EURASIP Journal on Applied Signal Processing 2004:13, 2025–2033

Algorithms for Blind Components Separation and Extraction from the Time-Frequency Distribution of Their Mixture

B. Barkat

School of Electrical and Electronic Engineering, Nanyang Technological University, Nanyang Avenue, 639798 Singapore
Email: ebarkat@ntu.edu.sg

K. Abed-Meraim

Signal and Image Processing Department, École National Supérieure des Télécommunications, Telecom Paris, 75013 Paris, France
Email: abed@tsi.enst.fr

Received 20 February 2003; Revised 29 November 2003; Recommended for Publication by Petar Djurić

We propose novel algorithms to select and extract separately all the components, using the time-frequency distribution (TFD), of a given multicomponent frequency-modulated (FM) signal. These algorithms do not use any a priori information about the various components. However, their performances highly depend on the cross-terms suppression ability and high time-frequency resolution of the considered TFD. To illustrate the usefulness of the proposed algorithms, we applied them for the estimation of the instantaneous frequency coefficients of a multicomponent signal and the results are compared with those of the higher-order ambiguity function (HAF) algorithm. Monte Carlo simulation results show the superiority of the proposed algorithms over the HAF.

Keywords and phrases: time-frequency signal analysis, components separation, polynomial phase signals, instantaneous frequency estimation.

1. INTRODUCTION

The joint time-frequency analysis has proved to be a powerful tool in the analysis of nonstationary signals, that is, signals whose spectral contents vary with time [1]. Such signals may be found in many engineering applications such as radar, sonar, telecommunications, and biomedical engineering. These signals can be classified in two groups: monocomponent and multicomponent.

In this paper, we focus our analysis on multicomponent signals. By a multicomponent signal, we mean a signal whose time-frequency representation presents multiple ridges in the time-frequency plane. Analytically, it may be defined as

$$s(t) = \sum_{i=1}^{M} s_i(t), \tag{1}$$

where each component $s_i(t)$, of the form

$$s_i(t) = a_i(t)e^{j\phi_i(t)}, \tag{2}$$

is assumed to have only one ridge, or one continuous curve, in the time-frequency plane. An example of a multicomponent signal, consisting of three components, is displayed in Figure 1.

Recovery of a particular component from a given multicomponent signal has always been a challenge for the time-frequency community. The objective of this paper is to address this particular problem. Specifically, we present two different algorithms in order to retrieve and extract separately the components from the time-frequency distribution (TFD) of their mixture signal. The motivation behind this can be found in situations where the user may be interested in the instantaneous frequency (IF) law of one of the components only. For instance, in telecommunications the received signal may be a mixture of several source signals (multiple access interference) but the user may wish to recover only one source signal (blind source separation) [2, 3]. In this context, by applying either of the proposed algorithms to the TFD of the received signal, we may be able to separate and recover the desired source signal.

The algorithms proposed here do not use any a priori information about the various components to be extracted. However, the first algorithm assumes that all components of the signal exist at the almost all time instants; while, the second algorithm assumes that all components are well separated in the time-frequency plane. Moreover, it is necessary that the used TFD, in addition to its high time-frequency

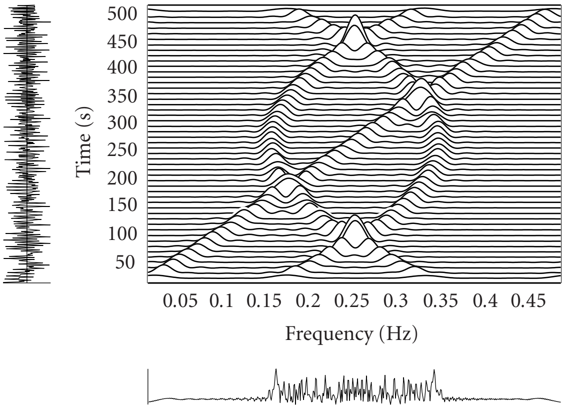

Figure 1: A time-frequency distribution of a multicomponent signal. $F_s = 1$ Hz, $N = 512$, Time resolution $= 1$.

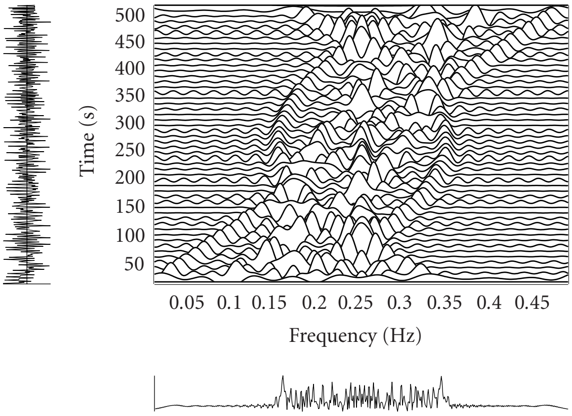

Figure 2: The WVD of the same multicomponent signal displayed in Figure 1. $F_s = 1$ Hz, $N = 512$, Time resolution $= 1$.

resolution, should be cross-terms free or at least be able to suppress them as much as possible.

Once the various components have been extracted, we can use available estimation techniques to obtain their desired characteristics [4]. In the literature, we can find other techniques for the estimation of multicomponent signals in noise [5, 6, 7]. Among these we can cite the higher-order ambiguity function (HAF) algorithm [7]. Explicitly, the algorithm in [7] was designed to estimate the phase parameters as well as the constant, or slowly varying, amplitudes of each component of a multicomponent signal. Each of these components is assumed to have a polynomial phase law. As an illustration, we present here a brief statistical performance comparison between one of the proposed algorithms and the HAF in the estimation of a multicomponent signal consisting of two quadratic polynomial phase signals embedded in noise. We note that our proposed algorithms can also be used in the estimation of other nonlinear, not necessarily polynomial, phase signals. Examples, using real-life as well as synthetic data, are presented in order to show the high accuracy of the proposed algorithms.

The paper is organized as follows. In Section 2, we discuss the choice of the appropriate TFD to be used in both algorithms. In Section 3, we present the first algorithm as well as the statistical comparison with the HAF algorithm. In Section 4, we present the second algorithm. Section 5 concludes the paper.

2. TIME-FREQUENCY DISTRIBUTION CHOICE

There exist many TFDs. The choice of a TFD depends on the specific application at hand and the representation properties that are desirable for this application. One of the well-known TFDs is the Wigner-Ville distribution (WVD) defined as [1]

$$W(t,f) = \int_{-\infty}^{+\infty} z\left(t+\frac{\tau}{2}\right)\cdot z^*\left(t-\frac{\tau}{2}\right)e^{-j2\pi f\tau}d\tau, \quad (3)$$

where $z(t)$ is the analytic version of the signal under consideration.

The WVD is known to have high resolution in both time and frequency; however, it suffers from the presence of cross-terms for a multicomponent signal. These cross-terms result from the interaction of different components of the signal. As an illustration, we consider the WVD of the multicomponent signal displayed in Figure 1. The WVD of such a signal is displayed in Figure 2. It is clear from this figure that the features of the signal are hidden making the WVD inappropriate for the analysis in this case.

In order to apply the proposed algorithms we need to have a "clean" TFD. That is, we need a distribution that can reveal the features of the signal as clearly as possible without any "ghost" component. For that, we need to apply a TFD that can get rid of the cross-terms while preserving a high time-frequency resolution. Thanks to the recent results in the design of TFDs, nowadays the user has a myriad of TFDs to choose from [8, 9, 10, 11]. As an example, in the sequel, we will use a newly developed high-resolution quadratic TFD. This distribution, called the B-distribution, is defined as [12]

$$S(t,f) = \iint_{-\infty}^{+\infty}\left(\frac{|\tau|}{\cosh(t')}\right)^{\sigma}\cdot\left[z\left(t-t'+\frac{\tau}{2}\right)\cdot z^*\left(t-t'-\frac{\tau}{2}\right)\right] \cdot e^{-j2\pi f\tau}dt'd\tau, \quad (4)$$

where $0 \le \sigma \le 1$ is a real parameter. The choice of the B-distribution, or its modified version [13], stems from the fact that it presents a good performance in terms of resolution and cross-terms suppression. Detailed performance evaluation, design criteria, and implementation can be found in [12, 13]. In Figure 1, it was this particular distribution that was used to display the time-frequency representation of the signal.

In the next sections, we will present the two proposed algorithms to select and extract a particular component (of a given multicomponent signal) using the B-distribution. However, we should stress here that any other clean, with high resolution, TFD can also be used. For instance, in [14] we used the S-distribution [10] to successfully extract the various components of the multicomponent signal.

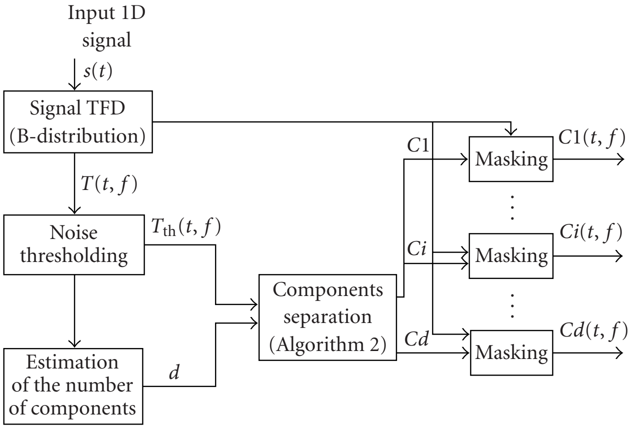

Figure 3: Flowchart of the proposed first algorithm.

3. PROPOSED FIRST ALGORITHM

The first proposed components-separation algorithm is illustrated in Figure 3, and Algorithms 1 and 2. Figure 3 provides the algorithm flowchart, Algorithm 1 summarizes the estimation technique of the number of components, and Algorithm 2 summarizes the components-separation technique.

The first step of the algorithm consists in noise thresholding to remove the undesired "low" energy peaks in the time-frequency domain[1]. This operation can be written as

$$T_{\text{th}}(t, f) = \begin{cases} T(t, f) & \text{if } T(t, f) > \epsilon, \\ 0 & \text{otherwise,} \end{cases} \tag{5}$$

where ϵ is a properly chosen threshold (in our simulations we used $\epsilon = 0.01 \max_{(t,f)} T(t, f)$).

The second step consists in estimating the number of components as shown next.

3.1. Estimation of the number of components

First, we assume that all components exist simultaneously at almost all time instants in the time-frequency plane. Second, we observe that, in general, for a noiseless and cross-terms free TFD, the number of components at a given time instant t_0 can be estimated as the number of peaks of the TFD slice $T(t_0, f)$. By searching and counting the peaks of each TFD slice, we end up with a set of numbers. The number corresponding to the maximum of the histogram of these numbers yields an estimate of the number of components in the signal. This simple procedure is detailed in Algorithm 1.

Note that the thresholding operation performed in the first step has an effect on the second step. Indeed, the TFD should present high peaks for the auto-terms compared to cross-terms and noise. In this situation, the threshold can easily remove all peaks that do not belong to auto-terms. However, in large noise situations the choice of the threshold value becomes more difficult and this may generate errors in the number of components.

```
(1) For each time instant t, where $t = 1, \dots, t_{\max}$, take a slice
    of the TFD $T(t, f)$.
(2) Search and count the number of peaks in each slice.
(3) Evaluate the histogram of the obtained set of peaks
    numbers.
(4) Estimate the total number of the signal components as
    the argument of the maximum of the above histogram.
```

Algorithm 1: Estimation of the number of components.

```
(1) Assign an index to each of the d components in an
    orderly manner.
(2) For each time instant t (starting from $t = 1$) find the
    components frequencies as the peaks positions of the
    TFD slice $T(t, f)$.
(3) Assign a peak to a particular component based on the
    smallest distance to the peaks of the previous slice
    $T(t - 1, f)$ (IFs continuous functions of time). For the
    special case of a crossing point (see step (4) how to
    detect it and its corresponding components), we assign
    the peak to both crossing components.
(4) If at a time instant t a crossing point exists (i.e., number
    of peaks smaller than the number of components), identify
    the crossing components using the smallest distance
    criterion by comparing the distances of the actual peaks
    to those of the previous slice.
(5) Permute the indices of the corresponding crossing
    components.
```

Algorithm 2: Components-separation procedure for the proposed first algorithm.

3.2. Components-separation procedure

The proposed algorithm assumes that (i) all components exist at all time instants in the time-frequency plane and (ii) any components intersection is a crossing point. Under these two assumptions, we note that if, at a time instant t_0, two components are crossing, then the number of peaks (at this particular slice $T(t_0, f)$) is smaller than the total number of components d. For practical implementation reasons, we decide that a crossing occurs when the number of peaks is smaller than d over a fixed number of consecutive slices. In this case, we implement the following procedure:

(1) choose a particular maximum point location in the slice where the crossing occurs;

(2) measure all distances from this point to the peaks locations of the previous slice (with no crossing);

(3) select the 2 smallest distances and add them;

(4) repeat steps (1) to (3) for all other maximum point locations in the slice where the crossing occurred;

[1] This noise thresholding is justified by the fact that the noise energy is spread over all time-frequency domain while the components energies are well localized around their respective IFs leading to high energy peaks for the latter (assuming no cross-terms).

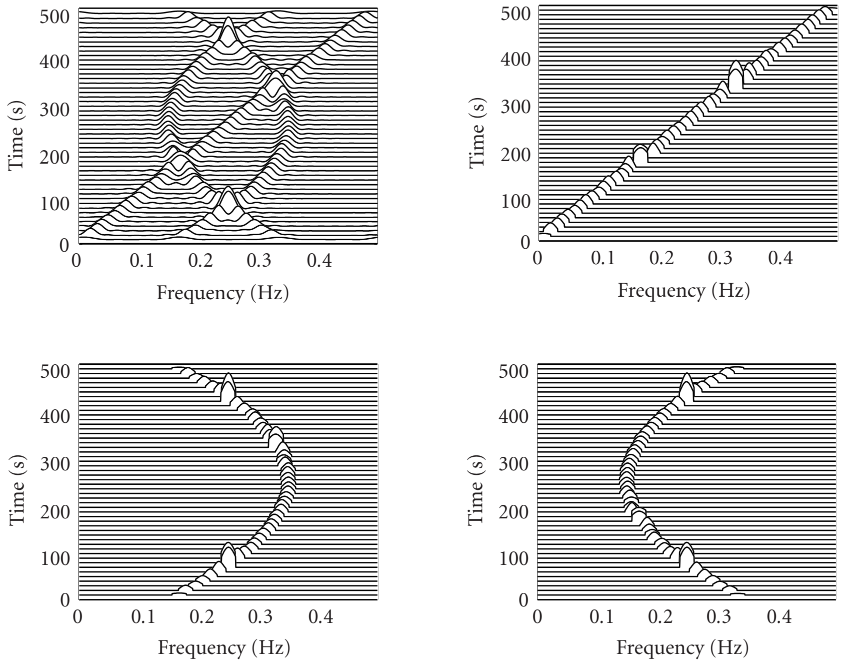

Figure 4: The B-distribution of the original signal (top left) as well as the extracted components using the proposed first algorithm.

(5) from the set of the smallest sums found above, the program selects the smallest value and the points associated to them. This will yield the location where the crossing occurred and the 2 components involved in the crossing.

Then, we use a simple numerical permutation operation of the 2 components involved in the crossing. The details of the proposed separation technique is outlined in Algorithm 2.

To validate the proposed algorithm, we reconsider the same multicomponent signal analysed earlier. This signal consists of a mixture of a unit modulus (with increasing frequency) quadratic frequency-modulated (FM) component, a unit modulus (with decreasing frequency) quadratic FM component, and a unit modulus (with increasing frequency) linear FM component. The mixture signal is added to a zero-mean white Gaussian noise with power equal to 0 dB. This means that the individual signal-to-noise ratio (SNR) defined as $\mathrm{SNR}_i = i$th component power/noise power is equal to 0 dB.

The B-distribution of the noisy signal as well as the components resulting from the separation algorithm are displayed in Figure 4.

A different signal consisting of 5 components was also analysed using the proposed algorithm. In particular, this signal is a mixture of two linear FM signals, a quadratic FM signal, a cubic FM signal, and a pure sinusoid. The mixture signal was embedded in 0 dB Gaussian noise. Similarly to the previous case, the individual SNR is also equal to 0 dB. Again, the algorithm was able to separate and extract each of these components. The results are displayed in Figure 5.

Note that a similar algorithm to the one above could be designed if the signal exists over all frequencies but not necessarily over all times. In this case, the slices are taken at particular frequencies and not time instants as we did here.

3.3. Performance evaluation and comparison

In this subsection, we evaluate the statistical performance of the proposed first algorithm and compare it to the performance of the HAF method [7]. For that, consider a discrete-time multicomponent signal consisting of two linear FM components embedded in additive white complex Gaussian noise $w(n)$:

$$y(n) = z_1 + z_2 + w(n), \quad n = 0, 1, \ldots, N-1, \tag{6}$$

where $z_1 = \exp\{j(a_1 n + a_2 n^2)\}$ and $z_2 = \exp\{j(b_1 n + b_2 n^2)\}$. The noise $w(n)$ is assumed to be an independent and identically distributed (i.i.d.) sequence with zero mean and variance equal to σ^2. The signals' IF coefficients are given by $a_1 = 0.4\pi$, $a_2 = 0.5\pi 10^{-3}$, $b_1 = 0.9\pi$, and $b_2 = -1.5\pi 10^{-3}$. The signal length is chosen equal to $N = 256$ with a sampling period equal to unity. We define the SNR as the total noiseless signal power over the noise power, namely,

$$\mathrm{SNR\,(dB)} = 10\log_{10}\left(\frac{|z_1|^2 + |z_2|^2}{\sigma^2}\right). \tag{7}$$

For a given SNR value, we put the noisy signal $y(n)$ through the proposed algorithm in order to extract the two respective components. The peaks of the extracted components (in the time-frequency domain) are then used to estimate the IFs of

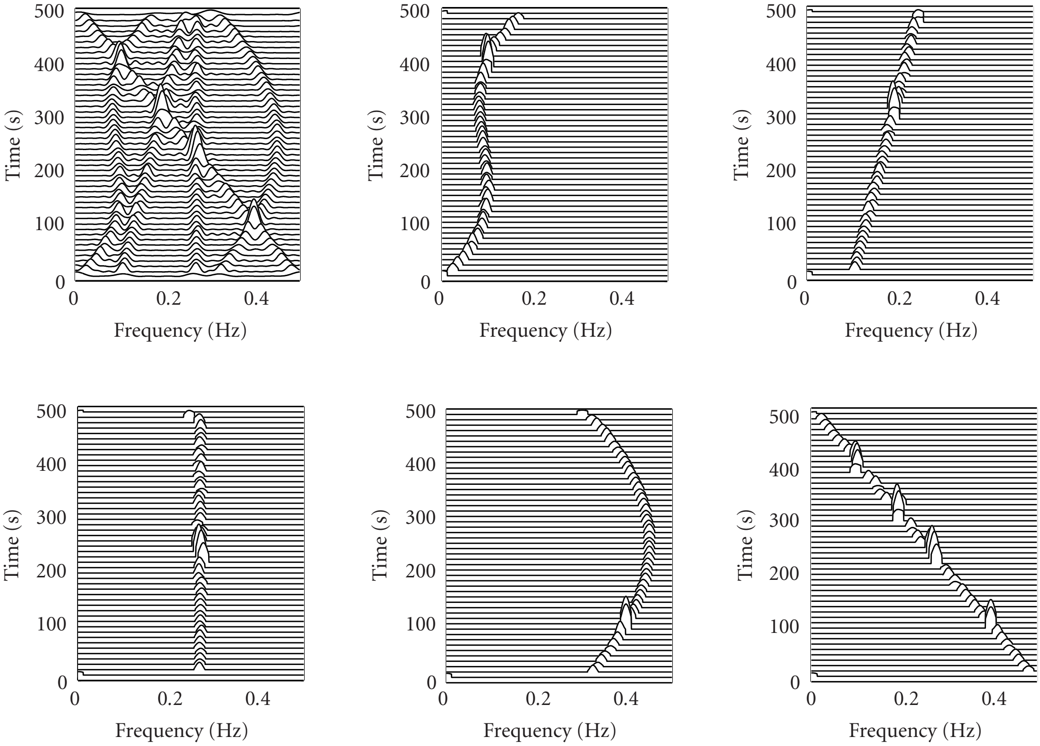

Figure 5: The B-distribution of a different multicomponent signal (top left) as well as the extracted components using the proposed first algorithm.

these linear FM components [4]. By recalling that the IF of $z_1(n)$ (estimated from the peak of the extracted component) is given by [4]:

$$f_{z_1}(n) = \frac{1}{2\pi} \cdot (a_1 + 2a_2 \cdot n), \quad n = 0, \dots, N-1, \tag{8}$$

and that of $z_2(n)$ (estimated from the peak of the other extracted component) is given by

$$f_{z_2}(n) = \frac{1}{2\pi} \cdot (b_1 + 2b_2 \cdot n), \quad n = 0, \dots, N-1, \tag{9}$$

we use a simple polynomial fit to obtain estimates of (a_1, a_2) from $f_{z_1}(n)$ and estimates of (b_1, b_2) from $f_{z_2}(n)$.

For comparison purposes, the same noisy signal $y(n)$ is also put through the HAF algorithm [7]. From this algorithm, we directly obtain the IF coefficients estimates [7]. These estimates are then used to evaluate the corresponding IFs estimates of the two linear FM components (using the above expressions). We note here that, in the comparison, we choose the coefficients to be half of those of [7] to contain the frequency in the range 0–0.5 Hz instead of the 0–1 Hz. Moreover, in the simulation, we used a second estimation stage as suggested in [7] to refine the phase parameter estimates.

In Figure 6, we display the estimated IFs of the two components. The dotted lines correspond to the HAF algorithm and the dashed lines correspond to the proposed first algorithm. The true IFs are represented by the continuous lines (superimposed with those of the proposed first algorithm). The superiority of the proposed algorithm over the HAF is obvious. In this particular example, the SNR was fixed equal to 0 dB.

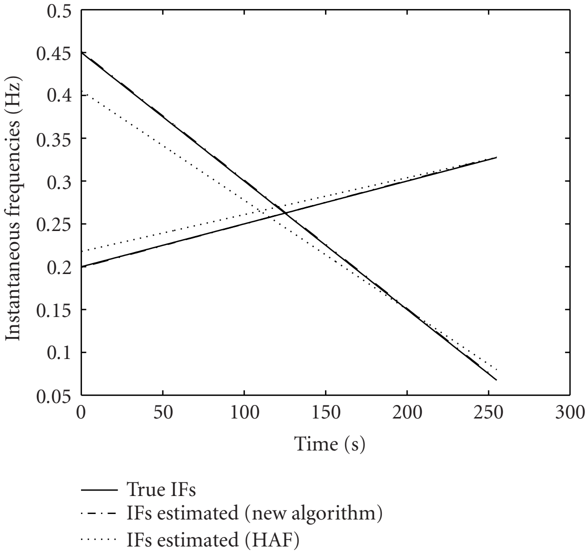

Figure 6: Estimated IFs of the two linear FM components. The dotted lines correspond to the HAF algorithm and the dashed-dotted lines correspond to the proposed first algorithm.

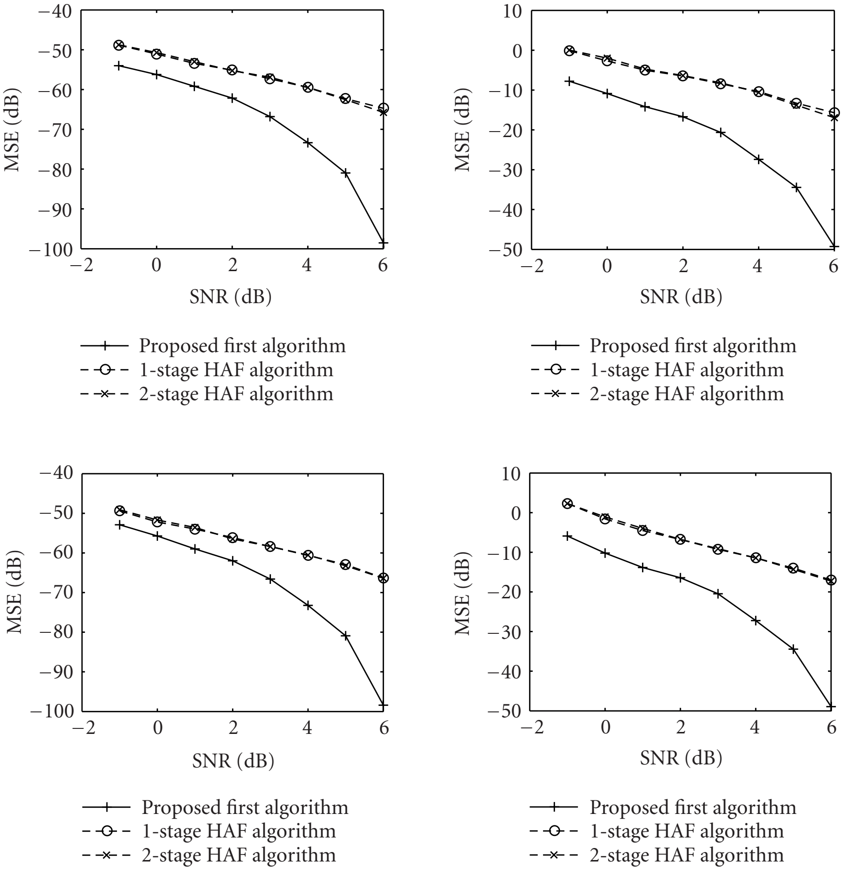

Figure 7: Mean squared error of the various phase parameters.

We re-ran the above experiment for various values of the SNR. For each SNR value, we ran 6000 realizations. The results of the Monte Carlo simulations, namely, the mean squared error of the phase parameters are displayed in Figure 7. The "∘" curves (resp., the "×" curves) correspond to the 1-stage (resp., 2-stage[2]) HAF algorithm; while, the "+" curves correspond to the proposed algorithm. These results confirm the superiority of the proposed first algorithm over the HAF.

4. PROPOSED SECOND ALGORITHM

In this second algorithm, the various components are extracted sequentially. That is, the algorithm extracts the first component (or part of it), then the next one, and so on until the last one. Normally, the overall energy in the TFD becomes smaller and smaller after each extraction and after the last component has been retrieved the energy should be a fraction of the original one. It is the energy criterion that stops the extraction algorithm. Then, a classification procedure is applied, as explained later.

The proposed second algorithm is illustrated in Figure 8. As can be seen, the second algorithm consists of three major phases. The first phase is to analyze the mixture, or multicomponent, signal using an appropriate TFD. By appropriate, we mean a cross-terms reduced TFD. In the sequel, we will consider the B-distribution but any other clean TFD can also be a candidate.

The second phase is the separation procedure. In this phase, the various components are extracted based on their peaks in the time-frequency plane. That is, the frequency and time occurrence of the highest peak are obtained first. Then, we look for the next highest peak in the *nearest neighborhood* of the previous found one (making sure to reset to zero, and some frequency range around it, the previous peak in order to avoid it again). We continue this until we reach the extreme end of the TFD or when the new obtained peak is smaller than a prefixed threshold (chosen to be equal to a fraction of the *first maximum*). The consecutive found peaks would constitute the first component. We repeat the procedure again to obtain a new component and so on until the remaining energy in the TFD matrix is smaller than a fraction of the initial TFD energy.

In general, the TFD is not maximum at its extremities. And since our proposed procedure starts at the maximum, it will consequently follow a component pattern from the

[2] As can be observed, for low and moderate SNRs, the performance gain due to the second stage of the HAF algorithm is not significant.

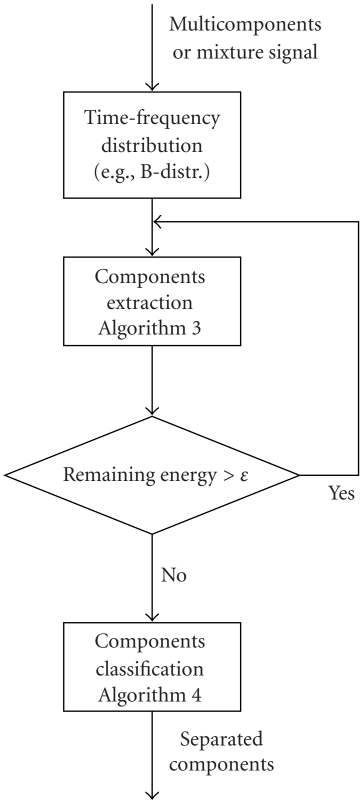

Figure 8: The flowchart of the proposed second algorithm.

maximum location to one end. This will constitute only one part of the component. The other part of the component will be taken in a different step of the iterative algorithm. For this reason, at the end of the second phase, we end up with a number of components which is higher than the actual number of components in the signal. Therefore, a classification procedure is necessary in order to group the halves (or parts) of the actual components together. This is performed in the third and last phase of the algorithm. Algorithm 3 gives the details of the second phase.

The classification technique (detailed in Algorithm 4) consists of grouping the components obtained from the second phase based on an appropriate measurement criterion. This criterion is chosen to be the minimum distance between two components. Indeed, if two components belong to the same actual component, their distance in the time-frequency plane should be smaller compared to any other obtained component. By applying the classification procedure once, we can group a certain number of the components and the resulting new number of components will be smaller than the one obtained from the second phase. We continue applying this classification until there is *no change* in the number of components. This last number corresponds to the actual number of components in the original mixture signal.

As an illustration, we consider the analysis of a real-life data sound emitted by a bat. The B-distribution of this multicomponent signal, which consists of three components, is displayed in Figure 9 (top left plot). Note that although there

```
(1) Initialization. Create an empty matrix called component to hold the results (its first row will hold the time and its second row will hold the corresponding frequency of the extracted component).
(2) Find the maximum energy point, $\begin{bmatrix} t_0 \\ f_0 \end{bmatrix}$, of the time-frequency distribution.
(3) Augment the matrix component by adding the point
    $\begin{bmatrix} t_0 \\ f_0 \end{bmatrix}$ as its first column.
(4) Set the TFD matrix $T(t_0, f)$ to zero, at time $t_0$, around the found maximum point, that is, $T(t_0, f) = 0$ for $f \in [f_0 - \Delta f, f_0 + \Delta f]$.
(5) Find the next maximum energy point, $\begin{bmatrix} t_0' \\ f_0' \end{bmatrix}$, of the TFD in the vicinity of the previous maximum. That is,
    $\begin{bmatrix} t_0' \\ f_0' \end{bmatrix} = \max_{(t,f)} T(t, f) \quad \text{where } t \in [t_0 - 1, t_0 + 1]$
    and $f \in [f_0 - F, f_0 + F]$,
    where $F$ is a chosen frequency window parameter.
(6) Augment the matrix component by adding the point
    $\begin{bmatrix} t_0' \\ f_0' \end{bmatrix}$ as its next column.
(7) Again, set the TFD to zero at time $t_0'$, around the found maximum, that is, $T(t_0', f) = 0$ for $f \in [f_0' - \Delta f, f_0' + \Delta f]$.
(8) As long as the time and frequency indices have not reached the boundaries of the TFD matrix and the TFD in the neighborhood of $\begin{bmatrix} t_0' \\ f_0' \end{bmatrix}$, defined in step (5), is not equal to zero, then, go back to step (5).
(9) Otherwise, go back to step (1) to extract a new component.
(10) Stop the algorithm when the remaining TFD energy is smaller than a threshold $\epsilon$.
```

Algorithm 3: Components-separation procedure for the second algorithm.

```
(1) Initialization. Set the number of components equal to that found in the extraction procedure of Algorithm 3.
(2) Do the following:
    (2.1) For all pairs of components $(C_i, C_j)$, compute the distance $d_{ij}$ between the two components;
    (2.2) If the distance between any pair of components verifies $d_{ij} < \epsilon_d$, then, merge the two components $(C_i, C_j)$, decrease by one the number of compo nents, and go back to step (2.1);
    (2.3) If all distances $d_{ij}$ are larger than $\epsilon_d$, then, stop the algorithm.
```

Algorithm 4: Classification procedure in the proposed second algorithm.

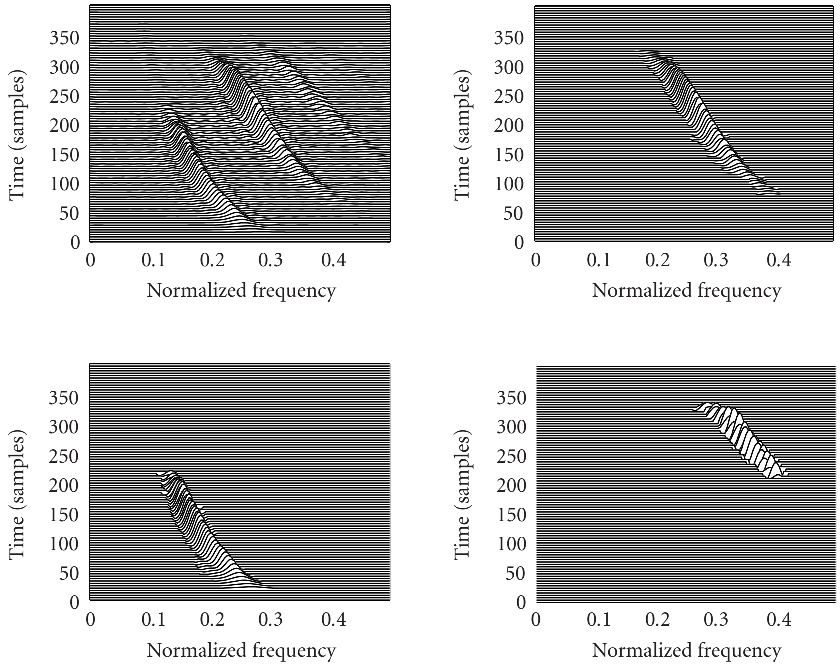

Figure 9: The B-distribution of a bat signal (top left) as well as the extracted components using the proposed second algorithm.

is an overlap of the various components either in time or in frequency, they are well separated in the time-frequency domain. Applying the proposed second algorithm, we are able to extract each of these components separately, as shown in Figure 9.

5. CONCLUSION

In this paper, we presented two novel blind (i.e., without a priori information) algorithms to extract separately all the components, using a "cross-terms free" TFD, of a given mixture signal. The first algorithm assumes that the components exist at almost all time instants; while, the second one assumes that the components are well separated in the time-frequency plane. Such components extraction can be used, for example, as a preprocessing step to estimate the polynomial phase parameters of a multicomponent FM signal. Examples, using real-life as well as synthetic data, were presented in order to validate the new algorithms. In addition, the first algorithm was compared with the HAF algorithm for the estimation of the IF coefficients of a multicomponent signal consisting of two linear FM components. Monte Carlo simulations showed the superiority of the proposed algorithm over the HAF.

REFERENCES

[1] L. Cohen, *Time-Frequency Analysis*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1995.

[2] M. G. Amin, W. Mu, and Y. Zhang, "Spatial and time-frequency signature estimation of nonstationary sources," in *Proc. IEEE 11th Workshop on Statistical Signal Processing*, pp. 313–316, Singapore, August 2001.

[3] L.-T. Nguyen, A. Belouchrani, K. Abed-Meraim, and B. Boashash, "Separating more sources than sensors using time-frequency distributions," in *Proc. IEEE 6th International Symposium on Signal Processing and Its Applications*, vol. 2, pp. 583–586, Kuala Lumpur, Malaysia, August 2001.

[4] B. Barkat and B. Boashash, "Instantaneous frequency estimation of polynomial FM signals using the peak of the PWVD: statistical performance in the presence of additive Gaussian noise," *IEEE Trans. Signal Processing*, vol. 47, no. 9, pp. 2480–2490, 1999.

[5] S. Barbarossa and V. Petrone, "Analysis of polynomial-phase signals by the integrated generalized ambiguity function," *IEEE Trans. Signal Processing*, vol. 45, no. 2, pp. 316–327, 1997.

[6] 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.

[7] S. Peleg and B. Friedlander, "Multicomponent signal analysis using the polynomial-phase transform," *IEEE Trans. on Aerospace and Electronics Systems*, vol. 32, no. 1, pp. 378–387, 1996.

[8] M. G. Amin and W. J. Williams, "High spectral resolution time-frequency distribution kernels," *IEEE Trans. Signal Processing*, vol. 46, no. 10, pp. 2796–2804, 1998.

[9] H.-I. Choi and W. J. Williams, "Improved time-frequency representation of multicomponent signals using exponential kernels," *IEEE Trans. Acoustics, Speech, and Signal Processing*, vol. 37, no. 6, pp. 862–871, 1989.

[10] L. Stanković, "S-class of time-frequency distributions," *IEE Proceedings Vision, Image and Signal Processing*, vol. 144, no. 2, pp. 57–64, 1997.

[11] L. Stanković, "On the realization of the polynomial Wigner-Ville distribution for multicomponent signals," *IEEE Signal Processing Letters*, vol. 5, no. 7, pp. 157–159, 1998.

[12] B. Barkat and B. Boashash, "A high-resolution quadratic time-frequency distribution for multicomponent signals analysis,"

IEEE Trans. Signal Processing, vol. 49, no. 10, pp. 2232–2239, 2001.

[13] B. Boashash and V. Sucic, "Resolution measure criteria for the objective assessment of the performance of quadratic time-frequency distributions," *IEEE Trans. Signal Processing*, vol. 51, no. 5, pp. 1253–1263, 2003.

[14] B. Barkat and K. Abed-Meraim, "Blind source separation using the time-frequency distribution of the mixture signal," in *Proc. IEEE 2nd International Symposium on Signal Processing and Information Technology*, pp. 663–666, Marrakesh, Morocco, December 2002.

B. Barkat received the degree of Ingenieur d'État in electronics from the École Nationale Polytechnique d'Alger (ENPA) in 1985 and the M.S. degree in control systems from the University of Colorado, Boulder, USA, in 1988. From 1989 to 1995, he held a Lecturer position in digital and advanced control systems at the University of Blida, Algeria. In 1999, he obtained the Ph.D. degree in signal processing from Queensland University of Technology (QUT), Brisbane, Australia. From September 1999 to November 2000 he was a Postdoctoral Research Fellow, first at QUT and then at Curtin University, Western Australia. Since November 2000, Barkat has been an Assistant Professor in the School of Electrical and Electronic Engineering at the Nanyang Technological University, Singapore. His research interests include time-frequency analysis, estimation and detection, statistical array processing, and signal processing for communications.

K. Abed-Meraim was born in 1967. He received the State Engineering degree from Ecole 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 the École Nationale Supérieure des Télécommunications (ENST), Paris, France, in 1995 (in the field of signal processing and communications). From 1995 to 1998, he has been on the research staff of the Electrical Engineering Department at 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. He is currently Associate Professor (since 1998) at the Signal and Image Processing Department at 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 Member and an Associate Editor for the IEEE Transactions on Signal Processing.