A New FastICA Algorithm with Symmetric Orthogonalization

Changyuan Fan
Department of Control Engineering
Chengdu University Of Information Technology
Chengdu, Sichuan Province, China
Email: Xiongxiong123@cuit.edu.cn

Baoqiang Wang, Hui Ju
Department of Control Engineering
Chengdu University Of Information Technology
Chengdu, Sichuan Province, China
Email: bqwang@cuit.edu.cnj, juhui@cuit.edu.cn

***Abstract*—The separation of independent components from an array of mixtures is an interesting but difficult problem in signal processing. This paper re-examine the fastICA algorithm proposed by Hyvarinen and Oja for independent component analysis. The ways needed in fastICA algorithm for decorrelation the separating matrix can be deflationary or symmetric orthogonalization. Using the one-unit fastICA algorithm, this paper proposes a method to choose a suitable initial separating matrix of the fastICA algorithm with symmetric orthonormalization. The formed new algorithm can get faster speed without losing the intrinsic advantage of the original symmetric fastICA algorithm. Simulation results confirm the validity of the new algorithm.**

I. INTRODUCTION

Independent component analysis is a statistical method for transforming an observed multidimensional random vector into components that are statistically as independent from each other as possible[1,2]. Up to now, many approaches have been proposed to solve problems in independent component analysis. One of the most-used method is the fastICA algorithm of Hyvarinen and Oja[3,4]. FastICA was originally introduced for the instantaneous noise free ICA model. The problem is to estimate the unknown mixing matrix A and n unknown independent sources s_i making up a random vector $s = (s_1 \cdots s_n)^T$, from the model

$$x = As \tag{1}$$

Vector x contains the mixtures and a sample of x is available. It is commonly assumed that the vector of the sources s has a mean of zero and a normalized-covariance matrix. The fundamental restriction of the model is that we can only estimate non-Gaussian independent components (except if just one of the independents is Gaussian). Under the linear mixing model (1), the Darmois-Skitovitch theorem[5] guarantees the identifiability of the original sources from the observations up to a permutation and scaling of them.

The problem of ICA can be somewhat simplified by performing a preliminary prewhitening of the data x. The observed vector x is linearly transformed to a vector $z = Vx$ such that $E\{zz^T\} = I$. This can be accomplished with principal component analysis. After the transformation we have

$$z = Vx = VAs \tag{2}$$

where both z and s are white; z because of the explicit whitening, and s by the assumptions of zero mean and unit covariance matrix. It is easy to show that in this case where $n \times n$ square matrix VA is orthogonal, which simplifies the estimation problem. If A were known, the sources could be solved from $s = (VA)^T z$.

The recover of the original sources is attempted through seeking a linear orthogonal transformation

$$y = Wz \tag{3}$$

such that $y = s$. In fact, it will not be possible to identify the original scaling of the sources, or the specific order of the source components, so it is sufficient for us to find a $y = Wz$ such that $y = Ps$ where P is a permutation matrix, *i.e.*, a matrix for which, given a sequence $\{j_i \mid 0 \le i \le n\}$ of n distinct integers $0 \le j_i \le n$, it has $u_{ij} = \pm 1$, if $j = j_i$ and $u_{ij} = 0$ otherwise.

Let us denote by w_i^T, $i = 1,2\cdots,n$ the rows of W. The FastICA algorithm is an iterative method to find the local maximum of a cost function defined by

$$J_G = \sum_{i=1}^{n} E\{G(w_i^T z)\} \tag{4}$$

with G an even symmetrical function. The symbol E stands for expectation, which in practice would be estimated by

This work is supported by CUIT foundation grant CSRF200503

sample mean over the whitened vectors z. Some choices for the cost function are given in [3]. A widely used cost function is the fourth-order cumulant or kurtosis, defined for any random variable v as

$$kurt(v) = E\{v^4\} - 3(E\{v^2\})^2 \tag{5}$$

With the constraint that the argument $w_i^T z = y_i$ has unit variance the cost function becomes

$$J_G^{kurt} = \sum_{i=1}^{n} E\left\{\left(w_i^T z\right)^4\right\} \tag{6}$$

For the one-unit case, in which only one of the rows of W is considered and orthogonalization is reduced to just normalization of the vector to unit length after each iteration step, the fastICA algorithm for the general cost function (4), the updating step is

$$\overline{w}_i = E\{zg(w_i^T z)\} - E\{g'(w_i^T z)\}w_i \tag{7}$$

with function g the derivative of G and g' the derivative of g. For the kurtosis cost function, the corresponding updating step is

$$\overline{w}_i = E\left\{z\left(w_i^T z\right)^3\right\} - 3w_i \tag{8}$$

To obtain the full matrix W, we need to run the one-unit algorithm n times and the vector $\overline{w}_i$ must be re-orthonormalized after the update because they lose their orthonormality in the updating step. The orthonormalization can be accomplished basically in two ways: either deflationary or symmetrical orthonormalization. The former is given by

$$w_p = w_p - \sum_{j=1}^{p-1}\left(w_p^T w_j\right)w_j \tag{9}$$

with p the previously estimated vectors number. The latter is given by

$$W = \left(\overline{W}\,\overline{W}^T\right)^{-\frac{1}{2}}\vec{W} \tag{10}$$

where $\overline{W}$ is the matrix with rows $\overline{w}_i^T$. This means that the updating step is first performed for all the n weight vectors, and then the matrix $\overline{W}$ is orthogonalized using (10).

The rest of the paper is organized as follows. In section II, the new fastICA algorithm is proposed and explained in detail. The efficacy of the new algorithm is verified by the simulation in section III, where comparison with the fastICA of symmetric orthonormalization is performed. Finally, section IV summarizes the paper.

II. The new FastICA Algorithm

One can know from section I the estimated components are obtained one by one in the fastICA algorithm with deflation orthonormalization. However in certain applications, it may be desirable to use a symmetric decorrelation, in which no vectors are "privileged" over others; This means that the vectors w_i are not estimated one by one; instead, they are estimated in parallel. One motivation for this is that the deflationary method has the drawback that estimation errors in the first vectors are cumulated in the subsequent ones by the orthonormalization. Another one is that the symmetric orthonormalization methods enable parallel computation of independent components. For convenience for further analysis, we investigate the fastICA algorithm symmetric orthonormalization.

Symmetric orthonormalization is done by first doing the iterative step of the one-unit algorithm on every vector w_i in parallel, and afterwards orthogonalizing all the w_i by special symmetric methods. In other words:

1) Initialize the w_i, $i = 1,2\cdots,n$.

2) Do an iteration of a one-unit algorithm on every w_i in parallel.

3) Do a symmetric orthogonalization of the matrix W.

4) If not converged, go back to step 3.

The symmetric orthonormalization of W can be accomplished by $W = \left(\overline{W}\,\overline{W}^T\right)^{-\frac{1}{2}}\vec{W}$ where the inverse square root $\left(\overline{W}\,\overline{W}^T\right)^{-\frac{1}{2}}$ is obtained from the eigenvalue decomposition of $WW^T = Qdiag(d_1,\cdots,d_n)Q^T$ as

$$\left(\overline{W}\,\overline{W}^T\right)^{-\frac{1}{2}} = Adiag\left(d_1^{-\frac{1}{2}},\cdots,d_n^{-\frac{1}{2}}\right)Q^T \tag{11}$$

How to construct the initial W in step 1 is of great influence on the convergence speed of the algorithm. An appropriate choice of the initial W can yield faster convergence. To deal with this problem we can take use of the one-unit fastICA algorithm. The detailed method is that using the one-unit algorithm to get the first row of the initial W and the rest rows are randomly selected provided the orthogonality of the matrix W is assured. It finds out in simulation the first line of the matrix W is invariant with the iteration. Hence the problem to seek for the full n rows of W by iteration is reduced to seek for the $n-1$ rows of W. Consequently, the computation is cut down and the speed is enhanced. More precisely, we give the following detailed steps:

1) Center the observed data x to make its mean zero.

2) Whiten the data x to give z.

3) Estimate one row of the full matrix W using the updating rule (8) followed by a norm normalization step. This step is just a preparing procedure for the succeeding process.

4) Set the initial W as a matrix whose first row is the result of step 1 and the other rows are randomly selected as long as the orthogonality is guaranteed.

5) Update the every row of W as (8) in parallel and then normalize symmetrically as (10).

6) If not converged, go back to step 5.

In some real-world applications, it may be preferable to use the fastICA algorithm with symmetric orthononalization, in which every vector is impartially treated and the parallel computaion of independent components is enabled. But as it is known the computation of matrix is boring, which is the unavoidable problem of the fastICA algorithm with symmetric orthononalization. The new algorithm uses the one-unit algorithm to obtain the suitable initial W to give a good preparation for the succeeding iteration learning and orthonormalization steps of the full matrix W. The result it brings out is faster and better convergence. The computation of the initial matrix W is small, however the advantage is evident. Thus the new algorithm reduces the complex matrix iteration computation.

III. Simulation Results

In this section the performance of the new algorithm is demonstrated using an example. The mixing matrix is arbitrarily chosen, which is

$$A = \begin{bmatrix} 0.2434 & 0.8636 & 0.9573 \\ 0.6320 & 0.3301 & 0.1045 \\ 0.9950 & 0.5094 & 0.2044 \end{bmatrix}$$

The whitening matrix V is obtained as

$$V = \begin{bmatrix} 0.9173 & 0.4694 & -0.6567 \\ 0.4694 & 41.2257 & -25.9921 \\ -0.6567 & -25.9921 & 17.6196 \end{bmatrix}$$

Using the new fastICA algorithm we set the initial matrix W_0 as a matrix whose first row is the result of one-unit algorithm and the other rows are randomly selected as long as the orthogonality is guarantee. Thus the initial matrix W_0 is chosen as

$$W_0 = \begin{bmatrix} 0.1139 & -0.2758 & 0.9544 \\ 0.0327 & 0.9612 & -0.2738 \\ 0.9929 & 0 & 0.1186 \end{bmatrix}$$

Here, we use a source-to-input matrix defined by $B = WVA$. Then $y = Bs$ can recover successfully the source signal given that B is approximate generalized permutation matrix. The final matrix W of the new algorithm is

$$W = \begin{bmatrix} 0.1139 & -0.2758 & -0.9544 \\ -0.7599 & 0.5946 & -0.2625 \\ 0.6399 & 0.7522 & -0.1419 \end{bmatrix}$$

and the corresponding source-to-input matrix B is

$$B = \begin{bmatrix} -1.0008 & 0.0201 & -0.0015 \\ 0.0374 & 0.0390 & -1.0004 \\ 0.0139 & 1.0001 & 0.0515 \end{bmatrix}$$

One can see from the result of the final matrix W and the initial matrix W_0 that the first row of the matrix W_0 is invariable after iteration. The original problem of seeking for the three rows of matrix W is reduced to seeking for two rows of W, which is the right result we want.

As an illustration, consider the waveforms in Fig. 1 and Fig. 2. The original signals are shown in Fig. 1, and the mixed signals are shown in Fig. 2. The problem is to estimate the source signals in Fig. 1, using only the data in Fig. 2. Fig. 3

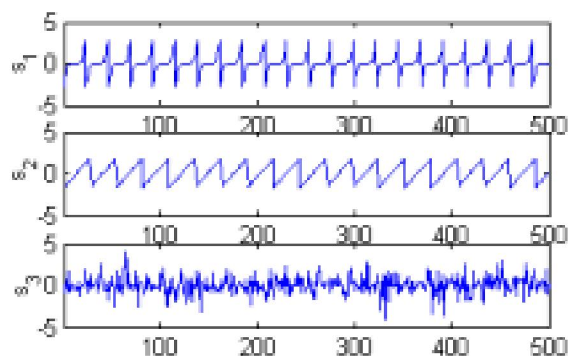

Figure 1. The original signals.

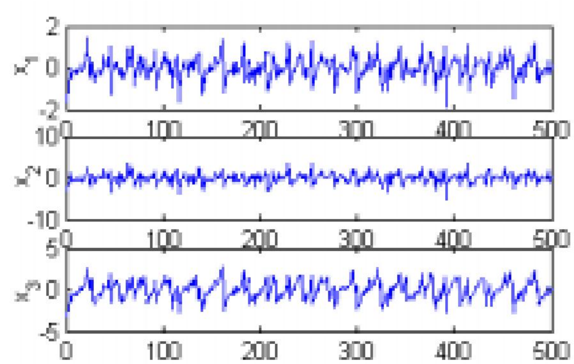

Figure 2. The observed mixture of the original signals.

presents the whitened signals and Fig. 4 demonstrates the recovered signals. From Fig.4 and Fig. 1 it is intuitively clear that the recovered signals y_1, y_2 and y_3 estimate $-s_1$, $-s_3$ and s_2, respectively, thus the separation is successful.

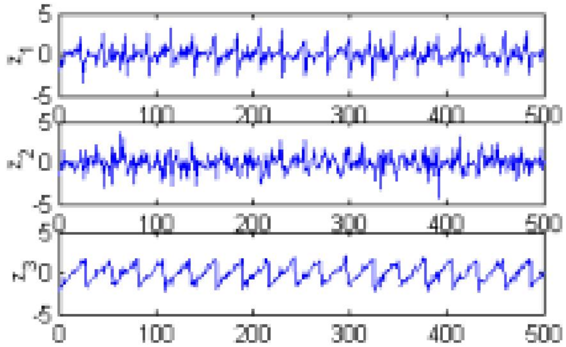

Figure 3. The prewhitened signals.

If the original fastICA algorithm with symmetric orthonormalization is used, the similar results can be obtained. However, the time needed is a litter longer compared to the

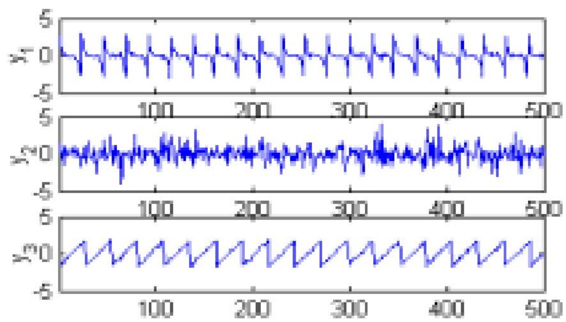

Figure 4. The recovered signals.

new algorithm, which is the case for only 3×3 case. The difference will be more evident with the dimension of the source signals gets larger. For example, the time needed for separating 6 source signals with 1000 samples in the new fastICA algorithm is 28.5470s compared with that in the original symmetric fastICA algorithm whose run time is 37.6452s.

IV. Conclusion

This paper considers the independent component analysis problem and discusses fastICA algorithm. The ways needed in fastICA algorithm for decorrelation the separating matrix can be deflationary or symmetric orthogonalization. In some applications, it may be preferable to use the fastICA algorithm with symmetric orthonormalization, in which every vector is impartially treated and the parallel computation of independent components is enabled. However the slow speed of the symmetric fastICA algorithm is boring. This paper deals with the problem to set the initial W as a matrix whose first row is the result of one-unit fastICA algorithm and the other rows are randomly selected as long as the orthogonality is guaranteed. The simulation results illustrate the performance of the new algorithm.

References

[1] E. Oja, "From neural learning to independent components", Neurocomputing, Vol. 22, pp. 187-199,1998.

[2] P. Comon, "Independent component analysis, A new concept?", Signal Processing, Vol. 36, pp. 287-314, 1994.

[3] A. Hyvarinen, "Fast and robust fixed-point algorithms for independent component analysis", IEEE Trans. on Neural Networks, Vol. 10, No. 3, pp. 626-634, 1999.

[4] A. Hyvarinen and E. Oja, "A fast fixed-point algorithm for independent component analysis", Neural Computation, Vol. 9, No. 7, pp. 1483-1492, 1997.

[5] X.R. Cao and R. Liu, "General approach to blind source separationg", IEEE Tran. on Signal Processing, Vol. 44, pp. 562-571, 1996.