

Separating More Sources Than Sensors Using Time-Frequency Distributions

Nguyen Linh-Trung

Service Système Télécommunications Spatiales, Centre National d'Études Spatiales, 18 avenue Edouard Belin, 31401 Toulouse, France
Email: linhtrung@ieee.org

Adel Belouchrani

Département d'Électronique, École Nationale Polytechnique, 10 avenue Hassen Badi, PB 182 EL Harrach, 16200 Algiers, Algeria
Email: adel.belouchrani@mail.enp.edu.dz

Karim Abed-Meraim

Département Traitement du Signal et des Images, École Nationale Supérieure des Télécommunications, 46 rue Barrault, 75634 Paris Cedex 13, France
Email: abed@tsi.enst.fr

Boualem Boashash

College of Engineering, University of Sharjah, P.O. Box 27272, Sharjah, United Arab Emirates
Email: boualem_boashash@yahoo.com

Received 8 July 2004; Revised 24 March 2005; Recommended for Publication by Kostas Berberidis

We examine the problem of blind separation of nonstationary sources in the underdetermined case, where there are more sources than sensors. Since time-frequency (TF) signal processing provides effective tools for dealing with nonstationary signals, we propose a new separation method that is based on time-frequency distributions (TFDs). The underlying assumption is that the original sources are disjoint in the time-frequency (TF) domain. The successful method recovers the sources by performing the following four main procedures. First, the spatial time-frequency distribution (STFD) matrices are computed from the observed mixtures. Next, the auto-source TF points are separated from cross-source TF points thanks to the special structure of these mixture STFD matrices. Then, the vectors that correspond to the selected auto-source points are clustered into different classes according to the spatial directions which differ among different sources; each class, now containing the auto-source points of only one source, gives an estimation of the TFD of this source. Finally, the source waveforms are recovered from their TFD estimates using TF synthesis. Simulated experiments indicate the success of the proposed algorithm in different scenarios. We also contribute with two other modified versions of the algorithm to better deal with auto-source point selection.

Keywords and phrases: underdetermined blind source separation, spatial time-frequency distribution, time-frequency synthesis, unsupervised vector clustering, nonstationary sources.

1. INTRODUCTION

Blind source separation (BSS) considers the estimation of multiple sources from multiple observations (mixtures) received by a set of sensors, where the observations have been linearly mixed by the transfer medium. The transfer medium between the sources and the mixtures forces each mixture to contain a combination of the sources. The term “blind” indicates that no a priori knowledge of both the sources and the structure of the transfer medium is available. To compensate for this lack of information, the sources are usually assumed to be statistically independent [1]. BSS is important

when precise modeling of the medium transfer is difficult or when no a priori information is available about the mixtures.

BSS is also known as blind array processing, signal copy, independent component analysis, or waveform preserving estimation. Useful theories and methods of BSS can be found in, for example, [1, 2, 3, 4, 5]. BSS has many applications in areas that involve the processing of signals from a sensor array. Typical examples of BSS are seen in (i) radar and sonar applications (source separation/recognition from antenna arrays, robust source localization from ill-calibrated arrays [6]), (ii) communications (multiuser detection [7]),

(iii) speech processing (speaker separation, also called the “cocktail party” problem, in presence of background noise and/or competing speakers, automatic voice recognition in noisy acoustic environments [8]), and (iv) biomedical signal processing (separation of electroencephalogram (EEG) signals [9]).

When sources are *nonstationary*, a time-frequency (TF) approach was introduced in [10] that combines spatial diversity with TF diversity by using spatial time-frequency distribution (STFD). The benefit of using STFD for nonstationary sources is the direct exploitation of the information it offers due to the signal nonstationarity. In contrast to other BSS approaches using second- and higher-order statistics (see [4] and references therein), this approach allows the separation of Gaussian sources with identical spectral shape but with different TF localization properties. Moreover, the effects of spreading the noise power using time-frequency distributions (TFDs), while localizing the source energy in the TF domain, amount to increasing the signal-to-noise ratio (SNR) [11]. Further developments following this approach can be seen in [12, 13].

Although many successful BSS algorithms, including the above TF approach, have been proposed in the determined case, they cannot work in the underdetermined case, where there are more sources than sensors. This problem, now referred to as underdetermined blind source separation (UBSS), is challenging by itself. The reason is that these algorithms rely mathematically on the invertibility of the mixing matrix, whereas this is no longer satisfied in the underdetermined case.

Therefore, our motivation in this paper is twofold: we examine the problem of blind separation of *nonstationary* sources in the *underdetermined* separation configuration, where there are more sources than sensors. By extending the above-mentioned TF approach, we propose here a TF-based UBSS (TF-UBSS) algorithm for nonstationary sources under the main assumption that the sources are disjoint in the TF domain. In particular, the TF disjoint assumption allows for the identification, and in turn the selection and grouping, of TF points that belong to individual sources.

UBSS has been studied in [8, 14, 15, 16, 17, 18, 19]. We make some distinctions between these UBSS approaches and our TF-UBSS method, which will be described later in Section 4. The works in [14, 15] treat discrete sources, that is, sources belonging to a known set, whereas our method has no such restriction. We also do not rely on the a priori knowledge of source probability density functions which are required in [16, 17, 18]. With respect to [8, 19], we follow a similar approach in which sources are assumed to be disjoint in the TF domain. A general UBSS framework in which the mixed signal representation is transformed into another representation in order to achieve the sparsity of original signals is provided in [19]. However, it only gives an example using the short-time Fourier transform (STFT) without providing any specific separation algorithm. Note that, in contrast to the quadratic TFDs used in our method, the STFT has a low TF resolution that leads to difficult interpretation about TF disjoint condition. Several different points compared with

[8] are in order. First, this method in [8], using STFT, also faces the problem of low TF resolution. Second, only two sensors are used in [8] and the extension to more sensors is not obvious, whereas our method is general as it applies to any number of sensors greater than two. Third, unlike our nonparametric approach, [8] follows a semiparametric approach, in which the applied model for the mixing matrix is specific to delay and amplitude, that may result in big errors if the model in use does not match with the actual situation of the mixing. Finally, other problems can be foreseen in [8], such as the inherent local minima convergence due to the use of multidimensional nonlinear optimization, or the “division-by-zero” problem, when performing normalization in the case of unequal-energy sources, because the first row of the mixing matrix has all ones.

The paper is organized as follows. Section 2 presents the data model and assumptions, especially the notion of TF disjoint sources. Section 3 describes the STFD matrices. Section 4 proposes the TF-UBSS algorithm, with detailed discussions in general and a simulated illustrative example in particular. Section 5 presents an enhanced version of the algorithm using a different TFD to achieve better selection of TF points. Section 6 provides another method to enhance the selection of TF points using image component extraction. Several measurements for numerical performance evaluation are given in Section 7. The last section is for concluding remarks and perspectives.

2. SIGNAL MODEL AND ASSUMPTIONS

Let $s_i(t)$, where $i = 1, \dots, n$, be n underlying source signals and denote $\mathbf{s}(t) = [s_1(t), s_2(t), \dots, s_n(t)]^T$. At the output of the sensor array are m observed mixture signals $x_j(t)$, where $j = 1, \dots, m$, that are represented by $\mathbf{x}(t) = [x_1(t), x_2(t), \dots, x_m(t)]^T$. Under the *instantaneous linear mixture* model, the mixture signals can be modeled as

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t) + \boldsymbol{\eta}(t), \quad (1)$$

where the mixing matrix \mathbf{A} , written out as $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n]$, represents the transfer between the source and the mixture, and $\boldsymbol{\eta}(t)$ is additive noise vector. The instantaneous means that \mathbf{A} does not depend on time. Given $\mathbf{x}(t)$, the goal of BSS is to estimate $\mathbf{s}(t)$. Schematically, the problem is illustrated in Figure 1. When $n > m$, we are said to be in the underdetermined case and now have the UBSS problem. In UBSS, the mixing matrix \mathbf{A} is no longer invertible [14]; thus any previous approaches in the determined BSS problem (i.e., $n \leq m$) are no longer applicable in general. Note that, as m approaches infinity, the quantity $(n - m)/n$ approaches zero (since $n > m$); thus, UBSS becomes determined BSS. Therefore, one may approximately use the usual methods in the determined BSS case to achieve the separation; in other words, this happens when $n - m$ is small compared to m .

We make the following two assumptions; the first is usually made in the context of BSS, and the second is the specific to the proposal of our TF-UBSS algorithm.

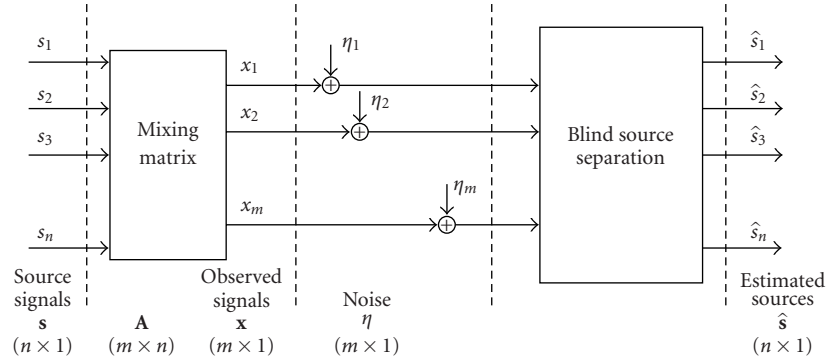


FIGURE 1: Schematic diagram of BSS.

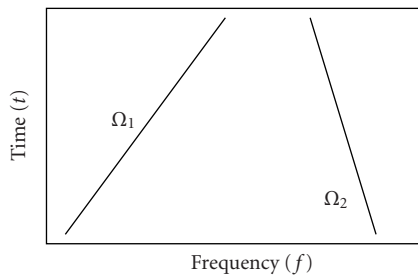


FIGURE 2: TF disjoint sources. The TF supports of two sources are disjoint in the TF domain.

(As1) The column vectors of matrix \mathbf{A} are assumed to be *pairwise* linearly independent; that is, for all $i \neq j$, \mathbf{a}_i and \mathbf{a}_j are linearly independent. If otherwise, for example, \mathbf{a}_1 and \mathbf{a}_2 are linearly dependent as such $\mathbf{a}_2 = \alpha \mathbf{a}_1$, then we can also express \mathbf{x} in the following form:

$$\mathbf{x}(t) = \tilde{\mathbf{A}}\tilde{\mathbf{s}}(t) + \boldsymbol{\eta}(t), \quad (2)$$

where $\tilde{\mathbf{A}} = [\mathbf{a}_1, \alpha \mathbf{a}_2, \dots, \mathbf{a}_n]$ and $\tilde{\mathbf{s}} = [s_1 + \alpha s_2, s_3, \dots, s_n]^T$. Hence, the separation of $s_1(t)$ and $s_2(t)$ is inherently impossible if no a priori knowledge of these sources is available. It is also known that BSS is only possible up to an unknown scaling and an unknown permutation [20]. We take the advantage of this indeterminacy to assume, without loss of generality, that the column vectors of $\tilde{\mathbf{A}}$ have a unit norm; that is, $\|\mathbf{a}_i\| = 1$, where the norm $\|\cdot\|$ is, hereafter, evaluated in the Frobenius sense.

(As2) The sources are assumed to have different structures and localization properties in the TF domain. More precisely, we assume the sources to be disjoint in the TF domain (Figure 2) as stated in the following definition.

Definition 1. Let $S_1(t, f)$ and $S_2(t, f)$ be TFDs of two source signals $s_1(t)$ and $s_2(t)$, respectively. Let also Ω_1 and Ω_2 be the corresponding TF supports¹ of S_1 and S_2 . The sources $s_1(t)$

¹ Ω_i is the TF support of $S_i(t, f)$ if the following is true: $S_i(t, f) \neq 0$ if and only if $(t, f) \in \Omega_i$.

and $s_2(t)$ are said to be *disjoint in the TF domain* if the following is satisfied:

$$\Omega_1 \cap \Omega_2 = \emptyset. \quad (3)$$

The above definition can be applied to any TFD. It is clear that the TF disjoint assumption is too restrictive and will almost never be satisfied exactly in practice. Fortunately, only approximate disjoint condition, called *quasi-disjoint*, is needed to achieve the separation, as will be shown in a later experiment (Section 4.3). Note that the source TF disjoint condition can be considered as a particular type of source sparse decomposition [16, 19, 21].

A physical example of TF (quasi)disjoint sources can be observed in a musical performance, for example, the duo performance of a base guitar and a lead guitar. As they are being played simultaneously, we hear two different musical pieces representing two different instantaneous frequency laws. The physical structures of the guitars are such that the base guitar mainly produces notes in a low frequency range, whereas the notes from the lead guitar are mainly in a high frequency range. However, the frequency range of the former has an overlap with that of the latter. At some instants during the performance, the two guitars may produce the notes belonging to this frequency overlap. Therefore, most of the time we hear the different notes and occasionally we hear the same notes. Consequently, the instantaneous frequency laws produced by these guitars for the whole duration of the performance have a few overlaps in the TF domain. In other words, the different sound sources produced by the guitars are quasi-disjoint in the TF domain.

Before describing the details about our algorithm, some remarks are listed in order regarding the TF disjoint assumption. First, one would think that if the sources are TF disjoint, then the source TF signatures can be separated using a simple TF masking procedure, instead of using any other more sophisticated methods. However, in the context of blind separation, such source TF signatures are unknown and are to be estimated. Our algorithm can estimate the source signatures. Second, one may extract all the existing TF components in the mixture by some method, such as the method in [22]. However, this method still fails to the source TF signatures because it is well possible that a source TF signature can

have multiple TF components. Our algorithm can determine which component belongs to which source, thus allowing the separation of multicomponent sources.

3. SPATIAL TIME-FREQUENCY DISTRIBUTIONS

We provide here some definitions that will be used throughout the paper.

Definition 2 (see [10]). Consider K signals $z_1(t), z_2(t), \dots, z_K(t)$, and denote $\mathbf{z}(t) = [z_1(t), \dots, z_K(t)]^T$. The $(K \times K)$ STFD matrix is defined in the following expression:

$$\mathbf{D}_{\mathbf{z}\mathbf{z}}(t, f) \triangleq \sum_{l=-\infty}^{\infty} \sum_{k=-\infty}^{\infty} \phi(k, l) \mathbf{z}(t+k+l) \mathbf{z}^H(t+k-l) e^{-j4\pi fl}, \quad (4)$$

where $\phi(k, l)$ is the TFD time-lag kernel and $(^H)$ denotes the complex conjugate transpose operator.

Note that $\mathbf{D}_{\mathbf{z}\mathbf{z}}(t, f)$ is a time-frequency varying matrix. When evaluated at a TF point (t_o, f_o) , its (i, j) elements are the values of $D_{z_i z_j}(t_o, f_o)$ obtained from the cross-TFD between $z_i(t)$ and $z_j(t)$ that is defined below:

$$\begin{aligned} D_{z_i z_j}(t, f) \\ \triangleq \sum_{l=-\infty}^{\infty} \sum_{k=-\infty}^{\infty} \phi(k, l) z_i(t+k+l) z_j^*(t+k-l) e^{-j4\pi fl}, \end{aligned} \quad (5)$$

where $(^*)$ is the conjugate operator.

By applying (4) to both sides of the linear data model in (1), and assuming a noise-free environment, we obtain the following expression, which transforms the UBSS model from the time-only domain to the dual TF domain:

$$\mathbf{D}_{\mathbf{x}\mathbf{x}}(t, f) = \mathbf{A} \mathbf{D}_{\mathbf{s}\mathbf{s}}(t, f) \mathbf{A}^H, \quad (6)$$

where $\mathbf{D}_{\mathbf{s}\mathbf{s}}(t, f)$ and $\mathbf{D}_{\mathbf{x}\mathbf{x}}(t, f)$ are the source and mixture STFD matrices, respectively. Our concern is how to exploit the TF sparsity, given by the TF disjoint assumption, from the transformed model (6) in order to allow for the separation of the underlying sources. We defer this concern until later and now define the notions of cross-source and auto-source STFDs, which are slightly modified from those defined in [12] for more clarity.

Definition 3. Consider two sources $z_1(t)$ and $z_2(t)$ and denote $\mathbf{z}(t) = [z_1(t), z_2(t)]^T$. Let $\rho(t, f)$ be the TFD used in the computation of the STFD matrix $\mathbf{D}_{\mathbf{z}\mathbf{z}}(t, f)$.

(i) (t_a, f_a) is called an *auto-source TF point* of a source $z_i(t)$ if its auto-TFD at this point, $\rho_{z_i z_i}(t_a, f_a)$, exhibits an energy concentration. The matrix $\mathbf{D}_{\mathbf{z}\mathbf{z}}(t_a, f_a)$ is called an *auto-source STFD matrix*.

(ii) (t_c, f_c) is called a *cross-source TF point* between sources $z_1(t)$ and $z_2(t)$ if their cross-TFD, defined as

$\rho_{z_1 z_2}(t, f) \triangleq \iiint_{-\infty}^{\infty} e^{j2\pi\nu(u-t)} \Gamma(\tau, \nu) \times z_1(u + (\tau/2)) z_2^*(u - (\tau/2)) e^{-j2\pi f \tau} d\nu du d\tau$ with $\Gamma(\tau, \nu)$ being the TFD lag-Doppler kernel, at this point exhibits an energy concentration. The matrix $\mathbf{D}_{\mathbf{z}\mathbf{z}}(t_c, f_c)$ is called a *cross-source STFD matrix*. When $z_1(t) = z_2(t)$, the cross-TFD becomes the auto-TFD.

Before making some remarks about the above definition, we recall the notions of “auto-term” and “cross-term” in the literature of TF signal processing. Given a signal with multiple instantaneous frequency (IF) components, an auto-term TF in the TF representation of this signal represents the “true” energy concentration of the signal in time and frequency. On the other hand, a cross-term TF represents a “ghost” energy concentration of the signal though the concentration may visually appear very high in the TF representation. This “ghost” effect comes from the TFD bilinear operation applying to the IF components [11]. Here, the TFD is applied to only one signal. In our context, we consider several source signals, and each of which may have multiple IF components. Several remarks are now in order.

(a) The energy concentration at an auto-source point can be “true” if $z_i(t)$ is monocomponent and can be “ghost” if multicomponent. The latter means that the auto-source point coincides with the cross-term point if the source is multicomponent. This will be illustrated in Experiment 1 (Figure 3l).

(b) At a cross-source point (t_c, f_c) , the cross-source STFD matrix $\mathbf{D}_{\mathbf{z}\mathbf{z}}(t_c, f_c)$ is off-diagonal (i.e., its diagonal entries are zero) since the off-diagonal elements of the matrix $\mathbf{D}_{\mathbf{z}\mathbf{z}}(t, f)$ are evaluated by the cross-TFD.

(c) At an auto-source point (t_a, f_a) , the auto-source STFD matrix $\mathbf{D}_{\mathbf{z}\mathbf{z}}(t_a, f_a)$ is diagonal (i.e., its diagonal entries are nonzero) since the diagonal elements of $\mathbf{D}_{\mathbf{z}\mathbf{z}}(t, f)$ are evaluated by the auto-TFD.

We now come back to our previously mentioned concern on the TF sparsity of the transformed model (6). Taking into account the above remarks and using the additional assumption that sources are disjoint in the TF domain, we have the following observations for the source STFD matrix $\mathbf{D}_{\mathbf{s}\mathbf{s}}(t, f)$. First, the diagonal elements of $\mathbf{D}_{\mathbf{s}\mathbf{s}}(t, f)$ at a cross-source point are *all equal to zero*. Second, $\mathbf{D}_{\mathbf{s}\mathbf{s}}(t, f)$ at an auto-source point is diagonal and, more than that, its diagonal elements are *all equal to zero except for one value* because there is only once source present at this auto-source point. Therefore, if Ω_i is the TF support of the source signal $s_i(t)$, then the following is achieved:

$$\mathbf{D}_{\mathbf{x}\mathbf{x}}(t, f) = D_{s_i s_i}(t, f) \mathbf{a}_i \mathbf{a}_i^H, \quad \forall (t, f) \in \Omega_i. \quad (7)$$

It is the particular structure in (7) that will be used for our TF-UBSS.

We note here that if the sources are nondisjoint in the TF domain, then there exists at least an auto-source point at which there are k sources present, where $1 < k$. In this case, if $k \leq m$, then exactly k diagonal elements of $\mathbf{D}_{\mathbf{s}\mathbf{s}}(t, f)$ at this point are nonzero. If, otherwise, $k > m$, all the m diagonal elements of $\mathbf{D}_{\mathbf{s}\mathbf{s}}(t, f)$ are nonzero. This observation may be used to provide a test on TF disjoint condition and further

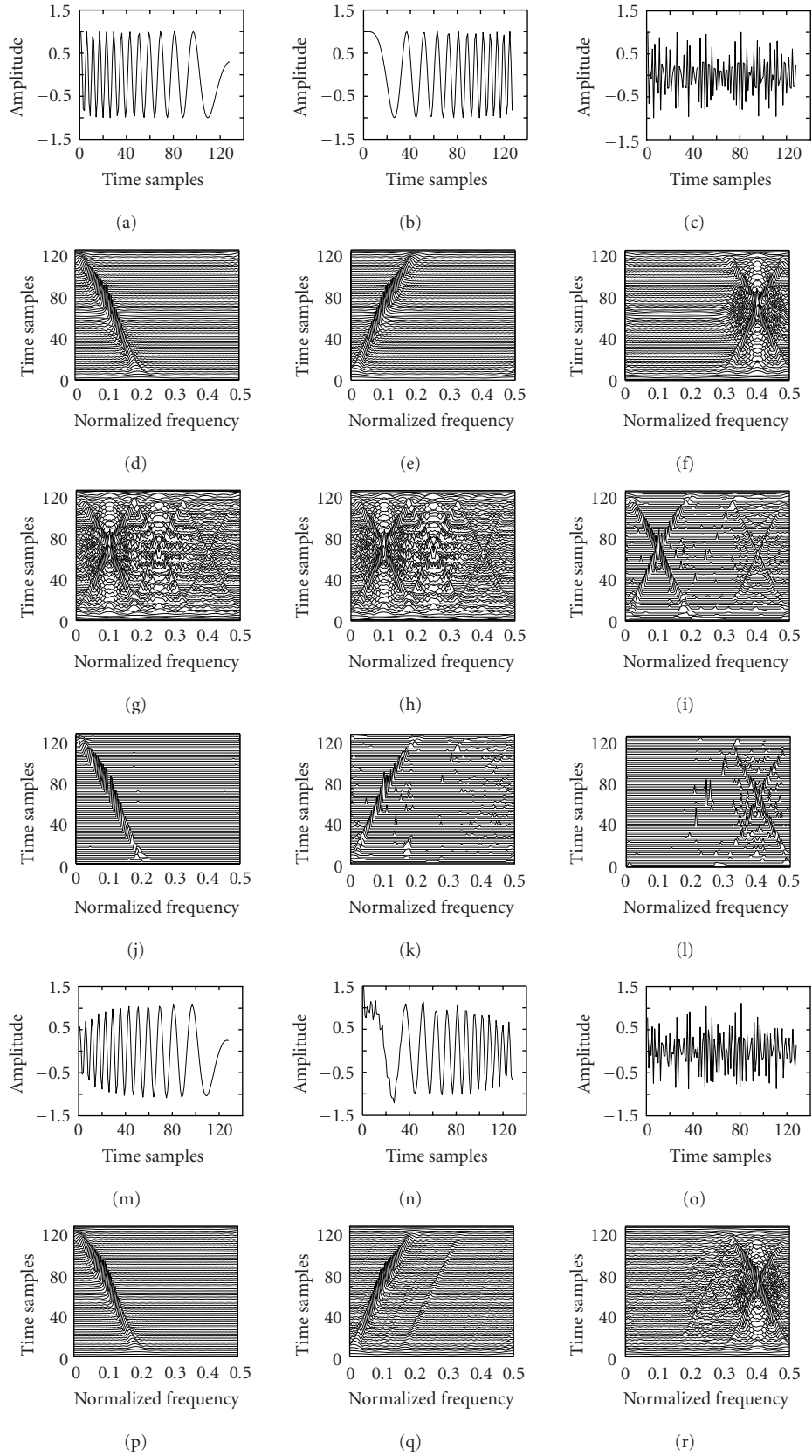


FIGURE 3: Experiment 1: TF-UBSS algorithm. (a) $s_1(t)$. (b) $s_2(t)$. (c) $s_3(t)$. (d) WVD of $s_1(t)$. (e) WVD of $s_2(t)$. (f) WVD of $s_3(t)$. (g) WVD of $x_1(t)$. (h) Auto- and cross-source points. (i) Auto-source points. (j) TF signature of $s_1(t)$. (k) TF signature of $s_2(t)$. (l) TF signature of $s_3(t)$. (m) $\hat{s}_1(t)$. (n) $\hat{s}_2(t)$. (o) $\hat{s}_3(t)$. (p) WVD of $\hat{s}_1(t)$. (q) WVD of $\hat{s}_2(t)$. (r) WVD of $\hat{s}_3(t)$.

Proc. 1: STFD computation and noise thresholding
 Proc. 2: auto-source TF point selection
 Proc. 3: clustering and source TFD estimation
 Proc. 4: source signal synthesis

ALGORITHM 1: TF-UBSS algorithm: procedures.

to study the UBSS for non-disjoint sources. The treatment of TF non-disjoint sources is out of the scope of this paper and is subject to subsequent research. This issue is of importance when dealing with speech signals that are not well localized in TF domain.

4. TF-UBSS ALGORITHM

Thanks to the structure in (7), the following observation is deduced for two auto-source points (t_1, f_1) and (t_2, f_2) corresponding to the same source $s_i(t)$:

$$\begin{aligned} \mathbf{D}_{\mathbf{xx}}(t_1, f_1) &= D_{s_i s_i}(t_1, f_1) \mathbf{a}_i \mathbf{a}_i^H, \\ \mathbf{D}_{\mathbf{xx}}(t_2, f_2) &= D_{s_i s_i}(t_2, f_2) \mathbf{a}_i \mathbf{a}_i^H. \end{aligned} \quad (8)$$

This observation implies that both $\mathbf{D}_{\mathbf{xx}}(t_1, f_1)$ and $\mathbf{D}_{\mathbf{xx}}(t_2, f_2)$ have the *same* principal eigenvector \mathbf{a}_i . Therefore, all the auto-source points whose mixture STFD matrices have the same principal eigenvector belong to the TF support of only one particular source. Recall that \mathbf{a}_i and \mathbf{a}_j have been assumed linearly independent for $i \neq j$.

This leads to the main idea of our TF-UBSS algorithm as follows. We first obtain only auto-source points from the TF representation. Next, we cluster all the auto-source points into different sets according to the difference in the principal eigenvector of the mixture STFD matrices. Each set of auto-source points now represents the TF support for a particular source. Knowing the support of the source, we can then obtain its TF signature which is, in other words, the TFD estimate of this source. Given the source TFD estimate, we can use TF synthesis to recover the source waveform.

4.1. Separation algorithm

The proposed TF-UBSS algorithm includes four main procedures as shown in Algorithm 1 and its schematic diagram is illustrated in Figure 4. Details of these procedures are given next.

(1) *STFD computation and noise thresholding.* Given L observation vectors $\mathbf{x}(1), \dots, \mathbf{x}(L)$, the STFD matrices $\mathbf{D}_{\mathbf{xx}}(t, f)$ defined according to (4) can be estimated using time-lag domain discrete implementation [11] as

$$\hat{\mathbf{D}}_{\mathbf{xx}}(l, k) = \sum_{p=-M}^M \sum_{q=-M}^M g(q-l, p) \mathbf{x}(q+p) \mathbf{x}^H(q-p) e^{-j4\pi p k / L}, \quad (9)$$

where $g(l, p)$ is a discrete time-lag kernel, $M = (L-1)/2$, and $l = 1, \dots, L$. The elements of $\mathbf{D}_{\mathbf{xx}}(l, k)$ are obtained from

the TFD as

$$\begin{aligned} [\hat{\mathbf{D}}_{\mathbf{xx}}(l, k)]_{ij} &= D_{x_i x_j}(l, k) \\ &= \sum_{p=-M}^M \sum_{q=-M}^M g(q-l, p) x_i(q+p) x_j^*(q-p) e^{-j4\pi p k / L}, \end{aligned} \quad (10)$$

where $i, j = 1, \dots, m$. In the later simulations (Experiment 1), we will use the Wigner-Ville distribution (WVD) for computing the STFD matrices. The WVD of an analytic signal $z(t)$ is defined as [11]

$$\rho_z^{\text{wvd}}(t, f) \triangleq \int_{-\infty}^{\infty} z\left(t + \frac{\tau}{2}\right) z^*\left(t - \frac{\tau}{2}\right) e^{-j2\pi f \tau} d\tau. \quad (11)$$

Its discrete implementation is of the form in (10) without the time-lag kernel $g(l, p)$.

These STFD matrices are next processed to extract the source signals. In order to reduce the computational complexity, by processing only ‘‘significant’’ STFD matrices, a noise thresholding step is then carried out to remove the points with negligible energy. More precisely, a threshold ϵ_1 (typically, $\epsilon_1 = 0.05$) is used to keep only the points $\{(t_s, f_s)\}$ with sufficient energy:

$$\begin{aligned} \text{if } \|\mathbf{D}_{\mathbf{xx}}(t_s, f_s)\| &> \epsilon_1, \\ \text{then keep } (t_s, f_s). \end{aligned} \quad (12)$$

(2) *Auto-source TF point selection.* The second procedure of the algorithm consists of separating the auto-source points from cross-source points using an appropriate testing criterion.

In the *determined* case, where the number of sources is smaller than or equal to the number of sensors, the mixing matrix \mathbf{A} is of full-column rank. A selection procedure, which exploits the off-diagonal structure of the cross-source STFD matrices, has been proposed in [12]. This selection procedure proceeds through two steps as follows.

(i) *Data whitening.* Let \mathbf{W} denote an $m \times n$ matrix such that $(\mathbf{W}\mathbf{A})(\mathbf{W}\mathbf{A})^H = \mathbf{U}\mathbf{U}^H = \mathbf{I}$, that is, $\mathbf{W}\mathbf{A}$ is an $n \times n$ unitary matrix. This matrix is referred to as the whitening matrix since it whitens the signal part of the observations. Pre- and post-multiplying the STFD matrices $\mathbf{D}_{\mathbf{xx}}(t, f)$ by \mathbf{W} lead to the whitened STFD matrices

$$\underline{\mathbf{D}}_{\mathbf{xx}}(t, f) = \mathbf{W}\mathbf{D}_{\mathbf{xx}}(t, f)\mathbf{W}^H = \mathbf{U}\mathbf{D}_{\mathbf{ss}}(t, f)\mathbf{U}^H. \quad (13)$$

In practice, \mathbf{W} is often computed as an inverse squared root of the sample estimate covariance matrix of the observation.

(ii) *Testing.* For a whitened cross-source STFD matrix $\underline{\mathbf{D}}_{\mathbf{xx}}(t_c, f_c)$, we have the following:

$$\begin{aligned} \text{trace}\{\underline{\mathbf{D}}_{\mathbf{xx}}(t_c, f_c)\} &= \text{trace}\{\mathbf{U}\mathbf{D}_{\mathbf{ss}}(t_c, f_c)\mathbf{U}^H\} \\ &= \text{trace}\{\mathbf{D}_{\mathbf{ss}}(t_c, f_c)\} \approx 0. \end{aligned} \quad (14)$$

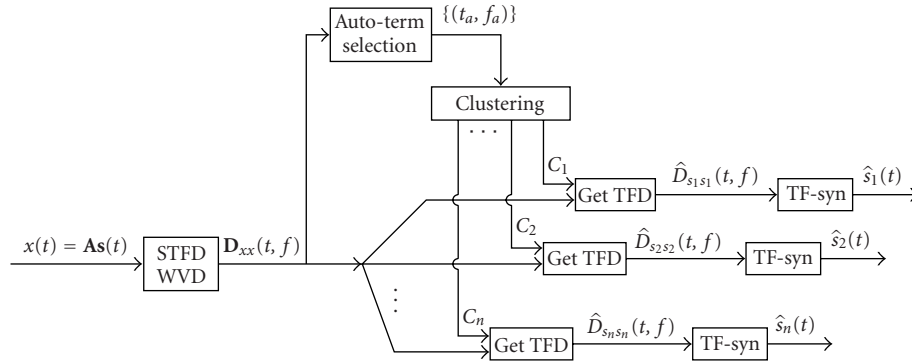


FIGURE 4: TF-UBSS algorithm: schematic diagram.

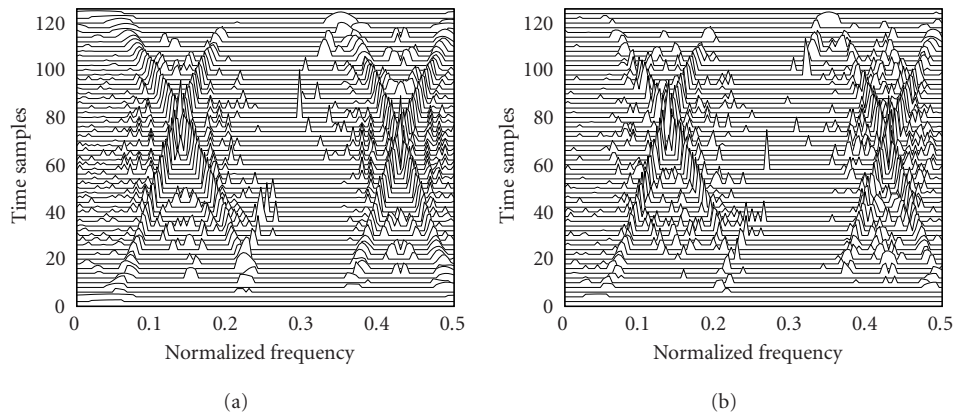


FIGURE 5: Comparison of auto-source selection methods: (a) approximation projection and (b) TF disjoint.

Hence, the cross-source points are to be removed using the testing criterion below:

$$\text{if } \frac{\text{trace} \{ \hat{\mathbf{D}}_{\mathbf{xx}}(t_c, f_c) \}}{\| \hat{\mathbf{D}}_{\mathbf{xx}}(t_c, f_c) \|} < \epsilon_2, \quad (15)$$

then (t_c, f_c) is a cross-source point,

where the threshold ϵ_2 is a positive scalar not greater than 1 (typically $\epsilon_2 = 0.8$).

In the underdetermined case, the matrix \mathbf{U} is nonsquare with more columns than rows, and consequently $\mathbf{U}^H \mathbf{U} \neq \mathbf{I}$ represents the *projection* matrix onto the row space of \mathbf{U} . Therefore, (14) becomes only an approximation, a good one if $(m - n)$ is “small” as observed in our simulation results (see Figure 3).

Another method, alternative to the above approximation projection method, may be used. Under the source TF disjoint assumption, each auto-source STFD matrix is of rank one, or at least has one “large” eigenvalue compared to its other eigenvalues. Therefore, one can use rank selection criteria, such as the minimum description length (MDL) or Akaike information criterion (AIC) [23] to select auto-source points as those corresponding to STFD matrices of selected rank equal to one. For simplicity, we use the following

criterion (see Figure 5):

$$\text{if } \left| \frac{\lambda_{\max} \{ \hat{\mathbf{D}}_{\mathbf{xx}}(t, f) \}}{\| \hat{\mathbf{D}}_{\mathbf{xx}}(t, f) \|} - 1 \right| > \epsilon_2, \quad (16)$$

then (t, f) is a cross-source point,

where ϵ_2 is a small positive scalar (typically, $\epsilon_2 = 0.3$) and $\lambda_{\max} \{ \cdot \}$ represents the largest eigenvalue of the matrix in the bracket.

Comparison of the above two methods for auto-source point selection based on approximation projection and TF disjoint assumption for the underdetermined case shows a similar performance (see Figure 5).

(3) *Clustering and source TFD estimation.* Once the auto-source points have been selected, a clustering procedure based on the sources spatial directions/signatures is performed. This clustering is based on the observation that two STFD matrices which correspond to the same source have the same principal eigenvector. Moreover, the corresponding principal eigenvalues are given by the desired source TFD. This implies that if we apply an appropriate clustering procedure to the set of auto-source points, we will be able to obtain the separate TF signatures of the sources. Specifically, we consider the following steps.

(i) For each auto-source point (t_a, f_a) , we compute from the auto-source STFD matrix $\mathbf{D}_{\mathbf{x}\mathbf{x}}(t_a, f_a)$ the main eigenvector $\mathbf{a}(t_a, f_a)$ and its corresponding eigenvalue $\lambda(t_a, f_a)$. Since $\mathbf{a}(t_a, f_a)$ is only estimated up to a random phase $e^{j\phi}$, $\phi \in [0, 2\pi)$, we force it to have, without loss of generality, its first element real and positive.

(ii) The set of all auto-source points $\{(t_a, f_a)\}$ is then clustered into different classes $\{\mathcal{C}_i\}$. Mathematically, we decide $\mathbf{a}(t_i, f_i)$ and $\mathbf{a}(t_j, f_j)$ to belong to the same class if

$$d(\mathbf{a}(t_i, f_i), \mathbf{a}(t_j, f_j)) < \epsilon_3, \quad (17)$$

where ϵ_3 is a properly chosen positive scalar and d is a distance measure. Different strategies for choosing the threshold ϵ_3 and the distance d or even the clustering method can be found in [24]. As an example, we use later in the experiments a distance measure in accordance with the angle difference as given by

$$d(\mathbf{a}_1, \mathbf{a}_2) = \arccos(\tilde{\mathbf{a}}_1^T \tilde{\mathbf{a}}_2), \quad i \neq j, \quad (18)$$

where $\tilde{\mathbf{a}} = [\text{Re}(\mathbf{a})^T; \text{Im}(\mathbf{a})^T]^T$ and $\|\tilde{\mathbf{a}}\| = 1$. Algorithmically, the clustering simply performs the following:

(a) *initialization*

set number of classes $K = 1$
assign \mathbf{a}_1 to class \mathcal{C}_1

(b) *unsupervised clustering*

while $i = 2 \rightarrow N_v$ (number of vectors)
for $k = 1 \rightarrow K$
calculate distance d between \mathbf{a}_i and \mathcal{C}_k
if $d < \epsilon_3$, then assign \mathbf{a}_i to \mathcal{C}_k
else, set $K = K + 1$ (new class created), then
assign \mathbf{a}_i to \mathcal{C}_K and exit “for” loop.

(iii) We set the number of sources equal to the number of classes then estimate the TFD of the source $s_i(t)$ from the class \mathcal{C}_i by assigning

$$\hat{D}_{s_i s_i}(t, f) = \begin{cases} \lambda(t_a, f_a), & \text{if } (t, f) = (t_a, f_a) \in \mathcal{C}_i, \\ 0, & \text{otherwise.} \end{cases} \quad (19)$$

(4) *Source signal synthesis.* The source $s_i(t)$ is then recovered from its TFD estimate $\hat{D}_{s_i s_i}$ using TF synthesis. The reconstruction of the waveform (in time) of a signal from its TFD (in time frequency) is made possible thanks to the following inversion property of the WVD [11]:

$$x(t) = \frac{1}{x^*(0)} \int_{-\infty}^{\infty} \rho_x^{\text{wvd}}\left(\frac{t}{2}, f\right) e^{j2\pi f t} df. \quad (20)$$

The above expression implies that the signal can be reconstructed to within a complex exponential constant $e^{j\alpha} = x^*(0)/|x(0)|$ given $|x(0)| \neq 0$.

A well-known synthesis algorithm was proposed in [25] to recover a signal from its estimated TFD where the TFD in use is the WVD. Since we use WVD to compute our

STFD matrices, we opt to use this synthesis algorithm for recovering our sources. Below, this algorithm is summarized from [25] to assist the understanding of our UBSS algorithm.

Let $\hat{D}_{ss}(t, f)$ be the TFD estimate of a signal $s(t)$. Our goal is to find the $\hat{s}(t)$ whose WVD, denoted by $\rho_s^{\text{wvd}}(t, f)$, best approximates $\hat{D}_{ss}(t, f)$ in the least square sense. That is, we want to minimize $J(s)$ given in the following:

$$J(s) = \int_{-\infty}^{\infty} \left| \hat{D}_{ss}(t, f) - \rho_s^{\text{wvd}}(t, f) \right|^2 df. \quad (21)$$

Let L be the signal length, $L_e = \lfloor (L+1)/2 \rfloor$, $L_o = \lfloor L/2 \rfloor$, $\mathbf{s}_o = [s_o(1)s_o(2) \cdots s_o(L_o)]^T$, and $\mathbf{s}_e = [s_e(1)s_e(2) \cdots s_e(L_e)]^T$. The above minimization then leads to the estimation of the odd and even samples of the signal as such \mathbf{s}_o and \mathbf{s}_e are equal to the normalized principal eigenvectors of the matrices \mathbf{C}_e and \mathbf{C}_o whose elements are computed by

$$c_e(q+1, p+1) = y(q+p, q-p) + y^*(q+p, p-q) \quad (22)$$

for $q, p = 0, \dots, L_e - 1$,

$$c_o(q, p) = y(q+p+1, q-p) + y^*(q+p+1, p-q) \quad (23)$$

for $q, p = 1, \dots, L_o$,

where $y(l, p)$ is the discrete inverse Fourier transform of $\hat{D}_{ss}(t, f)$. If the phase of the recovered signal is important, the phase can be corrected using the original signal $s(t)$ by first computing

$$\alpha_e = \tan^{-1} \left\{ \frac{\Re \left\{ \sum_{k=0}^{L_e-1} s(2k) s_e^*(k) \right\}}{\Im \left\{ \sum_{k=0}^{L_e-1} s(2k) s_e^*(k) \right\}} \right\}, \quad (24)$$

$$\alpha_o = \tan^{-1} \left\{ \frac{\Re \left\{ \sum_{k=1}^{L_o} s(2k-1) s_o^*(k) \right\}}{\Im \left\{ \sum_{k=1}^{L_o} s(2k-1) s_o^*(k) \right\}} \right\}$$

then doing the following assignments: $s_e(k) := s_e(k)e^{j\alpha_e}$ and $s_o(k) := s_o(k)e^{j\alpha_o}$. Above, $\Re \{\cdot\}$ and $\Im \{\cdot\}$ denote the real part and imaginary part, respectively.

4.2. Discussion

It is essential to address the following issues regarding the above proposed TF-UBSS algorithm.

(1) *Underdeterminacy.* The algorithm is *general* in the sense that it is not only specific to UBSS but can also be used for BSS. However, in this work, we only provide results for the challenging case of UBSS. In the simulated experiments that will be shown later, we choose $m = 2$. Obviously, to exploit the spatial diversity offered by a sensor array, the minimum value for m is two. This case, however, is the most difficult case given a fixed number of sources, in contrast to an intuition that $m = 2$ might be the simplest. This is due to the fact that more sensors will provide more spatial diversity, hence more information. On the other hand, we only use $n = 3$ sources which is the simplest case for UBSS given that $m = 2$. This selection serves our purpose to illustrate the

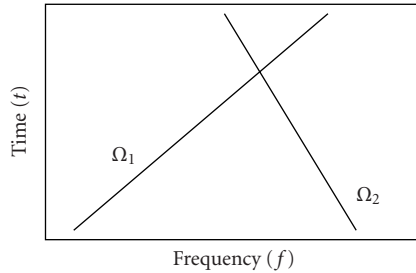


FIGURE 6: TF quasi-disjoint sources.

new approach, rather than to provide a very detailed performance analysis on the approach.

(2) *TF disjoint sources.* It is important to have disjoint sources in the TF domain in order to achieve the separation using the above TF-UBSS algorithm. However, this assumption is too restrictive and will almost never be satisfied exactly in practice. Nonetheless, as shown in the simulation in Section 4.3, it suffices that the source signals may need only to satisfy a TF quasi-disjoint condition for the signal separation to be achieved. The term “quasi” implies that “most” of the energy of one source is localized in the TF region that is disjoint from the TF regions of all other sources, as illustrated in Figure 6. Small overlaps of the two TF supports are allowable ($\Omega_1 \cap \Omega_2 \approx \emptyset$).

(3) *Choice of the TFD.* In general, the choice of a TFD depends on the specific application that is under consideration and the representation properties that are desirable for this application [11]. In our simulations, we chose the WVD in the computation of the STFD matrices for the following two reasons. First, the WVD is invertible up to a constant phase, as given in (20). Second, the WVD is the optimal TFD for linear frequency-modulated (LFM) signals [11] and these signals are used in our simulations. The proposed algorithm is sensitive to the choice of the TFD selected and is particular to the cross-term effect, as will be shown in Section 4.3. Hence, using a TFD that reduces this effect is important to obtain a good separation quality.

(4) *Noise thresholding.* The threshold used for removing the noisy points can be chosen based on the SNR and the possible structure of the mixed signals. Noise thresholding, however, is used mainly for the benefit of reducing the computational complexity, and so is not a critical factor in the proposed algorithm.

(5) *Auto-source point selection.* We have proposed three selection criteria to separate the auto-source points from the cross-source points in the TF plane. These criteria require a good choice of the thresholds and the TFD (a good choice of the TFD is proposed in Section 5). It is clear that if few auto-source points are missing, the signal reconstruction, via the implicit interpolation in the synthesis procedure, works in general. Now, if too few auto-source points are selected, the signal reconstruction then fails since too many curves/solutions are possible to fill the gaps in between the selected auto-source points. In that case, the only way to get the correct solution is to have strong a priori knowledge of the desired sources.

(6) *Vector clustering.* A simple algorithm for vector clustering was used in the simulations in order to illustrate the feasibility of UBSS. More sophisticated algorithms (see [24] and references therein) should be applied to achieve more robust separation.

(7) *Number of sources.* We have observed in the experiments that the number of classes, obtained from the clustering procedure, was greater than the actual number of sources. Simple thresholding scheme, based on energy leveling, was used to eliminate the classes with insignificant energy compared to others. These classes may or may not be considered as noise, depending on the nature of the sources in the particular application of interest.

On the other hand, the clustering may result in a number of sources smaller than the original one. This would be the case in two typical situations. The first one occurs when “effective” high-energy sources are mixed with other low-energy sources. The latter may be seen as “noise” in the separation procedure and, thus, cannot be estimated, while the effective sources are correctly estimated. If the low-energy sources are of interest, one can extract them by applying our algorithm in an iterative way in conjunction with a deflation technique [26]. The other situation occurs when two or more sources are “closely spaced,” that is, their corresponding column vectors in the mixing matrix are “closely” linear-dependent. In that case, the clustering technique easily fails to separate the sources. Note that this situation is difficult to solve even in the determined case when the mixing matrix is of full-column rank.

(8) *TF synthesis.* We have applied in our simulations a classical but seminal algorithm (without any TF masking) proposed in [25]. Other synthesis algorithms can be found in [11, 27, 28]. The successful reconstruction of source waveforms depends on the source signal type, the choice of TFD, the robustness of vector clustering procedure, and the performance of the TF synthesis algorithm itself.

On the other hand, instead of using TF synthesis, we may apply the time-varying notched filter approach as sketched in Figure 7 in which selection block is composed of all the steps from Procedure 4.1(1) to Procedure 4.1(3). More information on designing such a notched filter can be found in [29]. This approach is useful when the TF synthesis algorithm corresponding to the TFD used for STFD computation is not yet available.

(9) *Computational complexity.* The total cost of computation is broken down into separate costs corresponding to different procedures in the proposed algorithm. Major contributions to the total cost C_{total} come from the computations of (i) STFD matrices (C_1), (ii) the singular value decomposition (SVD) of the STFD matrices for separating auto-source points from cross-source points (C_2), (iii) clustering (C_3), and source synthesis (C_4). Note that, we use the values of SVD already obtained for the estimation of source TFDs:

$$C_{\text{total}} \approx C_1 + C_2 + C_3 + C_4. \quad (25)$$

Let N_a and N_c be the number of auto-source points and cross-source points, respectively, and denote by C_L the cost

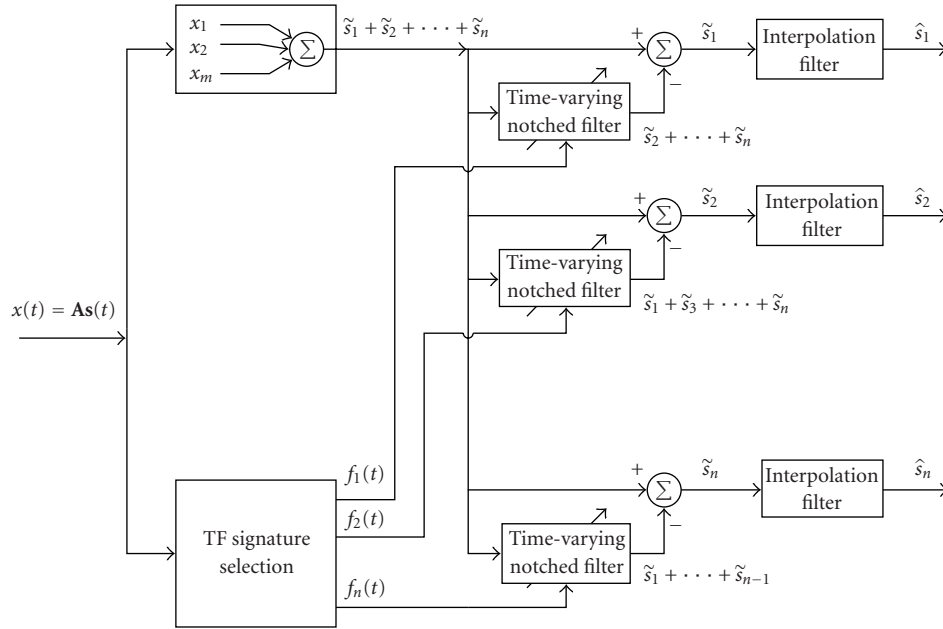


FIGURE 7: TF-UBSS algorithm using notch filters: schematic diagram.

for the TFD computation of a signal of length L . Then, the associated costs are

$$\begin{aligned}
 C_1 &= C_L \times \frac{m(m+1)}{2}, \\
 C_2 &= (N_a + N_c) \times \mathcal{O}(m^3), \\
 C_3 &\approx \frac{N_a(N_a+1)}{2}, \\
 C_4 &\approx n \times \mathcal{O}(L^3).
 \end{aligned} \tag{26}$$

Note that the computation of C_L depends on the TFD method, signal length L , and the number of FFT points used. If a sophisticated clustering method is used, then C_3 is expected to increase. Overall, C_2 and C_3 are the most expensive computations due to the high numbers of auto-source points and cross-source points present in the TF representation; obviously, these numbers depend on the number of sources.

4.3. Experiment 1: TF-UBSS algorithm

A uniform linear array of $m = 2$ sensors, having half wavelength spacing, is used. It receives $n = 3$ independent sources, as shown in Figures 3a, 3b, 3c, where the two sources $s_1(t)$ and $s_2(t)$ are monocomponent LFM signals and the remaining source $s_3(t)$ is a multicomponent signal. Each signal has the length $L = 128$. The transfer medium is assumed to be affected by the additive white Gaussian noise (AWGN), where the SNR is equal to 20 dB. The source signals arrive at different angles, 30° , 45° , and 60° , respectively.

The WVD was used to compute the STFD matrices. The “noisy” points appearing in the data mixture (Figure 3g) were first removed using energy thresholding; see Figure 3h (a significant number of points were indeed removed though

difficult to visualize). Next, the auto-source points were selected as shown in Figure 3i. Then, the output of vector clustering was the three separate classes that represent the three TF signatures of the sources (Figures 3j, 3k, 3l). Finally, the sources were reconstructed as shown in Figures 3m, 3n, 3o. The estimated sources and their WVD representations (Figures 3p, 3q, 3r) resembled the original sources (Figures 3a, 3b, 3c) and their WVD representations (Figures 3d, 3e, 3f), respectively.

By comparing the original with the estimates of source waveforms, we can conclude that the proposed UBSS algorithm was successful. Note that, however, an amplitude fading at the two ends of the recovered signals is due to the poor TFD energy concentration in the vicinity of the TF support boundaries. In addition, though significant cross-source points have been removed, there remain a small number of cross-source points in the clustered TF signatures. This indicates that the proposed method is sensitive to cross-term effect, hence motivates the use of a TFD that can effectively reduce the cross-terms (this will be shown in the next section).

In this experiment, we deliberately set up the sources such as they were quasi-disjoint in the TF domain ($s_1(t)$ and $s_2(t)$ overlap each other). The sources $s_1(t)$ from $s_2(t)$ were, however, still separated successfully by the proposed algorithm. In addition, the purpose of setting the third source to be multicomponent was to distinguish the proposed algorithm from any time-varying filtering approach. As confirmed by the simulation, the proposed algorithm did not falsely separate $s_3(t)$ into two monocomponent LFM sources, whereas a time-varying filtering approach would normally interpret this source as two separate monocomponent sources.

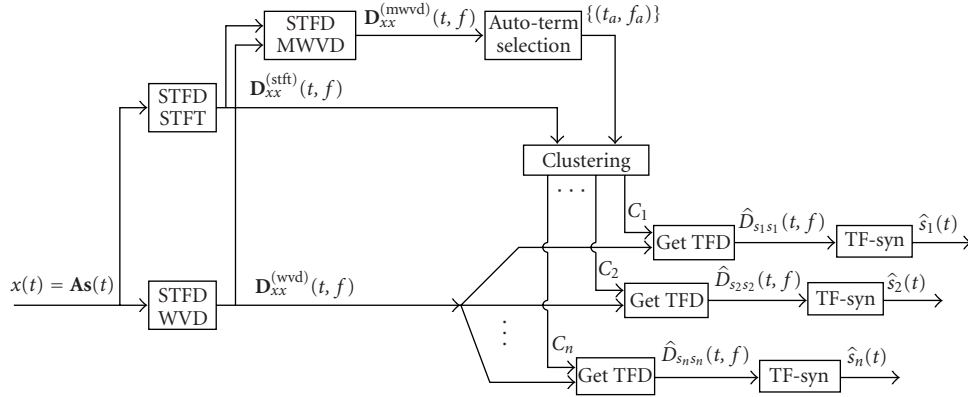


FIGURE 8: TF-UBSS algorithm using MWVD: schematic diagram.

5. TF-UBSS USING MWVD

The previous experiment showed that the proposed algorithm was successful in separating nonstationary sources in the UBSS case. However, as observed, there were undesirable (cross-source) points, in the TF signatures (Figures 3k and 3l), present along with the desired (auto-source) points for a particular source. Consequently, extra ridges appeared in the TFD of the estimated sources (Figures 3q and 3r). The presence of these extra ridges may lead to a wrong interpretation of the original source as if it has another IF law. Therefore, we need to seek a more robust solution. In this section, we propose a modified version of the proposed TF-UBSS algorithm that helps improve the auto-source point selection procedure, hence the performance of the separation.

It is essential to note that, apart from the observed problem in auto-source selection, other problems remain: (i) the thresholds used in the previous experiment were ad hoc and (ii) there were no treatments at the overlapping due to the quasi-disjoint scenario (i.e., a point at the intersection of two sources was only clustered “by chance” to belong to one of the sources). These issues need further development and is out of the scope of this paper.

5.1. Remarks

In improving the proposed algorithm, we first notice the following.

(i) Although the WVD is optimal for LFM signals, it suffers from the cross-term problem [11]. A subset of quadratic TFDs specifically designed for cross-term suppression is called the set of reduced interference distributions (RIDs) [11]. We choose to use the modified Wigner-Ville distribution (MWVD) which is defined as

$$\rho_x^{\text{mwvd}}(t, f) = \rho_x^{\text{wvd}}(t, f) \cdot \rho_x^{\text{spec}}(t, f), \quad (27)$$

where the spectrogram (SPEC) is given by

$$\begin{aligned} \rho_z^{\text{spec}}(t, f) &= \left| \rho_z^{\text{stft}}(t, f) \right|^2 \\ &= \left| \int_{-\infty}^{\infty} z(\tau) h(\tau - t) e^{-j2\pi f \tau} d\tau \right|^2. \end{aligned} \quad (28)$$

In (28), $\rho_z^{\text{stft}}(t, f)$ is the STFT and $h(t)$ is some window function.

The choice of the MWVD serves two purposes: the WVD keeps the high resolution and the optimality for LFM signals, and the SPEC is free of cross-terms. In addition, the implementation of the TF synthesis algorithm used in this paper is based on the WVD; thus we still need to perform the original computation of STFD matrices using the WVD. One would expect that if TF synthesis algorithms of RIDs are available then direct use of such algorithms would help improve the TF-UBSS algorithm quality as it leads to better selection of the auto-source points and a better result for clustering.

(ii) Previously, the inputs to the clustering procedure were the selected set of auto-source points and the WVD-based STFD matrices. However, as observed in Experiment 1, there were points where each of them is a superimposition of both an auto-source point and cross-source point. We propose a solution to this by applying the STFT in the clustering procedure. This is due to the fact that the STFT is the square root of the SPEC, hence is free of cross-source points.

5.2. Algorithm

Based on the above discussion, we are now able to set up the procedures of TF-UBSS algorithm using the MWVD for a refined auto-source point selection. The algorithm consists of the same overall procedures as those in Algorithm 1. A diagram of the algorithm is also shown in Figure 8.

(1) *STFD computation and noise thresholding.* We compute the STFD matrices of the observation vectors $\mathbf{x}(1), \dots, \mathbf{x}(L)$ using both the WVD and the STFT, denoted by $\mathbf{D}_{\mathbf{xx}}^{\text{wvd}}(t, f)$ and $\mathbf{D}_{\mathbf{xx}}^{\text{stft}}(t, f)$, respectively. $\mathbf{D}_{\mathbf{xx}}^{\text{wvd}}(t, f)$ is computed in the same manner as that in Procedure 4.1(1). The off-diagonal elements of $\mathbf{D}_{\mathbf{xx}}^{\text{stft}}(t, f)$ are zeros and its diagonal elements are given by

$$\left[\mathbf{D}_{\mathbf{xx}}^{\text{stft}}(l, k) \right]_{ij} = \begin{cases} \sum_{p=-M}^M x(p) h(p-l) e^{-j2\pi p k / L}, & \text{for } i = j, \\ 0, & \text{for } i \neq j, \end{cases} \quad (29)$$

where $M = (L - 1)/2$ and $h(l)$ is a window function. For example, in the case of rectangular type, the window function is defined as

$$h(l) = \begin{cases} \frac{1}{L_w}, & \text{for } |l| \leq \frac{(L_w - 1)}{2}, \\ 0, & \text{otherwise,} \end{cases} \quad (30)$$

where L_w is the window length.

Hence, the MWVD-based STFD matrices are obtained using the following expression:

$$\mathbf{D}_{\mathbf{xx}}^{\text{mwvd}}(t, f) = \mathbf{D}_{\mathbf{xx}}^{\text{wvd}}(t, f) \odot \left| \mathbf{D}_{\mathbf{xx}}^{\text{stft}}(t, f) \right|^2, \quad (31)$$

where \odot denotes the Hadamard product.

To reduce the complexity, among all the TF points in each time slice of the TFD, we keep only those with sufficient energy, according to the point with maximum energy along this time slice, by comparing against a threshold ϵ_1 (typically, $\epsilon_1 = 0.05$). More precisely, along a particular time slice i ,

$$\begin{aligned} \text{if } & \frac{\left\| \mathbf{D}_{\mathbf{xx}}(t_s^{(i)}, f_s) \right\|}{\max_f \left\{ \left\| \mathbf{D}_{\mathbf{xx}}(t^{(i)}, f) \right\| \right\}} > \epsilon_1, \\ \text{then keep point } & (t_s^{(i)}, f_s). \end{aligned} \quad (32)$$

Note that by removing the low-energy points in each time slice, rather than in the entire TF domain as in Procedure 4.1(1), we are able to pick up the points in the starting and ending time-slices.

(2) *Auto-source TF point selection.* This procedure is similar to that in Section 4.1(2) except that we use the MWVD instead of the WVD. Using the MWVD results in a more robust selection of the auto-source points (due to the reduced-interference property of this distribution). Note that, for the choice of the WVD, we have tested the use of both auto-source selection methods, that is, approximation projection and TF disjoint condition as explained in Section 4.1(2), and the results were similar as shown in Figure 5. When choosing the MWVD instead, both of these methods achieved a much better performance as will be shown in Figure 9g.

(3) *Clustering and source TFD estimation.* For each selected auto-source point (t_a, f_a) , we estimate the corresponding spatial direction by

$$\mathbf{a}(t_a, f_a) = \frac{\text{diag} \left\{ \mathbf{D}_{\mathbf{xx}}^{\text{stft}}(t_a, f_a) \right\}}{\left\| \text{diag} \left\{ \mathbf{D}_{\mathbf{xx}}^{\text{stft}}(t_a, f_a) \right\} \right\|}. \quad (33)$$

These vectors are then clustered into different classes using the same clustering procedure in Section 4.1(3). The source TFDs are estimated (up to a scalar constant) by assigning the following:

$$\hat{D}_{s_i}(t, f) = \begin{cases} \text{trace} \left\{ \mathbf{D}_{\mathbf{xx}}^{\text{wvd}}(t_a, f_a) \right\}, & \text{if } (t, f) = (t_a, f_a) \in \mathcal{C}_i, \\ 0, & \text{otherwise.} \end{cases} \quad (34)$$

(4) *Source signal synthesis.* This procedure is the same as Procedure 4.1(4).

5.3. Experiment 2: TF-UBSS using MWVD

In this second experiment, we validate the algorithm using, again, 2 monocomponent LFM signals (Figures 9a, 9b) and 1 multicomponent signal (Figure 9c). Figures 9d, 9e, and 9f, respectively, show the TFDs of the mixtures using the WVD, the MWVD, and the STFT. Obviously, most cross-source points were removed when using the MWVD. The auto-source selection procedure was then applied to further separate the cross-source points from auto-source points (Figures 9g, 9h, and 9i). Next, the direction vectors were found based on the STFD of the STFT as shown in Figure 9i. Then, these vectors were clustered into 3 different TF signatures representing the TFD estimates of the sources (Figures 9j, 9k, and 9l). Obviously, compared to the simulated results in Experiment 1, a much cleaner TF representation of the TFD estimates was obtained using this MWVD-based TF-UBSS algorithm.

As already mentioned in Section 4.1(2), we provide here a simulation test (Figure 5) showing the similar visual performances using two methods, based on either the approximation projection or the TF disjoint assumption, for the auto-source point selection procedure.

6. TF-UBSS WITH COMPONENT EXTRACTION

As an alternative to using the MWVD as proposed in the previous section for enhancing the auto-source point selection procedure, we propose in this section another solution that is based on image processing, by using a *component-extraction procedure*. The underlying idea of this solution comes from the observation that a monocomponent frequency-modulated (FM) signal is represented by a linear feature that corresponds to the ‘energy concentration points’ in the TF image. If we can obtain all the IF components of the sources from the mixture TF image, then for each source we will be able to group its IF components appropriately into its TF signature by using the clustering procedure in Section 4.1(3). Note that, as we have mentioned in Section 2, knowing only the IF components would not allow us to separate the sources since the sources can have several IF monocomponents.

For this component-extraction approach, we must make another assumption, in addition to those in Section 2, as follows: the sources are well localized in the TF domain. Visually, this assumption means that the sources should only show ridges on the TF domain.

6.1. Algorithm

The procedures of the TF-UBSS algorithm with image-processing-based component extraction are shown in Algorithm 2. Note that, Procedure 2 in Algorithm 1 is replaced by Procedure 2* in Algorithm 2.

(1) *STFD computation and noise thresholding.* This procedure is similar to that in Section 5.2(1).

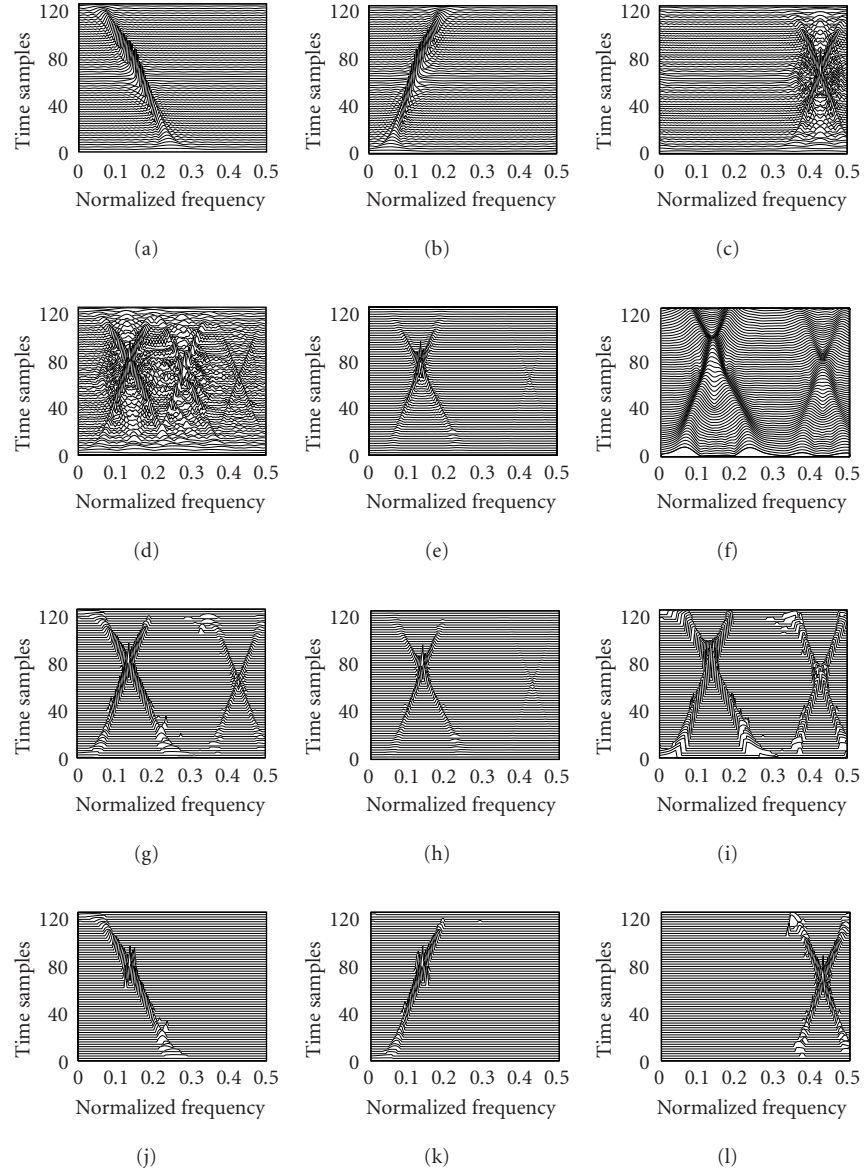


FIGURE 9: Experiment 2: TF-UBSS using MWVD. (a) WVD of $s_1(t)$. (b) WVD of $s_2(t)$. (c) WVD of $s_3(t)$. (d) WVD of mixture. (e) MWVD of mixture. (f) STFT of mixture. (g) Auto-source points displayed by WVD. (h) Auto-source points displayed by MWVD. (i) Auto-source points displayed by STFT. (j) TFD estimate of $s_1(t)$. (k) TFD estimate of $s_2(t)$. (l) TFD estimate of $s_3(t)$.

Proc. 1: STFD computation and noise thresholding Proc. 2*: image component extraction Proc. 3: clustering and source TFD estimation Proc. 4: source signal synthesis

ALGORITHM 2: Image-based TF-UBSS algorithm with component extraction: procedures.

In addition, we apply spatial averaging that mitigates further the cross-source points by a factor depending on their spatial signatures angle [30]. More precisely, we compute the

spatially averaged TFD as in

$$D_{\text{avg}}(t, f) = \text{trace} \{ \mathbf{D}_{\mathbf{xx}}(t, f) \} = \sum_{l=1}^m D_{x_l, x_l}(t, f). \quad (35)$$

The image (i.e., TF support) of this spatially averaged TFD will be used as the input to the image component extraction procedure, which we explain next.

(2) *Image component extraction.* A practical application of satellite image processing is to extract terrestrial roads from satellite images [31]. We apply this so-called “road network tracking” approach to extract the TF component from

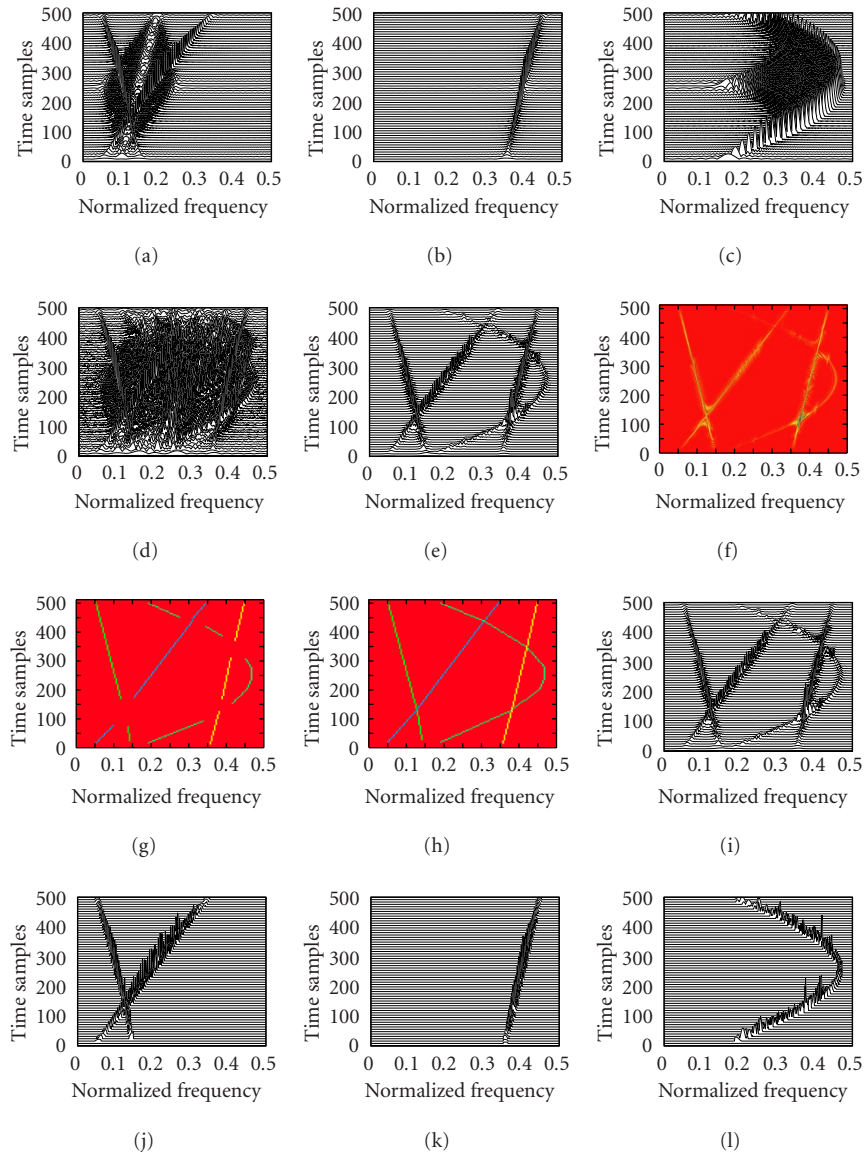


FIGURE 10: Experiment 3: TF-UBSS using component extraction for non-LFM signals. (a), (b), (c) WVD of $s_1(t)$, $s_2(t)$, $s_3(t)$; (d), (e) spatial-averaged TFD of the mixture outputs using WVD and MWVD; (f) convert STFD mixture to image; (g), (h) extraction of source components using image processing; (i) auto-source points of known components; (j), (k), (l) TFD estimates of the sources.

the TF image. This procedure includes three main steps, for which the mathematical details can be found in [31]: (i) preprocessing: because of the particularity of the TF image, a preprocessing is needed before applying the component extraction, (ii) line detection (or local optimization) giving local binary detection of the potential linear structures (segments) in the image, and (iii) road detection (or global optimization) giving a set of labeled segments.

(a) *Preprocessing*. First, the TF image is transformed into a real positive-valued image by forcing to zero all negative values² of the TFD and by using a gray scale in the range

[1, 256]. Also, line detectors are usually limited to a line width of 5 pixels. If the components being searched do not respect this limit (it is usually the case for a TF image), an image subsampling by block averaging is applied to reduce the pixel size. Despite the blurring effect, this filter presents the advantage of reducing the noise in the TF image. Moreover, this subsampling removes the unisotropic representation (containing horizontal lines) of the TF image (see Figures 10e, 10f).

(b) *Line detection (local optimization)*. A line detector is applied at each pixel of the image. We use the detector proposed in [31] for radar image processing. For a given direction, the detector response is based on the ratio of the means computed on both sides of the suspected line and the mean

²Negative values correspond mainly to undesired cross-terms or noise.

of the line itself. Height directions are studied and the best response is kept. The resulting image is then binarized using a simple thresholding. If statistics of the image are available (noise distribution, additive or multiplicative noise, etc.), a statistical study of the line detector performance can be made to choose a more adapted threshold (for instance, the threshold may correspond to a fixed false alarm rate in homogeneous areas).

(c) *Road detection (global optimization)*. This is a global step where constraints on the shape of the linear features are imposed to the global optimization in order to extract connected components and to suppress the false alarms [31]. It works on the already extracted segments by thinning and linearization. The segments are connected depending on proximity and alignment constraints (specially on the line curvature) to form coherent components. Small isolated segments are suppressed. The algorithm depends on the following thresholds: the maximum gap between two segments to connect them, the allowable angular difference between the two segments, and the minimum size of a component. The output from this global optimization is a labeled image of components.

(3) *Clustering and source TFD estimation*. This procedure is similar to that in Section 5.2(3). However, instead of clustering the set of spatial direction vectors corresponding to all the auto-source points, we cluster only the vectors representing the spatial directions of the components, which have been obtained in the previous procedure. The component spatial direction is estimated by averaging over the principal eigenvectors of the STFD matrices of all the points belonging to this component. More precisely, for each extracted component C , one estimates the corresponding spatial direction by the following:

$$\mathbf{a}_C = \frac{1}{\#\mathcal{L}_C} \sum_{i \in \mathcal{L}_C} \mathbf{a}(t_i, f_i), \quad (36)$$

where \mathcal{L}_C denotes the set of points of component C , $\#\mathcal{L}_C$ denotes the number of points in \mathcal{L}_C , and $\mathbf{a}(t_i, f_i)$ is the estimated principal eigenvector of the STFD matrix $\mathbf{D}_{\mathbf{x}\mathbf{x}}(t_i, f_i)$.

(4) *Source signal synthesis*. This is carried out in the same manner as that in Section 4.1(4).

6.2. Experiment 3: TF-UBSS with component extraction

To illustrate the performance of the TF-UBSS algorithm using component extraction, we present here a simulation example corresponding to the separation of $n = 3$ sources using $m = 2$ sensors. The first source $s_1(t)$ is a multicomponent LFM signal, the second is a monocomponent LFM signal, and the third is a quadratic FM signal. The SNR is equal to 10 dB. The simulated results are shown in Figure 10. Although no statistical analysis was provided here due to limited space, we can observe from these results a good performance of the algorithm. A conclusion can be drawn as such: due to the linear features of the TF image, the method using component extraction can give a better

performance regarding the extraction of the TF components that are present in all the underlying sources under the assumption that these components are FM-like signals.

7. NUMERICAL PERFORMANCE EVALUATION

In practice, several common performance criteria can be used for the evaluation of BSS algorithms, such as crosstalk (SNR, signal-to-interference-noise ratio (SINR)), distance to diagonal matrix, rejection level, global index, and mean squared error (MSE) (see [32] for a survey of these criteria). In our work, we apply the MSE criterion that is defined as follows:

$$\varepsilon_x = \frac{1}{N_r} \sum_{k=1}^{N_r} \frac{\|\hat{x}_k - x\|^2}{\|x\|^2}, \quad (37)$$

where N_r is the number of Monte Carlo simulation runs. $N_r = 100$ was used in all the performance simulation. The generic variable x in (37) represents the true value of the measure, whether being mixing matrix \mathbf{A} , IF $f_{in}(t)$, or signal waveform $s_i(t)$. The estimate of x is denoted by \hat{x} .

The experiment was set up with $m = 2$ sensors, having half wavelength spacing, and $n = 3$ sources generated by three monocomponent LFM signals, under the presence of AWGN. All signals have the same length of $L = 128$ samples. The corresponding IFs, in pair of starting and stopping normalized frequencies, were $[0.1, 0.05]$, $[0.33, 0.3]$, and $[0.45, 0.35]$, respectively. The source signals arrived at the sensor array at different angles, 30° , 45° , and 60° .

All the performance evaluations were done using the version of our TF-UBSS algorithm that uses the MWVD (see Section 5) since this version was shown to give better results than the one using the WVD (see Section 4.3). Note that we have corrected the permutation problem, inherent to BSS, in our simulation in order to run the numerical performance analysis. The plots of the source signal waveforms and their corresponding TFDs are shown in Figures 11a, 11b, 11c, 11d, 11e, and 11f. In addition, Figures 11g, 11h, 11i, 11j, 11k, and 11l represent the TFD estimates of the sources and their recovered waveforms.

7.1. On mixing matrix estimation

The first measure to be analyzed is the estimation of the mixing matrix. With the given angles of arrival (i.e., $\theta_1 = 30^\circ$, $\theta_2 = 45^\circ$, and $\theta_3 = 60^\circ$), the true mixing matrix is

$$\begin{aligned} \mathbf{A} &= [\mathbf{a}_1 \quad \mathbf{a}_2 \quad \cdots \quad \mathbf{a}_n] \\ &= \begin{bmatrix} e^{j\pi 0 \sin(\theta_1)} & \cdots & e^{j\pi 0 \sin(\theta_n)} \\ \cdots & \cdots & \cdots \\ e^{j\pi(m-1) \sin(\theta_1)} & \cdots & e^{j\pi(m-1) \sin(\theta_n)} \end{bmatrix} \\ &= \begin{bmatrix} 1 & 1 & 1 \\ 0 + j & -0.61 + 0.80j & -0.91 + 0.41j \end{bmatrix}. \end{aligned} \quad (38)$$

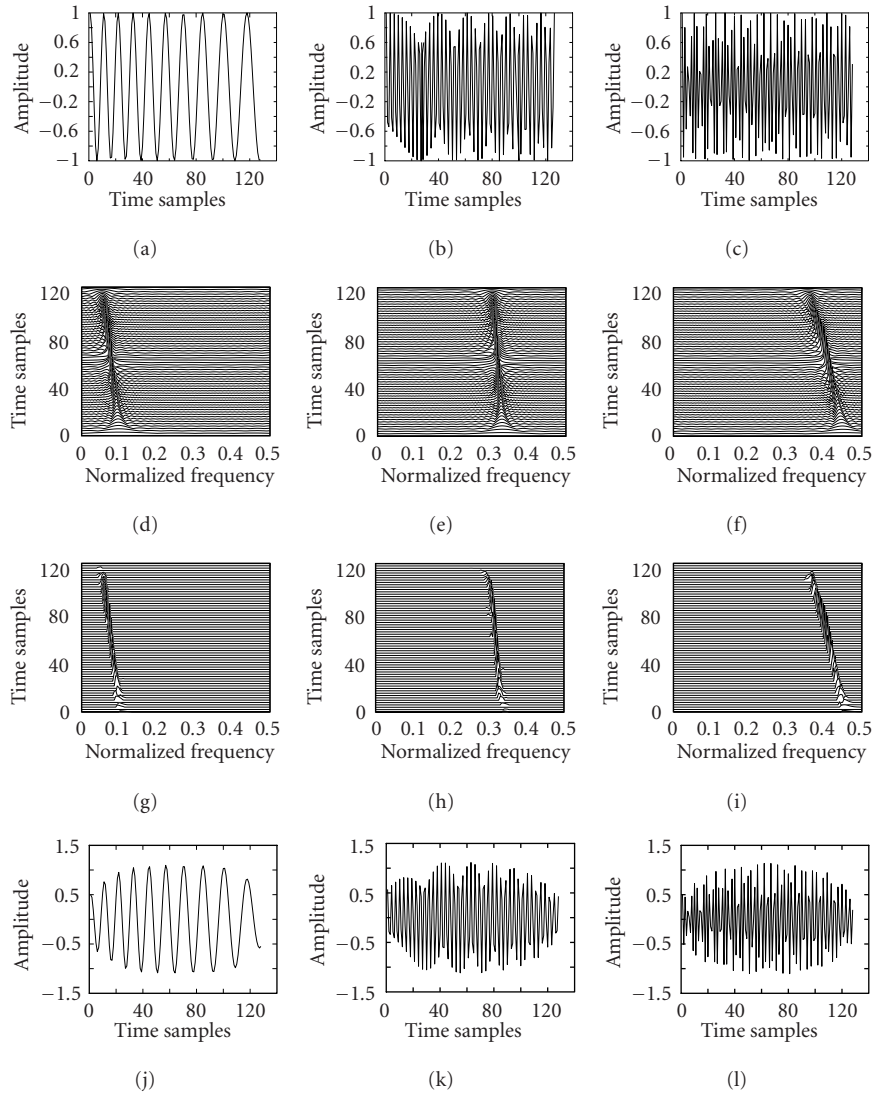


FIGURE 11: Signals used for performance evaluations.

For a particular simulation run, the spatial direction \mathbf{a}_i of the source $s_i(t)$ was estimated as the average of all “closely spaced” spatial directions at the auto-source points belonging to the obtained TF signature of $s_i(t)$. Mathematically, in a similar manner of (36), this writes

$$\hat{\mathbf{a}}_i = \frac{1}{\#\mathcal{L}_i} \sum_{p \in \mathcal{L}_i} \hat{\mathbf{a}}_p(t_p, f_p), \quad (39)$$

where \mathcal{L}_i denotes the set of auto-source points belonging to the TF signature of $s_i(t)$, $\#\mathcal{L}_i$ denotes the number of auto-source points in \mathcal{L}_i , and $\hat{\mathbf{a}}_p(t_p, f_p)$ is the estimated principal eigenvector of the STFD matrix $\mathbf{D}_{xx}(t_p, f_p)$ at the point (t_p, f_p) .

The performance of mixing matrix estimation was evaluated against different values of SNR as shown in Figure 12, showing a good obtained performance. In addition, the plot indicates that the estimation error decreases linearly as SNR increases.

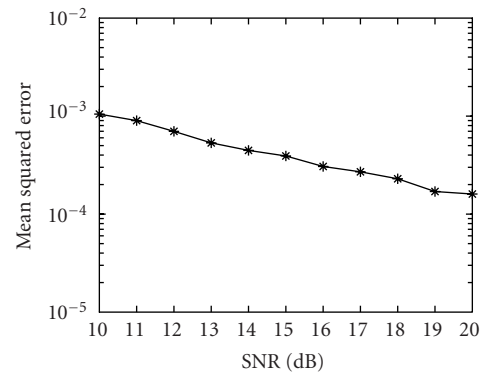


FIGURE 12: Error performance on mixing matrix estimation.

7.2. On auto-source selection

To address the performance on auto-source selection, we may choose to evaluate the performance of mixing matrix

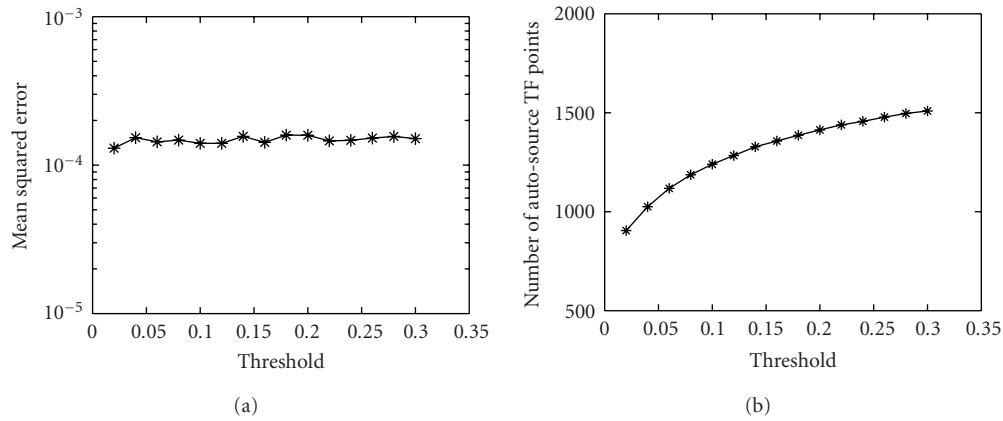


FIGURE 13: Performance on auto-source selection. (a) Estimation error on \mathbf{A} ; (b) number of auto-source points selected (over the total: $128 \times 128 = 16384$ points).

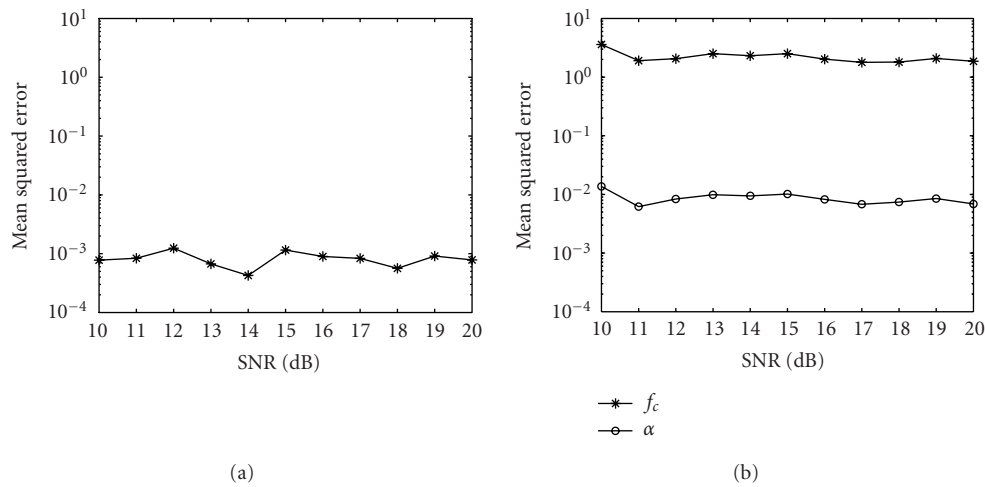


FIGURE 14: Error performance on IF estimation.

estimation as well as the number of selected points with respect to the threshold ϵ_2 (see (16)), while keeping the SNR value at 20 dB. The performance results were plotted in Figure 13. It is observed that the estimation of \mathbf{A} was not sensitive to ϵ_2 (Figure 13a). The number of selected auto-source points (Figure 13b) increased with an increase of ϵ_2 , but approached a constant value of around 1500 (the total number of points is $128 \times 128 = 16384$). Therefore, we may conclude that a typical value for ϵ_2 is 0.3.

7.3. On IF Estimation

The performance of IF estimation was, similarly, evaluated against different values of SNR. As usual, one measures the performance at a particular time instance [11]. We did choose to do the same and evaluated the frequency estimation at the middle time slide of the TF representation. Only the performance for the second LFM signal, with starting and stopping frequency pair of $[0.33, 0.3]$, was shown here

for the purpose of demonstration. The result, illustrated in Figure 14a, shows a good performance. Furthermore, it indicates that the IF estimation was not affected by the AWGN (at least for SNR greater than 10 dB). This comes from the fact that TFD spreads AWGN over the TF domain, and we were measuring over a high range of SNR (10–20 dB).

In addition, since the underlying signals are LFM, we can also use polynomial fitting in our estimation then evaluate the estimated polynomial coefficients. More precisely, using the following form of IF of an LFM signal [11]:

$$f_{\text{in}}(t) = f_c + \alpha t, \quad (40)$$

we measure the estimation errors on the center frequency f_c and the sweeping rate (slope) α , accordingly. The result (Figure 14b) shows that the estimation of the center frequency was very poor compared to that of the sweeping rate. The poor estimation of center frequency was indeed expected; since our underlying signal (second signal) is almost parallel

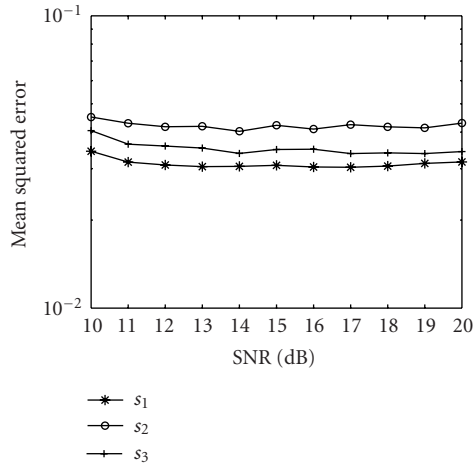


FIGURE 15: Error performance on source waveform estimation.

with the time axis (see Figure 11e), a small error in the sweeping rate caused a large error in the center frequency. Another observation is that the error in the sweeping rate was higher than the error evaluated as in Figure 14a. This was also expected since the points collected at the boundary of the TF representation were normally deviated from the true line of the IF (see Figure 11g, 11h, and 11i), causing some bias through the use of polynomial fitting.

As a conclusion for the performance of IF estimation, the method comparing the true IF and the estimated IF at a time slide gave better indication of the IF estimation, and this estimation was well performed.

7.4. On source waveform estimation

The performance of waveform estimation is shown in Figure 15. It indicates that the estimation was poorer compared to the estimation of mixing matrix and IF. This poorness was due to the boundary effect on the TF representation which has caused the loss in obtaining the TF points around the two ends of the signal (see Figures 11g, 11h, 11i). Hence, the signal waveforms were poorly estimated (see the two ends of estimated signals in Figures 11g, 11h, 11i). The poor performance of source waveform estimation suggests further investigation on the clustering procedure. This was due to the fact that some “small” clusters, which were considered as noisy clusters, were removed during the clustering procedure. A better clustering procedure may be able to correctly assign the points in these small clusters to the appropriate source TF signatures.

8. CONCLUSIONS

In this paper we have presented a new approach for blind separation of nonstationary sources using their TFDs. The proposed TF-UBSS algorithm is based on a vector clustering procedure that estimates the source TFDs by grouping together the TF points corresponding to “closely spaced” spatial directions. Simulation examples were provided to illustrate the approach performance. The work in this paper

represents a new research direction for solving the challenging UBSS problem. Many problems still remain under investigation, including the improvement on the vector clustering procedure, and extension to the convolutive mixture case. We note here that in the course of our study, two other approaches, namely, using the neural network [33] and Gap statistics [34], have been proposed to enhance the clustering procedure from the first version of our TF-UBSS algorithm, which had been first proposed in [35].

REFERENCES

- [1] J.-F. Cardoso, “Blind signal separation: statistical principles,” *Proc. IEEE*, vol. 86, no. 10, pp. 2009–2025, 1998.
- [2] P. Comon, “Independent component analysis, a new concept?” *Signal Processing*, vol. 36, no. 3, pp. 287–314, 1994.
- [3] T.-W. Lee, *Independent Component Analysis: Theory and Applications*, Kluwer Academic, Boston, Mass, USA, 1998.
- [4] V. Zarzoso and A. K. Nandi, “Blind source separation,” in *Blind Estimation Using Higher-Order Statistics*, A. K. Nandi, Ed., Kluwer Academic, Boston, Mass, USA, pp. 167–252, January 1999.
- [5] F. J. Theis, A. Jung, C. G. Puntonet, and E. W. Lang, “Linear geometric ICA: fundamentals and algorithms,” *Neural Computation*, vol. 15, no. 2, pp. 419–439, 2003.
- [6] V. Varadarajan and J. L. Krolik, “Multichannel system identification methods for sensor array calibration in uncertain multipath environments,” in *Proc. 11th IEEE Signal Processing Workshop on Statistical Signal Processing (SSP '01)*, pp. 297–300, Singapore, August 2001.
- [7] K. Abed-Meraim, S. Attallah, T. J. Lim, and M. O. Damen, “A blind interference canceller in DS-CDMA,” in *Proc. IEEE 6th International Symposium on Spread Spectrum Techniques and Applications*, vol. 2, pp. 358–362, Parsippany, NJ, USA, September 2000.
- [8] S. Rickard, R. Balan, and J. Rosca, “Real-time time-frequency based blind source separation,” in *Proc. 3rd International Conference on Independent Component Analysis and Blind Source Separation (ICA '01)*, pp. 651–656, San Diego, Calif, USA, December 2001.
- [9] A. Rouxel, D. Le Guennec, and O. Macchi, “Unsupervised adaptive separation of impulse signals applied to EEG analysis,” in *Proc. IEEE International Conference on Acoustics, Speech, Signal Processing (ICASSP '00)*, vol. 1, pp. 420–423, Istanbul, Turkey, June 2000.
- [10] A. Belouchrani and M. G. Amin, “Blind source separation based on time-frequency signal representations,” *IEEE Trans. Signal Processing*, vol. 46, no. 11, pp. 2888–2897, 1998.
- [11] B. Boashash, Ed., *Time-Frequency Signal Analysis and Processing: A comprehensive Reference*, Elsevier, Oxford, UK, 2003.
- [12] A. Belouchrani, K. Abed-Meraim, M. G. Amin, and A. M. Zoubir, “Joint anti-diagonalization for blind source separation,” in *Proc. IEEE International Conference on Acoustics, Speech, Signal Processing (ICASSP '01)*, vol. 5, pp. 2789–2792, Salt Lake City, Utah, USA, May 2001.
- [13] K. Abed-Meraim, A. Belouchrani, and R. Leyman, “Blind source separation using time frequency distributions,” in *Time-Frequency Signal Analysis and Processing*, Prentice-Hall, Englewood Cliffs, NJ, USA, to appear 2002.
- [14] A. Belouchrani and J.-F. Cardoso, “A maximum likelihood source separation for discrete sources,” in *Proc. European Signal Processing Conference (EUSIPCO '94)*, vol. 2, pp. 768–771, Edinburgh, Scotland, September 1994.
- [15] A. Taleb and C. Jutten, “On underdetermined source separation,” in *Proc. IEEE International Conference on Acoustics,*

- Speech, Signal Processing (ICASSP '99)*, vol. 3, pp. 1445–1448, Phoenix, Ariz, USA, March 1999.
- [16] T.-W. Lee, M. S. Lewicki, M. Girolami, and T. J. Sejnowski, “Blind source separation of more sources than mixtures using over complete representations,” *IEEE Signal Processing Lett.*, vol. 6, no. 4, pp. 87–90, 1999.
- [17] P. Comon and O. Grellier, “Non-linear inversion of underdetermined mixtures,” in *Proc. 1st International Workshop on Independent Component Analysis and Signal Separation (ICA '99)*, pp. 461–465, Aussois, France, January 1999.
- [18] K. I. Diamantaras, “Blind separation of multiple binary sources using a single linear mixture,” in *Proc. IEEE International Conference on Acoustics, Speech, Signal Processing (ICASSP '00)*, vol. 5, pp. 2889–2892, Istanbul, Turkey, June 2000.
- [19] P. Bofill and M. Zibulevsky, “Underdetermined blind source separation using sparse representations,” *Signal Processing*, vol. 81, no. 11, pp. 2353–2362, 2001.
- [20] A. Belouchrani, K. Abed-Meraim, J.-F. Cardoso, and E. Moulines, “A blind source separation technique using second-order statistics,” *IEEE Trans. Signal Processing*, vol. 45, no. 2, pp. 434–444, 1997.
- [21] M. Zibulevsky and B. A. Pearlmutter, “Blind source separation by sparse decomposition in a signal dictionary,” *Neural Computations*, vol. 13, no. 4, pp. 863–882, 2001.
- [22] B. Brakat and K. Abed-Meraim, “Blind source separation using the time-frequency distribution of the mixture signal,” in *Proc. 2nd IEEE International Symposium on Signal Processing and Information Technology (ISSPIT '02)*, vol. 2, pp. 663–666, Marrakech, Morocco, December 2002.
- [23] M. Wax and T. Kailath, “Detection of signals by information theoretic criteria,” *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 33, no. 2, pp. 387–392, 1985.
- [24] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*, Kluwer Academic, Boston, Mass, USA, 1991.
- [25] G. Boudreaux-Bartels and T. Parks, “Time-varying filtering and signal estimation using Wigner distribution synthesis techniques,” *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 34, no. 3, pp. 442–451, 1986.
- [26] N. Delfosse and P. Loubaton, “Adaptive separation of independent sources: a deflation approach,” in *Proc. IEEE International Conference on Acoustics, Speech, Signal Processing (ICASSP '94)*, vol. 4, pp. 41–44, Adelaide, SA, Australia, April 1994.
- [27] F. Hlawatsch and W. Krattenthaler, “Signal synthesis algorithms for bilinear time-frequency signal representations,” in *The Wigner Distribution—Theory and Applications in Signal Processing*, W. Mecklenbräuker and F. Hlawatsch, Eds., pp. 135–209, Elsevier, Amsterdam, The Netherlands, 1997.
- [28] A. Francos and M. Porat, “Analysis and synthesis of multicomponent signals using positive time-frequency distributions,” *IEEE Trans. Signal Processing*, vol. 47, no. 2, pp. 493–504, 1999.
- [29] M. G. Amin, “Interference mitigation in spread spectrum communication systems using time-frequency distributions,” *IEEE Trans. Signal Processing*, vol. 45, no. 1, pp. 90–101, 1997.
- [30] M. G. Amin, W. Mu, and Y. Zhang, “Spatial and time-frequency signature estimation of nonstationary sources,” in *Proc. 11th IEEE Signal Processing Workshop on Statistical Signal Processing (SSP '01)*, pp. 313–316, Singapore, August 2001.
- [31] F. Tupin, H. Maître, J.-F. Mangin, J.-M. Nicolas, and E. Pechersky, “Detection of linear features in SAR images: application to road network extraction,” *IEEE Trans. Geosci. Remote Sensing*, vol. 36, no. 2, pp. 434–453, 1998.
- [32] A. Mansour, M. Kawamoto, and N. Ohnishi, “A survey of performance indexes of ICA algorithms,” in *Proc. 21st IASTED International Conference Modelling, Identification and Control (MIC '02)*, pp. 660–666, Innsbruck, Austria, February 2002.
- [33] Y. Luo and J. A. Chambers, “Active source selection using gap statistics for underdetermined blind source separation,” in *Proc. 7th IEEE International Symposium on Signal Processing and Its Applications (ISSPA '03)*, vol. 1, pp. 137–140, Paris, France, July 2003.
- [34] Y. Luo and J. A. Chambers, “A modified underdetermined blind source separation method using competitive learning,” in *Proc. 3rd IEEE International Symposium on Image and Signal Processing Analysis (ISPA '03)*, pp. 966–969, Rome, Italy, September 2003.
- [35] L.-T. Nguyen, A. Belouchrani, K. Abed-Meraim, and B. Boashash, “Separating more sources than sensors using time-frequency distributions,” in *Proc. 6th IEEE International Symposium on Signal Processing and Its Applications (ISSPA '01)*, vol. 2, pp. 583–586, Kuala Lumpur, Malaysia, August 2001.

Nguyen Linh-Trung received his Engineering degree in electrical engineering in 1997 and the Ph.D. degree in signal processing in 2004, from the Queensland University of Technology (Brisbane, Australia). From 2001 to 2003, he visited the École Nationale Supérieure des Télécommunications (Paris, France) three times to work on the problem of time-frequency-based underdetermined blind source separation. Since September 2003, he has joined the Centre National d'Études Spatiales (Toulouse, France) as a Postdoctoral Research Fellow, where he currently investigates mechanisms for priority access in emergency communications over public satellite networks. His broad research interests are in time-frequency methods, signal processing for communications, and algebraic coding.



Adel Belouchrani received the State Engineering degree in 1991 from École Nationale Polytechnique (ENP), Algiers, Algeria, the M.S. degree in signal processing from the Institut National Polytechnique de Grenoble (INPG), France, in 1992, and the Ph.D. degree in signal and image processing from Télécom Paris (ENST), France, in 1995. He was a Visiting Scholar at the Electrical Engineering and Computer Sciences Department, University of California, Berkeley, from 1995 to 1996. He was with the Department of Electrical and Computer Engineering, Villanova University, Villanova, Pa, as a Research Associate from 1996 to 1997. He also served as a Consultant to Comcast, Inc., Philadelphia, Pa, during the same period. In February 1997, he was a Visiting Scientist at the Laboratory for Artificial Brain System, Riken, Japan. From August 1997 to October 1997, he was with Alcatel ETCA, Belgium. From May to June 2001, he was a Visiting Scholar at the Department of Cognitive Science of the University of California at San Diego, USA. From August to September 2001, he was a Visiting Researcher at Brain Signal Processing Laboratory, Riken, Japan. From July to September 2003, he had a sabbatical stay at the Signal and Image Processing Department of École Nationale Supérieure des Télécommunications (ENST), Paris (France). From December 2003 to January 2004, he was an Invited Professor at Telekom Malaysia RnD, Kuala Lumpur, Malaysia. He is currently and since 1998 with the Electrical Engineering Department of ENP as an Associate Professor. His research interests are in statistical



signal processing and (blind) array signal processing with applications in biomedical and communications, time-frequency analysis, time-frequency array signal processing, and wireless and spread spectrum communications.

Karim Abed-Meraim was born in 1967. He received the State Engineering degree from École Polytechnique, Paris, France, in 1990, the State Engineering degree from École Nationale Supérieure des Télécommunications (ENST), Paris, France, in 1992, the M.S. degree from Paris XI University, Orsay, France, in 1992, and the Ph.D. degree from École Nationale Supérieure des Télécommunications (EN-



ST), Paris, France, in 1995 (in the field of signal processing and communications). From 1995 to 1998, he was a research staff at the Electrical Engineering Department of the University of Melbourne where he worked on several research projects related to “blind system identification for wireless communications,” “blind source separation,” and “array processing for communications,” respectively. He is currently an Associate Professor (since 1998) at the Signal and Image Processing Department of ENST. His research interests are in signal processing for communications and include system identification, multiuser detection, space-time coding, adaptive filtering and tracking, array processing, and performance analysis. Dr. Abed-Meraim is an IEEE Senior Member and an Associate Editor for the IEEE Transactions on Signal Processing.

Boualem Boashash received the Award of Fellow of the IEEE for “pioneering contributions to time-frequency signal analysis and signal processing education.” He got the Diplôme d’ingénieur Physique-Electronique from the ICPI, Lyon, France, in 1978. He received a DEA (Master’s degree) in 1979 and Docteur-Ingénieur in 1982 from the Institut National Polytechnique de Grenoble, France. During the period 1979–1982, he was with Elf-Aquitaine Geophysical Research



Centre, Pau, France. In 1982, he joined the Institut National des Sciences Appliquées de Lyon, Lyon, France, as a Senior Lecturer. In 1984, he joined the EE Department of the University of Queensland, Australia, as a Lecturer, Senior Lecturer (1986), and Reader (1989). In 1990, he joined Bond University as a Professor of signal processing. In 1991, he moved to the Queensland University of Technology as the Foundation Professor of Signal Processing and Foundation Director of the Signal Processing Research Centre. He is currently at the University of Sharjah, College of Engineering, Sharjah, UAE. He has published over four hundred technical publications, three books, 30 book chapters, and 3 textbooks and supervised 42 Ph.D. students. Professor Boashash was the Technical Chairman of ICASSP 94. Since 1985, he has been the General Chairman and then Chairman of Steering Committee of the International Symposium on Signal Processing and its Applications (ISSPA) organized every two years.