A FAST APPROXIMATE JOINT DIAGONALIZATION ALGORITHM USING A CRITERION WITH A BLOCK DIAGONAL WEIGHT MATRIX

Petr Tichavský[1], *Arie Yeredor*[2] *and Jan Nielsen*[1]

[1]Institute of Information Theory and Automation,
Academy of Sciences of the Czech Republic, P.O.Box 18,
182 08 Prague 8, Czech Republic. E-mail: p.tichavsky@ieee.org

[2]School of Electrical Engineering, Tel-Aviv University
P.O.Box 39040, Tel-Aviv 69978, ISRAEL. E-mail: arie@eng.tau.ac.il

ABSTRACT

We propose a new algorithm for Approximate Joint Diagonalization (AJD) with two main advantages over existing state-of-the-art algorithms: Improved overall running speed, especially in large-scale (high-dimensional) problems; and an ability to incorporate specially structured weight-matrices into the AJD criterion. The algorithm is based on approximate Gauss iterations for successive reduction of a weighted Least Squares off-diagonality criterion. The proposed Matlab® implementation allows AJD of ten 100×100 matrices in 3-4 seconds (for the unweighted case) on a common PC (Pentium M, 1.86GHz, 2GB RAM), generally 3-5 times faster than the fastest competitor. The ability to incorporate weights allows fast large-scale realization of optimized versions of classical blind source separation algorithms, such as Second-Order Blind Identification (SOBI), whose weighted version (WASOBI) yields significantly improved separation performance.

Index Terms— Approximate joint diagonalization, blind source separation, autoregressive processes, WASOBI

1. INTRODUCTION

Approximate Joint Diagonalization (AJD) attempts to fit a given set of square $d \times d$ matrices $\widehat{\mathbf{R}}_{\mathbf{x}}[\tau]$, $\tau = 0, \ldots, M-1$ by structured matrices of the form

$$\mathbf{R}_{\mathbf{x}}[\tau] = \mathbf{A}\mathbf{R}_{\mathbf{s}}[\tau]\mathbf{A}^T \qquad \forall\tau \tag{1}$$

where $\mathbf{R}_{\mathbf{s}}[\tau] = \mathrm{diag}\{r_s^{(1)}[\tau], r_s^{(2)}[\tau], \cdots, r_s^{(d)}[\tau]\}$ are unknown diagonal matrices. For easy reference, the matrix $\mathbf{A}$ is termed the *mixing* matrix and its inverse $\mathbf{V} = \mathbf{A}^{-1}$ is termed the *demixing* matrix, implying the interpretation of these matrices in the context of Blind Source Separation (BSS).

Typically in BSS, $\mathbf{R}_{\mathbf{s}}[\tau]$ may denote, for example, lagged source covariance matrices; some cumulant slice matrices; source covariance matrices in different time windows; and other sets reflecting statistical or structural properties of the sources. $\widehat{\mathbf{R}}_{\mathbf{x}}[\tau]$ denote similar matrices pertaining to the observed mixtures.

Some of the earlier AJD algorithms (e.g., by Cardoso and Souloumiac [1]) assume that $\mathbf{A}$ (and $\mathbf{V}$) are unitary. However, more recent algorithms relax this constraint. Among these, current state-of-the-art computationally efficient algorithms seem to be Pham's Log-Likelihood based AJD [5] (termed LLAJD here), which is further constrained by the requirement that $\mathbf{R}_{\mathbf{s}}[\tau]$ must all be positive definite, FFDIAG by Ziehe *et al.* [7], QAJD by Vollgraf and Obermayer [9], and FAJD by Li and Zhang [10].

The AJD problem is commonly addressed by attempting to minimize some off-diagonality criterion applied to the transformed set $\widehat{\mathbf{V}}\widehat{\mathbf{R}}_{\mathbf{x}}[\tau]\widehat{\mathbf{V}}^T$, where $\widehat{\mathbf{V}}$ is the estimated demixing matrix, constrained so as to evade the trivial solution $\widehat{\mathbf{V}} = \mathbf{0}$. One possible constraint, used in [9] and also applied in this paper, is that $\widehat{\mathbf{V}}\widehat{\mathbf{R}}_{\mathbf{x}}[0]\widehat{\mathbf{V}}^T$ must have an all-ones main diagonal. This constraint usually corresponds (in the BSS context) to a unit-power constraint on the estimated sources, and is only applicable if the first matrix, $\widehat{\mathbf{R}}_{\mathbf{x}}[0]$, is positive definite (however, this is usually not a restrictive requirement in BSS).

In this paper we consider an off-diagonality criterion based on a weighted Least-Squares (WLS) approach, using a positive definite weight matrix $\mathbf{W}$, and propose an iterative AJD algorithm, which is an approximate minimizer of the criterion. The WLS criterion is a quadratic form of off-diagonal elements of $\widehat{\mathbf{V}}\widehat{\mathbf{R}}_{\mathbf{x}}[\tau]\widehat{\mathbf{V}}^T$

$$C_{\mathrm{WLS}}(\widehat{\mathbf{V}}) = [\widehat{\mathbf{r}}_{\mathbf{s}}(\widehat{\mathbf{V}})]^T \mathbf{W}\widehat{\mathbf{r}}_{\mathbf{s}}(\widehat{\mathbf{V}}) \tag{2}$$

where $\widehat{\mathbf{r}}_{\mathbf{s}}(\widehat{\mathbf{V}})$ is is an $Md(d-1)/2$-dimensional vector composed of all off-diagonal elements below the main diagonal of $\widehat{\mathbf{V}}\widehat{\mathbf{R}}_{\mathbf{x}}[\tau]\widehat{\mathbf{V}}^T$, $\tau = 0, \ldots, M-1$.

Note that elements of $\widehat{\mathbf{r}}_{\mathbf{s}}(\widehat{\mathbf{V}})$ are grouped according to their locations in the matrices (rather than according to their

This work was supported by Ministry of Education, Youth and Sports of the Czech Republic through the project 1M0572.

matrix-affiliation), namely

$$\widehat{\mathbf{r}}_{\mathbf{s}}(\widehat{\mathbf{V}}) = [\widehat{\mathbf{r}}_{21}^T, \widehat{\mathbf{r}}_{31}^T, \ldots, \widehat{\mathbf{r}}_{d1}^T, \widehat{\mathbf{r}}_{32}^T, \ldots, \widehat{\mathbf{r}}_{d2}^T, \ldots, \widehat{\mathbf{r}}_{d,d-1}^T]^T \quad (3)$$

where

$$\widehat{\mathbf{r}}_{k\ell}(\widehat{\mathbf{V}}) = [(\widehat{\mathbf{V}}\widehat{\mathbf{R}}_{\mathbf{x}}[0]\widehat{\mathbf{V}}^T)_{k\ell}, \ldots, (\widehat{\mathbf{V}}\widehat{\mathbf{R}}_{\mathbf{x}}[M-1]\widehat{\mathbf{V}}^T)_{k\ell}]^T \quad (4)$$

are M-dimensional vectors for $k, \ell = 1, \ldots, d$ and $k > \ell$. (The arguments $(\widehat{\mathbf{V}})$ are omitted for brevity in (3)).

The criterion (2) is useful whenever it is possible to characterize random variations of $\widehat{\mathbf{R}}_{\mathbf{s}}[\tau] \approx \mathbf{V}\widehat{\mathbf{R}}_{\mathbf{x}}[\tau]\mathbf{V}^T$ around their theoretical counterparts $\mathbf{R}_{\mathbf{s}}[\tau]$ in terms of their first and second moments. In such cases, an optimum weight matrix $\mathbf{W}$ is defined (assuming small errors) as the inverse of the covariance matrix of $\widehat{\mathbf{r}}_{\mathbf{s}}$. However, even if the optimum weight matrix is not known, a suitable ad-hoc choice of $\mathbf{W}$ can often lead to better separation results than the "default" choice of uniform weights ($\mathbf{W} = \mathbf{I}_{(Md(d-1)/2)\times(Md(d-1)/2)}$).

In this paper we shall not address the WLS criterion (2) in its full generality, but only in a form by which the weight matrix $\mathbf{W}$ is block diagonal, with $M \times M$ blocks $\mathbf{W}_{k\ell}$, $k, \ell = 1, \ldots, d$, $k > \ell$, each corresponding to the respective $\widehat{\mathbf{r}}_{k\ell}$. Due to the special structure (3), (4) of $\widehat{\mathbf{r}}_{\mathbf{s}}$, such block-diagonal weighting can be optimal whenever the vector-pairs $\widehat{\mathbf{r}}_{k\ell}$, $\widehat{\mathbf{r}}_{k'\ell'}$ are uncorrelated for all $(k, \ell) \neq (k', \ell')$ (although each vector $\widehat{\mathbf{r}}_{k\ell}$ may have, and usually has, correlated elements). Fortunately, it so happens in a BSS context, that due to the independence of the sources, such a block-decorrelation condition can often be encountered, see, e.g., the Weights-Adjusted Second Order Blind Identification (WASOBI) algorithm in [6].

The criterion (2) can then be expressed as

$$C_{\mathrm{WLS}}(\widehat{\mathbf{V}}) = \sum_{k>\ell}^{d} [\widehat{\mathbf{r}}_{k\ell}(\widehat{\mathbf{V}})]^T \mathbf{W}_{k\ell} \widehat{\mathbf{r}}_{k\ell}(\widehat{\mathbf{V}}) \,. \quad (5)$$

The "unweighted" or "uniformly weighted" version of our algorithm (with $\mathbf{W} = \mathbf{I}$) will be termed UWAJD for easy reference. In its unweighted form, our (W)LS criterion is approximately equivalent to the criteria used in FFDIAG, QAJD and FAJD, to which we shall compare the performance in the sequel. An alternative off-diagonality measure used in LLAJD (to which we shall also compare) is the log-likelihood based criterion [8],

$$C_{\mathrm{LLC}}(\widehat{\mathbf{V}}) = \sum_{\tau=0}^{M-1} \log \frac{\det \mathrm{diag}(\widehat{\mathbf{V}}\widehat{\mathbf{R}}_{\mathbf{x}}[\tau]\widehat{\mathbf{V}}^T)}{\det(\widehat{\mathbf{V}}\widehat{\mathbf{R}}_{\mathbf{x}}[\tau]h\mathbf{V}^T)} \,. \quad (6)$$

2. THE PROPOSED AJD ALGORITHM

The proposed algorithm is iterative and begins with as initial estimate $\widehat{\mathbf{V}}^{[0]} = (\widehat{\mathbf{R}}_{\mathbf{x}}[0])^{-1/2}$. Then, in each iteration i, we seek an estimate of the "residual mixing" matrix, $\widehat{\mathbf{A}}$, so as to fit the partially diagonalized matrices $\widehat{\mathbf{R}}_{\mathbf{y}}[\tau] = \widehat{\mathbf{V}}^{[i]}\widehat{\mathbf{R}}_{\mathbf{x}}[\tau](\widehat{\mathbf{V}}^{[i]})^T$ by matrix products of the form $\widehat{\mathbf{A}}\mathbf{D}_{\mathbf{y}}[\tau]\widehat{\mathbf{A}}^T$. Here $\widehat{\mathbf{A}}$ is constrained to have an all-ones main diagonal, and $\mathbf{D}_{\mathbf{y}}[\tau]$ is a diagonal matrix containing the main diagonal of $\widehat{\mathbf{R}}_{\mathbf{y}}[\tau]$, i.e. $\mathbf{D}_{\mathbf{y}}[\tau] = \mathrm{Diag}(\mathrm{diag}(\widehat{\mathbf{R}}_{\mathbf{y}}[\tau]))$ for $\tau = 0, \ldots, M-1$. Once $\widehat{\mathbf{A}}$ is found, the estimated demixing matrix is updated as $\widehat{\mathbf{V}}^{[i+1]} = \widehat{\mathbf{A}}^{-1}\widehat{\mathbf{V}}^{[i]}$ and subsequently re-normalized by proper scaling of rows of $\widehat{\mathbf{V}}^{[i+1]}$ so that

$$\mathrm{diag}(\widehat{\mathbf{V}}^{[i+1]}\widehat{\mathbf{R}}_{\mathbf{x}}[0](\widehat{\mathbf{V}}^{[i+1]})^T) = [1, \ldots, 1]^T \,.$$

Our approach for finding $\widehat{\mathbf{A}}$ in each iteration is the following. Put $\widehat{\mathbf{r}}_{\mathbf{y}} = \widehat{\mathbf{r}}_{\mathbf{s}}(\widehat{\mathbf{V}}^{[i]})$ as in (3) and let $\mathbf{d}$ be a vector composed of diagonal elements of $\widehat{\mathbf{R}}_{\mathbf{y}}[\tau]$, $\tau = 0, \ldots, M-1$, namely

$$\mathbf{d} = [\mathrm{diag}(\widehat{\mathbf{R}}_{\mathbf{y}}[0])^T, \ldots, \mathrm{diag}(\widehat{\mathbf{R}}_{\mathbf{y}}[M-1])^T]^T \,.$$

In order to find $\widehat{\mathbf{A}}$, we wish to minimize the WLS criterion

$$\widetilde{C}_{\mathrm{WLS}}(\boldsymbol{\theta}) = [\widehat{\mathbf{r}}_{\mathbf{y}} - \mathbf{f}(\boldsymbol{\theta})]^T \mathbf{W} [\widehat{\mathbf{r}}_{\mathbf{y}} - \mathbf{f}(\boldsymbol{\theta})] \quad (7)$$

where $\boldsymbol{\theta}$ is composed of off-diagonal elements of $\widehat{\mathbf{A}}$, and $\mathbf{f}(\boldsymbol{\theta})$ is a suitable quadratic form of these elements. In particular,

$$\mathbf{f}(\boldsymbol{\theta}) = [\mathbf{f}_{21}^T, \mathbf{f}_{31}^T, \ldots, \mathbf{f}_{d1}^T, \mathbf{f}_{32}^T, \ldots, \mathbf{f}_{d2}^T, \ldots, \mathbf{f}_{d,d-1}^T]^T \quad (8)$$

where

$$\begin{aligned} \mathbf{f}_{k\ell}(\boldsymbol{\theta}) &= [(\widehat{\mathbf{A}}\mathbf{D}[0]\widehat{\mathbf{A}}^T)_{k\ell}, \ldots, (\widehat{\mathbf{A}}\mathbf{D}[M-1]\widehat{\mathbf{A}}^T)_{k\ell}]^T \quad (9) \\ &= (\mathbf{I}_M \otimes (\widehat{\mathbf{A}}_{k,:} \star \widehat{\mathbf{A}}_{\ell,:}))\mathbf{d} \end{aligned}$$

for $k, \ell = 1, \ldots, d$ and $k > \ell$; $\star$, $\otimes$ denote the element-wise and Kronecker products (resp.), and $\widehat{\mathbf{A}}_{k,:}$ is the $k-$th row of $\widehat{\mathbf{A}}$ (the arguments $(\boldsymbol{\theta})$ are omitted for brevity in (8)).

Note that this WLS criterion is generally different from our initial WLS criterion (2). We shall elaborate on this difference shortly. Nevertheless, a Gauss iterative method [4] can be applied in order to minimize this nonlinear WLS criterion (7) as follows:

$$\boldsymbol{\theta}^{[j+1]} = \boldsymbol{\theta}^{[j]} + [\mathbf{F}_j^T \mathbf{W} \mathbf{F}_j]^{-1} \mathbf{F}_j^T \mathbf{W} [\widehat{\mathbf{r}}_{\mathbf{y}} - \mathbf{f}(\boldsymbol{\theta}^{[j]})] \quad (10)$$

where j is the nested Gauss-iteration index and $\mathbf{F}_j = \partial\mathbf{f}(\boldsymbol{\theta})/\partial\boldsymbol{\theta}|_{\boldsymbol{\theta}=\boldsymbol{\theta}^{[j]}}$ (assumed to have full rank). Exploiting the block-diagonality of $\mathbf{W}$, (10) can be rewritten as

$$\begin{aligned} \boldsymbol{\theta}^{[j+1]} &= \boldsymbol{\theta}^{[j]} + \left\{ \sum_{k>\ell}^{d} \mathbf{F}_{k\ell}^{[j]T} \mathbf{W}_{k\ell} \mathbf{F}_{k\ell}^{[j]} \right\}^{-1} \\ &\quad \cdot \left\{ \sum_{k>\ell}^{d} \mathbf{F}_{k\ell}^{[j]T} \mathbf{W}_{k\ell} \left(\widehat{\mathbf{r}}_{k\ell} - \mathbf{f}_{k\ell}(\boldsymbol{\theta}^{[j]}) \right) \right\} \quad (11) \end{aligned}$$

where $\mathbf{F}_{k\ell}^{[j]}$ is the derivative (matrix) of $\mathbf{f}_{k\ell}$ with respect to $\boldsymbol{\theta}$, whose elements can be easily shown to be given by

$$\left.\frac{\partial \mathbf{f}_{k\ell}(\boldsymbol{\theta})}{\partial \widehat{\mathbf{A}}_{pq}}\right|_{\boldsymbol{\theta}=\boldsymbol{\theta}^{[j]}} = (\delta_{kp}\widehat{\mathbf{A}}_{\ell q}^{[j]} + \delta_{\ell p}\widehat{\mathbf{A}}_{kq}^{[j]})(\mathbf{I}_M \otimes \mathbf{e}_q^T)\mathbf{d} \quad (12)$$

δ_{kp} denoting Kronecker's delta and $\mathbf{e}_q$ the q-th column of $\mathbf{I}_d$.

Assume now, that the initial condition (in the internal Gauss iterations) for $\boldsymbol{\theta}$ is selected as $\widehat{\mathbf{A}}^{[0]} = \mathbf{I}_d$. The linear system for $\boldsymbol{\theta}^{[1]}$ is nicely decoupled in this case, so that elements of $\widehat{\mathbf{A}}^{[1]}$ can be obtained by solving the following $d(d-1)/2$ systems of 2×2 for $k, \ell = 1, \ldots, d$, $k > \ell$:

$$\begin{bmatrix} \widehat{\mathbf{A}}^{[1]}_{k\ell} \\ \widehat{\mathbf{A}}^{[1]}_{\ell k} \end{bmatrix} = \begin{bmatrix} \widehat{\mathbf{r}}^T_{\ell\ell}\mathbf{W}_{k\ell}\widehat{\mathbf{r}}_{\ell\ell} & \widehat{\mathbf{r}}^T_{kk}\mathbf{W}_{k\ell}\widehat{\mathbf{r}}_{\ell\ell} \\ \widehat{\mathbf{r}}^T_{kk}\mathbf{W}_{k\ell}\widehat{\mathbf{r}}_{\ell\ell} & \widehat{\mathbf{r}}^T_{kk}\mathbf{W}_{k\ell}\widehat{\mathbf{r}}_{kk} \end{bmatrix}^{-1} \begin{bmatrix} \widehat{\mathbf{r}}^T_{\ell\ell}\mathbf{W}_{k\ell}\widehat{\mathbf{r}}_{k\ell} \\ \widehat{\mathbf{r}}^T_{kk}\mathbf{W}_{k\ell}\widehat{\mathbf{r}}_{k\ell} \end{bmatrix} \tag{13}$$

where we have used the relation $(\mathbf{I}_M \otimes \mathbf{e}_q^T)\mathbf{d} = \widehat{\mathbf{r}}_{qq}$. Computed for all pairs (k, ℓ), (13) realizes one update of $\widehat{\mathbf{A}}$ at complexity $O(M^2d^2)$. In the case of uniform weights, the complexity is only $O(Md^2)$.

To proceed, rather than employ (11) directly for $j = 1$ (which would no longer enjoy the decoupling induced by $\widehat{\mathbf{A}}^{[0]}$), we simply use $\widehat{\mathbf{A}}^{[1]}$ to update $\widehat{\mathbf{V}}^{[i]}$ as $\widehat{\mathbf{V}}^{[i+1]} = (\widehat{\mathbf{A}}^{[1]})^{-1}\widehat{\mathbf{V}}^{[i]}$. In other words, we merely employ a single Gauss iteration nested within each outer iteration.

Convergence of the algorithm is nearly quadratic, as inherited from the Gauss iterations. Simulations confirm very good global convergence even in high dimensions, outperforming all other algorithms in terms of speed, with comparable estimation accuracy (in the unweighted version).

As mentioned earlier, the proposed algorithm does not strictly minimize the WLS criterion (2). Indeed, we shall see in simulations that QAJD sometimes attains a lower value thereof. Nevertheless, the simulations also show that the difference between the UWAJD and QAJD solutions is usually very minor (and actually nearly vanishes when the residual LS diagonalization error is small).

3. SIMULATIONS

3.1. AJD with uniform weights

We generated $M = 10$ matrices of dimension $d \times d$ with $d = 10, 100$ as follows (in Matlab® notations): $\mathbf{R_s}[1] = \text{eye}(d)$, $\mathbf{R_s}[k] = \text{diag}(\text{rand}(d,1)+1)$ for $k = 2, \ldots, M$, and $\mathbf{R_x}[k] = \mathbf{A}\mathbf{R_s}[k]\mathbf{A}^T + \sigma/2(\mathbf{N}[k] + \mathbf{N}^T[k])$, where $\mathbf{A}$ is a mixing matrix, $\mathbf{N}[k]$ is a noise matrix generated as $\mathbf{N}[k] = \text{randn}(d,d)$, and σ is a free parameter. Since some algorithms in the comparative study (FFDIAG and LLAJD) are sensitive to badly conditioned mixing matrices, we chose $\mathbf{A}$ to be unitary, generated as $[\mathbf{A}, \text{Aux}] = \text{qr}(\text{randn}(d,d))$.

Fig. 1 shows typical convergence patterns: The criterion (5) with uniform weights, as well as Pham's criterion (6), vs. iteration number for the proposed algorithm (denoted UWAJD), for FFDIAG by Ziehe *et al.* [7], and for Pham's LLAJD [8] for $d = 10$ and $\sigma = 0.2$, and for $d = 100$ and $\sigma = 10^{-2}$. The computation times for $d = 100$ were 2.9s for UWAJD (15 iterations), 32s for LLAJD (15 iterations), 8.7s for FFDIAG (30 iterations) and 126s for QAJD (100 iterations). Algorithm FAJD was not run for $d = 100$, since its complexity is $O(Md^4)$ per iteration compared to $O(Md^3)$ of the other algorithms (namely, 100 times slower in this case).

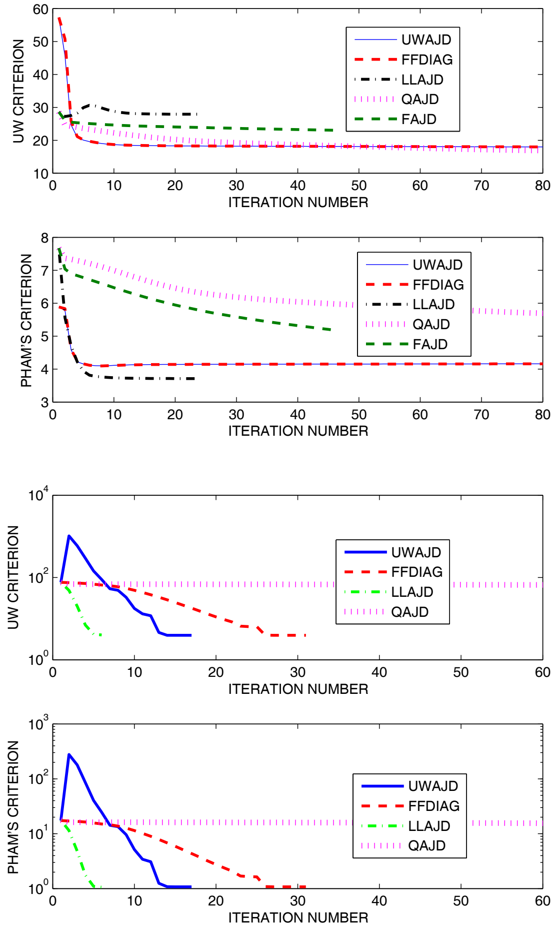

Fig. 1. The WLS (5) and Pham's (6) criteria, resp., vs. the iteration number for $\sigma = 0.2$, $d = 10$ and $\sigma = 0.01$ and $d = 100$, resp.

When running a similar experiment with sign-indefinite matrices (generated as $\mathbf{R_s}[k] = \text{diag}(1 - 2\,\text{rand}(d,1))$), we observed the same speed of convergence of UWAJD and FFDIAG, but a significantly faster convergence of QAJD (Pham's LLAJD had to be excluded from this experiment since it cannot accommodate sign-indefinite matrices), still with UWAJD being significantly faster than the others. Overall running time (for $d = 100$) of QAJD was 36.4 s.

To study the separation performance, we also measured the resulting implied Interference to Source Ratio (ISR) from the obtained overall mixing-unmixing matrix $\widehat{\mathbf{V}}\mathbf{A}$, averaged over 100 trials. We used sign-indefinite matrices and compared UWAJD, FFDIAG, QAJD and FAJD, obtaining negligible differences between the four.

3.2. Separation performance with non-uniform weights

The example presented in this subsections demonstrates an advantages of the non-uniform weighting of WAJD compared to the uniform weighting of UWAJD, FFDIAG and QAJD. Here the WAJD algorithm was used to implement WASOBI [2] according to the scheme in Fig. 2, so as to separate a linear mixture of 100 independent Gaussian AR(10) processes with distinct spectra. The $\mathbf{R}_{\mathbf{x}}[k]$ matrices are sample covariance matrices of the mixture at lags τ, for $\tau = 0, \ldots, 10$. The AR processes had poles at $p_k^{(i)} e^{\pm j\pi k/6}$, $k = 1, \ldots, 5$ where $p_k^{(i)} \in \{0.6\rho, 0.85\rho, 0.95\rho\}$ and ρ is a free parameter. Among the $3^5 = 243$ possible distinct AR processes, which differ in the modulus of at least one pair of complex conjugate poles, 100 processes were chosen for the test. The parameter ρ allows to tune the spectral dynamic range of the sources. For small ρ, the sources' power spectra are flat, similar to each other and therefore hard to separate. The spectra become more distinct as 0.95ρ approaches 1. Sources of length $N = 16000$ were mixed by random matrices (with a condition number ≤ 5) in 100 independent trials. Fig. 3 shows the inverted mean ISR, as well as the corresponding theoretical ISR bound (computed from the corresponding Cramér-Rao lower bound [11]), vs. ρ. The performance attained with non-uniform weighting is also shown for reference. The weighted algorithm is nearly statistically efficient (achieving the CRB) for $\rho > 0.65$ (with this data length). The ISR's of the algorithms FFDIAG and QAJD are not shown, as they are undistinguishable by naked eye from the results obtained by UWAJD. The average computation times were 3.4s for the initial separation by UWAJD. Each application of WAJD required about 3s and the entire procedure required 27s of CPU time (in Matlab® v.7 on the PC).

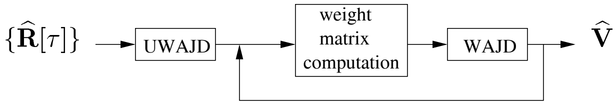

Fig. 2. A general scheme of WASOBI.

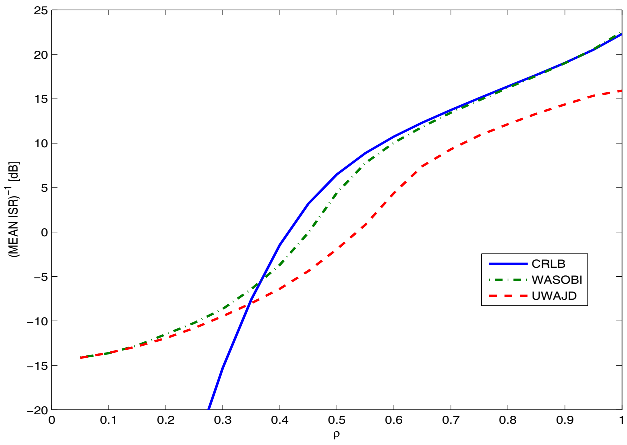

Fig. 3. Inverted mean ISR of 100 AR sources separated by WASOBI and UWAJD as a function of a parameter ρ.

4. CONCLUSIONS

The proposed algorithm outperforms all its competitors in terms of speed, with comparable estimation performance. In addition, it can incorporate weight matrices, which allow to optimize the performance in some applications. In particular, the algorithm allows fast implementation of WASOBI, proven to exhibit significantly improved performance in separation of stationary Gaussian processes with distinct spectra[2].

5. REFERENCES

[1] J.-F. Cardoso, A. Souloumiac, "Jacobi angles for simultaneous diagonalization", *SIAM Journal on Matrix Analysis and Applications*, vol. 17, pp. 161-164, 1996.

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

[3] E. Doron and A. Yeredor, "Asymptotically Optimal Blind Separation of Parametric Gaussian sources", *Proc. of ICA 2004*, C.G. Puntonet and A. Prieto (Eds.), pp. 390-397, 2004.

[4] H. W. Sorenson, *Parameter estimation*, New York:Dekker,1980.

[5] 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.

[6] P. Tichavský, E. Doron, A. Yeredor and J. Nielsen, "A Computationally Affordable Implementation of An Asymptotically Optimal BSS Algorithm for AR Sources", *Proc. EUSIPCO-2006*, Florence, Italy, September 4-8, 2006.

[7] A. Ziehe, P. Laskov, G. Nolte, and K.-R. Müller, "A Fast Algorithm for Joint Diagonalization with Non-orthogonal Transformations and its application to Blind Source Separation," *Journal of Machine Learning Research*, Vol. 5, pp. 777-800, 2004.

[8] D. T. Pham, "Joint Approximate Diagonalization of Positive Definite Matrices", *SIAM J. on Matrix Anal. and Appl.*, vol. 22, no. 4, pp. 1136-1152, 2001.

[9] R. Vollgraf and K. Obermayer, "Quadratic optimization for simultaneous matrix diagonalization", *IEEE Tr. Signal Processing*, vol. 54, no.9, pp. 3270-3278, Sep. 2006.

[10] Xi-Lin Li and X.D. Zhang, "Nonorthogonal joint diagonalization free of degenerate solutions", *IEEE Tr. Signal Processing*, vol. 55, no.5, pp. 1803-1814, May 2007.

[11] E. Doron, A. Yeredor and P. Tichavský, "Cramér-Rao Lower Bound for Blind Separation of Stationary Parametric Gaussian Sources", *IEEE Signal Processing Letters*, vol. 14, no.6, pp. 417-420, June 2007.

[12] P. Tichavský, matlab code for UWAJD, WAJD and WASOBI, `http://si.utia.cas.cz/Tichavsky.html`.