# Dual Multivariate Auto-Regressive Modeling in State Space for Temporal Signal Separation

Yiu-ming Cheung, *Member, IEEE*, and Lei Xu, *Fellow, IEEE*

***Abstract*—Many existing independent component analysis (ICA) approaches result in deteriorated performance in temporal source separation because they have not taken into consideration of the underlying temporal structure of sources. In this paper, we model temporal sources as a general multivariate auto-regressive (AR) process whereby an underlying multivariate AR process in observation space is obtained. In this dual AR modeling, the mixing process from temporal sources to observations is the same as the mixture from the nontemporal residuals of the source AR (SAR) process to that of the observation AR (OAR) process. We can therefore avoid the source temporal effects in performing ICA by learning the demixing system on the independently distributed OAR residuals rather than the time-correlated observations. Particularly, we implement this approach by modeling each source signal as a finite mixture of generalized autoregressive conditional heteroskedastic (GARCH) process. The adaptive algorithms are proposed to extract the OAR residuals appropriately online, together with learning the demixing system via a nontemporal ICA algorithm. The experiments have shown its superior performance on temporal source separation.**

***Index Terms*—Blind signal separation, dual auto-regressive processes, generalized autoregressive conditional heteroskedastic (GARCH) model, independent component analysis.**

## I. Introduction

BLIND signal separation (BSS) has recently received wide attention in the literature of signal processing and neural networks due to its attractive applications in many fields. For example, in medical signal processing, Makeig *et al.* [21] and Jung *et al.* [16] have shown that the BSS technique can extract electroencephalogram (EEG) activations and linearly decompose EEG artifacts such as line noise, eye blinks, and cardiac noise into independent components with sub-Gaussian and super-Gaussian distributions. Actually, Mckeown *et al.* [22] have used the BSS algorithms to investigate task-related human brain activity in functional magnetic resonance imaging (fMRI) data. Moreover, in wireless communications and speech recognition, Torkkola [28] has shown that the BSS technique can successfully separate the radio signals in fading channels of CDMA mobile system. Lee *et al.* [19] showed that the recognition rate of an automatic speech recognition system was increased after separating the speech signals. Apart from these applications, the BSS techniques are also applicable to unsupervised data classification [20], image feature extractions [5], time series analysis [3], [18], data mining [13], and so on. Hence, the exploration of BSS has been greatly attracting the researchers in the community during the past decade.

The blind signal separation with an instantaneous linear mixture can be formulated into the independent component analysis (ICA) problem: Suppose $k$ channels of non-Gaussian source signals that are statistically independent each other are sampled at discrete time $t$, denoted as $\mathbf{y}_t = [y_t^{(1)}, \ldots, y_t^{(k)}]^T$ with $t = 0, 1, 2, \ldots$. The sources are instantaneously and linearly mixed by an unknown full-column matrix $\mathbf{A}$ and observed as $\mathbf{x}_t = [x_t^{(1)}, \ldots, x_t^{(d)}]^T$:

$$\mathbf{x}_t = \mathbf{A}\mathbf{y}_t. \tag{1}$$

The objective of an ICA approach is to recover $\mathbf{y}_t$s up to a constant scale and any permutation of indices through the observations $\mathbf{x}_t$s by finding out a demixing matrix $\mathbf{W}$ such that

$$\hat{\mathbf{y}}_t = \mathbf{W}\mathbf{x}_t = \mathbf{W}\mathbf{A}\mathbf{y}_t = \mathbf{P}\mathbf{D}\mathbf{y}_t \tag{2}$$

where $\mathbf{P}$ is a $k \times k$ permutation matrix, $\mathbf{D}$ is a $k \times k$ diagonal matrix, and $\hat{\mathbf{y}}_t$ is the recovered signal of $\mathbf{y}_t$.

In the literature, a lot of ICA approaches based on different methodologies and theories have been proposed. Roughly, these methods can be separated into two categories: *one-step ICA approaches* and *two-step ICA approaches*. One-step approaches include maximum likelihood [27], negentropy maximization [14], INFOMAX [4], minimizing mutual information (MMI) [1], and learned parametric mixture (LPM) [32], [33], which perform independent component analysis upon the observed signals $\mathbf{x}_t$s without any preprocessing. In contrast, two-step ICA approaches, e.g., nonlinear principal component analysis (PCA) [24], [17] and cumulants-based methods [11], perform independent component analysis with two steps. The first step is to prewhiten the observations such that the second-order redundancy in the $\mathbf{x}_t$ is removed. Then, the second step uses higher order statistics to further reduce the remaining redundancy within the prewhitened observations. In general, these two-category approaches perform source separation without considering the internal time correlations in each source, thus resulting in deteriorated performance in separating temporal sources. Hereafter, we also call them *nontemporal* ICA approaches.

Recently, some ICA works have been done toward temporal sources. For example, considering the time delayed correlation matrix of observations with a preassigned delay parameter,

Manuscript received September 23, 2000; revised December 27, 2001. This work was supported by the Research Grant Council of the Hong Kong SAR under Project CUHK4169/00E and by a Faculty Research Grant of Hong Kong Baptist University under Project Code FRG/01-02/II-24. This paper was recommended by Associate Editor I. Gu.

Y. Cheung is with the Department of Computer Science, Hong Kong Baptist University, Hong Kong (e-mail: ymc@comp.hkbu.edu.hk).

L. Xu is with the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong (e-mail: lxu@cse.cuhk.edu.hk).

Digital Object Identifier 10.1109/TSMCB.2003.811132

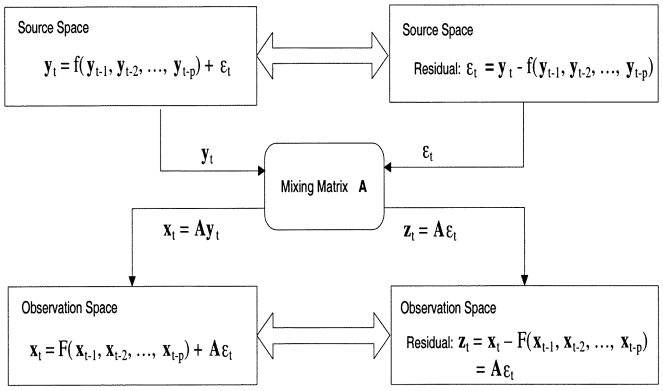



Fig. 1. Invariant mixing process in the ASSA approach, either between the source $\mathbf{y}_t$ and the observation $\mathbf{x}_t$ or between their residuals.

Molgedey *et al.* [23] formulated the ICA problem as an eigenvalue analysis which involves the simultaneous diagonalization of two symmetric matrices whose elements are measurable time delayed correlation functions. Attias [2] modeled each source as a linear combination of non-Gaussian white signals with the coefficients learned together with the demixing matrix by minimizing a Kullback–Leibler (KL) error function. Moreover, Pearlmutter *et al.* [25], [26] modeled the density function of a source as a mixture of logistic densities with the source mean being a linear function of the recent values of that source and tuned the density parameters by the maximum likelihood (ML) approach while learning $\mathbf{W}$ by the natural gradient descent method [1]. In recent years, Xu [30], [31] has developed a temporal Bayesian Ying–Yang (TBYY) learning theory that models temporal sources and observations in general state-space equations. Not only does the TBYY theory present a unified point of view on Kalman filter, hidden Markov model (HMM), and ICA with some extensions provided [31] but also solves: 1) temporal binary BSS problem via a higher order independent HMM and 2) temporal real BSS problem via temporal ICA (TICA) and temporal factor analysis developed from the TBYY learning theory. Particularly, it has been shown that not only non-Gaussian but also Gaussian process sources can be separated by the TICA through exploring the internal temporal dependence of each source. Furthermore, a degenerated case of the TICA becomes equivalent to the method proposed by Pearlmutter *et al.* [25], [26].

All the above studies belong to the one-step approach. In our preliminary papers [9], [10], we have proposed a specific two-step temporal approach called the *autoregressive-based signal separation approach* (ASSA). This method models $k$ independent channels in state space (or called *source space*) as a $p$-order multivariate autoregressive (AR) process, which results in obtaining a corresponding multivariate AR process in observation space. For convenience, we denote the AR processes in source and observation spaces by SAR and OAR, respectively. As shown in Fig. 1, in this dual AR modeling, the OAR residual is exactly the mixture of component-wise independent SAR residual by the same mixing matrix $\mathbf{A}$. Since the OAR residuals are also statistically independent in time domain, we can still apply a nontemporal ICA algorithm to learn $\mathbf{W}$ from OAR residuals. Hence, this two-step approach has actually provided a straightforward way to extend the power of nontemporal ICA algorithms to temporal source separation. The preliminary results in [9] and [10] have shown that the ASSA can successfully separate the temporal sources. The similar idea that estimates $\mathbf{W}$ based on observation innovations rather than $\mathbf{x}_t$s was also proposed in [15]. Generally, the performance of such an innovation-based method much depends on the estimation of the innovation process. However, [15] has not conducted the studies on how to appropriately estimate the innovation process, particularly on that such a process is nonstationary.

In this paper, we will elaborate a two-step approach in a broad view, which includes the extraction of OAR residuals, the learning of residual parameter, and the learning of the demixing matrix $\mathbf{W}$. Particularly, we study its specific case as a generalization of the ASSA approach [9], [10] with further improvements on two-fold. On the one hand, we generalize the AR source model to be a finite mixture of generalized autoregressive conditional heteroskedastic (GARCH) process [6]. The GARCH models the noise variance of each source varying over time. It is therefore believed that GARCH model is better to model nonstationary source signals. On the other hand, we present a ML learning algorithm, which includes the generalized least-mean-square (LMS) algorithms in [10] as a special case, to tune the parameters such that the OAR residuals are appropriately extracted online. The experiments have shown that the proposed method has a robust performance in separating nonstationary sources and outperforms an existing nontemporal ICA approach.

This paper is organized as follows. Section II gives out a general two-step approach to temporal signal separation, where the residual definition and the basic procedure of this method are both described. Section III studies this approach in a specific case in detail. We model each source as a finite mixture of GARCH process and, therefore, obtain a detailed ML learning algorithm to tune the parameters such that the OAR residuals are appropriately extracted on line. Furthermore, LPM, an existing nontemporal ICA algorithm used in this paper, is also briefly introduced in Section IV. We experimentally compare the performance of our proposed approach with the individual LPM algorithm in Section V and make a discussion in Section VI. Last, we draw a conclusion in Section VII.

## II. General Two-Step Temporal Approach

### A. General Principle of a Two-Step Temporal Approach

Suppose $k$ source signals $y_t^{(1)}, y_t^{(2)}, \ldots, y_t^{(k)}$ with $0 \le t \le N$ are statistically independent, and each of them can be generally modeled as an AR process

$$y_t^{(j)} = f_j\left(Y_{t-1}^{(j)} \middle| \theta_j\right) + \varepsilon_t^{(j)}, \qquad 1 \le j \le k \tag{3}$$

where $f_j(Y_{t-1}^{(j)}|\theta_j)$ with $Y_{t-1}^{(j)} = \{y_{t-1}^{(j)}, y_{t-2}^{(j)}, \ldots, y_0^{(j)}\}$ is a deterministic function of $Y_{t-1}^{(j)}$ with the parameter set $\theta_j$, and $\varepsilon_t^{(j)}$ is zero-mean SAR residual that is independent over time.

Here, we suppose that $\varepsilon_t^{(j)}$s are generally non-Gaussian distributed with at most one being Gaussian. For simplicity, we can also express (3) in matrix form as

$$\mathbf{y}_t = \mathbf{f}(\mathbf{Y}_{t-1}|\boldsymbol{\Theta}) + \boldsymbol{\varepsilon}_t \tag{4}$$

where $\mathbf{y}_t = [y_t^{(1)}, y_t^{(2)}, \ldots, y_t^{(k)}]^T$, $\mathbf{f} = [f_1, f_2, \ldots, f_k]^T$, $\mathbf{Y}_{t-1} = [\mathbf{y}_{t-1}^T, \mathbf{y}_{t-2}^T, \ldots, \mathbf{y}_0^T]^T$, $\boldsymbol{\Theta} = \{\theta_1, \theta_2, \ldots, \theta_k\}$, and $\boldsymbol{\varepsilon}_t = [\varepsilon_t^{(1)}, \varepsilon_t^{(2)}, \ldots, \varepsilon_t^{(k)}]^T$.

Through the ICA model of (1), we then have

$$\begin{aligned}\mathbf{x}_t &= \mathbf{A}\mathbf{y}_t \\ &= \mathbf{A}[\mathbf{f}(\mathbf{Y}_{t-1}|\boldsymbol{\Theta}) + \boldsymbol{\varepsilon}_t] \\ &= \mathbf{A}\mathbf{f}(\mathbf{Y}_{t-1}|\boldsymbol{\Theta}) + \mathbf{A}\boldsymbol{\varepsilon}_t.\end{aligned} \tag{5}$$

Since the mixing matrix $\mathbf{A}$ in (1) is full-column rank, there must exist at least a $\mathbf{W}$ such that (2) is held. Therefore, the term $\mathbf{A}\mathbf{f}(\mathbf{Y}_{t-1}|\boldsymbol{\Theta})$ in (5) can also be re-expressed as

$$\begin{aligned}\mathbf{A}\mathbf{f}(\mathbf{Y}_{t-1}|\boldsymbol{\Theta}) &= \mathbf{A}\mathbf{f}(\mathbf{y}_{t-1}, \mathbf{y}_{t-2}, \ldots, \mathbf{y}_0|\boldsymbol{\Theta}) \\ &= \mathbf{A}\mathbf{f}(\mathbf{W}\mathbf{x}_{t-1}, \mathbf{W}\mathbf{x}_{t-2}, \ldots, \mathbf{W}\mathbf{x}_0|\boldsymbol{\Theta}) \\ &= \mathbf{F}(\mathbf{X}_{t-1}|\boldsymbol{\Theta})\end{aligned} \tag{6}$$

with

$$\begin{aligned}\mathbf{F}(\mathbf{X}_{t-1}|\boldsymbol{\Theta}) &= \mathbf{A}\mathbf{f}(\mathbf{W}\mathbf{x}_{t-1}, \mathbf{W}\mathbf{x}_{t-2}, \ldots, \mathbf{W}\mathbf{x}_0|\boldsymbol{\Theta}) \\ \mathbf{X}_{t-1} &= [\mathbf{x}_{t-1}^T, \mathbf{x}_{t-2}^T, \ldots, \mathbf{x}_0^T]^T\end{aligned}$$

where $\mathbf{F}(\mathbf{X}_{t-1}|\boldsymbol{\Theta})$ is also a deterministic function of the past observations. Consequently, (5) becomes

$$\mathbf{x}_t = \mathbf{F}(\mathbf{X}_{t-1}|\boldsymbol{\Theta}) + \mathbf{A}\boldsymbol{\varepsilon}_t \tag{7}$$

which is actually an underlying AR process existing in observation space. We define the OAR residual $\mathbf{z}_t$ by

$$\mathbf{z}_t = \mathbf{x}_t - \mathbf{F}(\mathbf{X}_{t-1}|\boldsymbol{\Theta}) \tag{8}$$

and following from (7), we then have

$$\mathbf{z}_t = \mathbf{A}\boldsymbol{\varepsilon}_t. \tag{9}$$

It shows that the residual $\mathbf{z}_t$ is the linear instantaneous mixture of $k$ independent $\varepsilon_t^{(j)}$s with the same mixing matrix $\mathbf{A}$. Since $\boldsymbol{\varepsilon}_t$ and $\boldsymbol{\varepsilon}_{t-\tau}$, for any $t$ and $\tau \geq 1$ are statistically independent without temporal dependence, we can therefore estimate the demixing matrix $\mathbf{W}$ based on $\mathbf{z}_t$ via a nontemporal ICA algorithm. Consequently, at each time step $t$, we perform two steps as follows. **Step 1)** Extract $\mathbf{z}_t$ according to (8), and **Step 2)** based on (9), use a nontemporal ICA algorithm such as LPM one [32], [33] to adjust $\mathbf{W}$ by a small-step size with $\mathbf{z}_t$ as its input while adjusting $\boldsymbol{\Theta}$ by a small step as well. Here, two points should be noted. One is the selection of a nontemporal ICA algorithm invoked in this two-step approach. In general, we should choose an ICA algorithm with the computing complexity as small as possible, but it can separate any combination of sub-Gaussian and super-Gaussian source signals. In this paper, we will choose the existing LPM algorithm as an example, whose details will be described in Section IV. The other point is that this two-step approach much depends on the appropriate extraction of the OAR residuals $\mathbf{z}_t$s. In (8), the residual extraction involves two unknowns. One is the function form of $\mathbf{F}$, and the other is the unknown parameter set $\boldsymbol{\Theta}$. In general, the former can be determined after a source model $\mathbf{f}$ is explicitly specified. Therefore, the remaining critical task is how to appropriately estimate $\boldsymbol{\Theta}$. In the following subsection, we will give a general ML procedure to estimate $\boldsymbol{\Theta}$ adaptively.

### *B. General Procedures for OAR Residual Parameter Learning*

To estimate the OAR residual parameter $\boldsymbol{\Theta}$ in (8) via an ML algorithm, we need to determine the probability density function (pdf) of $\mathbf{z}_t$ based on that of $\boldsymbol{\varepsilon}_t$ through (9). Although we know nothing about the pdf of $\boldsymbol{\varepsilon}_t$ except for non-Gaussianity, it can be approximated by a universal density estimator. Here, we use a finite mixture of Gaussian densities, i.e.,

$$p(\boldsymbol{\varepsilon}_t) = \sum_{i=1}^{n} \gamma_i G(\boldsymbol{\varepsilon}_t|\mathbf{m}_i, \boldsymbol{\Sigma}_{t,i}), \quad \text{with } \gamma_i \geq 0, \sum_{i=1}^{n} \gamma_i = 1 \tag{10}$$

where $G(\mathbf{s}|\boldsymbol{m}, \boldsymbol{\Sigma})$ denotes the Gaussian density of the vector $\mathbf{s}$ with mean $\mathbf{m}$ and covariance matrix $\boldsymbol{\Sigma}$, and $n$ is a density mixture number. Since the components of $\boldsymbol{\varepsilon}_t$ are statistically independent, we let $\boldsymbol{\Sigma}_{t,i}$ be a diagonal matrix hereafter without loss of generality.

Intuitively, it can be understood from (10) that each individual component $G(\boldsymbol{\varepsilon}_t|\mathbf{m}_i, \boldsymbol{\Sigma}_{t,i})$ is the pdf of a dummy variable $\boldsymbol{\varrho}_i$, $\boldsymbol{\varepsilon}_t$ equals $\boldsymbol{\varrho}_i$ with the probability $\gamma_i$. That is, $\mathbf{z}_t$ is equal to $\mathbf{A}\boldsymbol{\varrho}_i$ with the probability $\gamma_i$. Since $\mathbf{A}\boldsymbol{\varrho}_i$ is also Gaussian distributed with the pdf $G(\boldsymbol{\varepsilon}_t|\mathbf{A}\mathbf{m}_i, \mathbf{A}\boldsymbol{\Sigma}_{t,i}\mathbf{A}^T)$, according to (9), we can therefore model the probability density function (pdf) of $\mathbf{z}_t$ by

$$p(\mathbf{z}_t) = \sum_{i=1}^{n} \gamma_i G(\mathbf{z}_t|\mathbf{A}\mathbf{m}_i, \mathbf{A}\boldsymbol{\Sigma}_{t,i}\mathbf{A}^T). \tag{11}$$

By (8), we then have

$$\begin{aligned}&p(\mathbf{x}_t|\mathbf{X}_{t-1}; \tilde{\boldsymbol{\Theta}}) \\ &\quad = \sum_{i=1}^{n} \gamma_i G(\mathbf{x}_t|\mathbf{F}(\mathbf{X}_{t-1}|\boldsymbol{\Theta}) + \mathbf{A}\mathbf{m}_i, \mathbf{A}\boldsymbol{\Sigma}_{t,i}\mathbf{A}^T)\end{aligned} \tag{12}$$

where $\tilde{\boldsymbol{\Theta}}$ consists of the parameters $\boldsymbol{\Theta}$, $\mathbf{A}$, $\gamma_i$s, $\mathbf{m}_i$s, and $\boldsymbol{\Sigma}_{t,i}$s.

Given a series of observations $\mathbf{x}_t$s with $t = 1, 2, \ldots, N$, the average log-likelihood function of the observed signals is

$$\begin{aligned}Q(\tilde{\boldsymbol{\Theta}}) &= \frac{1}{N} \ln p(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N; \tilde{\boldsymbol{\Theta}}) \\ &= \frac{1}{N} \ln p(\mathbf{x}_1|\mathbf{X}_0; \tilde{\boldsymbol{\Theta}}) \cdots p(\mathbf{x}_N|\mathbf{X}_{N-1}; \tilde{\boldsymbol{\Theta}}) \\ &= \frac{1}{N} \sum_{t=1}^{N} \ln p(\mathbf{x}_t|\mathbf{X}_{t-1}; \tilde{\boldsymbol{\Theta}}) \\ &= \frac{1}{N} \sum_{t=1}^{N} J_t(\tilde{\boldsymbol{\Theta}})\end{aligned} \tag{13}$$

where $J_t(\tilde{\boldsymbol{\Theta}}) = \ln p(\mathbf{x}_t|\mathbf{X}_{t-1}; \tilde{\boldsymbol{\Theta}})$, and $\mathbf{X}_0 = \mathbf{x}_0$. The ML estimate of $\tilde{\boldsymbol{\Theta}}$ can therefore be obtained by maximizing $Q(\tilde{\boldsymbol{\Theta}})$ through a constrained optimization algorithm in view of the limitation on $\gamma_i$s, as shown in (10). Alternatively, here, we let

$$\gamma_i = \frac{\exp(\beta_i)}{\sum_{r=1}^{n} \exp(\beta_r)}, \qquad 1 \leq i \leq n \tag{14}$$

in which the constraints of $\gamma_i$s are automatically satisfied, but the new variables $\beta_i$s are totally free. Consequently, instead of $\gamma_i$s, we can learn $\beta_i$s as well as other parameters in $\tilde{\Theta}$ by using an unconstrained optimization procedure such that $Q(\tilde{\Theta})$ is maximized. In this paper, we use a gradient ascent method to tune the parameters. That is, at each time step, we tune the parameters by a small-step size along the direction of maximizing

$$\begin{aligned} J_t(\Theta_1) &= J_t(\tilde{\Theta}) \\ &= \ln p(\mathbf{x}_t|\mathbf{X}_{t-1};\, \Theta_1) \\ &= \ln\left[\sum_{i=1}^{n} \gamma_i G(\mathbf{x}_t|\mathbf{F}(\mathbf{X}_{t-1}|\Theta)+\mathbf{A}\mathbf{m}_i,\, \mathbf{A}\Sigma_{t,\,i}\mathbf{A}^T)\right]. \end{aligned} \tag{15}$$

That is, we update

$$\Theta_1^{\text{new}} = \Theta_1^{\text{old}} + \eta \left.\frac{\partial J_t(\Theta_1)}{\partial \Theta_1}\right|_{\Theta_1^{\text{old}}} \tag{16}$$

where $\Theta_1 = \{\Theta,\, \mathbf{A},\, \beta_i\text{s},\, \mathbf{m}_i\text{s},\, \Sigma_{t,\,i}\text{s}\}$, and $\eta$ is a small positive learning rate. Under the circumstances, the detailed implementation of the previous **Steps 1** and **2** can be summarized, as shown in Table I.

TABLE I
IMPLEMENTATIONS OF THE DUAL AR MODELING APPROACH AT EACH TIME STEP $t$

| | |
|---|---|
| **Step 1** | Given $\Theta_1^{\text{old}}$ and $\mathbf{W}^{\text{old}}$, let<br>$\varepsilon_t = \mathbf{W}^{\text{old}}\mathbf{z}_t$, and $\hat{\mathbf{y}}_t = \mathbf{W}^{\text{old}}\mathbf{x}_t$.<br>where the OAR residual $\mathbf{z}_t$ is extracted by<br>$\mathbf{z}_t = \mathbf{x}_t - \mathbf{F}(\mathbf{X}_{t-1}|\Theta_1^{\text{old}})$. |
| **Step 2** | Given $\varepsilon_t$ and $\mathbf{z}_t$, update:<br>(i) $\mathbf{W}^{\text{new}}$ is given by a non-temporal ICA algorithm with $\mathbf{z}_t$ as its input;<br>(ii) $\Theta_1^{\text{new}} = \Theta_1^{\text{old}} + \eta \frac{\partial J_t(\Theta_1)}{\partial \Theta_1}\|_{\Theta_1^{\text{old}}}$. |

## III. SPECIFIC TWO-STEP APPROACH

As an example, this section will investigate the proposed approach by specifying the source model of (4) as *a finite mixture of GARCH process*. Consequently, a specific ML adaptive algorithm is obtained to realize **Step 1** in the previous section.

### A. GARCH Process and Source Modeling

Consider a scalar $p$-order AR model

$$y_t = \sum_{r=1}^{p} \lambda_r y_{t-r} + \varepsilon_t \tag{17}$$

where $\varepsilon_t$ denotes the zero-mean white residual with variance $\sigma_t^2$. Traditionally, $\sigma_t^2$ is regarded as a constant over time. However, [12], [6] has shown that a time-varied $\sigma_t^2$ over time instead of a constant is more useful in modeling nonstationary phenomena such as economic series. Particularly, Bollerslev [6] suggested that

$$\sigma_t^2 = \nu_0^2 + \sum_{r=1}^{n_q} \nu_r^2 \varepsilon_{t-r}^2 + \sum_{r=1}^{n_p} \psi_r^2 \sigma_{t-r}^2 \tag{18}$$

where $n_p \geq 0$, $n_q > 0$, $\nu_0^2$, $\nu_r^2$s, and $\psi_r^2$s are coefficients that need to be determined. Such a series of $\varepsilon_t$ is called a generalized autoregressive conditional heteroskedastic process, denoted as GARCH($n_q$, $n_p$).

When the pdf of $\varepsilon_t$ is modeled as a finite mixture of Gaussians, i.e.,

$$p(\varepsilon_t) = \sum_{i=1}^{n} \gamma_i G(\varepsilon_t|m_i,\, \sigma_{t,\,i}^2) \tag{19}$$

with

$$\sigma_{t,\,i}^2 = \nu_{i,\,0}^2 + \sum_{r=1}^{n_q} \nu_{i,\,r}^2 \varepsilon_{t-r}^2 + \sum_{r=1}^{n_p} \psi_{i,\,r}^2 \sigma_{t-r,\,i}^2 \tag{20}$$

we obtain a finite mixture of GARCH($n_q$, $n_p$) process [29].

In this paper, we further extend the above finite mixture of GARCH process to multivariate case. We model each source by a $p$-order AR process

$$y_t^{(j)} = \sum_{r=1}^{p} \lambda_r^{(j)} y_{t-r}^{(j)} + \varepsilon_t^{(j)} \tag{21}$$

with $1 \leq j \leq k$. Hence, (4) is explicitly specified as

$$\mathbf{y}_t = \sum_{r=1}^{p} \Lambda_r \mathbf{y}_{t-r} + \varepsilon_t \tag{22}$$

where $\Lambda_r$ is a diagonal matrix with $\lambda_r^{(j)}$, $j = 1, 2, \ldots, k$ as its diagonal elements due to the fact that the $k$ sources are statistically independent. The pdf of $\varepsilon_t$ in (22) is also given by (10), but the $(j, j)$th element of $\Sigma_{t,\,i}$, which is denoted as $(\sigma_{t,\,i}^{(j)})^2$, is described by (20). That is

$$\begin{aligned} \left(\sigma_{t,\,i}^{(j)}\right)^2 = \left(\nu_{i,\,0}^{(j)}\right)^2 &+ \sum_{r=1}^{n_q} \left(\nu_{i,\,r}^{(j)}\right)^2 \left(\varepsilon_{t-r}^{(j)}\right)^2 \\ &+ \sum_{r=1}^{n_p} \left(\psi_{i,\,r}^{(j)}\right)^2 \left(\sigma_{t-r,\,i}^{(j)}\right)^2. \end{aligned} \tag{23}$$

Hereafter, we also simply call this multivariate source model *a finite mixture of GARCH*($n_p$, $n_q$) *process* without further distinction.

### B. Parameter Estimation in a Finite GARCH Model

Using the source model in (22) and comparing with (4), it follows that

$$\mathbf{f}(\mathbf{Y}_{t-1}|\Theta) = \sum_{r=1}^{p} \Lambda_r \mathbf{y}_{t-r}. \tag{24}$$

Consequently, $\mathbf{F}(\mathbf{X}_{t-1}|\Theta)$ in (6) is specified as

$$\begin{aligned} \mathbf{F}(\mathbf{X}_{t-1}|\Theta) &= \mathbf{A}\mathbf{f}(\mathbf{Y}_{t-1}|\Theta) \\ &= \sum_{r=1}^{p} \mathbf{A}\Lambda_r \mathbf{W}\mathbf{x}_{t-r}. \end{aligned} \tag{25}$$

Hence, the OAR residual $\mathbf{z}_t$ in (8) becomes

$$\mathbf{z}_t = \mathbf{x}_t - \sum_{r=1}^{p} \mathbf{A}\mathbf{\Lambda}_r\mathbf{W}\mathbf{x}_{t-r}. \tag{26}$$

To adaptively extract $\mathbf{z}_t$ by (26), we need to estimate $\mathbf{A}$ online as well as other parameters. From (2), we know that $\mathbf{A}$ can be estimated by $\mathbf{W}^{-1}$, which, however, needs extensive computing costs to calculate the inverse of a matrix. Alternatively, we have noticed that each time we estimate $\mathbf{\Theta}$ with $\mathbf{W}$ being fixed, we can learn $\mathbf{\Theta}$ by regarding $\mathbf{A}$ and $\mathbf{W}$ as two constants. As a result, we further let (25) be

$$\begin{aligned}\mathbf{F}(\mathbf{X}_{t-1}|\mathbf{\Theta}) &= \sum_{r=1}^{p} \mathbf{A}\mathbf{\Lambda}_r\mathbf{W}\mathbf{x}_{t-r} = \sum_{r=1}^{p} \mathbf{B}_r\mathbf{x}_{t-r} \\ &= \mathbf{B}\mathbf{X}_{t-p}^{t-1}\end{aligned} \tag{27}$$

where $\mathbf{X}_{t-p}^{t-1} = (\mathbf{x}_{t-1}^T, \mathbf{x}_{t-2}^T, \ldots, \mathbf{x}_{t-p}^T)^T$, and $\mathbf{B} = (\mathbf{B}_1, \mathbf{B}_2, \ldots, \mathbf{B}_p)$ with $\mathbf{B}_r = \mathbf{A}\mathbf{\Lambda}_r\mathbf{W}$. $\mathbf{z}_t$ in (26) then becomes

$$\mathbf{z}_t = \mathbf{x}_t - \mathbf{B}\mathbf{X}_{t-p}^{t-1} \tag{28}$$

and $J_t(\mathbf{\Theta}_1)$ in (15) becomes

$$\begin{aligned}J_t(\mathbf{\Theta}_1) &= \ln p(\mathbf{x}_t|\mathbf{X}_{t-p}^{t-1};\, \mathbf{\Theta}_1) \\ &= \ln\left[\sum_{i=1}^{n} \gamma_i G(\mathbf{x}_t|\mathbf{B}\mathbf{X}_{t-p}^{t-1} + \mathbf{A}\mathbf{m}_i,\, \mathbf{A}\mathbf{\Sigma}_{t,i}\mathbf{A}^T)\right] \\ &= \ln\left[\sum_{i=1}^{n} \gamma_i G(\mathbf{x}_t|\mathbf{B}\mathbf{X}_{t-p}^{t-1} + \tilde{\mathbf{m}}_i,\, \tilde{\mathbf{\Sigma}}_{t,i})\right]\end{aligned} \tag{29}$$

where $\tilde{\mathbf{m}}_i = \mathbf{A}\mathbf{m}_i$, and $\tilde{\mathbf{\Sigma}}_{t,i} = \mathbf{A}\mathbf{\Sigma}_{t,i}\mathbf{A}^T$. We therefore directly learn $\mathbf{B}$, $\tilde{\mathbf{m}}_i$s and $\tilde{\mathbf{\Sigma}}_{t,i}$s to avoid estimating $\mathbf{A}$. Please notice that under the circumstances, $\mathbf{\Theta}_1$ consists of the parameters: {$\mathbf{B}$, $\beta_i$s, $\tilde{\mathbf{m}}_i$s, $\nu_{i,0}^{(j)}$, $\nu_{i,r}^{(j)}$s, and $\psi_{i,r}^{(j)}$s}. Consequently, the term $\partial J_t(\mathbf{\Theta}_1)/\partial\mathbf{\Theta}_1$ in **Step 2** of Table I is explicitly given as follows:

$$\begin{aligned}\frac{\partial J_t(\mathbf{\Theta}_1)}{\partial \mathbf{B}} &= \sum_{i=1}^{n} h_{t,i}\tilde{\mathbf{\Sigma}}_{t,i}^{-1}(\mathbf{z}_t - \tilde{\mathbf{m}}_i)\mathbf{X}_{t-p}^{t-1\,T} \\ \frac{\partial J_t(\mathbf{\Theta}_1)}{\partial \tilde{\mathbf{m}}_i} &= h_{t,i}\tilde{\mathbf{\Sigma}}_{t,i}^{-1}(\mathbf{z}_t - \tilde{\mathbf{m}}_i) \\ \frac{\partial J_t(\mathbf{\Theta}_1)}{\partial \beta_i} &= h_{t,i} - \gamma_i \\ \frac{\partial J_t(\mathbf{\Theta}_1)}{\partial \boldsymbol{\nu}_{i,0}} &= h_{t,i}dg(\mathbf{u}_{t,i}\mathbf{u}_{t,i}^T - \mathbf{\Sigma}_{t,i}^{-1})\boldsymbol{\nu}_{i,0} \\ \frac{\partial J_t(\mathbf{\Theta}_1)}{\partial \boldsymbol{\nu}_{i,r}} &= h_{t,i}dg(\mathbf{u}_{t,i}\mathbf{u}_{t,i}^T - \mathbf{\Sigma}_{t,i}^{-1})dg(\boldsymbol{\varepsilon}_{t-r}\boldsymbol{\varepsilon}_{t-r}^T)\boldsymbol{\nu}_{i,r} \\ \frac{\partial J_t(\mathbf{\Theta}_1)}{\partial \boldsymbol{\psi}_{i,r}} &= h_{t,i}dg[(\mathbf{u}_{t,i}\mathbf{u}_{t,i}^T - \mathbf{\Sigma}_{t,i}^{-1})\mathbf{\Sigma}_{t-r,i}]\boldsymbol{\psi}_{i,r}\end{aligned} \tag{30}$$

with

$$\begin{aligned}h_{t,i} &= \frac{\gamma_i G(\mathbf{x}_t|\mathbf{B}\mathbf{X}_{t-p}^{t-1} + \tilde{\mathbf{m}}_i, \tilde{\mathbf{\Sigma}}_{t,i})}{p(\mathbf{x}_t|\mathbf{X}_{t-p}^{t-1};\, \mathbf{\Theta}_1)} \\ \mathbf{u}_{t,i} &= \mathbf{\Sigma}_{t,i}^{-1}\mathbf{W}(\mathbf{z}_t - \tilde{\mathbf{m}}_i) \\ \boldsymbol{\nu}_{i,0} &= \left[\nu_{i,0}^{(1)}, \nu_{i,0}^{(2)}, \ldots, \nu_{i,0}^{(k)}\right]^T \\ \boldsymbol{\nu}_{i,r} &= \left[\nu_{i,r}^{(1)}, \nu_{i,r}^{(2)}, \ldots, \nu_{i,r}^{(k)}\right]^T, \qquad 1 \le r \le n_q \\ \boldsymbol{\psi}_{i,r} &= \left[\psi_{i,r}^{(1)}, \psi_{i,r}^{(2)}, \ldots, \psi_{i,r}^{(k)}\right]^T, \qquad 1 \le r \le n_p\end{aligned} \tag{31}$$

where $dg(\mathbf{X})$ denotes a diagonal matrix whose $(j,\, j)$th element is either the $(j,\, j)$th one of the square matrix $\mathbf{X}$, or the $j$th one as $\mathbf{X}$ is a vector, and $\delta_{ir}$ is the Kronecker delta function with

$$\delta_{ir} = \begin{cases} 1, & \text{if } i = r \\ 0, & \text{otherwise.} \end{cases} \tag{32}$$

We list the detailed derivation of (30) in the Appendix. Here, two points should be noted. One point is that we just need to estimate $\tilde{\mathbf{\Sigma}}_{t,i}^{-1}$ by $\mathbf{W}^T\mathbf{\Sigma}_{t,i}^{-1}\mathbf{W}$, where $\mathbf{\Sigma}_{t,i}^{-1}$ can be easily calculated because $\mathbf{\Sigma}_{t,i}$ is a diagonal matrix as described in (23). The other point is that the source process in (3) becomes a stationary process as $\{\boldsymbol{\nu}_{i,r}\}_{r=1}^{n_q}$ and $\{\boldsymbol{\psi}_{i,r}\}_{r=1}^{n_p}$ are all zero. In this case, $\tilde{\mathbf{\Sigma}}_{t,i}$ degenerates to $\tilde{\mathbf{\Sigma}}_i = \mathbf{W}^T\mathbf{\Sigma}_i^{-1}\mathbf{W}$ that is irrelevant to the time, and (11) actually becomes a classical finite mixture of Gaussian densities. Consequently, $\partial J_t(\mathbf{\Theta}_1)/\partial\mathbf{\Theta}_1$ in (30) can be further simplified as

$$\begin{aligned}\frac{\partial J_t(\mathbf{\Theta}_1)}{\partial \mathbf{B}} &= \sum_{i=1}^{n} h_{t,i}\mathbf{\Sigma}_i^{-1}(\mathbf{z}_t - \tilde{\mathbf{m}}_i)\mathbf{X}_{t-p}^{t-1\,T} \\ \frac{\partial J_t(\mathbf{\Theta}_1)}{\partial \tilde{\mathbf{m}}_i} &= h_{t,i}\mathbf{\Sigma}_i^{-1}(\mathbf{z}_t - \tilde{\mathbf{m}}_i) \\ \frac{\partial J_t(\mathbf{\Theta}_1)}{\partial \tilde{\mathbf{\Sigma}}_i} &= h_{t,i}[(\mathbf{z}_t - \tilde{\mathbf{m}}_i)(\mathbf{z}_t - \tilde{\mathbf{m}}_i) - \tilde{\mathbf{\Sigma}}_i] \\ \frac{\partial J_t(\mathbf{\Theta}_1)}{\partial \beta_i} &= h_{t,i} - \gamma_i.\end{aligned} \tag{33}$$

If we further let

$$\boldsymbol{\xi}_t = \mathbf{\Sigma}_{\mathbf{z}}^{-(1/2)}\mathbf{x}_t \tag{34}$$

from (28), we then have

$$\boldsymbol{\xi}_t = \mathbf{\Sigma}_{\mathbf{z}}^{-(1/2)}\mathbf{x}_t \tag{35}$$

$$= \mathbf{\Sigma}_{\mathbf{z}}^{-(1/2)}\mathbf{B}\mathbf{X}_{t-p}^{t-1} + \mathbf{\Sigma}_{\mathbf{z}}^{-(1/2)}\mathbf{z}_t \tag{36}$$

$$= \mathbf{C}\mathbf{X}_{t-p}^{t-1} + \mathbf{v}_t \tag{37}$$

where $\mathbf{\Sigma}_{\mathbf{z}}^{-(1/2)}$ is the square root of the inverse of $\mathbf{z}_t$s covariance, $\mathbf{C} = \mathbf{\Sigma}_{\mathbf{z}}^{-(1/2)}\mathbf{B}$, and $\mathbf{v}_t = \mathbf{\Sigma}_{\mathbf{z}}^{-(1/2)}\mathbf{z}_t$. In this way, the components of $\mathbf{v}_t$s are decorrelated with unit variance. Since each component of $\mathbf{v}_t$ is a linear combination of $\varepsilon_t^{(j)}$s, it can therefore be approximately regarded as a Gaussian variable by

the law of large number when the dimension $k$ of $\varepsilon_t$ is sufficiently large. Consequently, maximizing log-likelihood function of the observed signals in (13) is simplified to minimize the cost function

$$Q(\mathbf{B}, \mathbf{\Sigma_z}) = \frac{1}{N}\sum_{t=1}^{N}(\boldsymbol{\xi}_t - \mathbf{C}\mathbf{X}_{t-p}^{t-1})^T(\boldsymbol{\xi}_t - \mathbf{C}\mathbf{X}_{t-p}^{t-1}). \quad (38)$$

In this case, the extraction of OAR residuals is determined by the parameter $\mathbf{B}$ and $\mathbf{\Sigma_z}$ only, which can be estimated by a generalized LMS method. That is, at time step $t$, $\mathbf{B}$ and $\mathbf{\Sigma_z}$ are adaptively updated by

$$\begin{aligned}\mathbf{B}^{\text{new}} &= \mathbf{B}^{\text{old}} + \eta\mathbf{\Sigma_z}^{-1}\mathbf{z}_t\mathbf{X}_{t-p}^{t-1^T}\\ \mathbf{\Sigma_z}^{\text{new}} &= (1-\eta)\mathbf{\Sigma_z}^{\text{old}} + \eta\mathbf{z}_t\mathbf{z}_t^T. \end{aligned} \quad (39)$$

In (39), we only need to use $\mathbf{\Sigma_z}^{-1}$ in updating $\mathbf{B}$. To save computing cost and calculation stability, we can directly estimate $\mathbf{\Sigma_z}^{-1}$ instead of $\mathbf{\Sigma_z}$ with

$$\mathbf{\Sigma_z}^{-1^{\text{new}}} = \frac{\mathbf{\Sigma_z}^{-1^{\text{old}}}}{1-\eta}\left[\mathbf{I} - \frac{\eta\mathbf{z}_t\mathbf{z}_t^T\mathbf{\Sigma_z}^{-1^{\text{old}}}}{1-\eta+\eta\mathbf{z}_t\mathbf{\Sigma_z}^{-1^{\text{old}}}\mathbf{z}_t}\right] \quad (40)$$

where $\mathbf{I}$ is the $k \times k$ identity matrix.

## IV. LPM Algorithm

We choose the LPM algorithm [33], [32] among a variety of existing nontemporal ICA algorithms because it has the advantage of being capable of separating any combination of sub-Gaussian and super-Gaussian signals. Here, we briefly introduce the LPM concept and main implement steps only. See [32] and [33] for other details.

Suppose $k$ independent sources $y_t^{(1)}, y_t^{(2)}, \ldots, y_t^{(k)}$ in (1) are each independently and identically distributed. We therefore omit the time subscript in the remaining part of this section. LPM algorithm is to find out an appropriate $\mathbf{W}$ in (2) by minimizing the mutual information between $\hat{\mathbf{y}}$ and its components

$$\begin{aligned}L(\mathbf{W}) &= \int p(\hat{\mathbf{y}})\ln\frac{p(\hat{\mathbf{y}})}{\prod_{j=1}^{k} g_j\left(\hat{y}^{(j)}\right)}\,d\hat{\mathbf{y}}\\ &= \int p(\mathbf{x})\ln\frac{p(\mathbf{x})/|\det(\mathbf{W})|}{\prod_{j=1}^{k} g_j\left(\hat{y}^{(j)}\right)}\,d\mathbf{x}\\ &= -\sum_{j=1}^{k}\int p(\mathbf{x})\ln g_j\left(\hat{y}^{(j)}\right)d\mathbf{x} - \ln|\det(\mathbf{W})| - H(\mathbf{x})\end{aligned} \quad (41)$$

where $H(\mathbf{x})$ is the entropy of $\mathbf{x}$ irrelevant to $\mathbf{W}$, and $g_j(\hat{y}^{(j)})$ is an estimate of the marginal density of $\hat{y}^{(j)}$. LPM lets $g_j(\hat{y}^{(j)})$ be a mixture of densities

$$g_j\left(\hat{y}^{(j)}\right) = \sum_{i=1}^{n_j}\alpha_i^{(j)}q\left(\kappa_i^{(j)}\right) \quad (42)$$

TABLE II
Detailed Implementation Steps of LPM Algorithm at Each Time Step $t$

| | |
|---|---|
| **Step 1** | Given $\mathbf{W}^{\text{old}}$, calculate $\hat{\mathbf{y}} = \mathbf{W}^{\text{old}}\mathbf{x}$. |
| **Step 2** | Update $\mathbf{W}$ by<br>$\mathbf{W}^{\text{new}} = \mathbf{W}^{\text{old}} + \eta\Delta\mathbf{W}$ with<br>$\Delta\mathbf{W} = (\mathbf{I} + \phi(\hat{\mathbf{y}})\hat{\mathbf{y}}^T)\mathbf{W}^{\text{old}}$<br>where $\phi(\hat{\mathbf{y}}) = [\phi_1(\hat{y}^{(1)}), \phi(\hat{y}^{(2)}), \ldots, \phi_k(\hat{y}^{(k)})]^T$,<br>and $\phi_j(\hat{y}^{(j)}) = \frac{\partial \ln g_j(\hat{y}^{(j)})}{\partial \hat{y}^{(j)}}$ |
| **Step 3** | Update the parameters $\zeta_i^{(j)}, b_i^{(j)}, a_i^{(j)}$<br>with $i = 1, \ldots, n_j, 1 \le j \le k$ by:<br>$\zeta_i^{(j)^{\text{new}}} = \zeta_i^{(j)^{\text{old}}} + \eta\Delta\zeta_i^{(j)}$,<br>$b_i^{(j)^{\text{new}}} = b_i^{(j)^{\text{old}}} + \eta\Delta b_i^{(j)}$,<br>$a_i^{(j)^{\text{new}}} = a_i^{(j)^{\text{old}}} + \eta\Delta a_i^{(j)}$,<br>with<br>$\Delta\zeta_i^{(j)} = \frac{\sum_{m=1}^{n_j} b_m^{(j)}h'(\kappa_m^{(j)})\alpha_m^{(j)}(\delta_m^{(i)} - \alpha_i^{(j)})}{g_j(\hat{y}^{(j)})}$<br>$\Delta b_i^{(j)} = \frac{\alpha_i^{(j)}}{g_j(\hat{y}^{(j)})}[h'(\kappa_i^{(j)}) + h''(\kappa_i^{(j)})\kappa_i^{(j)}]$<br>$\Delta a_i^{(j)} = -\frac{1}{g_j(\hat{y}^{(j)})}\alpha_i^{(j)}(b_i^{(j)})^2h''(\kappa_i^{(j)})$ |

with

$$\begin{aligned}\kappa_i^{(j)} &= b_i^{(j)}\left(\hat{y}^{(j)} - a_i^{(j)}\right)\\ \sum_{i=1}^{n_y}\alpha_i^{(j)} &= 1, \quad \alpha_i^{(j)} = \frac{\exp\left(\zeta_i^{(j)}\right)}{\sum_{m=1}^{n_y}\exp\left(\zeta_m^{(j)}\right)}\end{aligned} \quad (43)$$

where $q(.)$ is a density function, $n_j$ is the number of components in the mixture, $\alpha_i^{(j)}$ is the weight of the component, $b_i^{(j)}$ controls the variant of the $j$th density, and $a_i^{(j)}$ is the bias or location of the center of the $j$th density. In particular, if $q(\kappa_i^{(j)}) = b_i^{(j)}h'(\kappa_i^{(j)})$ with

$$h'\left(\kappa_i^{(j)}\right) = \frac{\exp\left(-\kappa_i^{(j)}\right)}{\left[1+\exp\left(-\kappa_i^{(j)}\right)\right]^2}$$

we have

$$\begin{aligned}\phi_j\left(\hat{y}^{(j)}\right) &= \frac{\partial\ln g_j\left(\hat{y}^{(j)}\right)}{\partial\hat{y}^{(j)}}\\ &= \frac{1}{g_j\left(\hat{y}_t^{(j)}\right)}\sum_{i=1}^{n_j}\alpha_i^{(j)}b_i^{(j)}q'\left(\kappa_i^{(j)}\right)\end{aligned} \quad (44)$$

with

$$\begin{aligned}q'\left(\kappa_i^{(j)}\right) &= h''\left(\kappa_i^{(j)}\right)b_i^{(j)}\\ h''\left(\kappa_i^{(j)}\right) &= \frac{\exp\left(-2\kappa_i^{(j)}\right) - \exp\left(-\kappa_i^{(j)}\right)}{\left[1+\exp\left(-\kappa_i^{(j)}\right)\right]^3}.\end{aligned} \quad (45)$$

Given a series of inputs $\mathbf{x}$s, the LPM algorithm adaptively tunes $\mathbf{W}$ as well as $\{\gamma_i^{(j)}, b_i^{(j)}, a_i^{(j)}\}_{i=1}^{n_j}$, $j = 1, 2, \ldots, k$ toward minimizing the cost function in (41). Consequently, at

each time step $t$, LPM performs three steps, as summarized in Table II.

It should be noted that when the LPM algorithm is used in **Step 2** of Section II for (9), the inputs of LPM are actually $\mathbf{z}_t$s but not $\mathbf{x}_t$s.

## V. Experiments

We conducted two experiments to show the performance of our proposed two-step approach, where the LPM was used to implement its **Step 2**. In addition, we showed the performance of individual LPM algorithm with the observations as its inputs for comparison. In both experiments, we simply let the learning rate $\eta = 0.0001$ and arbitrarily choose the number of Gaussian density mixtures $n = n_j = 9$ with $j = 1, 2, \ldots, k$.

### A. Experiment 1

We used three GARCH(1,1) source signals generated, respectively, by

$$\begin{aligned} y_t^{(1)} &= 0.8 y_{t-1}^{(1)} + \varepsilon_t^{(1)} \\ y_t^{(2)} &= -0.7 y_{t-1}^{(2)} + \varepsilon_t^{(2)} \\ y_t^{(3)} &= 0.9 y_{t-1}^{(3)} + \varepsilon_t^{(3)} \end{aligned} \tag{46}$$

where $\varepsilon_t^{(1)}$, $\varepsilon_t^{(2)}$, and $\varepsilon_t^{(3)}$ were all zero-mean uniformly distributed with variance

$$\begin{aligned} \left(\sigma_t^{(1)}\right)^2 &= 0.1 + 0.2\left(\varepsilon_{t-1}^{(1)}\right)^2 + 0.5\left(\sigma_{t-1}^{(1)}\right)^2 \\ \left(\sigma_t^{(2)}\right)^2 &= 0.2 + 0.1\left(\varepsilon_{t-1}^{(2)}\right)^2 + 0.3\left(\sigma_{t-1}^{(2)}\right)^2 \\ \left(\sigma_t^{(3)}\right)^2 &= 0.1 + 0.1\left(\varepsilon_{t-1}^{(3)}\right)^2 + 0.6\left(\sigma_{t-1}^{(3)}\right)^2. \end{aligned} \tag{47}$$

In the experiments, we set the true mixing matrix

$$\mathbf{A} = \begin{pmatrix} 1.5 & 0.5 & 0.7 \\ 0.7 & -1 & 0.5 \\ 1.2 & 0.8 & 2.0 \end{pmatrix} \tag{48}$$

while randomly assigning the initial values of $\mathbf{B}$, $\mathbf{W}$, and other estimated parameters. We measured an algorithm performance by signal-to-noise ratio (SNR), defined by

$$\text{SNR}\left(y^{(j)}, \hat{y}^{(j)}\right) = 10\log_{10}\frac{\sigma_{y^{(j)}}^2}{\text{MSE}\left(y^{(j)}, \hat{y}^{(j)}\right)} \tag{49}$$

where $1 \le j \le k$, $\sigma_{y^{(j)}}^2$ is the variance of source signal $y^{(j)}$, and $\text{MSE}(y^{(j)}, \hat{y}^{(j)})$ is the mean square error between source signal $y^{(j)}$ and its recovered signal $\hat{y}^{(j)}$. Since $\sigma_{y^{(j)}}^2$ is irrelevant to the algorithm performance, we can further ignore it and use a simplified SNR with

$$\text{SNR}\left(y^{(j)}, \hat{y}^{(j)}\right) = -10\log_{10}\text{MSE}\left(y^{(j)}, \hat{y}^{(j)}\right). \tag{50}$$

Since the observations were sequentially observed without repeat, we calculated the MSE values on line once every 10 000 observation points and normalized the variances of both the sources and the recovered signals to be 1 to avoid the scaling problem in SNR calculations.

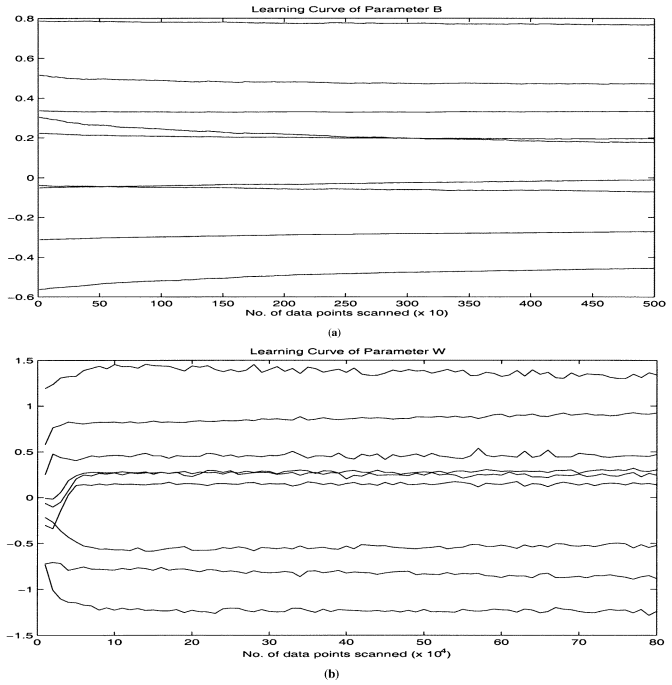



Fig. 2. (a) Learning curve of the parameter $\mathbf{B}$. (b) Curve of the demixing matrix $\mathbf{W}$.

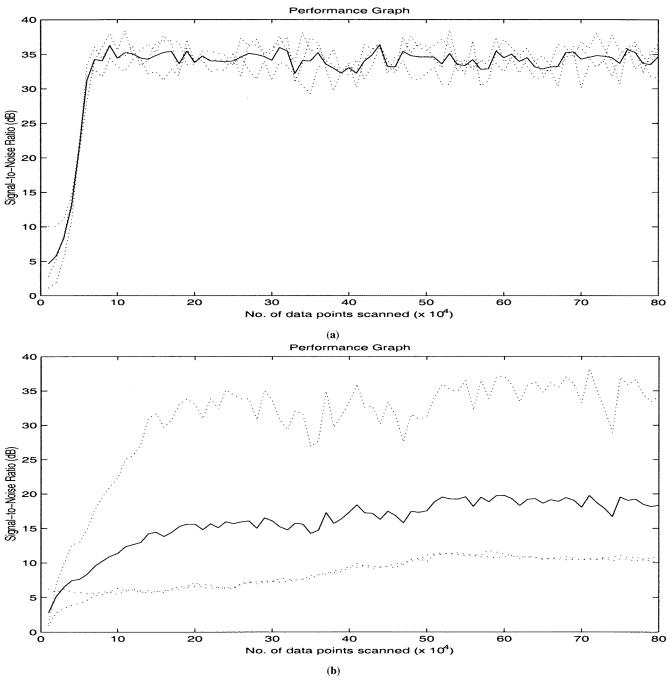



Fig. 3. (a) Performance of the proposed approach in Experiment 1. (b) That of the individual LPM algorithm. In (a) and (b), the dotted curve represents the SNR of each individual recovered signal, and the solid curve is their average SNR.

Fig. 2(a) and (b) show the learning curves of two major parameters $\mathbf{B}$ and $\mathbf{W}$, respectively. It can be seen that $\mathbf{B}$ quickly tends to converge after scanning about 5000 data points, although the learning rate $\eta$ is very small. In contrast, $\mathbf{W}$ converges after scanning about 80 000 data points. The reason that $\mathbf{W}$ converges slower than $\mathbf{B}$ is that $\mathbf{W}$ is tuned based on the extracted observation residuals, which, however, much depend

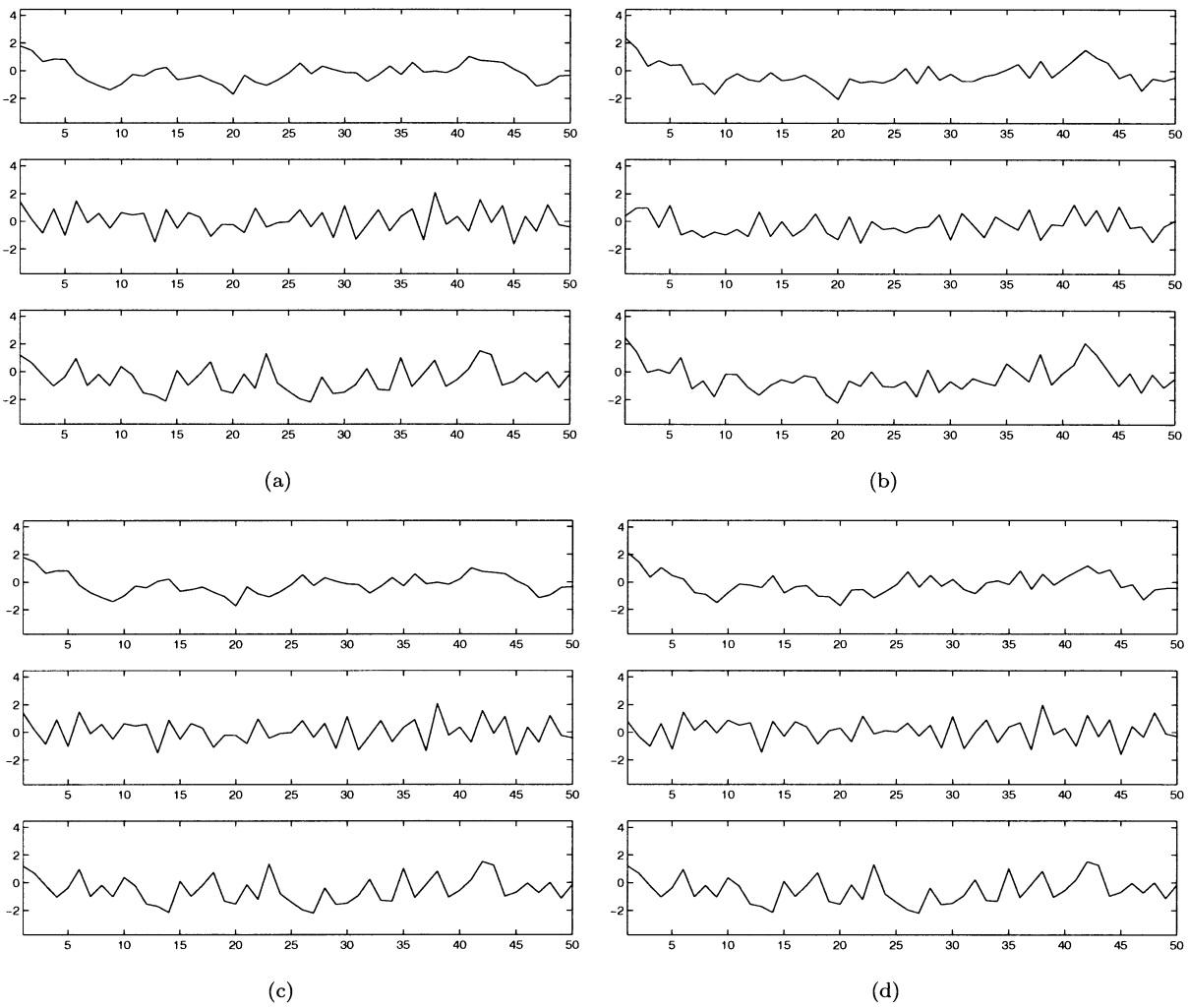



Fig. 4. (a), (b) Slide window of three source signals and their mixed signals, respectively, in Experiment 1. (c) and (d) Recovered signals obtained from the proposed approach and the individual LPM algorithm, respectively.

on the value of $\mathbf{B}$. After convergence, a snapshot value of $\mathbf{W}$ learned by the proposed approach is

$$\mathbf{W} = \begin{pmatrix} 1.2296 & 0.2439 & -0.5218 \\ 0.4698 & -1.2394 & 0.1388 \\ -0.8828 & 0.3018 & 0.9256 \end{pmatrix} \tag{51}$$

while $\mathbf{W}$ by the individual LPM is

$$\mathbf{W} = \begin{pmatrix} 0.8188 & -0.1074 & -0.2554 \\ 0.0739 & -0.7627 & 0.1526 \\ -0.6309 & 0.2179 & 0.6650 \end{pmatrix}. \tag{52}$$

Since a correct solution of $\mathbf{W}$ should satisfy that $\mathbf{W} \times \mathbf{A}$ is equal to $\mathbf{P} \times \mathbf{D}$, as described in (2), we therefore further investigated the values of $\mathbf{W} \times \mathbf{A}$. We found that the proposed approach gives

$$\mathbf{W} \times \mathbf{A} = \begin{pmatrix} \mathbf{1.5539} & 0.0085 & 0.0160 \\ 0.0037 & \mathbf{1.5854} & -0.0132 \\ -0.0023 & -0.0028 & \mathbf{1.3841} \end{pmatrix} \tag{53}$$

while the individual LPM gives

$$\mathbf{W} \times \mathbf{A} = \begin{pmatrix} \mathbf{0.8465} & \mathbf{0.3125} & 0.0087 \\ \mathbf{-0.2400} & \mathbf{0.9217} & -0.0245 \\ 0.0042 & -0.0014 & \mathbf{0.9972} \end{pmatrix}. \tag{54}$$

It can be seen that the former has successfully converged to a correct solution, but the latter is not. Actually, a snapshot of

$$\text{average SNR} = \frac{1}{k} \sum_{j=1}^{k} \text{SNR}\left(y^{(j)}, \hat{y}^{(j)}\right)$$

for the proposed approach is 34.69 dB, whereas the individual LPM is 15.74 dB only. Fig. 3 shows the learning curves of their performance, and Fig. 4 presents a slide window to show the separation results by these two algorithms. We found that the proposed approach has successfully separated the temporal sources, but the LPM has not.

### *B. Experiment 2*

We further investigated the performance of these two algorithms on real-world signals. We let the sources be three music

sound signals recorded at a 22-kHz sampling rate. After sequentially scanning 800 000 observation points, a snapshot value of $\mathbf{W} \times \mathbf{A}$ by the proposed approach is

$$\mathbf{W} \times \mathbf{A} = \begin{pmatrix} \mathbf{27.4976} & 0.1043 & 0.2997 \\ -0.7287 & \mathbf{10.9048} & 0.0188 \\ 0.3337 & 0.0292 & \mathbf{2.8709} \end{pmatrix} \tag{55}$$

while that by the individual LPM is

$$\mathbf{W} \times \mathbf{A} = \begin{pmatrix} \mathbf{4.1800} & -0.0035 & \mathbf{1.6341} \\ -0.2535 & \mathbf{4.6206} & -0.0689 \\ \mathbf{-1.2230} & 0.0855 & \mathbf{1.3361} \end{pmatrix} \tag{56}$$

where we found that the proposed approach has given a correct solution of $\mathbf{W}$, but the individual LPM does not. Actually, (56) has indicated that the latter cannot successfully separate the first audio signal and the third one. To clearly show this, we further chose the first audio source signal as an example to show its frequency spectrum as well as the separated results in Fig. 5, from which we can see that the separated signals given by the proposed approach has a similar frequency spectrum with the original one but that from the LPM is not. Fig. 6 gives out the separation results of these two algorithms on music sounds, respectively, and Fig. 7 shows their comparative performance graphs. It can be seen that the proposed approach again outperforms the LPM in this real-data case study.

## VI. Discussion

The proposed two-step approach as described in the previous sections supposes that the residuals $\varepsilon_t^{(j)}$s of the source process are generally non-Gaussian distributed with at most one being Gaussian. However, when two or more of $\varepsilon_t^{(j)}$s are or near Gaussian distributed, this approach will be not work any more because the independence property among a set of independently and identically distributed Gaussian variables is invariant in a rotation transformation (i.e., multiply an orthogonal matrix). In view of this, our recent paper [7] has further studied dual AR modeling on the case that $\boldsymbol{\varepsilon}_t$ is multivariate Gaussian distributed. Although we have presented a learning algorithm in [7] and further analyzed it in [8], the situation that some of $\varepsilon_t^{(1)}, \varepsilon_t^{(2)}, \ldots, \varepsilon_t^{(k)}$ are Gaussian distributed but not all has yet to be investigated.

Furthermore, this two-step approach uses the gradient-ascent learning rule to adjust the parameters, resulting in simple implementation but linear parameter convergent rate. Although the latter can be further improved by using second-order statistics information in updating the parameters at each time step, it needs the large amount of computing cost in calculating Hessian matrices, especially when the parameter dimension is large. Hence, one better way is to find an appropriate learning rule with convergent rate between 1 and 2 such that the algorithm is qualitatively similar to quadratic convergence, but only computes first derivative. We leave it for future studies.

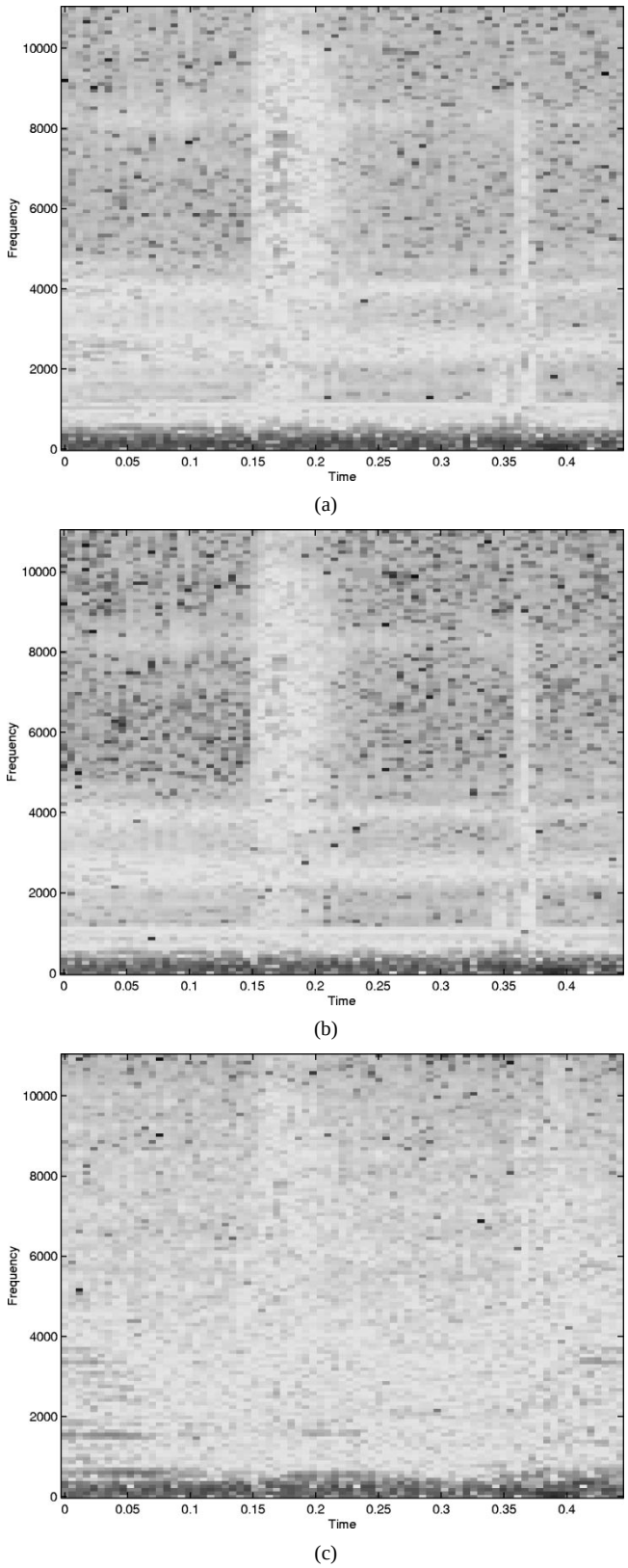



Fig. 5. (a) Frequency spectrum of an audio source signal. (b) Frequency spectrum of the corresponding recovered signal via our proposed approach. (c) Recovered signal from the individual LPM algorithm.

## VII. Conclusion

We have systematically presented a general two-step approach for temporal signal separation. This approach adaptively extracts the OAR residuals with their parameters learned by a ML algorithm, together with learning the demixing matrix on the OAR residuals by a nontemporal ICA algorithm. Particularly, we have implemented the two-step approach in

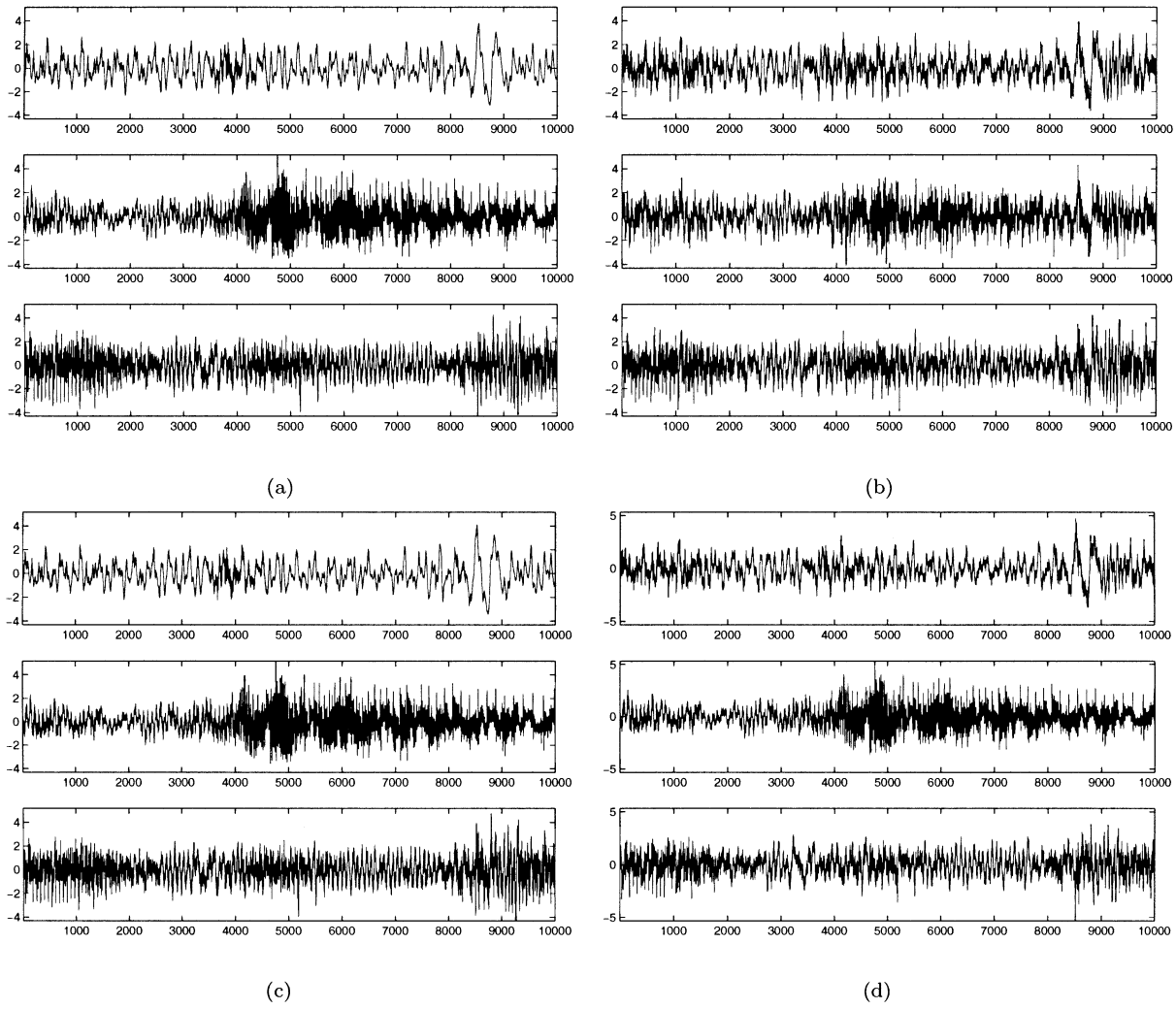



Fig. 6. (a) and (b) Slide window of three source signals and their mixed signals respectively in Experiment 2. (c) Recovered signals obtained from the proposed approach. (d) That from the individual LPM algorithm.

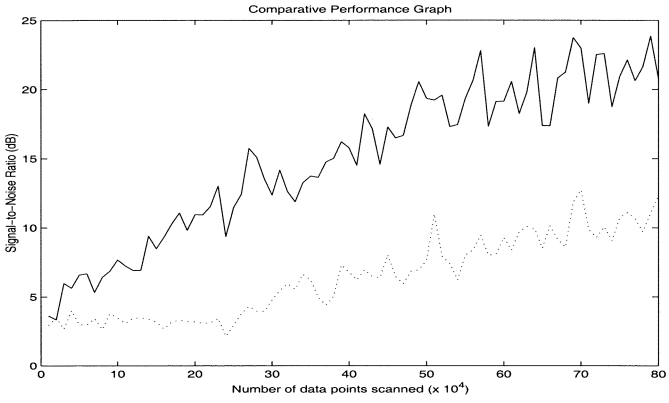



Fig. 7. Average SNR curves of our proposed approach (solid line) and the individual LPM (dotted line) on music sounds, respectively.

detail by modeling the sources as a finite mixture of GARCH process. We have therefore obtained an adaptive ML algorithm, which includes generalized LMS method as a special case. The experiments have shown that this new approach outperforms a nontemporal ICA algorithm in temporal source separation.

## APPENDIX
## DERIVATION OF $\partial J_t(\mathbf{\Theta}_1)/\partial \mathbf{\Theta}_1$ IN EQ. (30)

In the following, we denote $d_{\boldsymbol{\theta}} f$ as the differential of function $f$ with respect to $\boldsymbol{\theta}$. We will give the detailed derivations of (30) as follows.

From (29), we have

$$d_{\mathbf{B}} J_t(\mathbf{\Theta}_1) = \frac{1}{p(\mathbf{x}_t|\mathbf{X}_{t-p}^{t-1};\, \mathbf{\Theta}_1)}\, d_{\mathbf{B}} \sum_{i=1}^{n} \gamma_i G_{t,i} \tag{57}$$

$$d_{\tilde{\mathbf{m}}_i} J_t(\mathbf{\Theta}_1) = \frac{\gamma_i}{p(\mathbf{x}_t|\mathbf{X}_{t-p}^{t-1};\, \mathbf{\Theta}_1)}\, d_{\tilde{\mathbf{m}}_i} G_{t,i} \tag{58}$$

$$d_{\boldsymbol{\nu}_{i,0}} J_t(\mathbf{\Theta}_1) = \frac{\gamma_i}{p(\mathbf{x}_t|\mathbf{X}_{t-p}^{t-1};\, \mathbf{\Theta}_1)}\, d_{\boldsymbol{\nu}_{i,0}} G_{t,i} \tag{59}$$

$$d_{\boldsymbol{\nu}_{i,r}} J_t(\mathbf{\Theta}_1) = \frac{\gamma_i}{p(\mathbf{x}_t|\mathbf{X}_{t-p}^{t-1};\, \mathbf{\Theta}_1)}\, d_{\boldsymbol{\nu}_{i,r}} G_{t,i} \tag{60}$$

$$d_{\boldsymbol{\psi}_{i,r}} J_t(\mathbf{\Theta}_1) = \frac{\gamma_i}{p(\mathbf{x}_t|\mathbf{X}_{t-p}^{t-1};\, \mathbf{\Theta}_1)}\, d_{\boldsymbol{\psi}_{i,r}} G_{t,i} \tag{61}$$

$$d_{\beta_i} J_t(\boldsymbol{\Theta}_1) = \frac{\sum_{r=1}^{n} G_{t,r} d_{\beta_i} \gamma_r}{p(\mathbf{x}_t|\mathbf{X}_{t-p}^{t-1};\, \boldsymbol{\Theta}_1)} \tag{62}$$

where $G_{t,i} = G(\mathbf{x}_t|\mathbf{B}\mathbf{X}_{t-p}^{t-1} + \tilde{\mathbf{m}}_i,\, \tilde{\boldsymbol{\Sigma}}_{t,i})$.

Since

$$\begin{aligned} d_{\mathbf{B}} G_{t,i} &= G_{t,i} d_{\mathbf{B}} \left[ -\tfrac{1}{2} (\mathbf{z}_t - \tilde{\mathbf{m}}_i)^T \tilde{\boldsymbol{\Sigma}}_{t,i}^{-1} (\mathbf{z}_t - \tilde{\mathbf{m}}_i) \right] \\ &= \tfrac{1}{2} G_{t,i} \mathrm{Tr}[(d\mathbf{B}\mathbf{X}_{t-p}^{t-1})^T \tilde{\boldsymbol{\Sigma}}_{t,i}^{-1} \overline{\mathbf{z}}_{t,i} + \overline{\mathbf{z}}_{t,i}^T \tilde{\boldsymbol{\Sigma}}_{t,i}^{-1} d\mathbf{B}\mathbf{X}_{t-p}^{t-1}] \end{aligned} \tag{63}$$

with $\overline{\mathbf{z}}_{t,i} = \mathbf{z}_t - \tilde{\mathbf{m}}_i$, by using the *trace* property that $\mathrm{Tr}(\mathbf{CD}) = \mathrm{Tr}(\mathbf{DC})$ if and only if both of $\mathbf{CD}$ and $\mathbf{DC}$ are square matrix, we then further simplify (63) to

$$\begin{aligned} d_{\mathbf{B}} G_{t,i} &= G_{t,i}\, \mathrm{Tr}[\overline{\mathbf{z}}_{t,i}^T \tilde{\boldsymbol{\Sigma}}_{t,i}^{-1}\, d\mathbf{B}\mathbf{X}_{t-p}^{t-1}] \\ &= G_{t,i}\, \mathrm{Tr}[\mathbf{X}_{t-p}^{t-1} \overline{\mathbf{z}}_{t,i}^T \tilde{\boldsymbol{\Sigma}}_{t,i}^{-1}\, d\mathbf{B}]. \end{aligned} \tag{64}$$

Similarly, we can also obtain

$$\begin{aligned} & d_{\tilde{\mathbf{m}}_i} G_{t,i} \\ &= \tfrac{1}{2} G_{t,i} \mathrm{Tr}[d\tilde{\mathbf{m}}_i^T \tilde{\boldsymbol{\Sigma}}_{t,i}^{-1} \overline{\mathbf{z}}_{t,i} + \overline{\mathbf{z}}_{t,i}^T \tilde{\boldsymbol{\Sigma}}_{t,i}^{-1} d\tilde{\mathbf{m}}_i] \\ &= G_{t,i} \mathrm{Tr}[\overline{\mathbf{z}}_{t,i}^T \tilde{\boldsymbol{\Sigma}}_{t,i}^{-1} d\tilde{\mathbf{m}}_i] \end{aligned} \tag{65}$$

$$\begin{aligned} & d_{\boldsymbol{\nu}_{i,\iota}} G_{t,i} \\ &= G_{t,i} d_{\boldsymbol{\nu}_{i,\iota}} \left[ -\tfrac{1}{2} \ln |\tilde{\boldsymbol{\Sigma}}_{t,i}| - \tfrac{1}{2} \overline{\mathbf{z}}_{t,i}^T \tilde{\boldsymbol{\Sigma}}_{t,i}^{-1} \overline{\mathbf{z}}_{t,i} \right] \\ &= -\tfrac{1}{2} G_{t,i} d_{\boldsymbol{\nu}_{i,\iota}} [\ln |\tilde{\boldsymbol{\Sigma}}_{t,i}| + \overline{\mathbf{z}}_{t,i}^T \tilde{\boldsymbol{\Sigma}}_{t,i}^{-1} \overline{\mathbf{z}}_{t,i}] \\ &= -\tfrac{1}{2} G_{t,i} [\mathrm{Tr}(\tilde{\boldsymbol{\Sigma}}_{t,i}^{-1} d_{\boldsymbol{\nu}_{i,\iota}} \tilde{\boldsymbol{\Sigma}}_{t,i}) \\ &\qquad - \overline{\mathbf{z}}_{t,i}^T \tilde{\boldsymbol{\Sigma}}_{t,i}^{-1} d_{\boldsymbol{\nu}_{i,\iota}} \tilde{\boldsymbol{\Sigma}}_{t,i} \tilde{\boldsymbol{\Sigma}}_{t,i}^{-1} \overline{\mathbf{z}}_{t,i}] \\ &= -\tfrac{1}{2} G_{t,i} \mathrm{Tr}\{[\tilde{\boldsymbol{\Sigma}}_{t,i}^{-1} - \tilde{\boldsymbol{\Sigma}}_{t,i}^{-1} \overline{\mathbf{z}}_{t,i} \overline{\mathbf{z}}_{t,i}^T \tilde{\boldsymbol{\Sigma}}_{t,i}^{-1}] d_{\boldsymbol{\nu}_{i,\iota}} \tilde{\boldsymbol{\Sigma}}_{t,i}\} \end{aligned} \tag{66}$$

$$\begin{aligned} & d_{\boldsymbol{\psi}_{i,r}} G_{t,i} \\ &= -\tfrac{1}{2} G_{t,i} \mathrm{Tr}\{[\tilde{\boldsymbol{\Sigma}}_{t,i}^{-1} - \tilde{\boldsymbol{\Sigma}}_{t,i}^{-1} \overline{\mathbf{z}}_{t,i} \overline{\mathbf{z}}_{t,i}^T \tilde{\boldsymbol{\Sigma}}_{t,i}^{-1}] d_{\boldsymbol{\psi}_{i,r}} \tilde{\boldsymbol{\Sigma}}_{t,i}\} \end{aligned} \tag{67}$$

where $\iota = 0, 1, 2, \ldots, n_q$, and $r = 1, 2, \ldots, n_p$.

From (23), we know that

$$\begin{aligned} d_{\boldsymbol{\nu}_{i,0}} \tilde{\boldsymbol{\Sigma}}_{t,i} &= \mathbf{A} d_{\boldsymbol{\nu}_{i,0}} \boldsymbol{\Sigma}_{t,i} \mathbf{A}^T \\ &= 2\mathbf{A}\boldsymbol{\Omega}_{i,0}\mathbf{A}^T \end{aligned} \tag{68}$$

$$\begin{aligned} d_{\boldsymbol{\nu}_{i,r}} \tilde{\boldsymbol{\Sigma}}_{t,i} &= \mathbf{A} d_{\boldsymbol{\nu}_{i,r}} \boldsymbol{\Sigma}_{t,i} \mathbf{A}^T \\ &= \mathbf{A} dg[d\boldsymbol{\nu}_{i,r} \boldsymbol{\nu}_{i,r}^T + \boldsymbol{\nu}_{i,r} d\boldsymbol{\nu}_{i,r}^T] dg(\boldsymbol{\varepsilon}_{t-r} \boldsymbol{\varepsilon}_{t-r}^T) \mathbf{A}^T \\ &= 2\mathbf{A}\boldsymbol{\Gamma}_{i,r}\mathbf{A}^T \end{aligned} \tag{69}$$

$$\begin{aligned} d_{\boldsymbol{\psi}_{i,r}} \tilde{\boldsymbol{\Sigma}}_{t,i} &= \mathbf{A} d_{\boldsymbol{\psi}_{i,r}} \boldsymbol{\Sigma}_{t,i} \mathbf{A}^T \\ &= 2\mathbf{A}\boldsymbol{\Upsilon}_{i,r}\boldsymbol{\Sigma}_{t-r,i}\mathbf{A}^T \end{aligned} \tag{70}$$

with

$$\begin{aligned} \boldsymbol{\nu}_{i,r} &= \left[\nu_{i,r}^{(1)}, \nu_{i,r}^{(2)}, \ldots, \nu_{i,r}^{(k)}\right]^T, \qquad 1 \le r \le n_q \\ \boldsymbol{\psi}_{i,r} &= \left[\psi_{i,r}^{(1)}, \psi_{i,r}^{(2)}, \ldots, \psi_{i,r}^{(k)}\right]^T, \qquad 1 \le r \le n_p \end{aligned} \tag{71}$$

where $\boldsymbol{\Omega}_{i,r} = dg(d\boldsymbol{\nu}_{i,r}\boldsymbol{\nu}_{i,r}^T)$, $\boldsymbol{\Upsilon}_{i,r} = dg(d\boldsymbol{\psi}_{i,r}\boldsymbol{\psi}_{i,r}^T)$, and $\boldsymbol{\Gamma}_{i,r} = \boldsymbol{\Omega}_{i,r} dg(\boldsymbol{\varepsilon}_{t-r}\boldsymbol{\varepsilon}_{t-r}^T)$. Putting (68)– (70) into (66) and (67), we then have

$$d_{\boldsymbol{\nu}_{i,0}} G_{t,i} = -G_{t,i} \mathrm{Tr}\{\boldsymbol{\Psi}_{t,i} \mathbf{A} \boldsymbol{\Omega}_{i,0} \mathbf{A}^T\} \tag{72}$$

$$d_{\boldsymbol{\nu}_{i,r}} G_{t,i} = -G_{t,i} \mathrm{Tr}\{\boldsymbol{\Psi}_{t,i} \mathbf{A} \boldsymbol{\Gamma}_{i,r} \mathbf{A}^T\} \tag{73}$$

$$d_{\boldsymbol{\psi}_{i,r}} G_{t,i} = -G_{t,i} \mathrm{Tr}\{\boldsymbol{\Psi}_{t,i} \mathbf{A} \boldsymbol{\Upsilon}_{i,r} \boldsymbol{\Sigma}_{t-r,i} \mathbf{A}^T\} \tag{74}$$

where $\boldsymbol{\Psi}_{t,i} = \tilde{\boldsymbol{\Sigma}}_{t,i}^{-1} - \tilde{\boldsymbol{\Sigma}}_{t,i}^{-1}\, \overline{\mathbf{z}}_{t,i} \overline{\mathbf{z}}_{t,i}^T \tilde{\boldsymbol{\Sigma}}_{t,i}^{-1}$. Furthermore, based on (14), we have the following:

i) If $i = r$:

$$\begin{aligned} d_{\beta_i} \gamma_r &= \left\{ \frac{\exp(\beta_r) \sum_{m=1}^{n} \exp(\beta_m) - \exp(\beta_r)\exp(\beta_i)}{\left[\sum_{m=1}^{n} \exp(\beta_m)\right]^2} \right\} d\beta_i \\ &= (\gamma_r - \gamma_r \gamma_i)\, d\beta_i \\ &= \gamma_r (1 - \gamma_i)\, d\beta_i. \end{aligned} \tag{75}$$

ii) If $i \neq r$:

$$\begin{aligned} d_{\beta_i} \gamma_r &= -\frac{\exp(\beta_r)\exp(\beta_i)}{\left[\sum_{m=1}^{n} \exp(\beta_m)\right]^2}\, d\beta_i \\ &= -\gamma_r \gamma_i\, d\beta_i. \end{aligned} \tag{76}$$

By combining (75) and (76), we therefore obtain

$$d_{\beta_i} \gamma_r = \gamma_r (\delta_{ir} - \gamma_i)\, d\beta_i. \tag{77}$$

Consequently, by putting (64), (65), (72)– (74), and (77) into (57)– (62), we therefore obtain

$$\begin{aligned} d_{\mathbf{B}} J_t(\boldsymbol{\Theta}_1) &= \frac{\sum_{i=1}^{n} \gamma_i G_{t,i} \mathrm{Tr}(\mathbf{X}_{t-p}^{t-1} \overline{\mathbf{z}}_{t,i}^T \tilde{\boldsymbol{\Sigma}}_{t,i}^{-1}\, d\mathbf{B})}{p(\mathbf{x}_t|\mathbf{X}_{t-p}^{t-1};\, \boldsymbol{\Theta}_1)} \\ &= \sum_{i=1}^{n} h_{t,i} \mathrm{Tr}(\mathbf{X}_{t-p}^{t-1} \overline{\mathbf{z}}_{t,i}^T \tilde{\boldsymbol{\Sigma}}_{t,i}^{-1}\, d\mathbf{B}) \end{aligned} \tag{78}$$

$$\begin{aligned} d_{\tilde{\mathbf{m}}_i} J_t(\boldsymbol{\Theta}_1) &= \frac{\gamma_i}{p(\mathbf{x}_t|\mathbf{X}_{t-p}^{t-1};\, \boldsymbol{\Theta}_1)} G_{t,i} \mathrm{Tr}(\overline{\mathbf{z}}_{t,i}^T \tilde{\boldsymbol{\Sigma}}_{t,i}^{-1}\, d\tilde{\mathbf{m}}_i) \\ &= h_{t,i} \mathrm{Tr}(\overline{\mathbf{z}}_{t,i}^T \tilde{\boldsymbol{\Sigma}}_{t,i}^{-1}\, d\tilde{\mathbf{m}}_i) \end{aligned} \tag{79}$$

and

$$\begin{aligned} & d_{\boldsymbol{\nu}_{i,0}} J_t(\boldsymbol{\Theta}_1) \\ &= -\frac{\gamma_i G_{t,i} \mathrm{Tr}(\boldsymbol{\Psi}_{t,i} \mathbf{A} \boldsymbol{\Omega}_{i,0} \mathbf{A}^T)}{p(\mathbf{x}_t|\mathbf{X}_{t-p}^{t-1};\, \boldsymbol{\Theta}_1)} \\ &= -h_{t,i} \mathrm{Tr}[\mathbf{A}^T \tilde{\boldsymbol{\Sigma}}_{t,i}^{-1} (\tilde{\boldsymbol{\Sigma}}_{t,i} - \overline{\mathbf{z}}_{t,i} \overline{\mathbf{z}}_{t,i}^T) \tilde{\boldsymbol{\Sigma}}_{t,i}^{-1} \mathbf{A} \boldsymbol{\Omega}_{i,0}] \\ &= -h_{t,i} \mathrm{Tr}[\boldsymbol{\Sigma}_{t,i}^{-1} \mathbf{W} (\tilde{\boldsymbol{\Sigma}}_{t,i} - \overline{\mathbf{z}}_{t,i} \overline{\mathbf{z}}_{t,i}^T) \mathbf{W}^T \boldsymbol{\Sigma}_{t,i}^{-1} \boldsymbol{\Omega}_{i,0}] \\ &= -h_{t,i} \mathrm{Tr}[(\boldsymbol{\Sigma}_{t,i}^{-1} - \mathbf{u}_{t,i} \mathbf{u}_{t,i}^T) \boldsymbol{\Omega}_{i,0}] \\ &= h_{t,i} \mathrm{Tr}[dg(\mathbf{u}_{t,i} \mathbf{u}_{t,i}^T - \boldsymbol{\Sigma}_{t,i}^{-1})\, d\boldsymbol{\nu}_{i,0} \boldsymbol{\nu}_{i,0}^T] \\ &= h_{t,i} \mathrm{Tr}[\boldsymbol{\nu}_{i,0}^T\, dg(\mathbf{u}_{t,i} \mathbf{u}_{t,i}^T - \boldsymbol{\Sigma}_{t,i}^{-1})\, d\boldsymbol{\nu}_{i,0}] \end{aligned} \tag{80}$$

$$
\begin{aligned}
&d_{\boldsymbol{\nu}_{i,r}} J_t(\boldsymbol{\Theta}_1) \\
&\quad = -\frac{\gamma_i G_{t,i} \mathrm{Tr}(\boldsymbol{\Psi}_{t,i} \mathbf{A} \boldsymbol{\Gamma}_{i,r} \mathbf{A}^T)}{p(\mathbf{x}_t | \mathbf{X}_{t-p}^{t-1}; \boldsymbol{\Theta}_1)} \\
&\quad = -h_{t,i} \mathrm{Tr}[\mathbf{A}^T \tilde{\boldsymbol{\Sigma}}_{t,i}^{-1} (\tilde{\boldsymbol{\Sigma}}_{t,i} - \overline{\mathbf{z}}_{t,i} \overline{\mathbf{z}}_{t,i}^T) \tilde{\boldsymbol{\Sigma}}_{t,i}^{-1} \mathbf{A} \boldsymbol{\Gamma}_{i,r}] \\
&\quad = -h_{t,i} \mathrm{Tr}[\boldsymbol{\Sigma}_{t,i}^{-1} \mathbf{W} (\tilde{\boldsymbol{\Sigma}}_{t,i} - \overline{\mathbf{z}}_{t,i} \overline{\mathbf{z}}_{t,i}^T) \mathbf{W}^T \boldsymbol{\Sigma}_{t,i}^{-1} \boldsymbol{\Gamma}_{i,r}] \\
&\quad = -h_{t,i} \mathrm{Tr}(\boldsymbol{\Sigma}_{t,i}^{-1} \boldsymbol{\Omega}_{i,r} - \boldsymbol{\Sigma}_{t,i}^{-1} \mathbf{W} \overline{\mathbf{z}}_{t,i} \overline{\mathbf{z}}_{t,i}^T \mathbf{W}^T \boldsymbol{\Sigma}_{t,i}^{-1} \boldsymbol{\Gamma}_{i,r}) \\
&\quad = -h_{t,i} \mathrm{Tr}[(\boldsymbol{\Sigma}_{t,i}^{-1} - \mathbf{u}_{t,i} \mathbf{u}_{t,i}^T) \boldsymbol{\Gamma}_{i,r}] \\
&\quad = h_{t,i} \mathrm{Tr}[dg(\boldsymbol{\varepsilon}_{t-r} \boldsymbol{\varepsilon}_{t-r}^T)\, dg(\mathbf{u}_{t,i} \mathbf{u}_{t,i}^T - \boldsymbol{\Sigma}_{t,i}^{-1})\, d\boldsymbol{\nu}_{i,r} \boldsymbol{\nu}_{i,r}^T] \\
&\quad = h_{t,i} \mathrm{Tr}[\boldsymbol{\nu}_{i,r}^T\, dg(\boldsymbol{\varepsilon}_{t-r} \boldsymbol{\varepsilon}_{t-r}^T)\, dg(\mathbf{u}_{t,i} \mathbf{u}_{t,i}^T - \boldsymbol{\Sigma}_{t,i}^{-1})\, d\boldsymbol{\nu}_{i,r}]
\end{aligned} \tag{81}
$$

$$
\begin{aligned}
&d_{\boldsymbol{\psi}_{i,r}} J_t(\boldsymbol{\Theta}_1) \\
&\quad = -\frac{\gamma_i G_{t,i} \mathrm{Tr}(\boldsymbol{\Psi}_{t,i} \mathbf{A} \boldsymbol{\Upsilon}_{i,r} \boldsymbol{\Sigma}_{t-r,i} \mathbf{A}^T)}{p(\mathbf{x}_t | \mathbf{X}_{t-p}^{t-1}; \boldsymbol{\Theta}_1)} \\
&\quad = -h_{t,i} \mathrm{Tr}[\mathbf{A}^T \tilde{\boldsymbol{\Sigma}}_{t,i}^{-1} (\tilde{\boldsymbol{\Sigma}}_{t,i} - \overline{\mathbf{z}}_{t,i} \overline{\mathbf{z}}_{t,i}^T) \tilde{\boldsymbol{\Sigma}}_{t,i}^{-1} \mathbf{A} \boldsymbol{\Upsilon}_{i,r} \boldsymbol{\Sigma}_{t-r,i}] \\
&\quad = -h_{t,i} \mathrm{Tr}[\boldsymbol{\Sigma}_{t,i}^{-1} \mathbf{W} (\tilde{\boldsymbol{\Sigma}}_{t,i} - \overline{\mathbf{z}}_{t,i} \overline{\mathbf{z}}_{t,i}^T) \mathbf{W}^T \boldsymbol{\Sigma}_{t,i}^{-1} \boldsymbol{\Upsilon}_{i,r} \boldsymbol{\Sigma}_{t-r,i}] \\
&\quad = -h_{t,i} \mathrm{Tr}[\boldsymbol{\Sigma}_{t-r,i} (\boldsymbol{\Sigma}_{t,i}^{-1} - \mathbf{u}_{t,i} \mathbf{u}_{t,i}^T) \boldsymbol{\Upsilon}_{i,r}] \\
&\quad = h_{t,i} \mathrm{Tr}\{dg[\boldsymbol{\Sigma}_{t-r,i} (\mathbf{u}_{t,i} \mathbf{u}_{t,i}^T - \boldsymbol{\Sigma}_{t,i}^{-1})]\, d\boldsymbol{\psi}_{i,r} \boldsymbol{\psi}_{i,r}^T\} \\
&\quad = h_{t,i} \mathrm{Tr}\{\boldsymbol{\psi}_{i,r}^T\, dg[\boldsymbol{\Sigma}_{t-r,i} (\mathbf{u}_{t,i} \mathbf{u}_{t,i}^T - \boldsymbol{\Sigma}_{t,i}^{-1})]\, d\boldsymbol{\psi}_{i,r}\}
\end{aligned} \tag{82}
$$

with

$$
\begin{aligned}
d_{\beta_i} J_t(\boldsymbol{\Theta}_1) &= \frac{\sum_{r=1}^{n} G_{t,r} \gamma_r (\delta_{ir} - \gamma_i)\, d\beta_i}{p(\mathbf{x}_t | \mathbf{X}_{t-p}^{t-1}; \boldsymbol{\Theta}_1)} \\
&= \sum_{r=1}^{n} h_{t,r} (\delta_{ir} - \gamma_i)\, d\beta_i \\
&= (h_{t,i} - \gamma_i)\, d\beta_i
\end{aligned} \tag{83}
$$

where

$$
\begin{aligned}
h_{t,i} &= \frac{\gamma_i G_{t,i}}{p(\mathbf{x}_t | \mathbf{X}_{t-p}^{t-1}; \boldsymbol{\Theta}_1)} \\
\mathbf{u}_{t,i} &= \boldsymbol{\Sigma}_{t,i}^{-1} \mathbf{W} \overline{\mathbf{z}}_{t,i}.
\end{aligned} \tag{84}
$$

Based on the result that if $d_{\boldsymbol{\theta}} f = \mathrm{Tr}(\mathbf{S}\, d\boldsymbol{\theta})$, then $\partial f / \partial \boldsymbol{\theta} = \mathbf{S}^T$, from (78)–(83), we therefore have

$$
\begin{aligned}
\frac{\partial J_t(\boldsymbol{\Theta}_1)}{\partial \mathbf{B}} &= \sum_{i=1}^{n} h_{t,i} \tilde{\boldsymbol{\Sigma}}_{t,i}^{-1} (\mathbf{z}_t - \tilde{\mathbf{m}}_i) \mathbf{X}_{t-p}^{t-1\,T} \\
\frac{\partial J_t(\boldsymbol{\Theta}_1)}{\partial \tilde{\mathbf{m}}_i} &= h_{t,i} \tilde{\boldsymbol{\Sigma}}_{t,i}^{-1} (\mathbf{z}_t - \tilde{\mathbf{m}}_i) \\
\frac{\partial J_t(\boldsymbol{\Theta}_1)}{\partial \beta_i} &= \frac{\sum_{r=1}^{n} G_{t,r} \gamma_r (\delta_{ir} - \gamma_i)}{p(\mathbf{x}_t | \mathbf{X}_{t-p}^{t-1}; \boldsymbol{\Theta}_1)} = h_{t,i} - \gamma_i \\
\frac{\partial J_t(\boldsymbol{\Theta}_1)}{\partial \boldsymbol{\nu}_{i,0}} &= h_{t,i}\, dg(\mathbf{u}_{t,i} \mathbf{u}_{t,i}^T - \boldsymbol{\Sigma}_{t,i}^{-1}) \boldsymbol{\nu}_{i,0} \\
\frac{\partial J_t(\boldsymbol{\Theta}_1)}{\partial \boldsymbol{\nu}_{i,r}} &= h_{t,i}\, dg(\mathbf{u}_{t,i} \mathbf{u}_{t,i}^T - \boldsymbol{\Sigma}_{t,i}^{-1})\, dg(\boldsymbol{\varepsilon}_{t-r} \boldsymbol{\varepsilon}_{t-r}^T) \boldsymbol{\nu}_{i,r} \\
\frac{\partial J_t(\boldsymbol{\Theta}_1)}{\partial \boldsymbol{\psi}_{i,r}} &= h_{t,i}\, dg[(\mathbf{u}_{t,i} \mathbf{u}_{t,i}^T - \boldsymbol{\Sigma}_{t,i}^{-1}) \boldsymbol{\Sigma}_{t-r,i}] \boldsymbol{\psi}_{i,r}.
\end{aligned} \tag{85}
$$

## ACKNOWLEDGMENT

The authors would like to thank the Editor, Associate Editor, and anonymous reviewers for their valuable comments and suggestions.

## REFERENCES


[1] S. I. Amari, A. Cichocki, and H. H. Yang, "A new learning algorithm for blind separation of sources," in *Advances in Neural Information Processing*. Cambridge, MA: MIT Press, 1996, vol. 8, pp. 757–763.

[2] H. Attias, "Blind source separation and deconvolution: The dynamic component analysis algorithm," *Neural Comput.*, vol. 10, pp. 1373–1424, 1998.

[3] A. D. Back and A. S. Weigend, "A first application of independent component analysis to extracting structure from stock returns," *Int. J. Neural Syst.*, vol. 8, no. 4, pp. 473–484, 1997.

[4] A. J. Bell and T. J. Sejnowski, "An information-maximization approaches to blind separation and blind deconvolution," *Neural Comput.*, vol. 7, pp. 1129–1159, 1995.

[5] ——, "The independent components of natural scenes are edge filters," *Vis. Res.*, vol. 37, no. 23, pp. 3327–3338, 1997.

[6] T. Bollerslev, "Generalized autoregressive conditional heteroskedasticity," *J. Econometr.*, vol. 31, pp. 307–327, 1986.

[7] Y. M. Cheung, "Dual auto-regressive modeling approach to Gaussian process identification," in *Proc. IEEE Int. Conf. Multimedia Expo*, Tokyo, Japan, 2001, pp. 1256–1259.

[8] ——, "Temporal principal component analysis—Advances in dual auto-regressive modeling for blind Gaussian process identification," in *Proc. IEEE Int. Conf. Systems, Man, Cybernetics*, Hammamet, Tunisia, Oct. 6–9, 2002.

[9] Y. M. Cheung and L. Xu, "An auto-regressive based independent component analysis approach for temporal signal separation," in *Proc. Int. Conf. Speech Processing*, vol. 2, 1999, pp. 423–427.

[10] Y. M. Cheung, C. C. Cheung, and L. Xu, "Adaptive algorithms for auto-regressive based temporal signal separation," in *Proc. Int. Conf. Artificial Intelligence*, Monte Carlo Resort, Las Vegas, NV, 2000, pp. 1505–1511.

[11] P. Comon, "Independent component analysis—A new concept?," *Signal Process.*, vol. 36, no. 3, pp. 287–314, 1994.

[12] R. F. Engle, "Autoregressive conditional heteroskedasticity with estimates of the variance of United Kingdom inflation," *Econometrica*, vol. 50, no. 4, pp. 987–1007, 1982.

[13] M. Girolami, "An alternative perspective on adaptive independent component analysis algorithms," Dept. Comput. Inform. Syst., Paisley Univ., Glasgow, U.K., Tech. Rep. ISSN 1461–6122, 1997.

[14] M. Girolami and C. Fyfe, "Extraction of independent signal sources using a deflationary exploratory projection pursuit network with lateral inhibition," *Proc. Inst. Elect. Eng. Vision, Image, Signal Process.*, vol. 14, no. 5, pp. 299–306, 1997.

[15] A. Hyvarinen, "Independent component analysis for time-dependent stochastic process," in *Proc. Int. Conf. Artificial Neural Network*, 1998, pp. 135–140.

[16] T. P. Jung, C. Humphries, T. W. Lee, S. Makeig, M. Mckeown, V. Iragui, and T. J. Sejnowski, "Extended ICA removes artifacts from electroencephalographic recordings," *Adv. Neural Inform. Process. Syst.*, vol. 10, 1998.

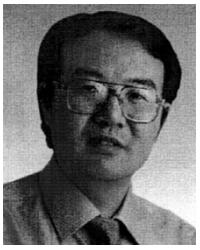
[17] J. Karhunen, E. Oja, L. Wang, R. Vigario, and J. Joutsensalo, "A class of neural networks for independent component analysis," *IEEE Trans. Neural Networks*, vol. 8, pp. 487–504, Apr. 1997.

[18] K. Kiviluoto and E. Oja, "Independent component analysis for parallel financial time series," in *Proc. Fifth Int. Conf. Neural Inform. Process.*, 1998, pp. 895–898.

[19] T. W. Lee, A. J. Bell, and R. Orglmeister, "Blind source separation of real-world signals," in *Proc. IEEE Int. Conf. Neural Networks*, 1997, pp. 2129–2135.
[20] T. W. Lee, M. S. Lewicki, and T. J. Sejnowski, "Unsupervised classification with non-Gaussian mixture models using ICA," *Adv. Neural Inform. Process. Syst.*, vol. 11, 1998.
[21] S. Makeig, A. J. Bell, T. Jung, and T. J. Sejnowski, "Independent component analysis of electroencephalographic data," *Adv. Neural Inform. Process. Syst.*, vol. 8, pp. 145–151, 1996.
[22] M. Mckeown, S. Makeig, G. Brown, T. P. Jung, S. Kindermann, T. W. Lee, and T. J. Sejnowski, "Spatially independent activity patterns in functional magnetic resonance imaging data during the stroop color-naming task," *Proc. Nat. Acad. Sci.*, vol. 95, pp. 803–810, 1998.
[23] L. Molgedey and H. G. Schuster, "Separation of a mixture of independent signals using time delayed correlations," *Phys. Rev. Lett.*, vol. 72, no. 23, pp. 3634–3637, 1994.
[24] E. Oja and J. Karhunen, "Signal separation by nonlinear Hebbian learning," in *Proc. Int. Conf. Neural Networks*, 1995, pp. 83–87.
[25] B. A. Pearlmutter and L. C. Parra, "A context-sensitive generalization of ICA," in *Proc. Int. Conf. Neural Inform. Process.*, vol. 1, 1996, pp. 151–157.
[26] B. A. Pearlmutter and L. C. Parra, "Maximum likelihood blind source separation: A context-sensitive generalization of ICA," in *Advances in Neural Information Processing Systems*. Cambridge, MA: MIT Press, 1997, vol. 9, pp. 613–619.
[27] D. T. Pham, P. Garrat, and C. Jutten, "Separation of a mixture of independent sources through a maximum likelihood approach," in *Proc. EUSIPCO*, 1992, pp. 771–774.
[28] K. Torkkola, "Blind separation of radio signals in fading channels," *Adv. Neural Inform. Process. Syst.*, vol. 10, 1998.
[29] W. C. Wong, F. Yip, and L. Xu, "Financial prediction by finite mixture GARCH model," in *Proc. Fifth Int. Conf. Neural Inform. Process.*, 1998, pp. 1351–1354.
[30] L. Xu, "Bayesian Ying-Yang system and theory as a unified statistical learning approach: (V) Temporal modeling for temporal perception and control," in *Invited paper, Proc. Int. Conf. Neural Inform. Process.*, vol. 2, 1998, pp. 877–884.
[31] ——, "Temporal BYY learning for state space approach, hidden Markov model and blind source separation," *IEEE Trans. Signal Processing*, vol. 48, pp. 2132–2144, July 2000.
[32] L. Xu, C. C. Cheung, and S. I. Amari, "Learned parametric mixture based ICA algorithm," *Neurocomput.*, vol. 22, pp. 69–80, 1998.
[33] L. Xu, C. C. Cheung, H. H. Yang, and S. I. Amari, "Independent component analysis by the information-theoretic approach with mixture of density," in *Proc. IEEE Int. Conf. Neural Networks*, vol. III, Houston, TX, June 9–12, 1997, pp. 1821–1826.

**Yiu-ming Cheung** (M'00) received the Ph.D. degree from Department of Computer Science and Engineering from the Chinese University of Hong Kong in 2000.

He then became a visiting assistant professor with the same institution. Currently, he is an assistant professor with the Department of Computer Science, Hong Kong Baptist University. His research interests include machine learning, signal processing, data mining, financial time series analysis, and portfolio management.



**Lei Xu** (F'01) received the Ph.D. degree from Tsinghua University, Beijing, China, in 1987.

He is a Professor with the Department of Computer Science and Engineering, Chinese University of Hong Kong (CUHK). He is also a Full Professor at Peking University, Beijing, and an Adjunct Professor at three other universities in China and the U.K. In 1987, he joined Peking University, where he became one of ten university-level exceptionally promoted young associate professors in 1988. He was promoted to Full Professor in 1992. From 1989 to 1993, he worked at several universities in Finland, Canada, and the United States, including Harvard University, Cambridge, MA, and the Massachusetts Institute of Technology, Cambridge. He joined CUHK in 1993 as a Senior Lecturer and then took the current Professor position in 1996. He has published over 240 academic papers, with a number of them well cited in literature.

Professor Xu has given a number of keynote/plenary/invited/tutorial talks in international major neural networks (NN) conferences, such as WCNN, IEEE-ICNN, IJCNN, ICONIP, etc. He is on the Governor Board of International NN Society, a past president of Asia-Pacific NN Assembly, and an associate editor for six international journals on NN, including *Neural Networks* and IEEE TRANSACTIONS ON NEURAL NETWORKS. He was a 1996 ICONIP program committee chair and a general chair of IDEAL'98 and IDEAL'00. He has also served as the program committee member in international major NN conferences in the past decade, including IJCNN in 1997 and 1999–2002), WCNN in 1995, and 1996, IEEE-ICNN in 1996), etc. He has received several Chinese national prestigious academic awards (including the National Nature Science Prize) as well as some international awards (including a 1995 INNS Leadership Award). He is a Fellow of the International Association on Pattern Recognition.