

# A null space method for over-complete blind source separation

Ray-Bing Chen<sup>a,\*</sup>, Ying Nian Wu<sup>b</sup>

<sup>a</sup>*Institute of Statistics, National University of Kaohsiung, Kaohsiung 811, Taiwan, ROC*

<sup>b</sup>*Department of Statistics, University of California, Los Angeles, CA 90095, USA*

Received 22 June 2006; received in revised form 6 March 2007; accepted 8 March 2007

Available online 13 March 2007

## Abstract

In blind source separation, there are  $M$  sources that produce sounds independently and continuously over time. These sounds are then recorded by  $m$  receivers. The sound recorded by each receiver at each time point is a linear superposition of the sounds produced by the  $M$  sources at the same time point. The problem of blind source separation is to recover the sounds of the sources from the sounds recorded by the receivers, without knowledge of the  $m \times M$  mixing matrix that transforms the sounds of the sources to the sounds of the receivers at each time point. Over-complete separation refers to the situation where the number of sources  $M$  is greater than the number of receivers  $m$ , so that the source sounds cannot be uniquely solved from the receiver sounds even if the mixing matrix is known. In this paper, we propose a null space representation for the over-complete blind source separation problem. This representation explicitly identifies the solution space of the source sounds in terms of the null space of the mixing matrix using singular value decomposition. Under this representation, the problem can be posed in the framework of Bayesian latent variable model, where the mixing matrix and the source sounds can be inferred based on their posterior distributions. We then propose a null space algorithm for Markov chain Monte Carlo posterior sampling. We illustrate the algorithm using several examples under two different statistical assumptions about the independent source sounds. The blind source separation problem is mathematically equivalent to the independent component analysis problem. So our method can be equally applied to over-complete independent component analysis for unsupervised learning of high-dimensional data.

© 2007 Elsevier B.V. All rights reserved.

**Keywords:** Bayesian latent variable model; Givens rotation; Independent component analysis; Langevin–Euler equation; Markov chain Monte Carlo; Singular value decomposition

## 1. Introduction

### 1.1. The problem and our method

Consider the situation where  $M$  sources at various locations produce sounds independently and continuously over time, and  $m$  recorders at different locations record the sounds produced by the  $M$  sources. The sound recorded by each receiver at each time point is a linear superposition of the sounds produced by the  $M$  sources at the same time point. The problem of source separation is to recover the  $M$  sequences of source sounds from the  $m$  sequences of observed receiver sounds. If the  $m \times M$  mixing matrix that transforms the  $M$  source sounds to the  $m$  receiver sounds at each time point is unknown, then the source separation problem is called blind source separation. The over-complete blind

\* Corresponding author.

*E-mail address:* [rbchen@nuk.edu.tw](mailto:rbchen@nuk.edu.tw) (R.-B. Chen).

source separation problem refers to the situation where the number of sources  $M$  is greater than the number of receivers  $m$ . For convenience of reference, we call the situation where  $M = m$  the complete case.

Blind source separation is mathematically equivalent to independent component analysis in unsupervised learning of high-dimensional data, such as image and speech data. In the latter problem, we observe a sample of  $m$ -dimensional vectors, which are often considered i.i.d. realizations of an  $m$ -dimensional random vector. The goal of independent component analysis is to represent the  $m$ -dimensional random vector as a linear transformation of an underlying  $M$ -dimensional random vector, whose  $M$  components are assumed to be independent. Equivalently, we seek to decompose the  $m$ -dimensional random vector as a linear superposition of  $M$  basis vectors in the  $m$ -dimensional space, where the coefficients of these basis vectors are assumed to be independent. Usually,  $m$  is assumed to be equal to  $M$ . If  $M > m$ , then the problem is called over-complete independent component analysis.

Independent component analysis is different from principal component analysis. In the former, we do not assume that the transformation matrix or the basis vectors are orthogonal, neither do we assume that the distributions of the independent components are Gaussian. As a matter of fact, the goal of independent component analysis is to capture higher order non-Gaussian statistical properties that are beyond the second order variance–covariance matrix.

One can map the independent component analysis problem to the blind source separation problem. The  $m$ -dimensional random vector can be considered the observed sounds recorded by the  $m$  receivers. The underlying  $M$  independent components can be considered as the unobserved sounds produced by the  $M$  sources. The difference is that in blind source separation, the  $M$  sources physically exist, and the linear superposition can be physically justified. In independent component analysis, however, the  $M$  independent components and the corresponding linear basis vectors are only used to interpret the multivariate data. In this article, we concentrate on blind source separation problem. Our method can be equally applied to the independent component analysis problem.

In the complete case where  $m = M$ , the source sounds can be uniquely identified from the receiver sounds if the squared mixing matrix that transforms the source sounds to the receiver sounds is known and invertible. As a result, the likelihood of the mixing matrix can be easily written down, and the mixing matrix can be estimated by maximum likelihood method. The situation is more complicated for the over-complete situation where  $M > m$ . Even if we know the mixing matrix, the source sounds cannot be uniquely identified from the receiver sounds.

In this article, we propose a simple and natural method to solve the over-complete blind source separation or independent component analysis problem. The key idea is to explicitly identify the solution space of the source sounds in terms of the null space of the mixing matrix using singular value decomposition. Under this representation, the coordinates of the solution in the null space are left unidentified, and are treated as latent variables. Then the problem can be formulated in the framework of Bayesian latent variable model, where the mixing matrix and the source sounds can be inferred based on their posterior distributions. We propose a null space algorithm for Markov chain Monte Carlo posterior sampling. The algorithm consists of two components. One is a null space diffusion algorithm that samples the source sounds given the mixing matrix by a sequence of Langevin–Euler diffusion moves. The other is the Givens rotation algorithm that samples the mixing matrix by a sequence of Givens rotation moves.

We illustrate the performance of our method using several examples, under two statistical assumptions about the independent source sounds. One assumption is that the sounds produced by each source over time follow i.i.d. double exponential distribution. The other assumption is that the sounds produced by each source over time follow autoregressive time series model. These two assumptions capture some statistical properties in human speeches and natural sounds, respectively.

## 1.2. Previous work

There has been extensive research on both blind source separation and independent component analysis in the complete case. Comon (1994) studied blind source separation problem. Bell and Sejnowski (1997) conducted independent component analysis of natural image patches, where the linear transformations that transform the image to the independent sources can be interpreted as edge filters.

Over-complete independent component analysis problem was studied by Lewicki and Olshausen (1999) and Lewicki and Sejnowski (2000). The algorithm of Lewicki and Sejnowski (2000) was applied to over-complete blind source separation by Lee et al. (1999). Girolami (2001) proposed an EM type algorithm for over-complete blind source separation.

In terms of modelling the independent sources, usually, the sounds produced by each source are assumed to be independent over time and the sounds are assumed to follow a heavy-tailed distribution such as double exponential distribution to reflect the non-Gaussian property of the source sounds (Bell and Sejnowski, 1995; Lee et al., 1999; Girolami, 2001). In terms of modelling temporal structures, Pearlmutter and Parra (1997) used auto-regressive model for source sounds. Choi and Cichocki (2000), Hosseini et al. (2003) also studied temporal modelling in blind source separation. However, only complete case was considered in these papers.

An important point that we want to make is that in complete case, the receiver sounds are assumed to be exact linear superpositions of source sounds without any observational errors. However, in the over-complete case, the receiver sounds are assumed to be linear superpositions of source sounds plus observational errors. Bell and Sejnowski (1997) argued against the error term in independent component analysis of image data in the complete case. In this article, we choose to study the over-complete case without the error term. In this sense, our work is a more direct generalization of the complete case to the over-complete case than the above-mentioned papers, and our method is complementary to the above-mentioned methods, in that it solves the blind source separation problem in the limiting situation where the variance of the errors goes to 0. As a matter of fact, in this limiting case, identifying the null space explicitly appears to be the only way to form likelihood or posterior distribution for inference. Our method can certainly be extended to the situation where there are observational errors. In this case, our method does not offer inferential advantages compared to the previous work, because the underlying models are the same. But the null space method can still be useful for Markov chain Monte Carlo computation for the situation where the variance of the errors is small, because under the null space representation, the coordinates of the source sounds become less correlated than the coordinates in the original representation.

The rest of the paper is organized as follows. Section 2 presents the null space representation and Bayesian latent variable model for the over-complete blind source separation problem. Section 3 presents the null space algorithm for sampling the source sounds and the mixing matrix from their posterior distributions. Section 4 illustrates the null space method using simulated and real examples with different statistical assumptions on source sounds. Section 5 concludes with a discussion about the issue of observational errors.

## 2. Null space representation

In this section, we first describe the statistical formulation of the blind source separation problem. Then we present the null space representation and Bayesian latent variable model for the over-complete case.

### 2.1. Statistical formulation of blind source separation

Let  $\mathbf{s}_t = (s_{1t}, \dots, s_{Mt})'$  be the sounds produced by the  $M$  sources at time  $t$ , where  $t = 1, \dots, T$ . Let  $\mathbf{x}_t = (x_{1t}, \dots, x_{mt})'$  be the sounds recorded by the  $m$  receivers at time  $t$ . We assume that each receiver sound is a linear superposition of the  $M$  source sounds, so that

$$\mathbf{x}_t = A\mathbf{s}_t, \quad t = 1, \dots, T, \quad (1)$$

where  $A$  is an  $m \times M$  mixing matrix. We assume that the  $M$  sources produce sounds independently of each other, i.e., the sequences  $(s_{it}, t = 1, \dots, T)$  are independent for  $i = 1, \dots, M$ . More specifically, the probability density of  $(\mathbf{s}_t, t = 1, \dots, T)$  is

$$f(\mathbf{s}_t, t = 1, \dots, T) = \prod_{i=1}^M f_i(s_{it}, t = 1, \dots, T), \quad (2)$$

where  $f_i$  is the probability density of the sound sequence produced by the  $i$ th source. There are two common models for  $f_i(s_{it}, t = 1, \dots, T)$ . One is the double exponential model, where  $(s_{it}, t = 1, \dots, T)$  are assumed to be independent over time, and  $s_{it}$  follows double exponential distribution with parameter  $\alpha_i$ . The other model is to assume that the sources form the  $d$ th-order Markov sequences and can be represented as

$$s_{it} = \phi_{i1}s_{i(t-1)} + \dots + \phi_{id}s_{i(t-d)} + \varepsilon_{it},$$

where  $\phi_{i1}, \dots, \phi_{id}$  are the coefficients, and  $\varepsilon_{it}$  are i.i.d. normal random variables with variance  $\sigma_i^2$ . This is the  $d$ th-order auto-regressive time series model. There is no intercept term in this auto-regressive model, because we assume that the unconditional mean of  $s_{it}$  is zero. The unconditional mean is the DC component of a source sequence. It has no effect on our perception of the sounds of the source sequences.

The goal of blind source separation is to estimate the mixing matrix  $A$  and to recover the source sounds ( $\mathbf{s}_t, t = 1, \dots, T$ ), based on the observed receiver sounds ( $\mathbf{x}_t, t = 1, \dots, T$ ). If there are unknown parameters in  $f_i(s_{it}, t = 1, \dots, T)$ , we also need to estimate these parameters.

In terms of the number of sources  $M$  and the number of receivers  $m$ , there are two cases. One is the complete case, where  $M = m$ . The other is the over-complete case, where  $M > m$ . In the rest of this subsection, we shall briefly review the solution to the complete case. Then in the next subsection, we shall present our null space method for the over-complete case.

In complete case  $M = m$ ,  $A$  is a squared matrix. If  $A$  is invertible, then the source sounds can be solved from the receiver sounds explicitly, i.e.,  $\mathbf{s}_t = A^{-1}\mathbf{x}_t = W\mathbf{x}_t$ , where  $W = A^{-1}$ . Under this transformation, one can easily write down the probability distribution of the observed receiver sounds:

$$p(\mathbf{x}_t, t = 1, \dots, T | W = A^{-1}) = f(W\mathbf{x}_t, t = 1, \dots, T) |W|^T, \quad (3)$$

where  $f$  is the density function of the source sounds in Eq. (2), and  $|W|$  is the determinant of  $W$ . Therefore, the log-likelihood of  $W$  is

$$l(W) = \log f(W\mathbf{x}_t, t = 1, \dots, T) + T \log |W|. \quad (4)$$

$W$  can be estimated by maximum likelihood method. After that, the source sounds can be recovered from the receiver sounds according to  $\mathbf{s}_t = W\mathbf{x}_t$ .

The above method can also be used for independent component analysis in unsupervised learning of the high-dimensional data ( $\mathbf{x}_t, t = 1, \dots, T$ ). For example, ( $\mathbf{x}_t, t = 1, \dots, T$ ) may be  $T$  image patches of  $12 \times 12$  pixels, so that  $\mathbf{x}_t$  is a 144-dimensional vector. The  $i$ th row of  $W$  can be interpreted as a linear filter that transforms  $\mathbf{x}_t$  to the  $i$ th independent component  $s_{it}$ . Unlike the blind source separation problem, the independent components here are not necessarily physically existing sources, and the linear transformation  $A$  often cannot be physically justified. Just like principal component analysis, independent component analysis is a method of interpreting the multivariate data. In comparison to principal component analysis, independent component analysis does not assume that  $A$  is an orthogonal matrix. More important, the distributions  $f_i$  are assumed to be non-Gaussian. As a matter of fact, one can perform independent component analysis after linearly transforming  $\mathbf{x}_t$  to diagonalize its variance–covariance matrix. That is, independent component analysis captures statistical properties that are beyond second-order variance–covariance matrix, which is the main target of principal component analysis and factor analysis.

## 2.2. Null space representation for over-complete case

In this paper, we are interested in the over-complete case where  $M > m$ . Unlike the complete case, we cannot uniquely identify  $\mathbf{s}_t$  from  $\mathbf{x}_t$  even if  $A$  is known. Therefore, extra work is needed in the over-complete case in order to obtain the counterparts of distribution (3) and log-likelihood (4).

There is a simple and natural way to handle this problem. Although there is no unique solution to Eq. (1), the linear space of all possible solutions can still be explicitly identified in terms of the null space of the mixing matrix  $A$ , using singular value decomposition of  $A$ .

To remind the reader of singular value decomposition and to facilitate a smooth reading of the paper, we would like to work through the following simplest case:  $A = (D, \mathbf{0})$ , where  $D = \text{diag}(d_i, i = 1, \dots, m)$  is an  $m \times m$  diagonal matrix, with  $d_1 \geq \dots \geq d_m > 0$ . Then Eq. (1) becomes

$$\begin{pmatrix} x_{1t} \\ \vdots \\ x_{mt} \end{pmatrix} = \begin{pmatrix} d_1 & \cdots & 0 & 0 & \cdots & 0 \\ & \ddots & & \vdots & \ddots & \vdots \\ 0 & \cdots & d_m & 0 & \cdots & 0 \end{pmatrix} \begin{pmatrix} s_{1t} \\ \vdots \\ s_{Mt} \end{pmatrix}. \quad (5)$$

Obviously, the first  $m$  sources,  $s_{it}$ ,  $i = 1, \dots, m$ , can be uniquely solved by  $s_{it} = x_{it}/d_i$ ,  $i = 1, \dots, m$ . But the rest of  $M - m$  sources,  $s_{it}$ ,  $m < i \leq M$ , are left unidentified and can be any values. Therefore, the solutions can be written as

$$\mathbf{s}_t = \begin{pmatrix} D^{-1} \\ \mathbf{0} \end{pmatrix} \begin{pmatrix} x_{1t} \\ \vdots \\ x_{mt} \end{pmatrix} + \begin{pmatrix} \mathbf{0} \\ I_{M-m} \end{pmatrix} \begin{pmatrix} c_{1t} \\ \vdots \\ c_{(M-m)t} \end{pmatrix} = \begin{pmatrix} D^{-1} \\ \mathbf{0} \end{pmatrix} \mathbf{x}_t + \begin{pmatrix} \mathbf{0} \\ I_{M-m} \end{pmatrix} c_t \tag{6}$$

for an arbitrary coefficient vector  $c_t$ , where  $I_{M-m}$  is the  $M - m$  dimensional identity matrix. In this simplest situation,  $(D^{-1}, \mathbf{0})'$  is the pseudo-inverse of the matrix  $A$ . The columns of  $(\mathbf{0}, I_{M-m})'$  are the  $M - m$  orthogonal basis vectors of the null space of the matrix  $A$ , and these basis vectors are all orthogonal to the rows of  $A$ . The  $M - m$  components of  $c_t$  are the coordinates of the solution with respect to the  $M - m$  basis vectors of the null space of  $A$ .

The above simplest case can be easily extended to the general situation. Suppose we can rotate  $\mathbf{x}_t$  to  $U'\mathbf{x}_t$ , and rotate  $\mathbf{s}_t$  to  $V'\mathbf{s}_t$ , where  $U$  is an  $m \times m$  orthogonal matrix, and  $V$  is an  $M \times M$  orthogonal matrix, so that the rotated vectors satisfy the simplest equation (5),

$$U'\mathbf{x}_t = (D \ \mathbf{0}) V'\mathbf{s}_t. \tag{7}$$

Fortunately, such  $U$ ,  $V$ , and  $D$  exist, and can be obtained by the singular value decomposition of  $A$ :

$$A = U(D \ \mathbf{0})V', \tag{8}$$

where  $d_1 \geq \dots \geq d_m > 0$  are the singular values of  $A$ . Then according to Eq. (6), the solution to Eq. (7) is

$$\mathbf{s}_t = V \begin{pmatrix} D^{-1} \\ \mathbf{0} \end{pmatrix} U'\mathbf{x}_t + V \begin{pmatrix} \mathbf{0} \\ I_{M-m} \end{pmatrix} c_t. \tag{9}$$

Let  $V = (V_1, V_2)$ , where  $V_1$  is the  $M \times m$  matrix composed of the first  $m$  columns of  $V$ , and  $V_2$  is the  $M \times (M - m)$  matrix composed of the rest  $M - m$  columns of  $V$ . Then we can write Eq. (9) as

$$\mathbf{s}_t = A^-\mathbf{x}_t + V_2c_t = \mathbf{s}_t^{(0)} + \Delta\mathbf{s}_t, \tag{10}$$

where  $A^- = V_1D^{-1}U'$  is the pseudo-inverse of  $A$ , and  $\mathbf{s}_t^{(0)} = A^-\mathbf{x}_t$  is a solution to Eq. (7) obtained by pseudo-inverse. The columns of  $V_2$  are the basis vectors of the null space of  $A$ , and they are orthogonal to the row vectors of  $A$ .  $c_t$  are the coordinates of the solution with respect to these basis vectors.  $\Delta\mathbf{s}_t = V_2c_t$  in Eq. (10) is the free displacement of the solution in the null space, with  $A\Delta\mathbf{s}_t = 0$ .

Eq. (9) tells us that  $\mathbf{s}_t$  can be uniquely identified from  $\mathbf{x}_t$  and  $c_t$  together. We can write Eq. (9) in a more compact form:

$$\mathbf{s}_t = V \begin{pmatrix} D^{-1}U' & \mathbf{0} \\ \mathbf{0} & I_{M-m} \end{pmatrix} \begin{pmatrix} \mathbf{x}_t \\ c_t \end{pmatrix}. \tag{11}$$

We call Eqs. (9)–(11) the null space representation. Eq. (11) is a direct generalization of  $\mathbf{s}_t = A^{-1}\mathbf{x}_t = W\mathbf{x}_t$  in the complete case of blind source separation or independent component analysis. The difference is that we must incorporate  $c_t$  in the over-complete case to account for the non-uniqueness of the solution for the source sounds. This generalization is simple and natural. But to the best of our knowledge, the null space representation had not been explicitly used in over-complete blind source separation or over-complete independent component analysis prior to our work.

### 2.3. Bayesian latent variable formulation

With null space representation (11), we can formulate the problem of the over-complete blind source separation as a latent variable problem, and derive the distribution of the observed receiver sounds and the likelihood of the mixing matrix  $A$  as counterparts of distribution (3) and log-likelihood (4) in the complete case.

Specifically, the joint distribution of  $(\mathbf{x}_t, c_t, t = 1, \dots, T)$  can be obtained from the distribution of source sounds (2):

$$p(\mathbf{x}_t, c_t, t = 1, \dots, T|A) = f(\mathbf{s}_t, t = 1, \dots, T)|D|^T, \quad (12)$$

where  $f$  is the density function of the source sounds, and  $\mathbf{s}_t$  is identified from  $(\mathbf{x}_t, c_t)$  via the null space representation (11).  $|D|^T$  is the Jacobian term.  $|D|$  is the determinant of  $D$ , and is simply the product of the  $m$  singular values of  $A$ .

Given the joint distribution (12), the distribution  $p(\mathbf{x}_t, t = 1, \dots, T|A)$  can be obtained theoretically by integrating out the latent variables  $(c_t, t = 1, \dots, T)$ . From this distribution, the likelihood of matrix  $A$  can then be formed. Therefore, the null space representation enables us to obtain the counterparts of distribution (3) and log-likelihood (4).

A natural computational tool for inferring  $A$  and  $(c_t, t = 1, \dots, T)$  is the EM algorithm (Dempster et al., 1977), where  $(c_t, t = 1, \dots, T)$  can also be called missing data, and the distribution  $p(\mathbf{x}_t, c_t, t = 1, \dots, T|A)$  in Eq. (12) is called complete-data model. But the E-step can be computationally intractable. So in this article, we adopt a Bayesian framework, and we use Markov chain Monte Carlo method for computational inference. Specifically, we put a prior distribution  $p(A)$  on  $A$ . Then we infer  $A$  and  $(c_t, t = 1, \dots, T)$  based on their joint posterior distribution

$$p(A, c_t, t = 1, \dots, T|\mathbf{x}_t, t = 1, \dots, T) \propto p(A)p(\mathbf{x}_t, c_t, t = 1, \dots, T|A). \quad (13)$$

The complete-data model  $p(\mathbf{x}_t, c_t, t = 1, \dots, T|A)$  in Eq. (12) involves  $U, V$ , and  $D$  of the singular value decomposition of  $A$ , and  $A$  does not appear explicitly as a single term. For convenience of specifying prior distribution and for computational tractability, we treat  $U, V$ , and  $D$  as individual parameters, and we further represent the two orthogonal matrices  $U$  and  $V$  by the products of Given rotations:

$$U = \prod_{ij} U_{ij}(\alpha_{ij}) \quad \text{and} \quad V = \prod_{ij} V_{ij}(\beta_{ij}), \quad (14)$$

where  $U_{ij}(\alpha_{ij})$  and  $V_{ij}(\beta_{ij})$  are  $m \times m$  and  $M \times M$  Givens rotation matrices, respectively. Specifically,  $U_{ij}(\alpha_{ij})$  is obtained by modifying the identity matrix so that the  $(i, i)$ ,  $(i, j)$ ,  $(j, i)$  and  $(j, j)$  elements of this matrix are, respectively,  $\cos \alpha_{ij}$ ,  $\sin \alpha_{ij}$ ,  $-\sin \alpha_{ij}$ , and  $\cos \alpha_{ij}$ , where  $\alpha_{ij} \in [0, 2\pi)$ . In the product representation of  $U$ , there are  $m(m-1)/2$  Givens rotation matrices with fixed order. So the matrix  $U$  is parametrized by  $m(m-1)/2$  rotation angles  $\alpha_{ij}$ . The same for representing  $V$  by the product of  $V_{ij}(\beta_{ij})$ .

For Bayesian inference, we put independent uniform prior distributions on  $\alpha_{ij}$  and  $\beta_{ij}$ . As to the diagonal matrix  $D$ , we put uniform prior distribution on  $\log d_i$  with the order constraint  $d_1 \geq \dots \geq d_m$ . This distribution is invariant under the scaling transformation of  $d_i$ .

In this Bayesian framework, we can use Markov chain Monte Carlo method to sample from the joint posterior distribution of  $\{\alpha_{ij}\}$ ,  $\{\beta_{ij}\}$ ,  $\{d_i\}$ , and  $\{c_t\}$ , which can be obtained from Eq. (13).

### 3. Null space algorithm

The Markov chain Monte Carlo sampling of the posterior distribution (13) can be accomplished by the data augmentation algorithm of Tanner and Wong (1987), which is a stochastic version of the EM algorithm. Specifically, the algorithm iterates the following two steps:

- (1) Sampling  $(c_t, t = 1, \dots, T)$  from  $p(c_t, t = 1, \dots, T|\mathbf{x}_t, t = 1, \dots, T, A)$ . Given  $A$  and  $(c_t, t = 1, \dots, T)$ , we recover the source sounds  $(\mathbf{s}_t, t = 1, \dots, T)$  using the null space representation (11).
- (2) Sampling  $A$  by drawing  $U, V$  and  $D$  from  $p(U, V, D|\mathbf{x}_t, c_t, t = 1, \dots, T)$ .

Both  $p(c_t, t = 1, \dots, T|\mathbf{x}_t, t = 1, \dots, T, A)$  and  $p(U, V, D|\mathbf{x}_t, c_t, t = 1, \dots, T)$  are proportional to the joint posterior distribution (13). Next, we describe the details of these two steps.

#### 3.1. Null space diffusions

We use Langevin–Euler diffusion process to sample  $(c_t, t = 1, \dots, T)$  from  $p(c_t, t = 1, \dots, T|\mathbf{x}_t, t = 1, \dots, T, A)$ . Since the null space representation is  $\mathbf{s}_t = V_1 D^{-1} U' \mathbf{x}_t + V_2 c_t$ , when  $c_t$  changes, the recovered  $\mathbf{s}_t$  moves in the null space of  $A$ . So we call this algorithm the null space diffusion algorithm.

The Langevin–Euler move is essentially a stochastic gradient descent move. Specifically, let us write  $p(c_t, t = 1, \dots, T | \mathbf{x}_t, t = 1, \dots, T, A)$  as  $\pi(\mathbf{c}) \propto \exp(-H(\mathbf{c}))$ , where  $\mathbf{c} = (c_1, \dots, c_T)$ . Suppose at step  $\tau$  of the Langevin–Euler process, the value of  $\mathbf{c}$  is  $\mathbf{c}(\tau)$  (note that  $\mathbf{c}(\tau)$  should not be confused with  $c_t$ .  $c_t$  is the null space coordinate for  $\mathbf{s}_t$ , whereas  $\mathbf{c}(\tau)$  is composed of the values of  $(c_1, \dots, c_T)$  at the  $\tau$ th step of Langevin–Euler move). Then the next step  $\mathbf{c}(\tau + 1)$  is

$$\mathbf{c}(\tau + 1) = \mathbf{c}(\tau) - \frac{h}{2} \frac{\partial H(\mathbf{c})}{\partial \mathbf{c}} \Big|_{\mathbf{c}=\mathbf{c}(\tau)} + \sqrt{h} Z_\tau, \tag{15}$$

where  $Z_\tau$  is a vector of independent standard normal random variables. The above Langevin–Euler move is to discretize the continuous time Langevin–Euler diffusion process with time intervals of length  $h$ . If  $h \rightarrow 0$ , the Langevin–Euler process (15) converges to the stationary distribution  $\pi(\mathbf{c})$ . In order to correct the discretization error caused by  $h > 0$ , we follow Roberts and Rosenthal (1998) to add the Metropolis–Hastings acceptance/rejection step, i.e., we accept  $\mathbf{c}(\tau + 1)$  with probability

$$\min \left\{ 1, \frac{\pi(\mathbf{c}(\tau + 1))q(\mathbf{c}(\tau + 1), \mathbf{c}(\tau))}{\pi(\mathbf{c}(\tau))q(\mathbf{c}(\tau), \mathbf{c}(\tau + 1))} \right\},$$

where

$$q(\mathbf{x}, \mathbf{y}) \propto \exp \left( -\frac{1}{2h} \left\| \mathbf{y} - \mathbf{x} + \frac{h}{2} \frac{\partial H(\mathbf{x})}{\partial \mathbf{x}} \right\|^2 \right)$$

and  $\| \cdot \|$  is the  $L_2$ -norm.

According to Roberts and Rosenthal (1998),  $h$  should be chosen so that the Metropolis–Hastings acceptance probability should be about 0.4–0.8. Roberts and Rosenthal (1998) also analyze the algorithm asymptotically as the dimensionality of the target distribution goes to infinity. In our case, the dimensionality is  $T$ , i.e., the length of the source sequences. According to their theoretical result,  $h$  should be proportional to  $T^{-1/3}$ .

### 3.2. Givens rotations

For the second step of the data augmentation algorithm, we need to sample the orthogonal matrices  $U$  and  $V$ , and the diagonal matrix  $D$ .

To sample  $U$  and  $V$ , we only need to sample the rotation angles  $\alpha_{ij}$  and  $\beta_{ij}$  in the Givens rotation representation (14) from the conditional posterior density

$$p(\{\alpha_{ij}\}, \{\beta_{ij}\} | \mathbf{x}_t, c_t, t = 1, \dots, T, D) \propto p \left( \mathbf{x}_t, c_t, t = 1, \dots, T | A = \prod_{ij} U_{ij}(\alpha_{ij})(D, \mathbf{0}) \prod_{ij} V_{ij}(\beta_{ij})' \right), \tag{16}$$

using the Metropolis–Hastings algorithm. We visit  $\alpha_{ij}$  and  $\beta_{ij}$  in random order. Each time, we propose to perturb the rotation angle by adding  $\alpha_{ij}$  or  $\beta_{ij}$  a random variable uniformly distributed in  $[-a, a]$  for a small value of  $a$ , where the addition is mod  $2\pi$  since  $\alpha_{ij}$  and  $\beta_{ij}$  belong to  $[0, 2\pi)$ . Then we accept the proposed perturbation according to the Metropolis–Hastings acceptance probability computed from Eq. (16).

For sampling  $D$ , as we mentioned earlier, we work on the log scale by sampling  $w_i = \log(d_i)$ . The posterior density of  $w_1, \dots, w_m$  is

$$p(w_i, i = 1, \dots, m | \mathbf{x}_t, c_t, t = 1, \dots, T, U, V) \propto f(\mathbf{s}_t, t = 1, \dots, T | A) \exp \left( -T \sum_{i=1}^m w_i \right). \tag{17}$$

The maximum a posteriori (MAP) estimate  $\hat{\mathbf{w}}$  of  $\mathbf{w} = (w_1, \dots, w_m)'$  is found by solving the following equation:

$$\frac{\partial \log f(\mathbf{s}_t, t = 1, \dots, T | A)}{\partial \mathbf{w}} = T \frac{\partial \sum_{i=1}^m w_i}{\partial \mathbf{w}}.$$

After locating the MAP, the posterior density of  $\mathbf{w}$  can be approximated by a multivariate normal density with the variance–covariance matrix  $\Sigma$ , where  $\Sigma$  is proportional to the inverse of the Fisher information matrix, i.e.,

$[\text{Var}(\partial \log f(\mathbf{s}_t, t = 1, \dots, T|A)/\partial \mathbf{w})]^{-1}$ . Besides, the Fisher information matrix, the second order derivative of  $\log f(\mathbf{s}_t, t = 1, \dots, T|A)$  or the observed information matrix can also be used for calculating the variance–covariance matrix. For long sequences, both the Fisher information and the observed information lead to accurate approximation to the variance–covariance matrix. Of course, we can make the sampling exact by adding a Metropolis–Hastings acceptance/rejection step.

#### 4. Experiments

In this section, we demonstrate the performance of our method using simulated and real examples. This section is divided into two parts according to two different models about the source sounds. In the first model, the sounds produced by each source follow i.i.d. double exponential distributions. In the second model, the sounds from each source follow auto-regressive time series model.

##### 4.1. Double exponential model

The double exponential model assumes that the sounds produced by each source are independent over time, therefore, it neglects the temporal structure in the sequence of sounds. But the double exponential distribution captures the heavy tails or large kurtosis in the marginal distributions of the sounds, especially human speeches. Specifically, the probability density of  $(\mathbf{s}_t, t = 1, \dots, T)$  is

$$f(\mathbf{s}_t, t = 1, \dots, T) = \prod_{t=1}^T f_t(\mathbf{s}_t) = \prod_{t=1}^T \prod_{i=1}^M f_{it}(s_{it}) \propto \exp \left\{ - \sum_{i=1}^M \left( \alpha_i \sum_{t=1}^T |s_{it}| \right) \right\}. \quad (18)$$

We first perform a simulation study. We generate source sounds independently from  $M = 3$  double exponential distributions with  $\alpha_1 = 2.3208$ ,  $\alpha_2 = 2.5707$  and  $\alpha_3 = 2.6843$ . The  $m = 2$  observed receiver sounds are obtained by

$$\begin{pmatrix} x_{1t} \\ x_{2t} \end{pmatrix} = \begin{pmatrix} 1 & 0.7071 & 0.7071 \\ 0 & 0.7071 & -0.7071 \end{pmatrix} \begin{pmatrix} s_{1t} \\ s_{2t} \\ s_{3t} \end{pmatrix}. \quad (19)$$

We perform blind source separation by running the null space algorithm for 2000 iterations. In our experiments, the discretization length  $h$  of the Langevin–Euler equation is empirically set at  $10^{-3}$ . We discard the first 1000 iterations and use the second 1000 iterations to compute the posterior means of the mixing matrix and the recovered source sounds. The posterior mean of the mixing matrix  $A$  is

$$\hat{A} = \begin{pmatrix} 1.0320 & 0.8267 & 0.5944 \\ -0.0024 & 0.6829 & -0.7201 \end{pmatrix}.$$

The 95% posterior intervals for the elements of  $A$  are also calculated based on the last 1000 iterations. Specifically, for each parameter, we sort the corresponding 1000 posterior samples for this parameter, and then the two end points of the posterior interval are taken to be the 0.025th and 0.975th sample percentiles. These intervals are

(0.9358, 1.1123)	(0.6874, 1.0580)	(0.4431, 0.6887)
(−0.1600, 0.1003)	(0.6127, 0.7516)	(−0.7780, −0.6526)

Except for the interval of  $A(1, 3)$ , all the other intervals cover the corresponding true values of  $A$ .

The observed sounds of the two receivers, the true sounds of the three sources, and the posterior means of the recovered sounds of the three sources are plotted in Fig. 1. To measure the performance of our method, we calculate the signal-to-noise ratio (SNR) by

$$\text{SNR}_{s_i}(\hat{s}_i)[\text{dB}] = 10 \log_{10} \frac{\|s_i\|^2}{\|s_i - \hat{s}_i\|^2}, \quad (20)$$

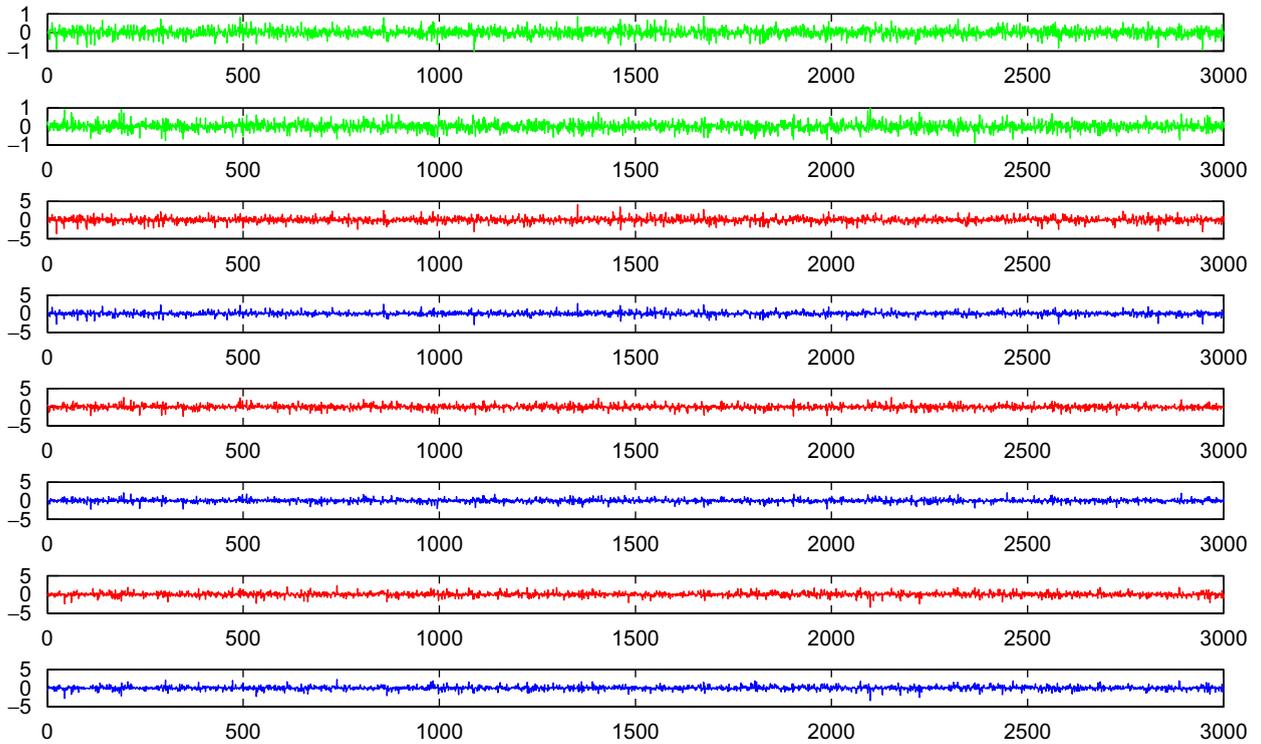


Fig. 1. Simulation study. The green lines are the receiver sounds, the red lines are the true source sounds, and the blue lines are the recovered source sounds. The SNR are 8.6915, 11.9470 and 11.0704 for the three sources, respectively.

where  $s_i = (s_{it}, t = 1, \dots, T), i = 1, \dots, M$ , are original sounds from the sources, and  $\hat{s}_i = (\hat{s}_{it}, t = 1, \dots, T), i = 1, \dots, M$ , are posterior means of the recovered source sounds. In this simulation study, the SNR are 8.6915, 11.9470 and 11.0704 for the three sources, respectively.

To make the Monte Carlo study more informative, we repeat the same experiment five times. For each replication, we re-generate the source sequences according to the same double exponential distributions, and mix them by the same mixing matrix. Then we use the same algorithm for posterior sampling. The results are shown in Tables 1 and 2. Even though each replication involves a different simulated data set, the inferential results for the five replications are similar to each other.

Next, we conduct another experiment with real sounds. The sources are three human speeches, and each source has 10,000 time points. The mixing matrix is set to be

$$A = \begin{pmatrix} 0 & 0.7071 & 0.7071 \\ 1 & 0.7071 & -0.7071 \end{pmatrix}.$$

We run the null space algorithm, and compute the posterior means of the mixing matrix and the source sounds using the last 1000 iterations. The posterior mean of  $A$  is

$$\hat{A} = \begin{pmatrix} 0.0025 & 0.7409 & 0.6637 \\ 1.2006 & 0.6867 & -0.6964 \end{pmatrix}.$$

We also calculate the 95% posterior intervals from the last 1000 iterations. The intervals for the elements of  $A$  are

(−0.0546, 0.0503)	(0.6751, 0.7968)	(0.6006, 0.7368)
(1.0579, 1.3132)	(0.6151, 0.7550)	(−0.8529, −0.6050)

Table 1  
The estimates of the elements of  $A$  and the corresponding 95% posterior intervals (in parentheses) for each of the five replications

Repetition	$A(1, 1)$	$A(1, 2)$	$A(1, 3)$	$A(2, 1)$	$A(2, 2)$	$A(2, 3)$
1	1.1828 (1.0962, 1.2582)	0.5480 (0.3894, 0.7170)	0.6417 (0.4525, 0.7842)	0.0970 (−0.0477, 0.2188)	0.7375 (0.6599, 0.8163)	−0.5606 (−0.6418, −0.4696)
2	1.1517 (1.0293, 1.2745)	0.7519 (0.6551, 0.8630)	0.5426 (0.3559, 0.6558)	−0.1051 (−0.2624, 0.0102)	0.7347 (0.6642, 0.8207)	0.5521 (−0.6339, 0.4508)
3	1.1685 (1.0327, 1.2986)	0.7024 (0.5915, 0.8325)	0.6308 (0.5260, 0.7171)	0.0048 (−0.0871, 0.0985)	0.7480 (0.6821, 0.8043)	−0.5827 (−0.6435, −0.5274)
4	1.1819 (1.0844, 1.2789)	0.5559 (0.4699, 0.6772)	0.7021 (0.5991, 0.8170)	0.0399 (−0.0548, 0.1252)	0.6335 (0.5799, 0.6967)	−0.6636 (−0.7301, −0.5946)
5	0.9739 (0.7779, 1.0857)	0.6243 (0.4561, 0.8234)	0.8200 (0.5382, 1.1807)	0.0651 (−0.1442, 0.2690)	0.7106 (0.6031, 0.8091)	−0.7172 (−0.8326, −0.5953)

Each replication involves a different data set simulated under the same parameter values.

Table 2  
The SNR values for each of the five replications

Repetition	1	2	3	4	5
Source 1	8.4933	8.1911	9.4173	9.5063	8.4283
Source 2	13.7309	11.4664	14.6939	13.8592	14.3899
Source 3	11.0415	10.9354	12.7340	12.7168	12.7896

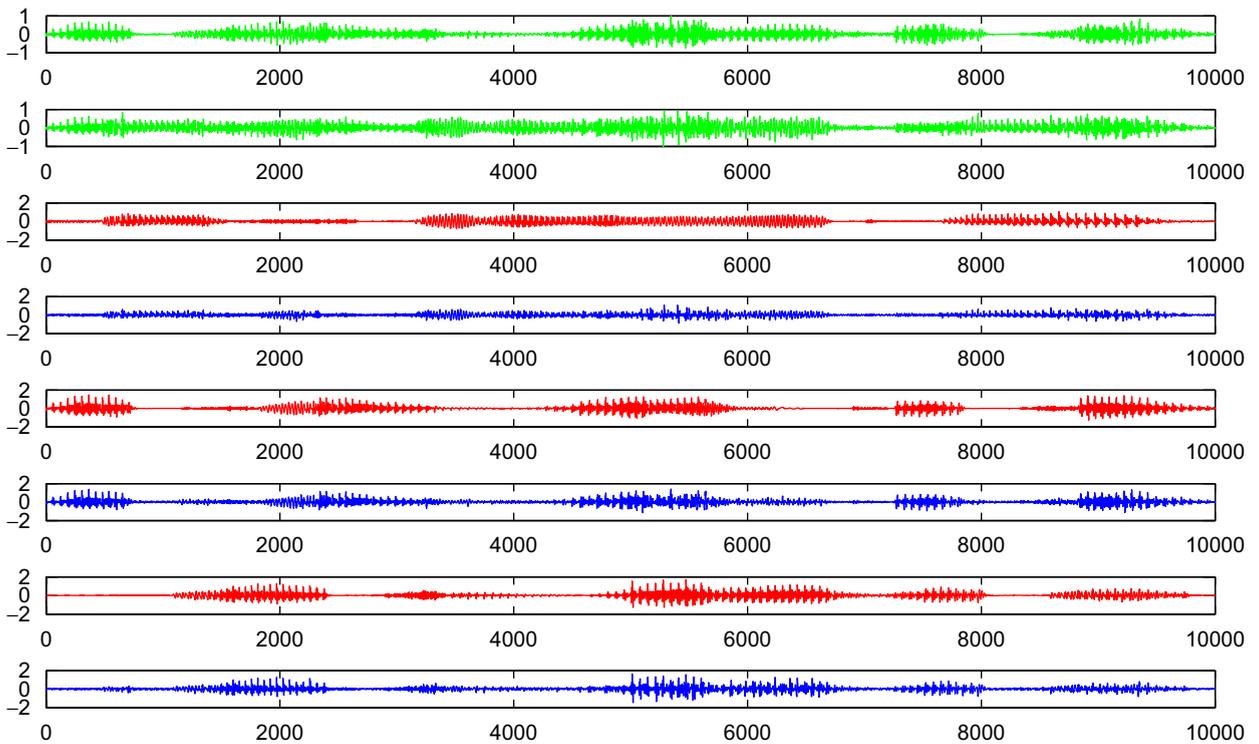


Fig. 2. Blind separation of human speeches. The green lines are the receiver sounds, the red lines are the true source sounds and the blue lines are the recovered source sounds. The SNR values are 7.2359, 13.7701 and 12.9859, respectively.

Except for the interval for  $A(2, 1)$ , all the other posterior intervals cover the corresponding true values of  $A$ . The SNR values for the recovered sources are 7.2359, 13.7701 and 12.9859, respectively. The observed sounds of the two receivers, the true sounds of the three sources, and the posterior means of the recovered sounds of the three sources are plotted in Fig. 2.

In these experiments, we assume that the parameters of the double exponential distributions are known. These parameters are scale parameters and they confound with the mixing matrix. More specifically, for  $i = 1, \dots, M$ , if we change the parameter  $\alpha_i$  of the double exponential model to  $h_i \alpha_i$ , and change  $A_{ki}$  to  $A_{ki} / h_i$ , for  $k = 1, \dots, m$ , then the distribution of the observed receiver sounds ( $\mathbf{x}_t, t = 1, \dots, T$ ) will remain the same. Therefore, we cannot estimate  $\alpha_i$  and  $A$  simultaneously. But this is not a real problem. As a matter of fact, we can fix  $\alpha_i$  at any values. Then the recovered source sounds will be proportional to the recovered source sounds with  $\alpha_i$  fixed at their true values. In practice, it is sufficient to recover the sounds in proportion. Moreover, each column of matrix  $A$  estimated with pre-fixed  $\alpha_i$  will also be proportional to the corresponding column of  $A$  estimated with  $\alpha_i$  fixed at their true values.

#### 4.2. Auto-regressive model

In this subsection, we consider the auto-regressive model that captures the temporal patterns of the source sounds. We assume that the sounds produced by each independent source over time follow an auto-regressive time series model of order  $d$ . Specifically, for each source  $i$ ,  $s_{it} = \phi_{i1}s_{i(t-1)} + \dots + \phi_{id}s_{i(t-d)} + \varepsilon_{it}$ , where  $\phi_{i1}, \dots, \phi_{id}$  are the auto-regressive coefficients, and  $\varepsilon_{it}$  are i.i.d. normal random variables with variance  $\sigma_i^2$ . Thus the probability density of  $(\mathbf{s}_t, t = 1, \dots, T)$  is

$$f(\mathbf{s}_t, t = 1, \dots, T) = \prod_{i=1}^M f_i(s_{it}, t = 1, \dots, T) \propto \prod_{i=1}^M \prod_{t>d} \exp\left(-\frac{\|s_{it} - \phi_{i1}s_{i(t-1)} - \dots - \phi_{id}s_{i(t-d)}\|^2}{2\sigma_i^2}\right). \quad (21)$$

For each source sequence, the initial values of the AR model,  $s_{i(1-d)}, \dots, s_{i0}$ , are taken to be zeros. The likelihood function is based on the distribution of  $s_{i1}, \dots, s_{iT}$  conditional on these initial values.

First we conduct a simulation study. We generate three independent sequences of source sounds with the same order  $d = 8$  over 2000 time points. Two observed receiver sounds are obtained by the following mixing equation:

$$\begin{pmatrix} x_{1t} \\ x_{2t} \end{pmatrix} = \begin{pmatrix} 1 & 0.7071 & 0.7071 \\ 0 & 0.7071 & -0.7071 \end{pmatrix} \begin{pmatrix} s_{1t} \\ s_{2t} \\ s_{3t} \end{pmatrix}. \quad (22)$$

We assume that the coefficients and the variances of the auto-regressive models are known. We run the null space algorithm, and use the last 1000 iterations for posterior estimation. The posterior mean of the mixing matrix  $A$  is

$$\hat{A} = \begin{pmatrix} 0.9903 & 0.6672 & 0.7850 \\ 0.0070 & 0.7331 & -0.6682 \end{pmatrix}.$$

The corresponding 95% posterior intervals for the elements of  $A$  are

(0.9276, 1.0458)	(0.5421, 0.7609)	(0.6775, 0.9010)
(-0.0645, 0.0797)	(0.6539, 0.8107)	(-0.7526, -0.5681)

All the posterior intervals cover the true values of the corresponding elements of  $A$ . The SNR are 4.9179, 18.0830 and 14.1024, respectively. The original source sounds and the recovered sounds are shown in Fig. 3.

To make the Monte Carlo study more informative, we repeat this experiment five times. For each replication, the source sequences are re-generated according to the same auto-regressive models, and the source sequences are mixed by the same mixing matrix  $A$ . We use the same algorithm for posterior calculations. The inferential results for the elements of  $A$  and the corresponding SNR values are shown in Tables 3 and 4. The inferential results for the five replications are similar to each other, even though each replication involves a different data set generated under the same conditions.

Next, we perform a simulation study where we assume that the auto-regressive coefficients are unknown. Following Chib and Greenberg (1994), we put diffused normal prior distributions on the auto-regressive coefficients. In each iteration of the null space algorithm, we add a step that samples these coefficients from their conditional posterior distribution given the recovered source sounds. In this simulation study, the source sounds are independently generated from three different auto-regressive processes with order  $d = 8$  over 3000 time points. The receiver sounds are obtained by mixing the source sounds:

$$\begin{pmatrix} x_{1t} \\ x_{2t} \end{pmatrix} = \begin{pmatrix} 1 & 0.7071 & -0.7071 \\ 0 & 0.7071 & 0.7071 \end{pmatrix} \begin{pmatrix} s_{1t} \\ s_{2t} \\ s_{3t} \end{pmatrix}. \quad (23)$$

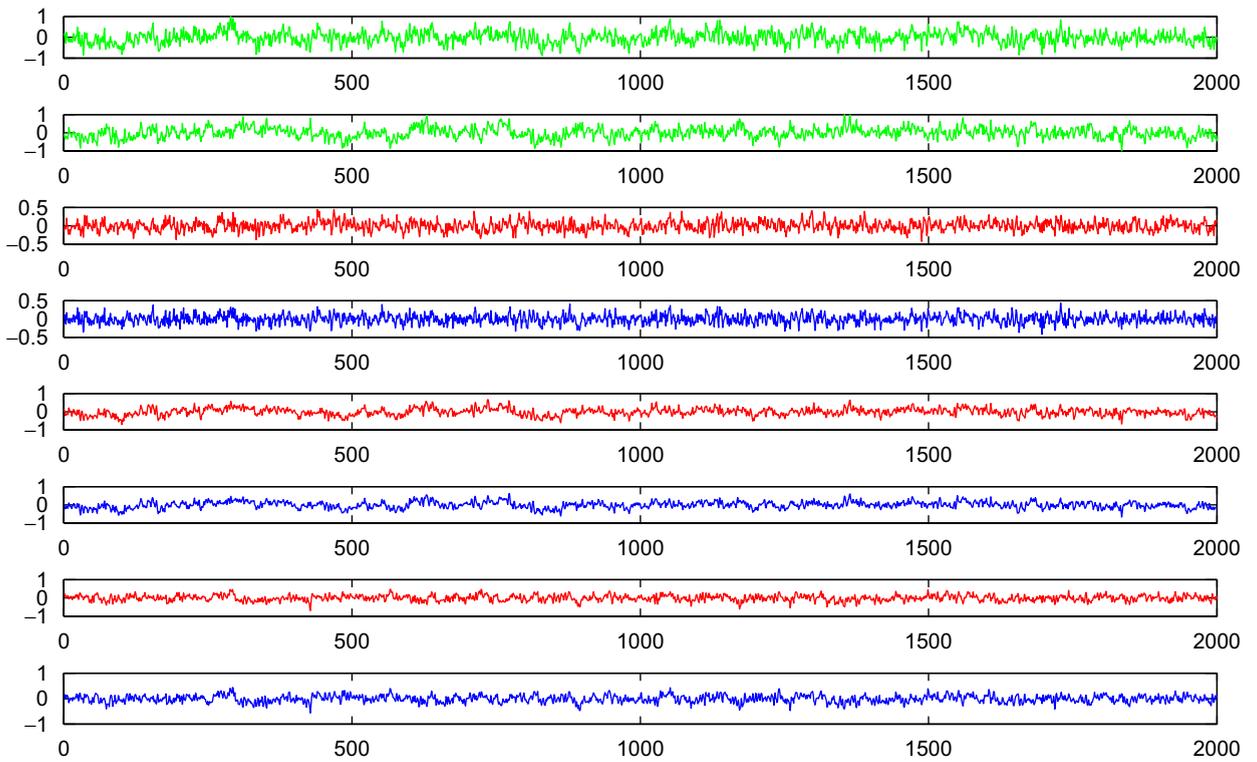


Fig. 3. Simulation study where the auto-regressive coefficients are assumed known. The green lines are the observations, the red lines are the true sources, and the blue lines are the recovered sources. The SNR values are 4.9179, 18.0830 and 14.1024, respectively.

To ensure identifiability, we fix the variances  $\sigma_i^2$  in the three auto-regressive processes at their true values. We also fix the order of the first coefficients of the three processes, and the initial values of the coefficients are all set to be zeros. We use the last 1000 iterations for posterior approximation. The estimate of the mixing matrix,  $\hat{A}$ , is

$$\begin{pmatrix} 0.7936 & 0.6994 & -0.8565 \\ -0.0591 & 0.7992 & 0.5579 \end{pmatrix},$$

and the corresponding 95% posterior intervals are

(0.5685,1.2507)	(0.5552, 0.8483)	(-1.1285, -0.2779)
(-0.3886,0.1185)	(0.6818, 0.8849)	(0.2856,0.7334)

The SNR values for the recovered signals are 3.7129, 16.7582 and 11.9918, respectively. Fig. 4 displays the original sources and the recovered sources. We also show the posterior means and 95% posterior intervals of the auto-regressive coefficients in Table 5.

Finally, we conduct an experiment using three natural sounds: fire, thunder and water with 2000 time points for each sound. These three sources are mixed via the mixing matrix

$$A = \begin{pmatrix} 1 & 0.7071 & 0.7071 \\ 0 & 0.7071 & -0.7071 \end{pmatrix}.$$

We model these three source sequences by auto-regressive processes with order  $d = 8$  with unknown coefficients and variances.

Note that there is a similar identifiability issue to the one discussed at the end of the previous subsection. Specifically, if we change  $\sigma_i$  to  $h_i\sigma_i$  for  $i = 1, \dots, M$ , and change  $A_{ki}$  to  $A_{ki}/h_i$  for  $k = 1, \dots, m$ , the distribution of the observed

Table 3

The estimates of the elements of  $A$  and the corresponding 95% posterior intervals (in parentheses) for each of the five replications

Repetition	$A(1, 1)$	$A(1, 2)$	$A(1, 3)$	$A(2, 1)$	$A(2, 2)$	$A(2, 3)$
1	0.9294 (0.8456, 1.0123)	0.6504 (0.5417, 0.7346)	0.8396 (0.7170, 0.9608)	0.0933 (0.0144, 0.1789)	0.7313 (0.6658, 0.7967)	-0.6832 (-0.7571, -0.5984)
2	1.0186 (0.9123, 1.0407)	0.5909 (0.6331, 0.8047)	0.7834 (0.5959, 0.8162)	0.0531 (-0.1002, 0.0629)	0.7075 (0.6164, 0.7382)	-0.6667 (-0.7598, -0.6339)
3	1.0089 (0.9367, 1.0723)	0.7382 (0.6340, 0.8292)	0.6698 (0.5588, 0.7836)	-0.0298 (-0.1162, 0.0478)	0.7187 (0.6461, 0.7924)	-0.7250 (-0.7975, -0.6468)
4	0.9419 (0.8689, 1.0103)	0.6889 (0.5875, 0.7882)	0.7317 (0.6113, 0.8535)	0.0452 (-0.0453, 0.1433)	0.6858 (0.5978, 0.7687)	-0.7508 (-0.8321, -0.6622)
5	1.0338 (0.9709, 1.0988)	0.6586 (0.5655, 0.7443)	0.7174 (0.6042, 0.8286)	0.0076 (-0.0701, 0.0814)	0.7433 (0.6696, 0.8053)	-0.6797 (-0.7563, -0.6056)

Each replication involves a different data set generated under the same conditions.

Table 4  
The SNR values for each of the five replications

Repetition	1	2	3	4	5
Source 1	3.1090	3.6594	3.5728	2.6680	2.4091
Source 2	21.1426	16.3795	20.6911	19.4474	18.2808
Source 3	16.6774	16.4046	15.2829	14.5320	13.1357

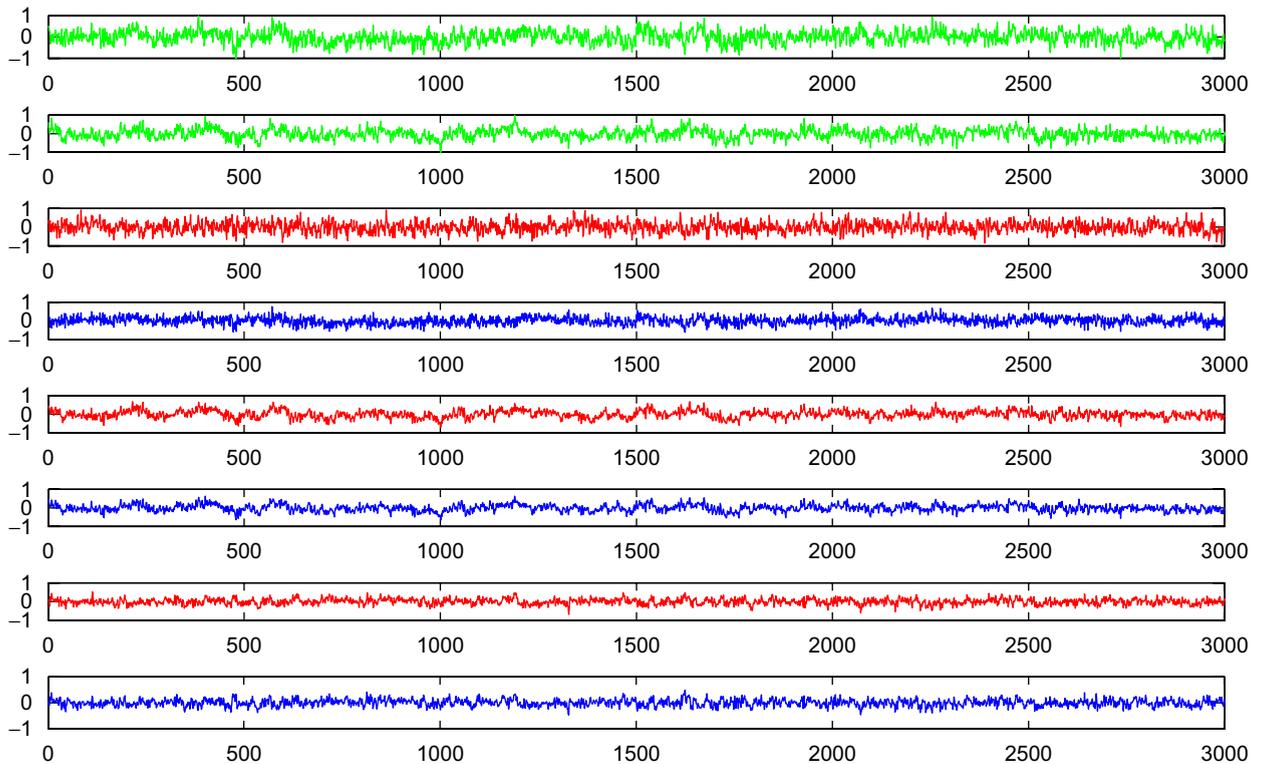


Fig. 4. Simulation study where the auto-regressive coefficients are assumed unknown. The green lines are the observations, the red lines are the true sources and the blue lines are the recovered sources. The SNR values for the recovered signals are 3.7129, 16.7582 and 11.9918, respectively.

receiver sounds ( $\mathbf{x}_t$ ,  $t = 1, \dots, T$ ) will not change. In this experiment, we fix  $\sigma_i^2 = 1$  for  $i = 1, 2, 3$ . Then the recovered source sounds will be proportional to the recovered source sounds with  $\sigma_i^2$  fixed at their true values, and it is sufficient to recover source sounds in proportion. Moreover, the estimated ratio  $A_{2i}/A_{1i}$  for  $i = 1, 2, 3$  will not depend on the values of  $\sigma_i^2$ . We run the null space algorithm and use the last 1000 iterations for posterior calculation. The posterior means of the ratios  $A_{2i}/A_{1i}$  for  $i = 1, 2, 3$  are  $(-0.0822, 0.9814, -1.0093)$ , which are very close to the true ratios,  $(0, 1, -1)$ . The 95% posterior intervals of these ratios are  $(-0.1510, -0.0305)$ ,  $(0.9529, 1.0106)$  and  $(-1.8139, -0.8460)$ . As to the recovered source sounds, after normalizing the marginal variances of the true sources and the recovered sources, respectively, we compute the SNR values, which are 8.2252, 32.2237 and 21.4048, respectively. The original sources and the recovered sources are shown in Fig. 5.

In our current experiments, the lag order of the auto-regressive model is empirically set at 8. For long sequences, we suggest to use larger lag order to better capture the temporal structures of the source sequences.

Table 5

The true values of the auto-regressive coefficients of the original sources and the posterior means and 95% posterior intervals (in parentheses) of these coefficients

True	$\phi_1$	$\phi_2$	$\phi_3$	$\phi_4$	$\phi_5$	$\phi_6$	$\phi_7$	$\phi_8$
Source 1	0.6597	-0.2889	0.2393	-0.1783	0.1632	-0.1837	0.1891	-0.0539
Source 2	1.0619	-0.7131	0.5600	-0.3641	0.3553	-0.2324	0.2071	0.0052
Source 3	0.8884	-0.4649	0.3086	-0.1618	0.2279	-0.0850	0.0268	-0.0082
<i>Estimated</i>								
Recover 1	0.5429 (0.32,0.77)	-0.2382 (-0.40,-0.09)	0.2927 (0.12,0.44)	-0.1724 (-0.31,0.00)	0.1902 (0.04,0.38)	-0.1758 (-0.32,-0.02)	0.1895 (-0.06,0.43)	-0.0564 (-0.06,0.23)
Recover 2	1.0288 (0.97,1.10)	-0.6624 (-0.77,-0.59)	0.4888 (0.42,0.58)	-0.2874 (-0.37,-0.22)	0.2979 (0.23,0.37)	-0.1910 (-0.26,-0.13)	0.1942 (0.12,0.26)	-0.0096 (-0.05,0.03)
Recover 3	0.8905 (0.81,0.99)	-0.4736 (-0.68,-0.37)	0.3135 (0.21,0.58)	-0.1789 (-0.44,-0.01)	0.2079 (0.02,0.43)	-0.0755 (-0.35,0.20)	0.0328 (-0.22,0.35)	-0.0247 (-0.17,0.05)

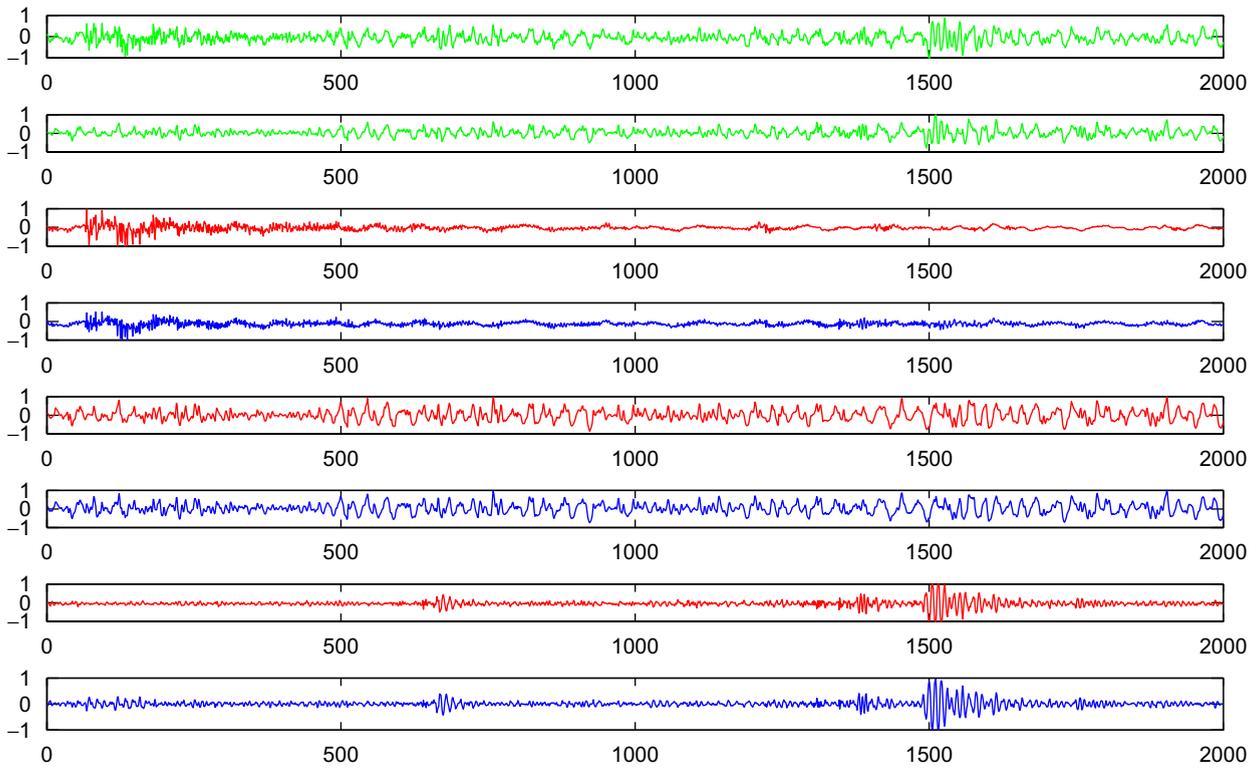


Fig. 5. Blind separation of three natural sound sequences. The green lines are the observations, the red lines are the true sources and the blue lines are the recovered sources. The SNR values are 8.2252, 32.2237 and 21.4048, respectively.

## 5. Discussion

Unlike the complete case where no error term is assumed, previous work reviewed in Section 1.2 on over-complete blind source separation or over-complete independent component analysis always assumes an error term in addition to linear mixing,

$$\mathbf{x}_t = A\mathbf{s}_t + \varepsilon_t, \quad t = 1, \dots, T, \quad (24)$$

where  $\varepsilon_t \sim N(0, \sigma^2 I_m)$  independently for  $t = 1, \dots, T$ . For this model, the likelihood  $p(\mathbf{x}_t, t = 1, \dots, T|A)$  or the posterior distribution  $p(A, \mathbf{s}_t, t = 1, \dots, T|\mathbf{x}_t, t = 1, \dots, T)$  can be formed without resorting to null space representation (11) or singular value decomposition (8). The Markov chain Monte Carlo posterior sampling can be accomplished by iteratively sampling  $(\mathbf{s}_t, t = 1, \dots, T)$  from the conditional posterior distribution  $p(\mathbf{s}_t, t = 1, \dots, T|A, \mathbf{x}_t, t = 1, \dots, T)$ , and then sampling  $A$  from the conditional posterior distribution  $p(A|\mathbf{s}_t, \mathbf{x}_t, t = 1, \dots, T)$ .

Compared to the previous work, what we are trying to solve is the limiting case where  $\sigma^2 \rightarrow 0$ . When  $\sigma^2 = 0$ , the previous methods do not directly apply. In terms of generalizing the complete blind source separation or independent component analysis to the over-complete case, our method is a more direct generalization than previous work, because no error term is assumed in the complete case.

Our method can certainly be applied to model (24), but it does not offer inferential advantage over previous methods, because the underlying models are the same. We believe that our method can offer computational advantage in terms of Markov chain Monte Carlo posterior sampling when  $\sigma^2$  is very small. Specifically, we need to modify the null space representation (11) to

$$\mathbf{s}_t = A^- \mathbf{x}_t + V_2 c_t + V_1 e_t, \quad (25)$$

where  $e_t$  is an  $m$ -dimensional vector composed of coordinates in the space spanned by  $V_1$ , which is also the space spanned by the rows of  $A$ . In representation (25),  $A^{-1}\mathbf{x}_t + V_2c_t$  solves  $A\mathbf{s}_t = \mathbf{x}_t$  exactly, and the additional  $V_1e_t$  term represents the additional degrees of freedom allowed by the variance  $\sigma^2$  of the error term  $\varepsilon_t$  in Eq. (24). If  $\sigma^2$  is small,  $\mathbf{s}_t$  is still highly constrained in Eq. (24). In terms of representation (25), the constraint caused by small  $\sigma^2$  only affects  $e_t$ , but not  $c_t$ , because  $c_t$  can change arbitrarily without changing  $A\mathbf{s}_t$ . Therefore, if we are to sample from the conditional posterior distribution  $p(\mathbf{s}_t, t = 1, \dots, T | A, \mathbf{x}_t, t = 1, \dots, T)$  when  $\sigma^2$  is small, the Gibbs sampler that samples  $p(\mathbf{s}_t, t = 1, \dots, T | A, \mathbf{x}_t, t = 1, \dots, T)$  can be slow because the components of  $\mathbf{s}_t$  are highly correlated. In contrast, under representation (25), the correlations between  $c_t$  and  $e_t$  can be much smaller, because  $c_t$  is free from the constraint caused by the small  $\sigma^2$ . Therefore, sampling  $c_t$  and  $e_t$  using the Gibbs sampler can be much faster than sampling  $\mathbf{s}_t$  directly. For the same reason, when  $\sigma^2$  is small, sampling  $p(A | \mathbf{s}_t, \mathbf{x}_t, t = 1, \dots, T)$  can also be difficult because of the constraint. But sampling  $U$ ,  $V$  and  $D$  does not suffer from this constraint. We shall explore this issue in future work.

We would also like to point out that previous work on over-complete blind source separation usually assumes double exponential model for the source sounds. The auto-regressive model has only been used for blind source separation in complete case, but not in over-complete case. In addition, previous work only gives the point estimate of the mixing matrix, without quantifying inferential uncertainties.

## Acknowledgments

The authors are grateful to the anonymous referees for the helpful comments and suggestions. The work reported in article is supported in part by the National Science Council of Taiwan, R.O.C. under grant NSC 93-2118-M-390-002.

## References

- Bell, A.J., Sejnowski, T.J., 1995. An information-maximization approach to blind separation and blind deconvolution. *Neural Comput.* 7, 1129–1159.
- Bell, A.J., Sejnowski, T.J., 1997. The ‘Independent components’ of natural scenes are edge filters. *Vis. Res.* 37, 3327–3338.
- Chib, S., Greenberg, E., 1994. Bayes inference in regression models with ARMA ( $p, q$ ) errors. *J. Econometrics* 64, 183–206.
- Choi, S., Cichocki, A., 2000. Blind separation of nonstationary sources in noisy mixtures. *Electron. Lett.* 36 (9), 848–849.
- Comon, P., 1994. Independent component analysis—a new concept? *Signal Process.* 36 (3), 287–314.
- Dempster, A.P., Laird, N.M., Rubin, D.B., 1977. Maximum likelihood from incomplete data via the EM algorithm. *J. Roy. Statist. Soc. B* 39, 1–38.
- Girolami, M., 2001. A variational method for learning sparse and overcomplete representations. *Neural Comput.* 13, 2517–2532.
- Hosseini, S., Jutten, C., Pham, D.T., 2003. Markovian source separation. *IEEE Trans. Signal Process.* 51 (12), 3009–3019.
- Lee, T.-W., Lewicki, M.S., Girolami, M., Sejnowski, T.J., 1999. Blind source separation of more sources than mixtures using overcomplete representations. *IEEE Signal Process. Lett.* 6 (4), 87–90.
- Lewicki, M.S., Olshausen, B.A., 1999. Probabilistic framework for the adaptation and comparison of image codes. *J. Opt. Soc. Amer. A* 16, 1587–1601.
- Lewicki, M.S., Sejnowski, T.J., 2000. Learning overcomplete representations. *Neural Comput.* 12, 337–365.
- Pearlmutter, B., Parra, L., 1997. Maximum likelihood blind source separation: a context-sensitive generalization of ICA. *Adv. Neural Inform. Process. System* 9, 613–619.
- Roberts, G.O., Rosenthal, J.S., 1998. Optimal sacling of discrete approximations to Langevin diffusions. *J. Roy. Statist. Soc. B* 60, 255–268.
- Tanner, M., Wong, W., 1987. The calculation of posterior distributions by data augmentation. *J. Amer. Statist. Assoc.* 82, 528–550.