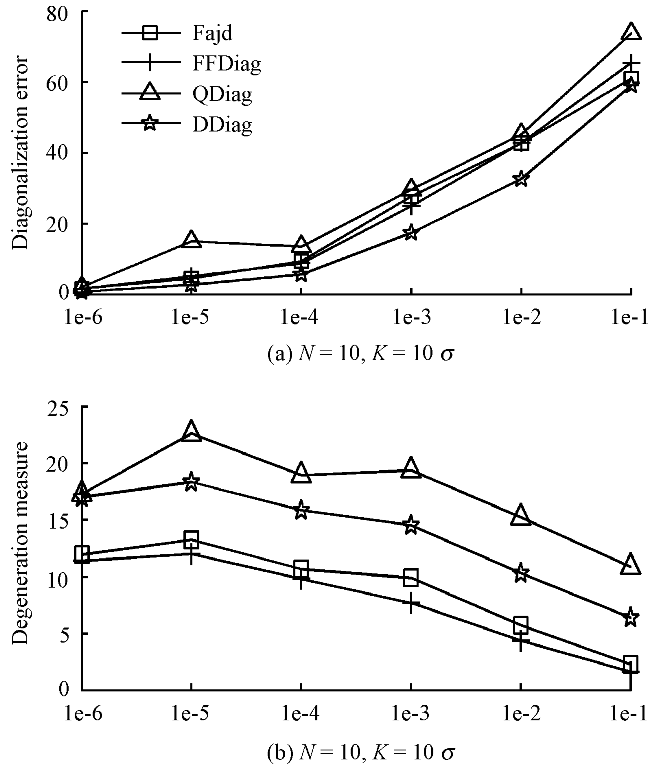

Fig. 1. Diagonalization error and degeneration measure with increasing σ.

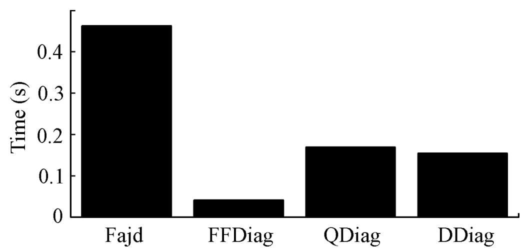

Fig. 2. Time consumption of each algorithm.

Fig. 2 is their time consumption when they run on an AMD Athlon 64 (1.8 GHz) processor with 448 MB of RAM. From Fig. 2, it can be seen that FFDiag is still the fastest; DDiag is slightly faster than QDiag, but much faster than Fajd.

Experiment 2: Applications in BSS. BSS has received much attention for more than two decades, and many algorithms have been presented [8,12]. BSS is one of the most important applications of JD, in this module we will evaluate their performances in BSS. The sources are from ICALAB benchmarks [13], and Speech8 and Aacspeech16 are picked out. Acspeech16 contains 16 typical speech signals which have a temporal structure but they are not precisely independent. In each run total 20 autocorrelation matrices are generated (see Ref. [1] for details), and time delay is 1. Non-singular mixing matrix $\boldsymbol{A}$ is generated randomly in each run. Cross-talking error is widely used to evaluate the separation accuracy [8]. For cross-talking error is

Table 1
Average cross-talking error in 100 independent runs

Algorithms	FFDiag	Fajd	QDiag	DDiag
Speech8	0.3078	0.1171	0.1003	0.0968
Acspeech16	0.6232	1.0162	0.3709	0.3622

equivalent to (8), here (8) is used to evaluate the separation quality for convenience: Suppose $\boldsymbol{W}$ is the diagonalizer, $\boldsymbol{WA}$ should be as close to a generalized permutation matrix as possible. If $error(\boldsymbol{WA}) = 0$, the sources are accurately recovered. Their average performance in 100 independent runs is shown in Table 1. Table 1 also shows that DDiag performs slightly better than the others.

5. Conclusions

This paper analyzes the main difficulties and ambiguities in JD, and proposes a well-defined objective function for JD. Aiming at WMP, non-orthogonal JD with diagonal constraints (DDiag) are proposed. By comparison, it has been found that FFDiag, Fajd, QDiag and DDiag perform very well for WMP. Fajd can avoid trivial and singular solutions strictly, however, its convergences are relatively slow. Especially, a hidden trouble exists in its optimization. DDiag converges fast, and provides the chance to obtain the best diagonalization qualities. However, when N increases, the optimization is relatively harder, although a nice diagonalization quality always can be obtained if we are sufficiently patient and careful in the set of parameters. We are expecting a better optimization method for model (5). Also, JD can only be applied in linear instantaneous mixing models and convolutive BSS (CBSS) is, till full of challenges. In Ref. [12], the authors addressed a novel CBSS algorithm that exploits the sparsity of source signals in frequency domains. This algorithm is maximum a posteriori-based and has good performance. We hope to develop new JD-based algorithms for CBSS in the future.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (Grant Nos. U0635001, 60674033, 60774094), and the Natural Science Fund of Guangdong Province, China (No. 05006508).

Appendix

Suppose $\boldsymbol{W}$ is an $N \times N$ square real matrix, and $\|\boldsymbol{W}\|_F = \sqrt{N}$. Then $|\det \boldsymbol{W}\mathrm{t}| \leqslant 1$, and $|\det \boldsymbol{W}| = 1$ if and only if $\boldsymbol{WW}^T = 1$.

Proof. If $\boldsymbol{W}$ is singular, $\det \boldsymbol{W}{=}0 < 1$. Let $\boldsymbol{W}$ be non-singular and $\boldsymbol{C} = \boldsymbol{WW}^T$, notice that $\|\boldsymbol{W}\|_F = \sqrt{N}$, then $trace(\boldsymbol{C}) = N$, $\boldsymbol{C}$ is positively definite. Suppose $\lambda_1, \lambda_2, \cdots, \lambda_N$ are the eigenvalues of $\boldsymbol{C}$, then

$$trace(\boldsymbol{C}) = \sum_{i=1}^{N} \lambda_i = N$$

$$\det(\boldsymbol{C}) = \prod_{i=1}^{N} \lambda_i$$

Because

$$\det(\boldsymbol{C}) = \prod_{i=1}^{N} \lambda_i \leqslant \left(\frac{\lambda_1 + \lambda_2 + \cdots + \lambda_N}{N}\right)^N = 1$$

Thus $\det(\boldsymbol{C}) = |\det(\boldsymbol{WW}^T)| = |\det(\boldsymbol{W})|^2 \leqslant 1$, so $|\det(W)| \leqslant 1$ and $|\det(\boldsymbol{W})| = 1$ if and only if $\lambda_1 = \lambda_2 = \cdots = \lambda_N = 1$, that is, $\boldsymbol{C} = \boldsymbol{WW}^T = \boldsymbol{I}$. Q. E. D.

References

[1] Belouchrani A, AbedMeraim K, Cardoso JF, et al. A blind source separation technique using second-order statistics. IEEE Trans Signal Process 1977;45:434–44.
[2] Yeredor A. Non-orthogonal joint diagonalization in the least-squares sense with application in blind source separation. IEEE Trans Signal Process 2002;50:1545–53.
[3] Sheinvald J. On blind beamforming for multiple non-Gaussian signals and the constant-modulus algorithm. IEEE Trans Signal Process 1988;46:1878–85.
[4] Cardoso JF, Souloumiac A. Jacobi angles for simultaneous diagonalization. Siam J Matrix Anal Appl 1996;17:161–4.
[5] Ziehe A, Laskov P, Nolte G, et al. A fast algorithm for joint diagonalization with non-orthogonal transformations and its application to blind source separation. J Mach Learn Res 2004;5:777–800.
[6] Vollgraf R, Obermayer K. Quadratic optimization for simultaneous matrix diagonalization. IEEE Trans Signal Process 2006;54:3270–8.
[7] Li XL, Zhang XD. Nonorthogonal joint diagonalization free of degenerate solution. IEEE Trans Signal Process 2007;55:1803–14.
[8] Cichocki A, Amari S. Adaptive Blind Signal and Image Processing: Learning Algorithms and Applications. New York: John Wiley & Sons, Ltd.; 2002.
[9] Pham DT. Joint approximate diagonalization of positive definite Hermitian matrices. Siam J Matrix Anal Appl 2001;22:1136–52.
[10] Xie SL, He ZS, Fu YL. A note on Stone's conjecture of blind signal separation. Neural Comput 2005;17:321–30.
[11] John M, Rahbar K. Joint diagonalization of correlation matrices by using newton methods with application to blind signal separation. IEEE, sensor array and multichannel signal processing workshop proceedings, 2002, p. 403–407.
[12] He ZS, Xie SL, Ding SX, et al. Convolutive blind source separation in the frequency domain based on sparse representation. IEEE Trans Audio Speech Lang Proc 2007;15:1551–63.
[13] Cichocki A, Amari S, Siwek K, et al. ICALABA toolboxes [OnLine]. Available from: http://www.bsp.brain.riken.jp/ICALAB (accessed 28.3.2008).