UNDERDETERMINED BLIND SOURCE SEPARATION BASED ON CONTINUOUS DENSITY HIDDEN MARKOV MODELS

Xiaoming Zhu and Keshab K. Parhi, Fellow, IEEE

University of Minnesota, Twin Cities
Department of Electrical and Computer Engineering, 200 Union St SE, Minneapolis, MN, 55455
Email: {xmzhu, parhi}@umn.edu

ABSTRACT
In this paper, a novel method is developed to solve the problem of underdetermined blind source separation, where the number of mixtures is smaller than that of sources. Generalized Gaussian Distributions (GGDs) are used to model the source signals and generative Continuous Density Hidden Markov Models (CDHMMs) are derived to track the nonstationarity inside the source signals. Each source signal can switch between several states such that the separation performance can be significantly improved. The model parameters are trained through the Expectation Maximization (EM) algorithm and the source signals are estimated via the Maximum a Posteriori (MAP) approach. Compared with the results of $L_1$-norm solution, our proposed algorithm has obtained much better output signal-to-noise ratio (SNR) and the separation results are more realistic.

Index terms—Underdetermined blind source separation, Hidden Markov Model, Generalized Gaussian Distribution, nonstationary signals.

1. INTRODUCTION

Blind source separation (BSS) is a topic that has received much attention recently in many applications [1]. The problem of BSS can be defined as follows. Given $M$-dimension mixing vector $x_t = [x_1(t), \ldots, x_M(t)]^T$ and $N$-dimension source vector $s_t = [s_1(t), \ldots, s_N(t)]^T$,

$$x_t = As_t + n_t$$  \hspace{1cm} (1)

where $A$ is an $M \times N$ full rank mixing matrix and $n_t = [n_1(t), \ldots, n_M(t)]^T$ is the additive Gaussian white noise with zero mean.

Underdetermined BSS ($M < N$) is generally more difficult to solve than determined BSS ($M \geq N$) because infinite solutions exist in the former case. Let $A^\dagger = A^T (AA^T)^{-1}$ be the Moore-Penrose pseudo inverse and $A^\perp$ be the null space of of matrix $A$. The complete solutions can be written as

$$\hat{s}_t = A^\dagger x_t + A^\perp v_t$$  \hspace{1cm} (2)

where $v_t$ is any vector of size $N - M$ spanning the null space of $A$. As the solution is not unique, the source signals must be estimated from the infinite solutions based on some constraints even with the knowledge of mixing matrix.

Generally, a two-step approach was suggested to solve the underdetermined BSS [2]. First, the mixing matrix can be estimated by several methods such as time-frequency domain clustering [3] and potential function based method [4]. Second, the source signals are estimated by maximizing a posteriori (MAP) probability of $s$, which is given as

$$\hat{s}^* = \arg \max_s P(s|x, A) = \arg \max_s P(s)$$  \hspace{1cm} (3)

where $\hat{s}^*$ is the most probable decomposition of the mixtures. The second equality in (3) follows the Baye’s rule and the fact that $x$ is fully determined by $s$ and $A$. If the source signal is modeled by Laplace distribution, equation (3) is exactly minimum $L_1$-norm solution [5] [6], or equivalently shortest path decomposition [3].

Fast computation for minimum $L_1$-norm solution is that the decomposition of mixtures only uses limited combination of column vectors in matrix $A$ [5] [7]. Generally, a two-step approach was suggested to solve the underdetermined BSS [2]. First, the mixing matrix can be estimated by several methods such as time-frequency domain clustering [3] and potential function based method [4]. Second, the source signals are estimated by maximizing a posteriori (MAP) probability of $s$, which is given as

$$\hat{s}^* = \arg \max_s P(s|x, A) = \arg \max_s P(s)$$  \hspace{1cm} (3)

where $\hat{s}^*$ is the most probable decomposition of the mixtures. The second equality in (3) follows the Baye’s rule and the fact that $x$ is fully determined by $s$ and $A$. If the source signal is modeled by Laplace distribution, equation (3) is exactly minimum $L_1$-norm solution [5] [6], or equivalently shortest path decomposition [3].

The main drawback of minimum $L_1$-norm solution is that the decomposition of mixtures only uses limited combination of column vectors in matrix $A$ [5] [7]. The main drawback of minimum $L_1$-norm solution is that the decomposition of mixtures only uses limited combination of column vectors in matrix $A$ [5] [7]. Fast computation for minimum $L_1$-norm solution is available in [7]. If the source signal is modeled by Gaussian distribution, equation (3) is minimum $L_2$-norm solution, whose performance is not quite satisfactory [8].

In this paper, we only focus on the second step to examine how well the source signals can be estimated based on the knowledge of mixing matrix $A$. We propose to use Generalized Gaussian Distribution (GGD) to model each source signal. The parameters of GGD can be adaptively changed by applying Continuous Density Hidden Markov Model (CDHMM) to each source signal. All the parameters are estimated by EM algorithm and the source signals are recovered by MAP algorithm. The experimental results show that our proposed method has much better separation results than $L_1$-norm solution. Our method can even deal with the situation where some source signals appear or disappear abruptly, while $L_1$-norm solution may not.
2. PROPOSED ALGORITHM

2.1. Generalized Gaussian Distribution

We propose to use Generalized Gaussian Distributions (GGDs) to model the source signals, which can be represented as

\[ P(s) = \prod_{i=1}^{N} \frac{r_i b_i}{2 \Gamma(1/r_i)} \exp\left(\frac{-r_i s_i^r}{b_i}\right) \]  \hspace{1cm} (4)

where \( \Gamma(\cdot) \) is the gamma function, \( r_i \) is the shape parameter, \( b_i \) is the scaling factor and \( \psi_i \triangleq \{r_i, b_i\}, i = 1, \ldots, N \). Substituting equation (4) into (3), we have the following optimization equation

\[ s_t^* = \arg\min_s \sum_{i=1}^{N} \{(b_i \psi_i)^{r_i} r_i \} \quad \text{arg min} \ \|s_t\|_p^p \]  \hspace{1cm} (5)

where \( r = [r_1, \ldots, r_N]^T \) and \( b = [b_1, \ldots, b_N]^T \). The solution becomes minimum weighted \( L_p \)-norm solution which is newly defined in the above equation. Also, we assume that each source may have \( K \) states and each state is characterized by parameter set \( \psi_{i,k} = \{r_{i,k}, b_{i,k}\} \), \( k = 1, \ldots, K \). The source signals can transit between these states such that the nonstationarity can be well modeled. To learn the states transition and model parameter set, we need to apply Hidden Markov Model to each source signal and use EM algorithm to estimate the model parameter set.

2.2. Continuous Density Hidden Markov Model

In this section, EM algorithm is derived to estimate the model parameter set and MAP approach is used to recover the source signals. The proposed algorithm iterates between model estimation and source recovery until it is converged.

2.2.1. Estimate Model Parameters

Each source signal has a corresponding CDHMM, the parameters of which need to be estimated individually. Without loss of generality, the following description assumes that the parameters of source \( s_i \) is estimated. The subscript \( i \) denotes the \( i \)th source is omitted for simplicity. The \( i \) and \( j \) that occur in the rest of this subsection represents the index of states.

Define the state space \( \mathcal{S} = \{S_1, \ldots, S_K\} \), observation sequence \( O = [s_1, \ldots, s_T]^T \) and hidden state sequence \( Q = [q_1, \ldots, q_T]^T \), where \( s_t \) is the estimated source signal at time \( t \). The model parameters are defined as follows

\[ \pi = [\pi_1, \ldots, \pi_K]^T \]  \hspace{1cm} (6)

\[ P = [p_{ij}]_{K \times K} = [P(q_j|q_i)]_{K \times K} \]  \hspace{1cm} (7)

\[ \psi = \{(r_1, b_1), \ldots, (r_K, b_K)\} \]  \hspace{1cm} (8)

\[ \lambda = \{\pi, P, \psi\} \]  \hspace{1cm} (9)

where \( \pi \) is the initial state probability vector, \( P \) is the transition matrix, \( \psi \) is the model parameter set and \( \lambda \) is the concatenated parameter set. The conditional joint probability of observation sequence and hidden state sequence is given by

\[ P(O, Q|\lambda) = \pi_1 \prod_{t=1}^{T-1} \pi_{q_t q_{t+1}} \prod_{t=1}^{T} P(s_t|q_t, \lambda) \]  \hspace{1cm} (10)

where \( \lambda \) is the new model parameter set to be estimated. By Jensen’s inequality, the above equation can be maximized by optimizing the following auxiliary function [9] defined as

\[ J(\lambda, \lambda) = \sum_{q} P(O|Q, \lambda) \log[P(O|Q, \lambda)] \]  \hspace{1cm} (11)

\[ = J_1(\pi, \lambda) + J_2(P, \lambda) + J_3(\psi, \lambda) \]  \hspace{1cm} (12)

The first two equations in (13) can be optimized by scaled forward-backward algorithm proposed in [10]. In this paper, we only optimize the last equation in (13). Define \( \gamma_{t}(i) = P(q_t = S_i|O, \lambda) \) the probability of being in state \( S_i \) at time \( t \), given the observation sequence \( O \) and the model \( \lambda \). Substituting (4) into the last equation of (13) and using the definition of \( \gamma_{t}(i) \), we get

\[ J_3(\psi, \lambda) = \sum_{j=1}^{K} \sum_{t=1}^{T} \{ P(O|q_t) \psi_i \} \log P(s_t|q_t, \lambda) \]  \hspace{1cm} (13)

We can use gradient method to adaptively estimate the parameter set \( \{r_k, b_k\} \) for state \( S_k \). The partial derivatives with respect to \( r_k \) and \( b_k \) are then given by:

\[ \frac{\partial J_3(\psi, \lambda)}{\partial r_k} = \sum_{i=1}^{K} \sum_{1 \leq q_i = S_k} \gamma_{t}(i) \{1/r_k + \phi(1/r_k)/r_k^2 \} \]  \hspace{1cm} (15)

\[ -\ln(b_k | s_t) (b_k | s_t)^{r_k} \}

where \( \phi(\cdot) = \Gamma'(\cdot)/\Gamma(\cdot) \) is the digamma function. Therefore, all parameters of CDHMM have been estimated for source \( s_i \). For other source signals, the similar procedure is applied.

2.2.2. Scaled Viterbi Algorithm & States Grouping

Now that all the parameters of CDHMM are estimated, scaled Viterbi algorithm can be used to find the best state
sequence for each source signal such that the parameter set \( \psi_i = \{ r_i,k, b_i,k \}, i = 1, \ldots, N, k = 1, \ldots, K \) can be well chosen at time \( t \) to increase performance. The details of scaled Viterbi algorithm can be found in [10]. Generally speaking, the source signal will stay in one state for a period before transiting into another state. The path tracking in Viterbi algorithm does not take this into consideration. Thus we need to group the states by the following criterion such that the state sequence looks more natural.

**Criterion 2.1** Given a rectangular window \( \omega_{t,L} \) of length \( L \) centered at time \( t \), if more than \( \rho L \) (0 \( \leq \) \( \rho \leq 1 \)) signals within that window are in state \( S_i \), then the period of \( L \) is in state \( S_i \).

Note: If the signals in window \( \omega_{t,L} \) do not satisfy criterion 2.1, the next window \( \omega_{t+1,L} \) can be overlapped with \( \omega_{t,L} \). Otherwise, the next window \( \omega_{t+1,L} \) should not be overlapped with \( \omega_{t,L} \) to ensure the states grouped in the previous window will not affect the next window.

### 2.2.3. Recovery of the Source Signals

Given all the model parameters and state sequences, the source signals can be recovered by optimizing equation (5) under the constraint that \( A \hat{s}_t = x_t \). The constrained optimization problem can also be solved by the unconstrained optimization problem through equation (2) as follows. Notice that in (2), the constraint that \( A \hat{s}_t = x_t \) is automatically satisfied.

Let \( \hat{s}_t = A^1 x_t + A^2 \hat{v}_t \) and \( J = || \hat{s}_t ||^2 \) defined in (5), then the unconstrained optimization problem can be expressed as

\[
\hat{v}_t^* = \arg \min_{v_t} J \quad \text{and} \quad \hat{s}_t^* = A^1 x_t + A^2 \hat{v}_t^* .
\]

By gradient method, we have the following update equations:

\[
v_t(n+1) = v_t(n) - \mu \frac{\partial J}{\partial v_t} = v_t(n) - \mu (A^1)^T u_t \quad (17)
\]

where \( u_t = [ r_1 b_1^T | \hat{s}_{1,t} | r_1^{-2} \hat{s}_{1,t} | r_{N-1} b_{N-1}^T | \hat{s}_{N,t} | r_{N-2} \hat{s}_{N,t} ]^T \) and \( \{ r_i, b_i \}, i = 1 \ldots, N \), is the model parameter set for source \( s_i \), chosen from one of the states according to the state sequence derived from scaled Viterbi algorithm. Compared with constraint optimization, the number of variables in unconstrained optimization is reduced from \( N \) to \( N - M \), which reduces the computational complexity.

### 3. NUMERICAL EVALUATIONS

In this experiment, we use four segments of speech, 3 seconds in duration, recorded by a sampling frequency of 16kHz. Source 1 and source 3 are female voices and source 2 and source 4 are male voices. The mixing matrix is given by

\[
A = \begin{bmatrix}
\cos 30^\circ & \cos 60^\circ & \cos 110^\circ & \cos 160^\circ \\
\sin 30^\circ & \sin 60^\circ & \sin 110^\circ & \sin 160^\circ
\end{bmatrix} . \quad (18)
\]

For each source signal, we use a two-state Hidden Markov Model to track the nonstationarity inside the signal. We use the output SNR defined as below to compare the separation results between the proposed algorithm and \( L_1 \)-norm optimization,

\[
\text{SNR}_{O,i} = 10 \log \frac{E[|s_i|^2]}{E[|s_i - \hat{s}_i|^2]} , \quad i = 1, \ldots, N . \quad (19)
\]

The initial parameters are given as follows:

\[
\begin{align*}
\hat{r}_1 & = [ r_{1,1}, \ldots, r_{N,1} ]^T = [1, \ldots, 1]^T \quad (20) \\
\hat{r}_2 & = [ r_{1,2}, \ldots, r_{N,2} ]^T = [1.5, \ldots, 1.5]^T \quad (21) \\
\hat{b}_1 & = [ b_{1,1}, \ldots, b_{N,1} ]^T = [1, \ldots, 1]^T \quad (22) \\
\hat{b}_2 & = [ b_{1,2}, \ldots, r_{N,2} ]^T = [5, \ldots, 5]^T \quad (23) \\
L & = 320, \ \rho = 0.8 . \quad (24)
\end{align*}
\]

First, we will compare the performance of two algorithms under a noise-free mixing environment. Next, we will compare the performance of the proposed algorithm with \( L_1 \)-norm optimization in the situation where source 1 appears and source 4 disappears abruptly to verify that our proposed algorithm can well handle this situation.

Table 1 summarizes the output SNR between the proposed algorithm and \( L_1 \)-norm optimization. The last row in the table means that the HMMs are pre-trained by the original source signals instead of the estimated source signals, which gives us an idea of the upbound that the proposed method can achieve. We can see that the performance of our proposed algorithm is quite close to the upper bound. As the HMMs in our proposed algorithm are trained by estimated source signals, the performance loss is inevitable in this case. The minimum output SNR gain from \( L_1 \)-norm optimization to our proposed algorithm is 1.65dB and the maximum is 2.9dB. Fig. 1 shows the separation results of source 1 and source 4 in the time domain. The separated results verify that our proposed algorithm achieves much better results than traditional \( L_1 \)-norm optimization and the nonstationarity inside the source signals is well preserved.

<table>
<thead>
<tr>
<th>Methods (dB)</th>
<th>Source 1</th>
<th>Source 2</th>
<th>Source 3</th>
<th>Source 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>( L_1 )-norm</td>
<td>3.0018</td>
<td>3.0139</td>
<td>4.1813</td>
<td>4.1709</td>
</tr>
<tr>
<td>Proposed</td>
<td>4.6509</td>
<td>5.1166</td>
<td>7.0832</td>
<td>6.6058</td>
</tr>
<tr>
<td>Optimal HMM</td>
<td>4.9781</td>
<td>5.6080</td>
<td>7.7170</td>
<td>7.0516</td>
</tr>
</tbody>
</table>

Next, we examine an interesting situation where source 1 appears abruptly at \( t = T/2 \) and source 4 disappears abruptly at \( t = T/2 \). In this case, source 1 and source 4 are even more nonstationary than in the previous case. This happens so frequently in the real environment that we may not even notice. The same setup is used as in the noise-free example.

Fig. 2 shows the separation results of source 1 and source 4 in the time domain. Table 2 summarizes the output SNR of
Fig. 1. The separation result of source 1 and 4 in the time domain. Top row: original sources. Middle row: estimated sources by the proposed algorithm. Bottom row: estimated sources by $L_1$-norm optimization.

these two algorithms. We can see that the proposed algorithm achieve even better improvement than the first noise-free experiment. This is because by using HMM, the nonstationarity can be tracked and the GGD parameters can be adaptively adjusted accordingly. Therefore, the silent period of source 1 and source 4 can be captured so that the output SNR can be increased significantly.

Fig. 2. The separation result of source 1 and source 4 in the time domain. Top row: original source signals. Middle row: estimated signals by proposed algorithm. Bottom row: estimated signals by $L_1$-norm optimization.

Table 2. Comparison of output SNR between $L_1$-norm optimization and the proposed algorithm.

<table>
<thead>
<tr>
<th>Methods (dB)</th>
<th>Source 1</th>
<th>Source 2</th>
<th>Source 3</th>
<th>Source 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>$L_1$-norm</td>
<td>3.8427</td>
<td>5.8183</td>
<td>6.7102</td>
<td>3.3661</td>
</tr>
<tr>
<td>Proposed</td>
<td>5.9804</td>
<td>8.4353</td>
<td>11.5217</td>
<td>10.4649</td>
</tr>
</tbody>
</table>

4. CONCLUSION

This paper presents a novel algorithm for underdetermined blind source separation based on Continuous Density Hidden Markov Model. The source signals are modeled by Generalized Gaussian Distributions, whose parameters are adaptively estimated by EM algorithm. MAP approach is used to recover the source signals. As the nonstationarity can be tracked by CDHMM and the parameters of GGD are estimated iteratively, our proposed method has achieved significant improvement compared with traditional $L_1$-norm optimization method. Experimental results also show that the proposed algorithm can well handle the case when some source signals appear or disappear abruptly, which $L_1$-norm optimization cannot.

5. REFERENCES


