



# Fast and Simple Iterative Algorithm of Lp-norm Minimization for Under-determined Speech Separation

Yasuharu Hirasawa, Naoki Yasuraoka, Toru Takahashi, Tetsuya Ogata, Hiroshi G. Okuno

Graduate School of Informatics, Kyoto University, Kyoto, Japan  
 {hirasawa, yasuraok, tall, ogata, okuno}@kuis.kyoto-u.ac.jp

## Abstract

This paper presents an efficient algorithm to solve Lp-norm minimization problem for under-determined speech separation; that is, for the case that there are more sound sources than microphones. We employ an auxiliary function method in order to derive update rules under the assumption that the amplitude of each sound source follows generalized Gaussian distribution. Experiments reveal that our method solves the L1-norm minimization problem ten times faster than a general solver, and also solves Lp-norm minimization problem efficiently, especially when the parameter p is small; when p is not more than 0.7, it runs in real-time without loss of separation quality.

**Index Terms:** speech separation, under-determined condition, Lp-norm minimization, auxiliary function method

## 1. Introduction

Since conventional speech recognition systems do not work properly when microphones are distant from a talker, distant-talking speech recognition is now attracting a lot of attention. It is expected to play a vital role in human-computer interaction. The reasons for low recognition accuracy are, for example, interference from other talkers, background noise, and sound reflection off walls.

One approach to realize a robust recognition in these situations is to pre-process the input sound mixtures using a sound separation technique. Since our daily environment potentially includes an infinite number of sound sources, these systems frequently face an *under-determined* condition, which means that there are more sound sources than microphones. For example, with a common stereo recording system, only three sound sources are enough to cause an under-determined condition. To develop a robust recognition system, dealing with the under-determined condition is essential and inevitable. In this paper, we focus on under-determined speech separation.

Under-determined source separation methods are roughly categorized into three groups. The first group does not estimate the mixing matrix explicitly; for example, Sawada's clustering and permutation method [1]. The second group simultaneously estimates the mixing matrix and separation results; for example, Ozerov's method models a sound source using non-negative matrix factorization [2]. The last group estimates separation results using a given or pre-estimated mixing matrix, such as Bofill's L1-norm minimization method [3].

We focus on the Lp-norm minimization method [4], which is an extension of L1-norm minimization. Since we use information of the mixing matrix, we do not need to assume that each time-frequency region is dominated by only one sound source, while most methods in the first group assume this. In addition, since Lp-norm minimization can be solved independently

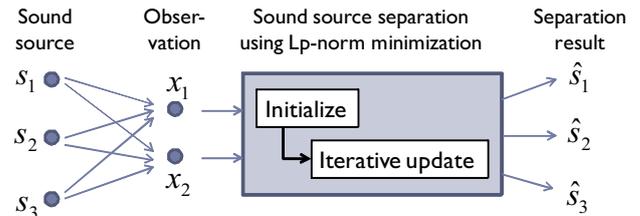


Figure 1: Under-determined sound source separation

in each time-frequency region, we can use parallel computation very easily, unlike the methods in the second group. On the other hand, the independence of time-frequency regions means that we need to solve many small problems in a short period. Since conventional approaches are too slow to achieve real-time separation, many researchers use an approximate method, which does not have theoretical validity.

In this paper, we present a fast iterative algorithm to solve the Lp-norm minimization. Update rules are derived using the auxiliary function method [5]. This makes our update rules general and simple; we can derive update rules for various prior distributions, and we can implement our algorithm quickly.

We conduct experiments to confirm the advantage of our algorithm. Our method is evaluated from the viewpoint of separation speed and separation quality compared with the conventional solver and the mask-based approximation method.

## 2. Under-determined speech separation using Lp-norm minimization

### 2.1. Problem setting

In this subsection, the general problem setting of under-determined speech separation is presented. The input is  $I$  sound mixtures of  $J$  simultaneous utterances ( $I < J$ ), and the output is the estimated speech signal of  $J$  talkers. The mixing process is usually assumed linear and time invariant; that is, the mixing process can be written as

$$\mathbf{x}_{fn} = \mathbf{A}_f \mathbf{s}_{fn}, \quad (1)$$

where  $\mathbf{x}_{fn} \in \mathbb{C}^I$  and  $\mathbf{s}_{fn} \in \mathbb{C}^J$  are observations and original sounds of the  $f$ -th frequency bin and  $n$ -th time frame, and  $\mathbf{A}_f \in \mathbb{C}^{I \times J}$  is the mixing matrix of the  $f$ -th frequency bin.

### 2.2. Difficulty of under-determined separation

First, we consider the *determined* condition; that is, the number of sound sources are equal to that of the microphones ( $I = J$ ). The basic approach of sound separation in the determined condition is to estimate the time-invariant separation

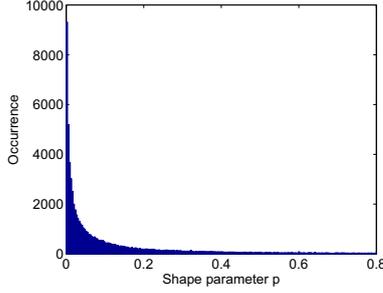


Figure 2: Histogram of the amplitude of speech signals

matrix  $\mathbf{W}_f \in \mathbb{C}^{J \times I}$  and separate the sound mixtures using the following formula:

$$\hat{\mathbf{s}}_{fn} = \mathbf{W}_f \mathbf{x}_{fn}. \quad (2)$$

For example, beamformer and independent component analysis (ICA) use this approach.

When we consider the under-determined condition, we cannot separate a sound mixture using the time-invariant separation matrix. Since the separation matrix  $\mathbf{W}_f$  is a  $J \times I$  matrix, columns must be linear dependent in the under-determined condition ( $I < J$ ). When a separation matrix is linear dependent, its output sound is also linear dependent, and this means that each separation result has a strong correlation even though the original sound sources are independent of each other.

This is a big difference between determined and under-determined separation, and this is why we need to use other methods to separate under-determined sound mixtures. For simplicity of notation,  $f$  and  $n$  are omitted in the rest of this paper since we separate each time-frequency region independently.

### 2.3. Lp-norm minimization

The Lp-norm minimization is derived from the assumption that the amplitude of each sound source follows a zero-mean generalized Gaussian distribution, which is written as

$$p(s; \alpha, p) = \frac{\alpha p}{2\Gamma(p-1)} \exp(-\alpha^p |s|^p), \quad (3)$$

where  $\alpha$  is an inversed scale parameter and  $p$  is a shape parameter. Figure 2 shows an amplitude histogram of speech signals. Since the distribution of the amplitude of each sound source is super-Gaussian, we regard  $p$  ranges from 0 to 2, exclusively. This method also assumes that the mixing matrix  $\mathbf{A}$  is known. Please note that we do not need to know the true mixing matrix and can use the estimated mixing matrix.

From the prior distribution shown in Eq. (3), the logarithm of the joint prior distribution can be written as

$$\log p(\mathbf{s}; \boldsymbol{\alpha}, \mathbf{p}) = - \sum_{j=1}^J \alpha_j^{p_j} |s_j|^{p_j} + C, \quad (4)$$

where  $C$  is a constant value. After observation, since we can use mixing equation (1), we can calculate the logarithm of the posterior distribution as

$$\log p(\mathbf{s}|\mathbf{x}; \boldsymbol{\alpha}, \mathbf{p}) = \begin{cases} - \sum_{j=1}^J \alpha_j^{p_j} |s_j|^{p_j} + C' & (\mathbf{x} = \mathbf{A}\mathbf{s}) \\ -\infty & (\text{otherwise}) \end{cases} \quad (5)$$

where  $C'$  is another constant value.

Equation (5) indicates that the maximum a posterior estimation is reduced to the constrained weighted Lp-norm minimization problem. Since objective variable  $s_j$  is complex-valued, this problem is classified into the second-order cone programming problem (SOCP) [6] even when  $p = 1$ . General SOCP solvers are too slow to separate sound mixtures in real-time and are very complicated to implement. In the next section, we derive simple update rules when  $p$  ranges from 0 to 2 by using the auxiliary function method.

## 3. Derivation of update rules

### 3.1. Auxiliary function method

Before we derive the update rules, we explain the auxiliary function method. The auxiliary function method is used for the minimization (or maximization) problem and derives update rules, which monotonically decrease the original cost. The basic idea of this method is to introduce auxiliary function  $Q^+(\theta, \phi)$ , whose lower bound is the same as that of the original cost function  $Q(\theta)$ , and derive update rules over the auxiliary function. Note that  $\phi$  is called an auxiliary variable.

More formally, the auxiliary function must satisfy the following property:

1.  $Q(\theta) = \underset{\phi}{\operatorname{argmin}} Q^+(\theta, \phi)$ .

In addition, the following properties should be satisfied in order to use the auxiliary function method:

2.  $\phi_{new} = \underset{\phi}{\operatorname{argmin}} Q^+(\theta, \phi)$  is analytically solvable
3.  $\theta_{new} = \underset{\theta}{\operatorname{argmin}} Q^+(\theta, \phi)$  is analytically solvable

Using these properties, we can update original parameters  $\theta$  with two steps:

- update  $\phi$  using property 2, and
- update  $\theta$  using property 3.

Now we can prove that these two steps decrease original cost. After the first step,  $Q(\theta) = Q^+(\theta, \phi_{new})$  from property 1. After the second step,  $Q^+(\theta_{new}, \phi_{new}) \leq Q^+(\theta, \phi_{new})$  from property 2. After that,  $Q(\theta_{new}) \leq Q^+(\theta_{new}, \phi_{new})$  from property 1 again. Now we have  $Q(\theta_{new}) \leq Q^+(\theta_{new}, \phi_{new}) \leq Q^+(\theta, \phi_{new}) = Q(\theta)$ . This means that we can ensure that the cost will converge to a local minimum.

### 3.2. Derivation of update rules

As Eq. (5) suggests, we want to minimize the cost function

$$Q(\mathbf{s}) = \sum_{j=1}^J \alpha_j^{p_j} |s_j|^{p_j} \quad (6)$$

under the constraint of mixing equation

$$\mathbf{x}_i = \sum_{j=1}^J a_{ij} s_j, \quad (7)$$

which comes from Eq. (1). Since  $p_j$  ranges from 0 to 2, we can use a quadratic function as the auxiliary function (Fig. 3). This idea was given by Kameoka to add the sparsity prior for Complex NMF [7].

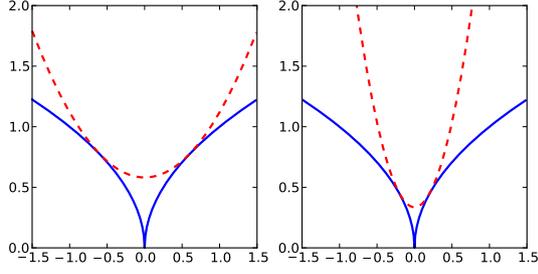


Figure 3: Blue lines show the original cost function ( $p = 0.5$ ), and red dotted lines show the auxiliary function whose  $\gamma$  is 0.6 (left panel) and 0.2 (right panel)

First, we use an auxiliary function defined as

$$Q^+(s, \gamma) = \sum_{j=1}^J \frac{\alpha_j^{p_j}}{2} \gamma_j^{p_j-2} |s_j|^2 + \frac{2-p_j}{2} \alpha_j^{p_j} \gamma_j^{p_j}, \quad (8)$$

where  $\gamma_j$  is an auxiliary variable. This cost function satisfies the three properties described in 3.1 and is equal to the original cost function when

$$\gamma_j = |s_j|. \quad (9)$$

Second, in order to add the constraint of mixing equation (7), we use Lagrange's multiplier and get the function

$$Q^+(s, \gamma) + \sum_{i=1}^I \lambda_i \left( x_i^* - \sum_{j=1}^J a_{ij}^* s_j^* \right), \quad (10)$$

where \* means complex conjugate. We can derive update rules by partial derivation on this function.

When we solve  $\frac{\partial Q^+}{\partial s_j^*} = 0$ , the update rule of separated signal  $s_j$  is written as

$$s_j = \frac{2}{\alpha_j^{p_j} p_j} \gamma_j^{2-p_j} \sum_{i=1}^I \lambda_i a_{ij}^*. \quad (11)$$

Also, by substituting Eq. (11) into mixing equation (7) and the transformation, we have

$$x_i = \sum_{i'=1}^I \lambda_{i'} \sum_{j=1}^J \frac{2}{\alpha_j^{p_j} p_j} \gamma_j^{2-p_j} a_{ij} a_{i'j}^*. \quad (12)$$

This equation shows that we can obtain  $\lambda_i$  by solving the linear equation as

$$\begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_I \end{bmatrix} = \begin{bmatrix} \sum_j \frac{2a_{1j}a_{1j}^*}{\alpha_j^{p_j} p_j} \gamma_j^{2-p_j} & \cdots & \sum_j \frac{2a_{1j}a_{Ij}^*}{\alpha_j^{p_j} p_j} \gamma_j^{2-p_j} \\ \vdots & \ddots & \vdots \\ \sum_j \frac{2a_{Ij}a_{1j}^*}{\alpha_j^{p_j} p_j} \gamma_j^{2-p_j} & \cdots & \sum_j \frac{2a_{Ij}a_{Ij}^*}{\alpha_j^{p_j} p_j} \gamma_j^{2-p_j} \end{bmatrix}^{-1} \begin{bmatrix} x_1 \\ \vdots \\ x_I \end{bmatrix}. \quad (13)$$

This equation looks a little complicated; however, we can calculate it easily because the fractions in the matrix are constant over the iterations, and the size of matrix is  $I$ ; that is, this matrix is just 2-by-2 when we use stereo recording.

In summary, we can minimize the original cost function using the following steps:

1. Initialize  $s_j$  with random non-zero complex value

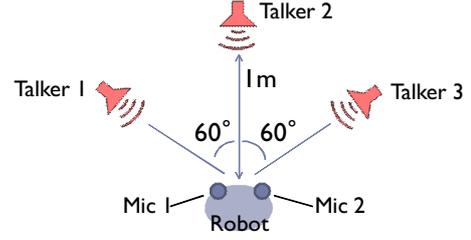


Figure 4: Location of three talkers and two microphones

2. Update auxiliary variable by using Eq. (9)
3. Update Lagrange's multiplier by using Eq. (13)
4. Update separation result by using Eq. (11)
5. If the result is not precise enough, go back to step 2

Please note that the Lp-norm minimization problem is a non-convex optimization problem when  $p$  is less than one. In that case, the separation results may converge to a poor local minimum since our method monotonically decreases the cost function through iterations. In our preliminary experiment, we found that 83% of our separation results were converged to the global optimum.

### 3.3. Update rules for other cost functions

Since our auxiliary function can be applied to many kinds of cost functions, we can change prior distribution. For example, some research has suggested that the distribution of the amplitude of speech signals is similar to a gamma distribution whose shape parameter is 0.5 [8]. Thus, we can use gamma distribution instead of the generalized Gaussian distribution, in which case, the cost function can be written as

$$Q(s) = \sum_{j=1}^J \beta_j |s_j| + (1 - q_j) \log |s_j|, \quad (14)$$

where  $q_j$  is the shape parameter and  $\beta_j$  is the inversed scale parameter of gamma distribution. We can obtain similar update rules by using a quadratic auxiliary function like in Fig. 3.

## 4. Experiments

### 4.1. Experiment condition

We synthesize sound mixtures that simulate under-determined simultaneous utterances and separate them. The locations of three talkers and two microphones used in our simulation are shown in Fig. 4. Impulse responses are recorded in an anechoic chamber, and their sampling frequency is 16 kHz. The STFT frame length and the STFT shift width are 1024 (64 ms) and 256 points (16 ms), respectively. We use 200 male- and female-utterances from Japanese Newspaper Article Sentences (JNAS) database.

We conducted two kinds of evaluation. The first one evaluates the separation speed of our method, and the second one compares the separation quality when we change the shape of the prior distribution. Since we do not have prior information about each source in this experiment, we use identical distributions for all sound sources, i.e.  $p_1 = p_2 = p_3 = p$  and  $\alpha_1 = \alpha_2 = \alpha_3 = \alpha$ . We use an open-source speech recognition software called Julius and use MFCC,  $\Delta$ MFCC, and  $\Delta$ Power as acoustic features.

Table 1: Separation time and real time factor ( $p = 1$ )

Method	Time (sec)	RTF
Proposed method	1670	1.37
SOCP solver	18500	15.2
Mask-based approximation	15.6	0.013

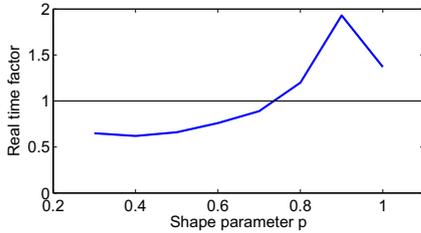


Figure 5: Real time factor of separation

## 4.2. Results and discussion

First, we compare the separation speed of our method with a general SOCP solver, mosek, and the mask-based approximation method, which assumes at most  $I$  sources are dominant in each time-frequency region. Since the accuracy of the solver output is about 40 dB, we finish the iteration when the error is less than 40 dB.

Table 1 shows the time taken to separate the sound mixtures, which contain 1218 seconds. To make this table, we assume that  $p = 1$  because the SOCP solver can be used only when  $p = 1$ . As you can see, our proposed method runs ten times faster than the general SOCP solver.

Figure 5 shows the real time factor of our method when we change the shape parameter  $p$ . This table reveals that computation time strongly depends on the parameter, and when  $p$  is about 0.9, it takes the longest to separate. This suggests that our method runs in real-time when  $p$  is small.

Second, we check the separation quality from the viewpoint of speech recognition correctness, and we change the shape parameter  $p$  of the prior distribution. Figure 6 presents the average recognition correctness of the separated signals. The horizontal line and the red dot shows the correctness when we use the gamma prior and use the mask-based method, respectively. Figure 6 shows the area whose  $p$  is not more than one since recognition correctness gets very bad when  $p$  is more than one.

As the figure shows, separation quality improves when  $p$  is less than one, i.e. when we assume sparse distribution. However, we can also see that the difference in speech recognition correctness is small. We can also see that the separation qualities are similar between the mask-based method and the method using the gamma prior. This is because both separation results contains at most  $I$  non-zero signals in each time-frequency region. In the gamma prior case, this is not assumed explicitly; however, this is true since its cost function (14) contains a log term, which makes the cost  $-\infty$  when  $|s_j| \rightarrow 0$ .

## 5. Conclusion and future work

In this paper, we propose a fast algorithm to calculate Lp-norm minimization for under-determined source separation. We use the auxiliary function method and derive the convergence-guaranteed update rules. These are three update rules, and they

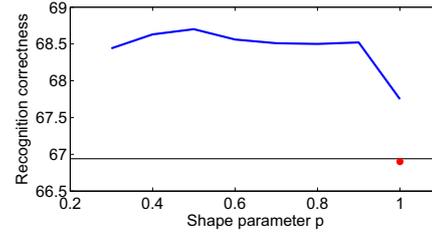


Figure 6: Speech recognition correctness of output signals. Horizontal line and red dot show the correctness when using the gamma prior and the mask-based method, respectively.

are simple enough for quick implementation. Experiments reveal that our method can calculate an exact solution for L1-norm minimization ten times faster than the conventional solution. In addition, it is confirmed that Lp-norm minimization can be solved in real-time when  $p$  is small, and its separation quality is better than L1-norm minimization.

For future work, we need to evaluate separation quality when we use the estimated mixing matrix instead of the true mixing matrix. We also need to consider reverberation in order to use the Lp-norm minimization in a real environment. In addition, we want to find better prior distributions than the generalized Gaussian distributions from the viewpoint of separation performance and separation speed.

## 6. Acknowledgment

Part of this study was supported by a Grant-in-Aid for Scientific Research (S), Global COE Program, and JST-ANR Research Program on BINAHR (Binaural Active Audition for Humanoid Robots). In addition, the authors would like to thank Dr. Patrick Danès of CNRS-LAAS for his valuable comments.

## 7. References

- [1] H. Sawada, S. Araki, and S. Makino, "Underdetermined convolutive blind source separation via frequency bin-wise clustering and permutation alignment," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 19, no. 3, pp. 516–527, 2011.
- [2] A. Ozerov and C. Févotte, "Multichannel nonnegative matrix factorization in convolutive mixtures for audio source separation," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 18, no. 3, pp. 550–563, 2010.
- [3] P. Bofill and M. Zibulevsky, "Underdetermined blind source separation using sparse representations," *Signal processing*, vol. 81, no. 11, pp. 2353–2362, 2001.
- [4] E. Vincent, "Complex nonconvex lp norm minimization for underdetermined source separation," in *Proc. of the 7th international conference on Independent component analysis and signal separation*, 2007, pp. 430–437.
- [5] D. Lee and H. Seung, "Algorithms for non-negative matrix factorization," *Advances in neural information processing systems*, vol. 13, pp. 556–562, 2001.
- [6] S. Winter, H. Sawada, and S. Makino, "On real and complex valued l1-norm minimization for overcomplete blind source separation," in *Proc. of IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2005, pp. 86–89.
- [7] H. Kameoka, N. Ono, K. Kashino, and S. Sagayama, "Complex NMF: A new sparse representation for acoustic signals," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2009, pp. 3437–3440.
- [8] S. Gazor and W. Zhang, "Speech probability distribution," *Signal Processing Letters, IEEE*, vol. 10, no. 7, pp. 204–207, 2003.