OPTIMUM ADAPTIVE BLIND SOURCE SEPARATION ALGORITHMS

George V. Moustakides

Institut National de Reserche en Informatique et en Automatique (INRIA), France.

ABSTRACT

Adaptive blind source separation algorithms are conventionally composed of two parts. The first, using second order statistics, is responsible for whitening the measured signals, whereas the second, based on nonlinear statistics, imposes independence and achieves the final separation. In this work we show that this two-part scheme is in fact not necessary. By proposing a general nonlinear adaptation model, we find conditions that lead to source separation and guarantee an overall desirable symmetric behavior of the algorithm. Furthermore, using a local performance measure, we optimize the general adaptation scheme and obtain algorithms that have optimum convergence rate. Finally we show that the proposed optimum schemes, except for trivial cases, cannot be put under the two-part classical scheme of the literature, suggesting that the latter is suboptimum.

1. INTRODUCTION

Blind source separation is the problem of recovering unobserved signals from their observed mixtures. The basic property one relies on to solve this problem is the assumption that the original unobserved signals are mutually independent. The term "blind" refers to the fact that there is no training involved and the algorithms that achieve separation use only the available measured mixtures.

The simplest blind source separation model considers a vector $S(n) = [s_1(n)\ s_2(n)\ \cdots\ s_K(n)]^t$ of K statistically independent signals that are instantaneously, linearly mixed, producing an observation vector $X(n)$ of the same length. The mixture is performed with the help of an unknown matrix $\mathbf{A}$ as follows

$$X(n) = \mathbf{A}S(n) \tag{1}$$

and the only assumption imposed on $\mathbf{A}$ is to be invertible.

In this paper we focus on blind adaptive techniques for the solution of the problem. Therefore let us assume that we are given sequentially measurements $X(n)$ that satisfy the mixture model in (1) and we are interested in estimating the original signal vector $S(n)$ by estimating the inverse of the matrix $\mathbf{A}$. More specifically we are interested in estimating a matrix $\mathbf{B}$ such that $\mathbf{C} = \mathbf{BA}$ is nonmixing, that is, $\mathbf{C}$ is a permutation like matrix with its nonzero entries being ± 1 instead of unity. The fact that $\mathbf{C}$ is not the identity matrix expresses a possible ambiguity in sign and in the ordering of the sources. However both drawbacks are not considered significant and in most cases in practice they can be corrected by simple means.

Blind adaptive algorithms that estimate $\mathbf{B}$, have the following form in the literature [2]

$$\hat{S}(n) = \mathbf{B}(n-1)X(n) \tag{2}$$

$$\mathbf{B}(n) = \mathbf{B}(n-1) - \mu\mathbf{H}(\hat{S}(n))\mathbf{B}(n-1), \tag{3}$$

where $\hat{S}(n)$ denotes the estimate of the signal vector $S(n)$ at time n; $\mathbf{B}(n)$ the estimate of $\mathbf{B}$ at time n; $\mathbf{I}$ the identity matrix and the matrix function $\mathbf{H}(Y)$ is of the form

$$\mathbf{H}(Y) = [YY^t - \mathbf{I}] + [YG^t(Y) - G(Y)Y^t]. \tag{4}$$

where, if $Y = [y_1\ \cdots\ y_K]^t$ then, $G(Y)$ denotes a vector function of the form $G(Y) = [g_1(y_1)\ g_2(y_2)\ \cdots\ g_K(y_K)]^t$, with $g_i(y)$ scalar nonlinear functions. We clearly distinguish the two parts in (4), the first involving only second order statistics and responsible for whitening the mixture of the signals and the second involving nonlinear statistics and responsible for the final separation of the sources.

For this algorithm to work we need to impose certain additional constraints. In particular we need to assume that all signals have a symmetric pdf of unit variance; and that all $g_i(y)$ are odd symmetric nonlinear functions of y. Under these assumptions and if *at most one* signal is Gaussian, then we have perfect reconstruction of the sources, provided of course that the algorithm does not diverge.

2. A GENERAL ALGORITHMIC MODEL

Let us now consider the case where $\mathbf{H}(Y)$ is a general matrix function, not necessarily of the form of (4), and examine under what conditions the corresponding algorithm can successfully separate sources. Clearly we are anticipating that the general model will lead to more efficient algorithmic structures than the classical scheme of the literature.

The algorithm we are interested in is described by (2), (3), with $\mathbf{H}(\cdot)$ being a general matrix function. As it is frequently remarked in the literature, it is more convenient,

0-7803-7402-9/02/$17.00 ©2002 IEEE

from an analysis point of view, to study a normalized version of this algorithm. Specifically by multiplying (3) with the matrix $\mathbf{A}$ from the right, the algorithm can be equivalently written as

$$\hat{S}(n) = \mathbf{C}(n-1)S(n) \quad (5)$$
$$\mathbf{C}(n) = \mathbf{C}(n-1) - \mu\mathbf{H}(\hat{S}(n))\mathbf{C}(n-1), \quad (6)$$

where $\mathbf{C}(n) = \mathbf{B}(n)\mathbf{A}$. Here we need to examine the convergence properties of the matrix $\mathbf{C}(n)$ towards one of the possible limiting, permutation like, matrices $\mathbf{C}$ mentioned in the introduction. In fact we must ensure, by a proper choice of $\mathbf{H}(\cdot)$, that these matrices become possible limits of the corresponding adaptation. Furthermore, we need to also to ensure a uniform overall convergence performance of the algorithm, independently of the limit the algorithm is converging to.

For simplicity we limit ourselves to the case of two signals, i.e. $K = 2$; extending our results to the general case is straightforward. For the two-signal case the matrix function $\mathbf{H}(\cdot)$ contains four elements (functions) $h_{ij}(y_1, y_2)$, $i, j = 1, 2$; the vector $\hat{S}(n) = [\hat{s}_1(n)\ \hat{s}_2(n)]^t$; while the signal vector $S(n) = [s_1(n)\ s_2(n)]^t$, where we recall that $s_i(n)$, $i = 1, 2$, are the two statistically independent source signals.

When $K = 2$ there are eight different $\mathbf{C}$ matrices that are acceptable limits for the adaptive algorithm, namely

$$\mathbf{C} = \begin{bmatrix} \pm 1 & 0 \\ 0 & \pm 1 \end{bmatrix}, \text{ or } \begin{bmatrix} 0 & \pm 1 \\ \pm 1 & 0 \end{bmatrix}, \quad (7)$$

corresponding to all different combinations in signs and ordering. Let us now impose a structure on $\mathbf{H}(\cdot)$ or equivalently on its elements $h_{ij}(y_1, y_2)$, $i, j = 1, 2$, in order to ensure separation of sources.

2.1. Correct Adaptation Limit

The first step is to ensure that the algorithm can converge to any of the eight possible $\mathbf{C}$ matrices defined in (7). From stochastic approximation theory [1] we know that the adaptation in (5), (6) can converge, *in the mean*, to one of the stable equilibrium matrices $\mathbf{C}_\infty$ that satisfy the equation

$$\mathbf{C}_\infty = \mathbf{C}_\infty - \mu\mathrm{E}\{\mathbf{H}(\mathbf{C}_\infty S(n))\}\mathbf{C}_\infty,$$

where $\mathrm{E}\{\cdot\}$ denotes expectation. Equivalently, if we consider only nonsingular solutions, we can write

$$\mathrm{E}\{\mathbf{H}(\mathbf{C}_\infty S(n))\} = 0. \quad (8)$$

We therefore conclude that, in order for the matrices in (7), to become possible limits of the adaptation we need to impose the constraint

$$\mathrm{E}\{\mathbf{H}(\mathbf{C}S(n))\} = 0,$$

for all $\mathbf{C}$ matrices listed in (7). One can easily show that this is equivalent to the following conditions applied to the elements of the matrix function $\mathbf{H}(\cdot)$

$$\begin{aligned} &\mathrm{E}\{h_{ij}(\pm s_1(n), \pm s_2(n))\} = \\ &\mathrm{E}\{h_{ij}(\pm s_2(n), \pm s_1(n))\} = 0, \ \ i, j = 1, 2. \end{aligned} \quad (9)$$

2.2. Symmetry in Trajectories

Since the eight possible limits are considered equivalent, we would like our algorithm to behave in a symmetric way regardless of the limit it converges to. In other words, if there is a trajectory leading to one limit, we would like to exist symmetric trajectories, with the same probability of appearance, leading to all other possible limits. It turns out that, because of the symmetry of the source pdfs, one can easily impose this symmetric behavior on the algorithm by selecting the following structure for the matrix function $\mathbf{H}(\cdot)$

$$\mathbf{H}(y_1, y_2) = \begin{bmatrix} h(y_1, y_2) & q(y_1, y_2) \\ q(y_2, y_1) & h(y_2, y_1) \end{bmatrix} \quad (10)$$

where $h(y_1, y_2)$, $q(y_1, y_2)$ functions that satisfy

$$h(y_1, y_2) \neq h(y_2, y_1), \ \ q(y_1, y_2) \neq q(y_2, y_1) \quad (11)$$

and

$$h(-y_1, y_2) = h(y_1, -y_2) = h(y_1, y_2) \quad (12)$$
$$q(-y_1, y_2) = q(y_1, -y_2) = -q(y_1, y_2). \quad (13)$$

With the help of (12), (13) the corresponding conditions in (9) of the previous subsection, become

$$\mathrm{E}\{h(s_1, s_2)\} = \mathrm{E}\{h(s_2, s_1)\} = 0 \quad (14)$$
$$\mathrm{E}\{q(s_1, s_2)\} = \mathrm{E}\{q(s_2, s_1)\} = 0, \quad (15)$$

where for simplicity we have dropped the time index n in the signals $s_i(n)$. Condition (14) remains a requirement, however Condition (15) *is always true* if $q(y_1, y_2)$ satisfies the odd symmetry property in (13). Summarizing, if the matrix function $\mathbf{H}(\cdot)$ satisfies Conditions (10)-(14) then the algorithm can converge to one of the possible $\mathbf{C}$ matrices and its convergence behavior is the same regardless of the limit it converges to.

Let us examine the classical adaptation scheme used in the literature and defined in (4). We realize that it is a special case of our general model with $h(y_1, y_2) = y_1^2 - 1$ and $q(y_1, y_2) = y_1 y_2 + g(y_1)y_2 - g(y_2)y_1$, satisfying all five conditions mentioned before.

3. AN ANALYTIC PERFORMANCE MEASURE

In an adaptive algorithm there are two quantities that are of primal importance, namely speed of convergence and steady

state estimation error. We will propose analytic expressions for both of them, for the case of small step size μ (which is the case of interest in practice). Because of the small gain assumption it will be possible to make use of powerful results coming from stochastic approximation theory, concerning performance of adaptive algorithms [1].

3.1. Speed of Convergence

The algorithm defined by (5), (6) is nonlinear and can converge, in the mean, to one of the possible matrices $\mathbf{C}$, under the assumptions of the previous section. Since its behavior is nonlinear, in order to measure its convergence speed, we can define an asymptotic convergence rate which, with the help of the stochastic approximation theory, can be shown to satisfy the following equality

$$-\lim_{n\to\infty}\frac{\log(\|\mathrm{E}\{\mathbf{C}(n)\}-\mathbf{C}\|)}{n}=\mu\min_i\{\mathrm{Re}(\lambda_i)\}+o(\mu),$$

where λ_i are the eigenvalues of the matrices $\mathbf{Q}_1$, $\mathbf{Q}_2$, with

$$\begin{aligned}\mathbf{Q}_1 &= \mathrm{E}\left\{\begin{bmatrix} h(s_1,s_2)\\ h(s_2,s_1)\end{bmatrix}[s_1 l_1(s_1)-1\ s_2 l_2(s_2)-1]\right\}\\ \mathbf{Q}_2 &= \mathrm{E}\left\{\begin{bmatrix} q(s_1,s_2)\\ q(s_2,s_1)\end{bmatrix}[s_1 l_2(s_2)\ s_2 l_1(s_1)]\right\},\end{aligned}$$

$l_i(y)=-\frac{f_i'(y)}{f_i(y)}$, and $f_i(y)$ being the pdf of source i. It is also known that for local stability of the equilibrium matrices $\mathbf{C}$, we need to have $\min_i\{\mathrm{Re}(\lambda_i)\}>0$.

3.2. Steady State Error Power

The second important quantity is the estimation error power at steady state, that is, $\lim_{n\to\infty}\mathrm{E}\{\|\hat{S}(n)-S(n)\|^2\}$. Using the independence of the sources and the fact that all sources are of unit variance, one can show that

$$\mathrm{E}\{\|\hat{S}(n)-S(n)\|^2\}=\mathrm{E}\{\|\mathbf{C}(n)-\mathbf{C}\|^2\}.$$

Again, because the step size μ is small, from stochastic approximation theory we have that

$$\lim_{n\to\infty}\mathrm{E}\{\|\mathbf{C}(n)-\mathbf{C}\|^2\}=\mu\,\mathrm{tr}\{\mathbf{P}_1+\mathbf{P}_2\}+o(\mu),$$

where $\mathrm{tr}\{\cdot\}$ denotes trace and $\mathbf{P}_1$, $\mathbf{P}_2$, are matrices satisfying the following Lyapunov equations

$$\mathbf{Q}_i\mathbf{P}_i+\mathbf{P}_i\mathbf{Q}_i^t=\mathbf{R}_i,\ \ i=1,2,$$

with

$$\begin{aligned}\mathbf{R}_1 &= \mathrm{E}\left\{\begin{bmatrix} h(s_1,s_2)\\ h(s_2,s_1)\end{bmatrix}[h(s_1,s_2)\ h(s_2,s_1)]\right\}\\ \mathbf{R}_2 &= \mathrm{E}\left\{\begin{bmatrix} q(s_1,s_2)\\ q(s_2,s_1)\end{bmatrix}[q(s_1,s_2)\ q(s_2,s_1)]\right\}.\end{aligned}$$

3.3. Efficacy of Adaptive Algorithms

When we compare adaptive algorithms there is no reason to make this comparison using the same step size μ in the algorithms involved. Since μ interferes directly in the convergence rate and the asymptotic steady state behavior, it is clear that the proper selection of μ, in each algorithm, is crucial for the comparison process.

A *fair* comparison method [3] consists in selecting the step size in each algorithm so that all algorithms under comparison have the same steady state error power (say σ^2); then examine which algorithm converges faster, that is, has the maximum convergence rate. Indeed if we select the step size in the way just described then, the rate of convergence of an algorithm, takes the form $\sigma^2\mathrm{Eff}$ where σ^2 is the common steady state error power and Eff is the *Efficacy* of the algorithm defined as

$$\mathrm{Eff}=\frac{\min_i\{\mathrm{Re}(\lambda_i)\}}{\mathrm{tr}\{\mathbf{P}_1+\mathbf{P}_2\}}.$$

Clearly, when we compare algorithms fairly, then the one with the maximum efficacy is considered as the best, since it has the largest convergence rate (for the same steady state error power).

There are several interesting points related to the efficacy. Notice first that this measure is independent of the step size μ and is only a function of the algorithm we like to analyze. Second, one can show that ratio of efficacies is approximately equal to ratio of steps required by the corresponding algorithms to converge. In other words if an algorithm has an efficacy which is $\alpha\%$ larger than the efficacy of another algorithm, then the first algorithm requires $\alpha\%$ less iterations to converge than the second.

4. ASYMPTOTICALLY OPTIMUM ALGORITHMS

Since in the previous section we defined an analytic performance measure, we can now attempt to maximize it in order to find optimum (fastest converging) algorithms.

Unfortunately the corresponding optimization problem we end up with, when we consider the general case, turns out to be complicated. There is however a special, and extremely interesting, case where the solution is possible. Specifically, if the sources have the same pdf, that is, if $f_1(y)=f_2(y)=f(y)$ then the two functions $h(y_1,y_2)$ and $q(y_1,y_2)$ that maximize the efficacy can be shown to be equal to

$$\begin{aligned}h(y_1,y_2) &= a(y_1 l(y_1)-1)\\ q(y_1,y_2) &= by_1 l(y_2)+cy_2 l(y_1)\end{aligned}$$

where $l(y)=\frac{f'(y)}{f(y)}$ and the constants a,b,c, are such that the two matrices $\mathbf{Q}_1,\mathbf{Q}_2$ have all their eigenvalues λ_i equal to unity.

Consider now the classical scheme with $h(y_1, y_2) = y_1^2 - 1$ and $q(y_1, y_2) = y_1 y_2 + g(y_1)y_2 - g(y_2)y_1$. If we compute its efficacy, it will of course depend on the selection of the function $g(y)$. Maximizing the efficacy over $g(y)$, it is clear that, optimizes the classical model. This optimization leads to

$$q(y_1, y_2) = y_1 y_2 + d\Big(y_1 l(y_2) - y_2 l(y_1)\Big)$$

(notice that $h(y_1, y_2)$ is given in this case). Again, constant d is selected so that the corresponding $\mathbf{Q}_2$ matrix has all eigenvalues equal to the (single multiple) eigenvalue of $\mathbf{Q}_1$.

Comparing the proposed optimum scheme to the classical one, we can realize that the only case where the two schemes coincide is when $l(y) = y$, or equivalently when the sources are Gaussian. However, this case is of no interest since, as it is well known, Gaussian sources cannot be separated. We therefore conclude that whitening the sources is not necessarily the best thing to do.

5. EXAMPLES AND SIMULATIONS

Let us consider the case of generalized Gaussian sources, with common pdf given by the following relation

$$f(y) = D(k)e^{-\left(\frac{|y|}{A(k)}\right)^k}, \quad k \geq 0,$$

where $A(k), D(k)$ are such that $f(y)$ integrates to unity and has unit variance.

We can now compute the optimum $h(y_1, y_2), q(y_1, y_2)$ functions for the proposed and the classical scheme and the corresponding efficacies as a function of the parameter k, Fig. 1 depicts these results. We can see that the proposed

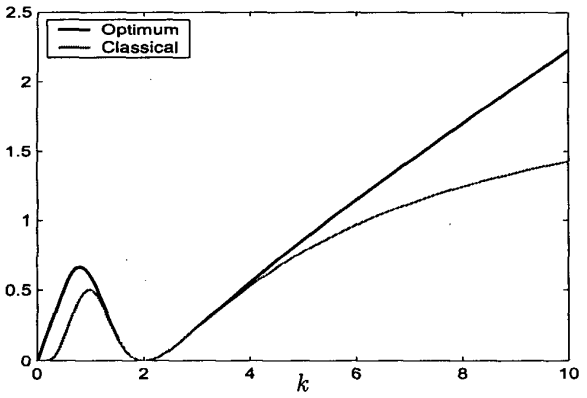

Fig. 1. Efficacies of proposed optimum and classical scheme, for generalized Gaussian sources.

scheme has always a larger efficacy (as it should), and that there are cases where the performance of the proposed algorithm is significantly superior to the classical one.

For example, when $k = 0.6$ the ratio of the two efficacies becomes 2.3, suggesting that the proposed scheme requires 2.3 times less steps to convergence than the classical scheme. Indeed this fact can be verified with the simulations presented in Fig. 2, where we plot the estimation error

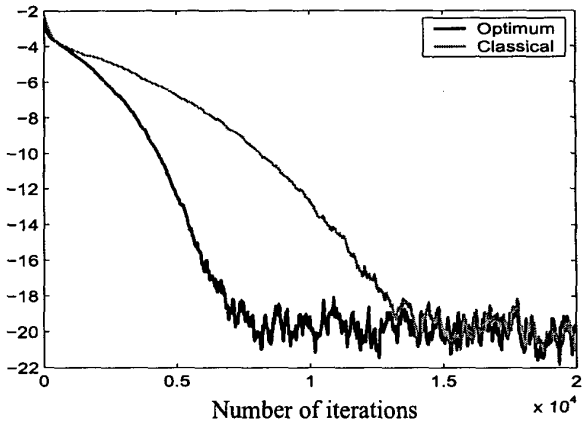

Fig. 2. Estimation error powers of proposed optimum and classical scheme, for generalized Gaussian sources with $k = 0.6$.

power of the two algorithms in db. The step sizes were selected so that the steady state error power becomes -20 db in both algorithms. The exact accordance of simulations and theory is quite evident.

6. CONCLUSION

We have presented a general class of adaptive algorithms that is capable of solving the source separation problem. With the help of an asymptotic performance measure we optimized the proposed algorithmic scheme and obtained algorithms that have optimum convergence speed. These algorithms, except the uninteresting Gaussian case, were shown to be different than the popular classical scheme used in the literature, suggesting that the latter is suboptimum.

7. REFERENCES

[1] A. Benveniste, et.al., *Adaptive Algorithms and Stochastic Approximation*, New York: Springer-Verlag, 1990.

[2] J-F Cardoso, "Blind signal separation: statistical principles," *Proceedings of the IEEE*, vol. 86, no. 10, pp. 2009-2025, Oct. 1998.

[3] G.V. Moustakides, "Locally optimum adaptive signal processing algorithms," *IEEE Transactions on Signal Processing*, vol. 46, no. 12, pp. 3315-3325, Dec. 1998.