# A new General Weighted Least-Squares Algorithm for Approximate Joint Diagonalization

M. Congedo, C. Jutten, R. Sameni, C. Gouy-Pailler

GIPSA-lab (Grenoble Images Parole Signal Automatique) :
UMR5216 – CNRS (Centre National de la Recherche Scientifique) - Université Joseph Fourier, Université Pierre Mendès-France, Université Stendhal, INPG (Institut Polytechnique de Grenoble), Grenoble, FRANCE

marco.congedo@gmail.com

**Abstract**

Independent component analysis (ICA) and other blind source separation (BSS) methods are important processing tools for multi-channel processing of electroencephalographic data and have found numerous applications for brain-computer interfaces. A number of solutions to the BSS problem are achieved by approximate joint diagonalization (AJD) algorithms, thus the goodness of the solution depends on them. We present a new least-squares AJD algorithm with adaptive weighting on the separating vectors. We show that it has good properties while keeping the greatest generality among AJD algorithms; no constraint is imposed either on the input matrices or on the joint diagonalizer to be estimated. The new cost function allows interesting extensions that are now under consideration.



## 1 Introduction

Given a set of matrices $\mathbb{C}:\{\boldsymbol{C}_k, k=1\ldots K\}$, $K>2$, the approximate joint diagonalization (AJD) consists in finding a matrix $\boldsymbol{B}$ such that all $K$ products $\boldsymbol{B}\boldsymbol{C}_k\boldsymbol{B}^T$ result in matrices as close as possible to diagonal form. The AJD is an important algebraic tool extending the generalized eigenvalue problem (two-matrix diagonalization). As such, it is enjoying considerable interest and several efficient algorithms have been proposed [1-9]. In the context of brain-computer interface (BCI) the AJD provides a natural extension of the common spatial pattern to multi-class feature extraction [10]. Furthermore, since many matrices can be jointly diagonalized, one may optimize the spatial filter not only with respect to the signal diversity across classes [10], but also with respect to other kinds of signal diversity such as coloration and non-stationarity [11].

Recently a least-squares (LS) AJD algorithm has been proposed almost simultaneously in [6] and [8]. This algorithm does not impose restrictions either on the input matrices $\boldsymbol{C}_k$ (e.g., real, positive-definite, symmetric, etc.) or on the joint diagonalizer $\boldsymbol{B}$ (e.g., orthogonality), thus it is the most flexible among existing AJD algorithms. In [7] a similar LS idea has been used to perform simultaneous joint diagonalization and zero-diagonalization on two matrix sets, an approach that suits time-frequency data expansions. More generally, AJD algorithms are well adapted to expansion of the signal in several dimensions, enhancing the ability of capturing the source of diversity in a given data-set, hence offering a powerful approach for feature extraction. We anticipate that AJD algorithms will acquire a prominent role in feature extraction methods for BCI and we feel that a general approach may prove advantageous, which motivated us pursue further LS algorithms. The criterion used in [6] and [8] is

$$\Im^{OFF}(\boldsymbol{B}) = \sum_k \left\| Off\left(\boldsymbol{B}\boldsymbol{C}_k\boldsymbol{B}^T\right)\right\|^2, \qquad (1)$$

where $\|\ \|$ indicates the Frobenius norm and the *Off* operator zeros the diagonal entries of the matrix argument. The minimization of this criterion with respect to $\boldsymbol{B}$ evidently yields an AJD solution in the LS sense.

## 2 Method

For simplicity of exposition in the following we assume that the $N$-dimensional input matrices $\boldsymbol{C}_k$ are real and square, but not necessarily symmetric. The non-square/complex case is easily derived thereupon. We propose a weighted and normalized version of (1) given by the minimization of

$$\wp^{Off}\left(\boldsymbol{B}\right)=\frac{\sum_k\left\|Off\left(\boldsymbol{WBC}_k\boldsymbol{B}^T\boldsymbol{W}\right)\right\|^2}{\sum_k\left\|\left(\boldsymbol{WBC}_k\boldsymbol{B}^T\boldsymbol{W}\right)\right\|^2}, \tag{2}$$

where $\boldsymbol{W}$ is an $N$-dimensional diagonal matrix holding the weights for each row vector of $\boldsymbol{B}$. Since

$\sum_k\left\|\left(\boldsymbol{WBC}_k\boldsymbol{B}^T\boldsymbol{W}\right)\right\|^2=\sum_k\left\|Off\left(\boldsymbol{WBC}_k\boldsymbol{B}^T\boldsymbol{W}\right)\right\|^2+\sum_k\left\|Diag\left(\boldsymbol{WBC}_k\boldsymbol{B}^T\boldsymbol{W}\right)\right\|^2$, where the *Diag* operator zeros the off-diagonal entries of the matrix argument, the minimization of (2) is equivalent to the maximization of

$$\wp^{Diag}\left(\boldsymbol{B}\right)=\frac{\sum_k\left\|Diag\left(\boldsymbol{WBC}_k\boldsymbol{B}^T\boldsymbol{W}\right)\right\|^2}{\sum_k\left\|\left(\boldsymbol{WBC}_k\boldsymbol{B}^T\boldsymbol{W}\right)\right\|^2}. \tag{3}$$

Denoting by $\boldsymbol{b}_i^T$ the i$^{th}$ row vector of $\boldsymbol{B}$ and by $\boldsymbol{b}_i$ its transpose (still the row vector but in column representation) and following [7] we expand (3) such as

$$\sum_k\left\|Diag\left(\boldsymbol{WBC}_k\boldsymbol{B}^T\boldsymbol{W}\right)\right\|^2=\sum_{k=1}^K\sum_{i=1}^N\left(w_i\boldsymbol{b}_i^T\boldsymbol{C}_k\boldsymbol{b}_i\,w_i\right)^2=\sum_{i=1}^N w_i\boldsymbol{b}_i^T\left[\sum_{k=1}^K\left(\boldsymbol{C}_k\boldsymbol{b}_i\,w_i^2\boldsymbol{b}_i^T\boldsymbol{C}_k\right)\right]\boldsymbol{b}_i w_i \tag{4}$$

and

$$\sum_k\left\|\left(\boldsymbol{WBC}_k\boldsymbol{B}^T\boldsymbol{W}\right)\right\|^2=\sum_{k=1}^K\sum_{i=1}^N\sum_{j=1}^N\left(w_i\,\boldsymbol{b}_i^T\boldsymbol{C}_k\boldsymbol{b}_j w_j\right)^2=\sum_{i=1}^N w_i\boldsymbol{b}_i^T\left[\sum_{k=1}^K\left(\boldsymbol{C}_k\boldsymbol{B}^T\boldsymbol{W}^2\boldsymbol{B}\boldsymbol{C}_k^T\right)\right]\boldsymbol{b}_i w_i\,. \tag{5}$$

Now by defining

$$\boldsymbol{M}_i=\sum_{k=1}^K\left(\boldsymbol{C}_k\boldsymbol{b}_i\,w_i^2\boldsymbol{b}_i^T\boldsymbol{C}_k\right)\text{ and} \tag{6}$$

$$\boldsymbol{M}=\sum_{k=1}^K\left(\boldsymbol{C}_k\boldsymbol{B}^T\boldsymbol{W}^2\boldsymbol{B}\boldsymbol{C}_k^T\right) \tag{7}$$

and substituting (4) and (5) in $\wp^{Diag}\left(\boldsymbol{B}\right)$ (3), we can write

$$\wp^{Diag}\left(\boldsymbol{B}\right)=\sum_i\frac{w_i\boldsymbol{b}_i^T\boldsymbol{M}_i\boldsymbol{b}_i\,w_i}{tr\left(\boldsymbol{WBM}\,\boldsymbol{B}^T\boldsymbol{W}\right)}. \tag{8}$$

Similarly as in [6-8] the optimization of $\boldsymbol{B}$ according to (8) may proceed iteratively row-by-row. For each vector of $\boldsymbol{B}$ a step consists in sphering $\boldsymbol{M}$ (fixing the denominator) and finding the optimal

direction $\boldsymbol{b}_i$ maximizing $\boldsymbol{b}_i^T \boldsymbol{M}_i \boldsymbol{b}_i$ . Updating $\boldsymbol{b}_i$ will results in different $\boldsymbol{M}$ and $\boldsymbol{M}_i$, to which a new $\boldsymbol{b}_i$ will correspond and so on iteratively. The process sequentially applies to all $N$ vectors of $\boldsymbol{B}$ within each iteration, resulting in mutual restrictions. The following SWDiag (sphered weighted diagonalization) algorithm makes use of adaptive weighting:

Initialize $\boldsymbol{B}$ by a clever guess or by $\boldsymbol{I}$ (the identity matrix) if no guess is available. Initialize $\boldsymbol{W}$ by $\boldsymbol{I}$.

***While not*** Convergence ***do***

***For*** $i$=1 to $N$ ***do twice***

(A: Sphering): Find $\boldsymbol{H}$ such that $\boldsymbol{HMH}^T = \boldsymbol{I}$

(B: Optimal Direction): find the principal eigenvector $\boldsymbol{u}_i$ and associated eigenvalue $\lambda_i$ of $\boldsymbol{HM}_i \boldsymbol{H}^T$

Update the $i^{\text{th}}$ row of $\boldsymbol{B}$ as $\boldsymbol{b}_i^T \leftarrow \boldsymbol{u}_i^T \boldsymbol{H}$

***End For***

Update all diagonal elements of $\boldsymbol{W}$ as $w_i \leftarrow \lambda_i^{-1/2}$ and normalize them so as $\sum_i w_i^2 = N, \ i = \{1 \ldots N\}$

***End While***

This family of algorithms has good convergence properties (see [6-8]). Note that $\boldsymbol{M}$(7) and $\boldsymbol{M}_i$ (6) are updated at each pass of the *for* loop. If each pass of the *for* loop is not repeated twice, as suggested, the algorithm still converges, but the stopping criterion (see below) displays a "saw" (non-monothonically decreasing) behavior. The eigenvalues associated with the principal eigenvectors of Step B are by definition comprised between 0 and 1.0 and equals 1.0 iff the off criterion is zero, which happens if the input matrices can be diagonalized exactly, that is, if they have exactly the same eigenstructure. If not, or more in general due to sampling error, which will always happen in practice, the eigenvalues will converge to a value smaller than 1.0. This ensures numerical stability of the algorithm and provides the rationale for the weighting scheme: at each iteration the diagonalization achieved by each row vector of $\boldsymbol{B}$ is proportional to the magnitude of the associated eigenvalue. In Eq. (7) $\boldsymbol{C}_k \boldsymbol{B}^T \boldsymbol{W}^2 \boldsymbol{B} \boldsymbol{C}_k^T$ can be written as $\boldsymbol{C}_k \sum_i w_i^2 \boldsymbol{b}_i \boldsymbol{b}_i^T \boldsymbol{C}_k^T$, thus we see that the adaptive weighting emphasizes the search of vectors attaining a lower eigenvalue at the expense of those attaining an higher eigenvalue, which steers the algorithm toward a more balanced solution. See also the discussion on balanced solutions in [6]. As for the stopping criterion of the algorithm, we stop as soon as the change of the $N$ eigenvalues $\boldsymbol{\lambda}_i$ is negligible.

Each eigenvector (optimal direction) in step B can be successfully updated toward convergence if matrix $\boldsymbol{M}_i$ does not have multiple maximum eigenvalues. In this case the optimal direction eigenvector cannot be found uniquely. This is also the case of the LS algorithm of [6] and [8], which minimizes

$$\Im^{OFF}(\boldsymbol{B}) = \sum_k \left\| Off\left(\boldsymbol{B}\boldsymbol{C}_k\boldsymbol{B}^T\right) \right\|^2 \text{ with constraint } Diag\left(\boldsymbol{B}\boldsymbol{E}\boldsymbol{B}^T\right) = \boldsymbol{I}, \qquad (9)$$

where $\boldsymbol{E}$ is any positive definite matrix. Since the matrix $\boldsymbol{E}$ is disjoint to matrix $\boldsymbol{B}$, their algorithm consists in performing the sphering once at the beginning and then iteratively finding the optimal directions by minor component analysis and scaling to match the constraint. However, if after sphering there are multiple minor eigenvalues this algorithm is definitely trapped and fails. On the other hand in our optimization scheme the matrices $\boldsymbol{HM}_i \boldsymbol{H}^T$ change at each pass due to the fact that the sphering step (Step A) depends on the previous estimation of $\boldsymbol{B}$ (7), thus our algorithm may fail only if the multiplicity of maximum eigenvalues happens close to convergence, whence the changes caused by the sphering update are small and cannot correct the multiplicity issue anymore.

# 3 Results

We compared our SWDiag algorithm and its unweighted version SDiag (obtained setting all weights to 1.0 and not updating them at each iteration) to the well-established FDiag algorithm of [5] and QDiag of [6]. Referring to (9), we use $\boldsymbol{E}=\boldsymbol{I}$ for QDiag. We performed simulations using synthetic input matrices and a real-data example.

For the synthetic matrices simulation we generated 12 6-dimensional square diagonal matrices with each diagonal entry distributed as a chi-squares random variable with one degree of freedom. Each of these matrices, named $\boldsymbol{D}_k$, may represent the error-free covariance matrix of six independent standard Gaussian processes (zero mean and unit variance). The 12 input matrices were obtained as

$$\boldsymbol{C}_k=\boldsymbol{A}\boldsymbol{D}_k\boldsymbol{A}^T, \tag{10}$$

$k$:{1…12}, where each entry of the 6-dimensional square mixing matrix $\boldsymbol{A}$ is randomly distributed as a standard Gaussian.

We considered three cases:

- *No perturbation:* the exact AJD problem as described by (10)

- *Perturbation of the Mixing Matrix*: input matrices were generated as $\boldsymbol{C}_k=\boldsymbol{A}_k\boldsymbol{D}_k\boldsymbol{A}_k^T$, where each entry of the mixing matrix $\boldsymbol{A}$ in (10) is perturbated as $\boldsymbol{A}_{kij} \leftarrow \boldsymbol{A}_{ij} + \varphi\,\zeta\,\boldsymbol{A}_{ij}$, where φ is +1 or -1 with equal probability and ζ is uniformly distributed in [0.001…0.1], for all $k$=1…$K$ and for all $i, j$=1…N.

- *Perturbation of Independence:* with probability 0.2 each off-diagonal symmetric pair of the input matrices $\boldsymbol{D}_k$ is perturbated as $D_{kij}=D_{kji} \leftarrow \varphi\,(\sqrt{D_{kii}}\sqrt{D_{kjj}}) / \delta$, where φ is +1 or -1 with equal probability and δ uniformly distributed in {1…8}, for all $k$ and $i>j$.

Given true mixing $\boldsymbol{A}$, each AJD algorithm estimates demixing $\boldsymbol{B}$, which should approximate the inverse of $\boldsymbol{A}$ out of row scaling (including sign) and permutation. Then, matrix $\boldsymbol{G}=\boldsymbol{B}\boldsymbol{A}$ should equal a scaled permutation matrix. At each repetition we computed the performance index such as

$$\text{Performance Index}=\frac{2(N\text{-}1)\sum_{\mathrm{i}}\sum_{\mathrm{j}}G_{ij}^2}{\sum_{\mathrm{i}}\max_j\left(G_{ij}^2\right)+\sum_{\mathrm{j}}\max_i\left(G_{ij}^2\right)}, \tag{11}$$

which is positive and reaches its maximum 1.0 iff $\boldsymbol{G}$ has only one non-null elements in each row and column. The mean and standard deviation across 500 repetitions are reported in table 1.

Table 1 : Mean and standard deviation (within parentheses) of the performance index (11) across 500 repetitions of the synthetic input matrices simulation with and without perturbation. The higher the index the better.
*: QDiag resulted in a false solution 87 out of 500 repetitions in this case.

| Perturbation | **FDiag** | **QDiag** | **SDiag** | **WSDiag** |
|---|---|---|---|---|
| **None** | 0.9999 (0.0000) | 0.9999 (0.0011) | 1.0 (0.0) | 1.0 (0.0) |
| **Mixing** | 0.9926 (0.0107) | 0.9927 (0.0104) | 0.9915 (0.0119) | 0.9928 (0.0102) |
| **Independence** | 0.8057 (0.0676) | * | 0.7961 (0.0698) | 0.8002 (0.0683) |

The real data example concerns an eyes open EEG recording comprising 19 electrodes and 11 seconds sampled at 128 samples per second. The recording (Figure 1 top) displays a rapid sequence of

eye blinks and bilateral jaw muscle contamination visible at temporal leads T3 and T4. We performed AJD of 44 Fourier co-spectral matrices corresponding to frequencies 1 Hz to 44 Hz in 1 Hz steps. EEG data was previously whitened and the 16 most energetic components were retained. Such an AJD procedure corresponds to exploiting the different coloration of EEG source components. In fact, the AJD of cospectral matrices successfully estimates the inverse of the mixing matrix if the source components have non-proportional power spectra (characteristic coloration). Out of random permutations, FDiag, QDiag and WSDiag gave very similar results (Fig 1).

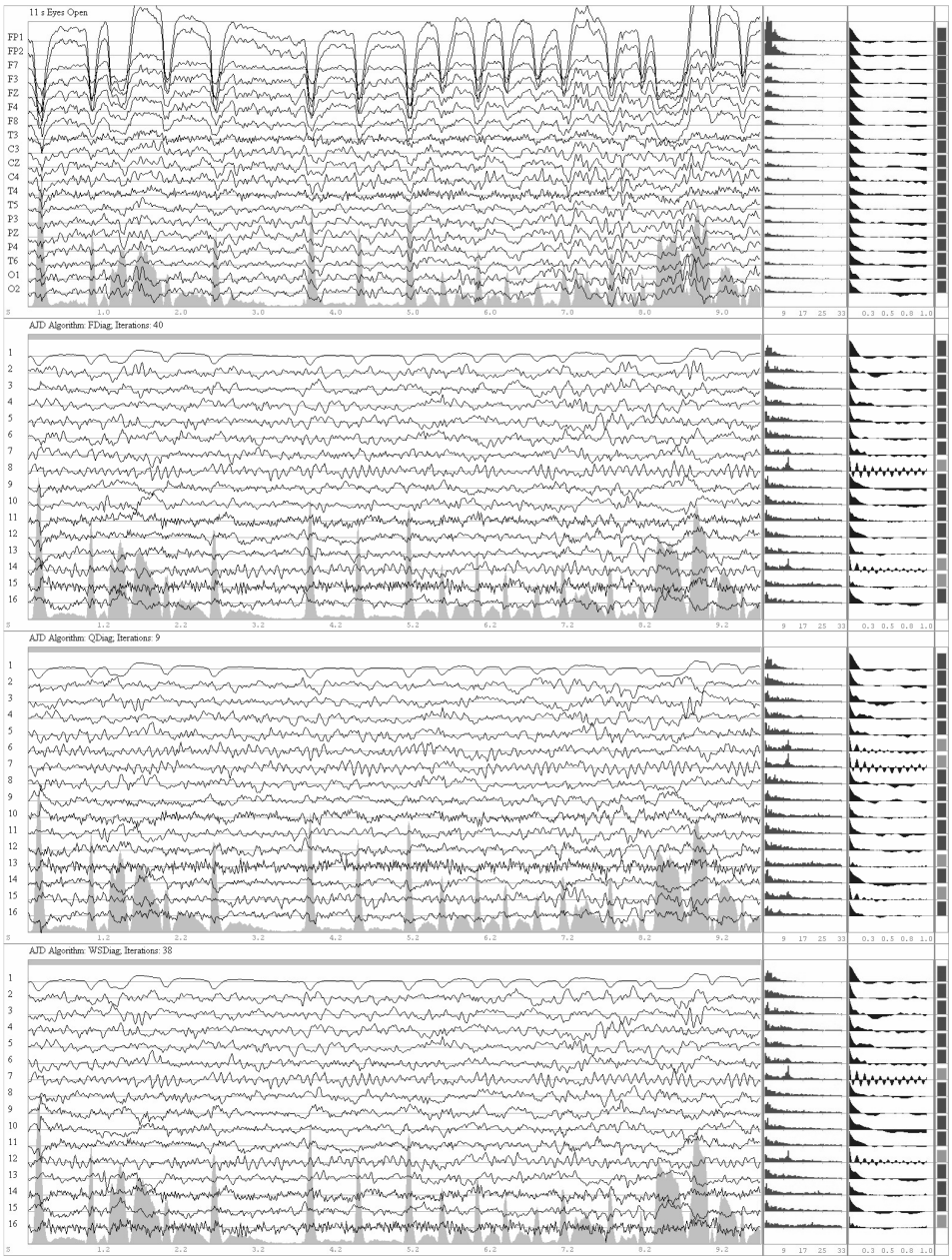


Figure 1. Top: about 9 s of a 11 s epoch extracted from the raw EEG recording of a 12 y.o. male. From left to right, electrode labels according to the 10-20 international system, raw EEG tracing (upward deflection is negative potential; the space between two horizontal centering lines is 70 µV), average power spectrum (from zero to 32 Hz; arbitrary units) and autocorrelation function (the space between two horizontal centering lines is autocorrelation = 1 in the upward direction and -1 in the downward direction). The gray shaded area in the background of EEG tracings is the global field power, the sum of the square of potentials across electrodes for each sample (arbitrary units). The next three plots are the sources estimated using FDiag, QDiag and WSDiag on the same set of Fourier cospectral matrices. For all methods sources were standardized (unit variance) and plotted on the same scale.

## 4 Discussion

We have presented a new least-squares approximate joint diagonalization algorithm with adaptive weighting for the row vectors of the matrix to be estimated. Simulations on synthetic input matrices and a real-data example indicates the good performance of WSDiag when compared to FDiag and QDiag. Our new LS optimization scheme allows interesting manipulations, besides the adaptive weighting here proposed, which are now under investigation. We are currently considering weighting also the input matrices and solving block diagonalization problems. We are also working on the convergence properties of the algorithms and on its link to cost function (3).

## 5 Conclusion

The proposed AJD algorithm may prove useful for the extraction of electroencephalographic features. Application of source separation methods making use of AJD algorithms has been recently introduced in the brain-computer interface field [10-11] and appears a promising approach.

## References

[1] J.-F. Cardoso and A. Souloumiac, Jacobi Angles for Simultaneous Diagonalization. *SIAM Journal on Matrix Analysis and Applications*, 17(1):161-164, 1996.

[2] D. T. Pham. Joint Approximate Diagonalization of Positive Definite Matrices. *SIAM Journal on Matrix Analysis and Applications*, 22(4):1136-1152, 2001.

[3] E. Moreau, A generalization of joint-diagonalization criteria for source separation, *IEEE Transactions on Signal Processing*, 49(3):530-541, 2001.

[4] A. Yeredor. Non-orthogonal joint diagonalization in the least-squaress sense with application in blind source separation Signal Processing, *IEEE Transactions on Signal Processing*, 50(7):1545-1553, 2002.

[5] 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*, 5:801-818, 2004.

[6] R. Vollgraf and K. Obermayer, Quadratic optimization for simultaneous matrix diagonalization, *IEEE Transactions on Signal Processing*, 54(9):3270-3278, 2006.

[7] E. M. Fadaili, N. T. Moreau and E. Moreau, Nonorthogonal Joint Diagonalization/Zero Diagonalization for Source Separation Based on Time-Frequency Distributions, *IEEE Transactions on Signal Processing*, 55(5): 1673-1687, 2007.

[8] S. Degerine and E. Kane, A Comparative Study of Approximate Joint Diagonalization Algorithms for Blind Source Separation in Presence of Additive Noise, *IEEE Transactions on Signal Processing*, 55(6):3022-3031, 2007.

[9] X.-L. Li and X.D. Zhang. Nonorthogonal Joint Diagonalization Free of Degenerate Solution. *IEEE Transactions on Signal Processing*, 55(5):1803-1814, 2007.

[10] M. Grosse-Wentrup and M. Buss. Multi-class Common Spatial Pattern and Information Theoretic Feature Extraction. *IEEE Transactions on Biomedical Engineering* (in press).

[11] C. Gouy-Pailler, M. Congedo, M. Brunner, C. Jutten, and G. Pfurtscheller. Multi-Class Independent Common Spatial Patterns: Exploiting Energy Variations of Brain Sources. *Proceedings of the 4th International BCI Workshop,* (in press).