Non-negative Matrix Factorization using Stable Alternating Direction Method of Multipliers for Source Separation

Shaofei Zhang*, Dongyan Huang†, Lei Xie*, Eng Siong Chng‡, Haizhou Li†‡ and Minghui Dong†
* School of Computer Science, Northwestern Polytechnical University, Xi'an, China
E-mail: {sfzhang,lxie}@nwpu-aslp.org
† Institute for Infocomm Research, A*STAR, Singapore
E-mail: {huang,hli,mhdong}@i2r.a-star.edu.sg
‡ School of Computer Engineering, Nanyang Technological University, Singapore
E-mail: ASESChng@ntu.edu.sg

***Abstract*—Nonnegative matrix factorization (NMF) is a popular method for source separation. In this paper, an alternating direction method of multipliers (ADMM) for NMF is studied, which deals with the NMF problem using the cost function of beta-divergence. Our study shows that this algorithm outperforms state-of-the-art algorithms on synthetic data sets, but it presents unstable behavior and low accuracy on real data sets. Therefore, we propose two different stable ADMM algorithms for NMF to solve this problem. They differ slightly in the multiplicative factor utilized in the update rules. One algorithm is to adapt the step size to guarantee the convergence while the other minimizes the beta-divergence with a pivot element weighting iterative method (PEWI). Experimental results demonstrate that the proposed algorithms are more stable and accurate. Particularly, PEWI based ADMM shows superior performance in the source separation task.**

I. Introduction

NMF [1] has been applied to various applications such as polyphonic music transcription [2] and source separation [3]. Given a data matrix V of dimensions $M \times N$ with non-negative entries, NMF aims at finding two low-rank matrices W and H such that

$$V \approx WH. \tag{1}$$

The common approach is to minimize the difference between V and WH using Euclidean distance (EUD):

$$\underset{W\geq 0, H\geq 0}{\text{minimize}} \quad \frac{1}{2}\sum_{m=1}^{M}\sum_{n=1}^{N}(V_{mn}-(WH)_{mn})^2. \tag{2}$$

In practice, the problem is convex in W and H separately, so many algorithms adopt an alternating minimization approach. The most popular approach is a simple multiplicative update method proposed by Lee and Seung [4], but the convergence of the algorithm to a stationary point has not yet been proven [5]. Based on alternating nonnegative least squares, several algorithms have been proposed and showed good performance such as the project gradient method [6] and the block principal pivoting method [7]. These algorithms possess a property that every produced limit point is a stationary point [6].

The mentioned algorithms rely on special properties of EUD, and they are not universal for different applications. Recently, Sun *et al.* [8] proposed an ADMM based universal update framework (Universal ADMM) which has faster convergence and better accuracy than the state-of-the-art algorithms on synthetic data sets. However, we discovered that for real non-negative data, e.g., speech spectrum, Universal ADMM results in lower performance than our expectation in terms of stability and accuracy. Through the analysis of our experimental results, we find that this phenomenon happens due to the ill-condition of matrices while updating dictionary matrix and activation matrix.

In this paper, in order to make ADMM to work properly on real non-negative data, two different stable ADMM algorithms are proposed to solve the ill-condition problem for NMF. Algorithm 1 adapts step size to prevent ill-condition indirectly, while Algorithm 2 solves the ill-conditioned issue using an PEWI method directly. We apply the proposed algorithms to NMF-based source separation. Experimental results show that, compared with Universal ADMM, the proposed algorithms are more stable and accurate, and particularly, Algorithm 2 outperforms the others in stability and accuracy. It can achieve the best source separation results. For further improvement of accuracy, we also analyze the correlation of distribution of signals, the cost function and the accuracy of ADMM algorithms.

II. ADMM Algorithm

ADMM is a simple but powerful algorithm that is well suited to distributed convex optimization. It can blend the decomposability of dual ascent with the superior convergence properties of the method of multipliers [9]. The algorithm solves problems in the form

$$\begin{aligned} \underset{x,z}{\text{minimize}} \quad & f(x)+g(z) \\ \text{subject to} \quad & Ax+Bz=c, \end{aligned} \tag{3}$$

where f and g are convex functions defined on closed sets, The augmented Lagrangian for problem (3) is

$$L_\rho = f(x) + g(z) + y^T(Ax + Bz - c) + (\rho/2)\|Ax + Bz - c\|_2^2, \tag{4}$$

where ρ is dual step size, y is the dual variable or Lagrange multiplier. ADMM involves the following iterations

$$\begin{aligned} x^{k+1} &:= \text{argmin}_x L_\rho(x, z^k, y^k) \\ z^{k+1} &:= \text{argmin}_z L_\rho(x^{k+1}, z, y^k) \\ y^{k+1} &:= y^k + \rho(Ax^{k+1} + Bz^{k+1} - c). \end{aligned} \tag{5}$$

Decomposability: From Eq. (3), we find that ADMM algorithm treats the separated variables as different variables at the beginning, the same with constraints, that guarantees the decomposability of ADMM algorithm. It is different from dual decomposition which treats separated variables as parts of primal variables, and all the parts should be gathered in order to update dual variables.

Convergence Properties: For ADMM, the optimality conditions for Eq. (3) are primal feasibility and dual feasibility, i.e.,

$$Ax + Bz - c = 0, \quad \nabla f(x) + A^T y = 0 \quad \nabla g(z) + B^T y = 0,$$

respectively. Since z^{k+1} minimizes $L_\rho(x^{k+1}, z, y^k)$, we have

$$\begin{aligned} 0 &= \nabla g(z^{k+1}) + B^T y^k + \rho B^T(Ax^{k+1} + Bz^{k+1} - c) \\ &= \nabla g(z^{k+1}) + B^T y^{k+1}, \end{aligned}$$

i.e., with ADMM dual variable update, $(x^{k+1}, z^{k+1}, y^{k+1})$ satisfies the second dual feasibility condition, and primal and first dual feasibility are achieved as $k \to \infty$. A detailed convergence proof can be found in [10].

III. Conventional ADMM for NMF

In distributed optimization, ADMM can blend the decomposability of dual ascent with the superior convergence properties of the method of multipliers [9]. In order to apply ADMM algorithm to extensive applications using different divergences, Sun and Févotte [8] developed an ADMM-based update framework which extended the NMF problem to β-divergence [11]. They proposed to split divergence with dictionary and activation which makes optimization simpler and much more universal for different divergences.

A. Universal ADMM

In the approach proposed by Sun and Févotte [8], the NMF problem can be rewritten as

$$\begin{aligned} \text{minimize} \quad & D_\beta(V|X) \\ \text{subject to} \quad & X = WH. \\ & W = W_+, H = H_+ \\ & W_+ \geq 0, H_+ \geq 0, \end{aligned} \tag{6}$$

where D_β represents a general family of divergence functions known as β-divergence. This approach introduces new variables W_+ and H_+ to which the nonnegativity constrains are applied and a new variable X to split the divergence with WH which makes the optimization problem simpler and also more universal for different divergences. The corresponding augmented Lagrangian is as followings

$$\begin{aligned} &L_\rho(X, W, H, W_+, H_+, \alpha_X, \alpha_W, \alpha_H) = \\ &D_\beta(V|X) + \langle \alpha_X, X - WH\rangle + \frac{\rho}{2}\|X - WH\|_F^2 \\ &+\langle \alpha_W, W - W_+\rangle + \frac{\rho}{2}\|W - W_+\|_F^2 \\ &+\langle \alpha_H, H - H_+\rangle + \frac{\rho}{2}\|H - H_+\|_F^2. \end{aligned} \tag{7}$$

The $\alpha_X, \alpha_W, \alpha_H$ are three dual variables. The detailed update roles can be found in [8]. In particular, the root formula and Cardan formula are adapted to compute the update rule of X corresponding to β=1 and β=0, respectively.

B. Existing Problem

In Universal ADMM algorithm updates, updating dictionary matrix W and activation matrix H requires solving system $Ax = b$ [8]. Since the matrix A in updates is square and nonsingular, the common solution adopts $x = A^{-1}b$, however, if this system is ill-conditioned, this common method may yield unstable and imprecise results. During our experiments on real data, we observe that Universal ADMM results in unstable and low-accuracy solution shown in Fig.2 in Section 5.1.

The problem can be investigated by studying the stability of $x = A^{-1}b$. Assuming that x is the solution of original system and $x + \Delta x$ is the solution when b change from b to $b + \Delta b$. According to properties of norm, we can write

$$\frac{\|\Delta x\|}{\|x\|} \leq \left\|A^{-1}\right\| \cdot \|A\| \cdot \frac{\|\Delta b\|}{\|b\|}. \tag{8}$$

Likewise, if the coefficient matrix A changes from A to $A + \Delta A$ while b is fixed, the solution is $x + \Delta x$, then the changes of solution can be expressed in the following manner:

$$\frac{\|\Delta x\|}{\|x\|} \leq \frac{\left\|A^{-1}\right\| \cdot \|A\| \cdot \frac{\|\Delta A\|}{\|A\|}}{1 - \|A^{-1}\| \cdot \|A\| \cdot \frac{\|\Delta A\|}{\|A\|}}. \tag{9}$$

We set $K(A) = \left\|A^{-1}\right\| \cdot \|A\|$ which is called the condition number of matrix A, the $K(A)$ is in fact a measure of the relative sensitivity of solution x to changes in right-hand vector b and coefficient matrix A. If $K(A)$ becomes large, the system is regarded as being ill-conditioned, i.e., small changes of right-hand vector b or coefficient matrix A result in a large change in the solution. In practice, we test the condition number of system which solves W and H in each iteration, the result shows that the point of solution mutation coincides with large value of the condition number.

For speech data, the possible explanation is that since the speech energy usually focuses on certain frequency bands, the column features of dictionary W which captures prototypical spectra will become similar, i.e., the column vectors in coefficient matrix A of linear systems which solve dictionary matrix

W are strongly correlated. When small changes occur in A, a large change in x appears, e.g., for matrix A

$$A = \begin{pmatrix} 1000 & 1000 \\ 0 & 0.001 \end{pmatrix},$$

if the right-hand vector $b = [1000, 0]^T$, the solution $x = [1, 0]^T$, while small change occurs on b, e.g., $b = [1000, 0.001]$, the solution abruptly changes to be $x = [0, 1]^T$. On the other hand, since the speech energy focus on certain frequency bands, SVD factorization or eigenvalue factorization of speech spectrum results in great difference between the maximum eigenvalue and the minimum eigenvalue. Assuming that two eigenvectors of matrix A are x_1, x_2, two eigenvalues are λ_1, λ_2 respectively. For vector b, we can write

$$\begin{aligned} b &= mx_1 + nx_2 = \frac{m}{\lambda_1}\lambda_1 x_1 + \frac{n}{\lambda_2}\lambda_2 x_2 \\ &= A(\frac{m}{\lambda 1}x_1 + \frac{n}{\lambda_2}x_2) = Ax. \end{aligned}$$

If λ_1 is much greater than λ_2 and b changes in x_1 direction, i.e., m changes, the solution x does not change significantly; otherwise, when b changes in x_2 direction, i.e., n changes, the solution x changes significantly. These two aspects have a direct relationship with the large value and mutation of condition number of linear system [12].

IV. Stable ADMM for NMF

As discussed in the previous section, the instability of ADMM essentially owes to large value and mutation of the condition number of coefficient matrix A. Thus we propose two methods to solve the ill-condition of coefficient matrix. The first one avoids the ill-condition by heuristically adapting the step size to change the convergence trend. The second one solves the ill-conditioned system using PEWI algorithm directly.

A. Algorithm 1: Changing the Convergence Trend

The basic form of Universal ADMM updates [8] on W and H can be written as

$$X_k := f(Y_{k-1}) \backslash g_{k-1}(\rho), \tag{10}$$

where X_k is W or H in the k^{th} iteration which can be considered as the solution x. Y_{k-1} is H or W in the $(k-1)^{th}$ iteration, respectively. $f(Y_{k-1})$ can be regarded as the coefficient matrix A. $g_{k-1}(\rho)$ can be regarded as right-hand vector b, and the $\backslash$ denotes a least-squares solution to the system of equations $Ax = b$ [8].

The stability of the solution X_k (Eq. 10) depends on the ill-condition degree of $f(Y_{k-1})$ and the change degree of $g_{k-1}(\rho)$. Therefore, we propose a heuristic idea that if ill-conditioned degree of $f(Y(k-1))$ is out of the given $interval$, $Y(k-2)$ is adapted to update $X(k)$, let $Y(k-1) = Y(k-2)$. At the same time, the value of ρ is changed by adding and subtracting a constant Δ, i.e., new ρ^* is in $(\rho, \rho - \Delta, \rho + \Delta)$. Contrasting the norm value, ρ^* is selected to correspond to the minimal norm value $\|g(\rho^*, k-1) - g(\rho^*, k-2)\|$. The detailed pseudo-code is given in Algorithm 1.

Algorithm 1 Changing the Convergence Trend

Require:
$thrh$: neighborhood value.
$cond(f(Y_0)) * (1 \pm thrh)$: initial given $interval$.
ρ : initial dual step size.
Δ: changing step size of ρ.
Ensure:
if $cond(f(Y_k)) \notin interval$ **then**
$Y_{k-1} = Y_{k-2}$;
$\rho^* = [\rho, \rho + \Delta, \rho - \Delta]$;
$min = 1e9$;
for all $i \in \rho^*$ **do**
$D = \|g(i, k-1) - g(i, k-2)\|$
if $D < min$ **then**
$\rho = i$; $min = D$;
end if
end for
end if

This heuristic idea can change the convergence trend and stabilize ADMM. However, it is also obvious that this method slows down the convergence speed simultaneously by using the prior to the last iteration result and decreasing change degree of b to update x, as shown in Fig.3 and 4.

B. Algorithm 2: Pivot Element Weighting Iterative Method

In order to guarantee the stability of ADMM while improving the convergence speed, we adopt a more simple and efficient method named PEWI for solving ill-conditioned linear systems. The form of pivot element weighting is as follows:

$$A + \alpha E, \tag{11}$$

where A is a positive definite square matrix, as $f(Y_{k-1})$ in Algorithm 1, E is the identity matrix, and α is a weighting value. It has been proven that if A is a symmetric positive definite matrix and $\alpha > 0$, cond$(A + \alpha E) <$ cond(A) [13]. Therefore, we can rewrite the ill-conditioned system $Ax = b$ as $(A + \alpha E)x = b + \alpha x$ which guarantees the small condition number. Then we can construct the iterative formula

$$(A + \alpha E)x_{k+1} = b + \alpha x_k. \tag{12}$$

Let $x_{k+1} = x_k + e_k$, then

$$(A + \alpha E)e_k = b - Ax_k, \tag{13}$$

if $e_k \to 0$, and $x_{k+1} \to$ optimal solution, the proof of iterative convergence can be found in [13]. In practice, we first set an initial upper $limit$ and the solution x_0, if the ill-conditioned degree of matrix A is out of the upper $limit$, we weight the pivot element by a constant Δ until the matrix A is well-conditioned, using Eq.(10) to iterate until the norm of e_k meets a certain accuracy eps. The detailed pseudo-code is given in Algorithm 2. In particular, since the PEWI method improves the condition number and guarantees the condition number in a given interval, it gradually makes the solution out of the

Algorithm 2 Pivot Element Weighting Iterating

Require:

$thrh$: neighborhood value.

$cond(A) * (1 + thrh)$: initial given upper $limit$.

Δ: changing step size. $n = 0$: counter.

eps : accuracy constant.

x_0: initial solution.

Ensure:

while $cond(A)) > limit$ **do**

$A = A + \Delta * E$;

$n = n + 1$;

end while

$x = A\ (b + n * \Delta) * x_0$;

while $|x - x_0| > eps$ **do**

x_0 = x;

$x = A\ (b + n * \Delta) * x_0$;

end while

bad solution set. The iteration number of PEWI until well-condition gradually decreases, and the complexity of algorithm can be accepted.

V. Source Separation via NMF

During the past decades, many methods have been proposed for source separation including independent component analysis (ICA) [14], principle component analysis (PCA) [15] and NMF [16] *etc.* Comparing to PCA and ICA, NMF has been the most promising method in source separation owing to the parts-based decomposition and non-negative constraint [16], [17]. In this paper, we use a fully-supervised "factorize-train" method [18]. Fig.1 shows a general pipeline of separation.

As shown in Fig.1, the "factorize-train" method firstly transforms the time-field mixed signal x_{mix} and individual clean source $x_{indiv1,2}$ to frequency-field using short-time Fourier transform (STFT) and calculates the dictionary matrix of clean source using magnitude $|X_{indiv1,2}|$ by NMF respectively. Then all the dictionaries including W_{indiv1} and W_{indiv2} are concatenated as a dictionary W to factorize the mixed signal in order to determine the activation matrix H. Because of the parts-based decomposition of NMF, the dictionary matrix captures the prototypical spectral in each column vector that contains the latent information of multiple sources, and the activation matrix captures the weight of each dictionary vector through the time axis. Based on this representation, because the dictionary indices are known for each source, the separation can be performed by a filter which is referred to as a masking filter. Combining the phase of mixed signal $\angle X_{mix}$ and the separated magnitude Y_1 and Y_2, the separated time-field signals y_1 and y_2 can be obtained by inverse short-time Fourier transform (ISTFT). When we get the separated signals, we use the blind source separation evaluation (BSS Eval) toolkit[19], especially, we use the source-to-distortion ratio (SDR) to evaluate the source separation results.

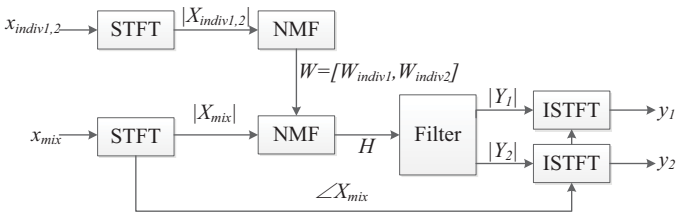

Fig. 1. Pipeline of separating two sources via NMF.

VI. Experiments

We carried out our experiments to evaluate the proposed algorithms. We consider two values of ρ using the cost function of KL divergence to test algorithms on both synthetic and real non-negative data. We set the iteration number to 1e5. Firstly, we examine the convergence performance of Universal ADMM to identify the limitations of Universal ADMM. Secondly, we compare the convergence performance of Algorithm 1, Algorithm 2, and Universal ADMM on real non-negative data, e.g., the spectrum data, as well as their application to source separation task to show the stability of the proposed algorithms. Finally, we discuss the correlation among the distribution of signals, the cost function, and the accuracy of the ADMM algorithms.

A. Universal ADMM

We evaluate the performance of Universal ADMM on both synthetic and real non-negative data considering KL divergence and two initial values of $\rho = (1, 2.5)$. The synthetic data V is constructed as $V = W * H$ where the initial W and H are generated as the absolute of values of Gaussian noise [1], the dimensions are $M = 513, K = 25, N = 185$. Real spectrum data uses the speech magnitude spectrum with size $V = 513 * 185$. As shown in Fig 2, for synthetic data, Universal ADMM has a stable convergence trend and produces high accuracy [2]. However, for real spectrum data, it is not stable and has low accuracy.

B. Stable ADMM against Universal ADMM

The simulation results show the instability and low accuracy of Universal ADMM on real data sets. Therefore, we compare the performance of Algorithm 1, Algorithm 2, and Universal ADMM on real spectrum data. For Algorithm 1, we set $thrh = 0.5$ and the step size as ρ: $\Delta = 10$ for current $\rho > 1$ and $\Delta = 0.1$ for current $\rho < 1$. For Algorithm 2, we set $thrh = 0.02$, $\Delta = 2$, $eps = 1e-7$ and x_0 is unit vector. As shown in Fig.3 and 4, the proposed algorithms demonstrate more stable than Universal ADMM, and particularly, PEWI based ADMM yields a positive performance in both stability and accuracy.

We also compare the proposed algorithms with the Universal ADMM on source separation task using KL-divergence

[1] In MATLAB notation: V=abs(randn(M,K))*abs(randn(K,N)).

[2] The proposed algorithms focuses on solving the ill-conditioned system, while synthetic data is not ill-conditioned, the proposed algorithms coincide with general solver used in Universal ADMM.

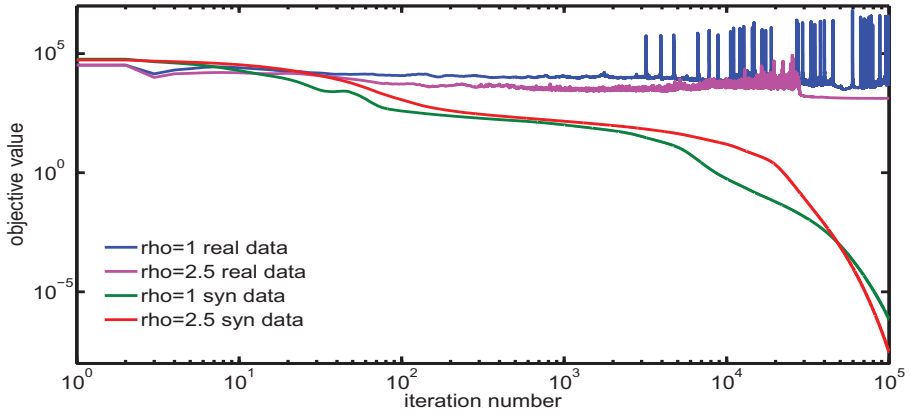

Fig. 2. Universal ADMM performance on synthetic data and real spectrum data: KL-divergence as the cost function and two settings of $\rho = (1, 2.5)$.

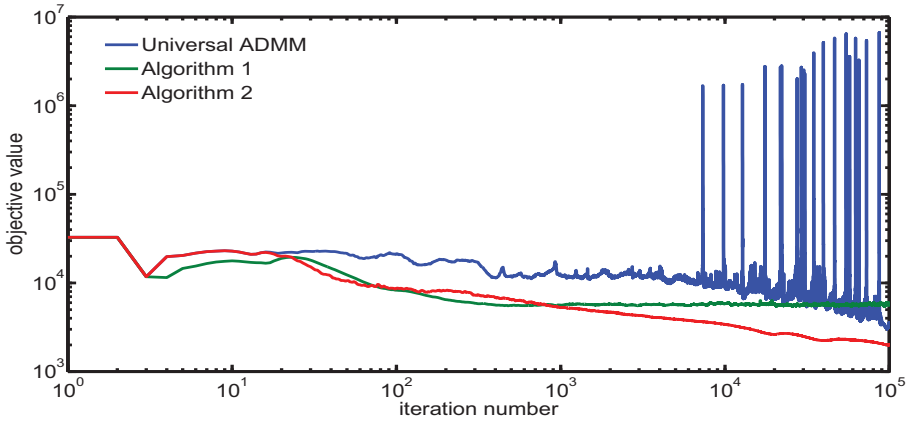

Fig. 3. Comparison of proposed ADMM performance with Universal ADMM on real spectrum data: KL-divergence as cost function and $\rho = 1$.

TABLE I
SDR OF SEPARATED SPEECH USING DIFFERENT ADMM ALGRITHMS ON SPEECH FROM TIMIT AND NOISES FROM NOISE92.

	(fcmg0, f16)	(fcmg0, m109)	(fcmg0, engine)	(mcal0, f16)	(mcal0, m109)	(mcal0, engine)
Algorithm 1	9.916	17.127	9.088	7.713	12.161	6.741
Algorithm 2	**10.118**	**17.592**	**9.381**	**7.986**	**12.215**	**6.839**
Universal ADMM	8.917	15.568	8.822	6.812	11.658	6.512

and ρ set to 1. We generate a noisy speech sample with 0db of length 4-sec using speech from TIMIT and noise from NOISE92 with the same initialization for all algorithms and set the iteration number to 5000. We adopt the source separation pipeline shown in Fig.1, and the SDR of the results are shown in Table 1. We can observe that the proposed algorithms achieve superior performance in comparison to Universal ADMM. Particularly, Algorithm 2 outperforms the others.

The simulation results show that Algorithm 2 improves the accuracy to some extent against others on real spectrum data; but it is still lower than its accuracy performance on the synthetic data. Thus, we compare the distribution of different signals and analyse the correlation of distribution of signals, the cost function, and the accuracy of the ADMM algorithms to identify the issue. We consider two synthetic data sizes $(M, K, N) = (513, 25, 185)$(V1) and $(200, 100, 1000)$(V2). We chose the magnitude spectrum of speech, brown noise, and noisy speech with size $V = 513 * 185$. As shown in Fig.5, the distribution of synthetic data approximates the normal distribution while the real spectrum data approximates the Laplace distribution. For data with a normal distribution, while applying to the cost functions like KL-divergence, it can turn the KL-divergence into a Euclidean distance which coincides with the distribution of the signals. However, since the real spectrum data is non-stationary signal, it is desired to develop sparse NMF using ADMM with the cost function of KL-divergence to further improve the accuracy performance of algorithms.

C. Discussion on Accuracy of Algorithms

The simulation results show that Algorithm 2 improves the accuracy to some extent against Algorithm 1 and Universal ADMM on real spectrum data; but it is still lower than its

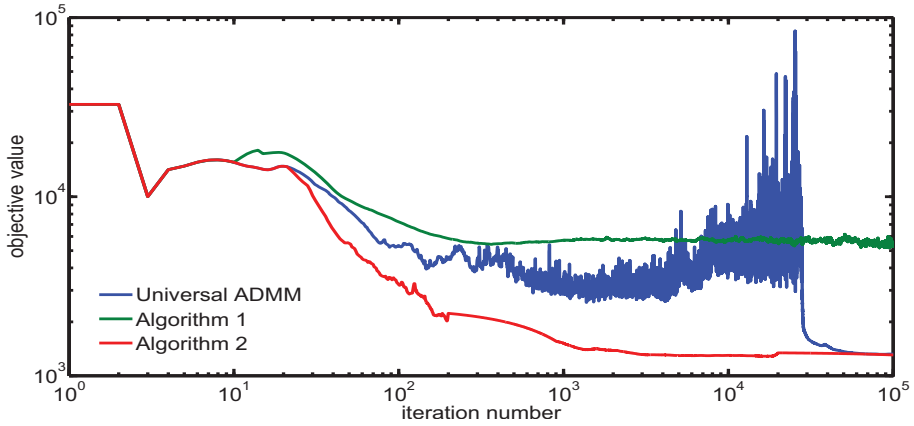

Fig. 4. Comparison of proposed ADMM performance with Universal ADMM on real spectrum data: KL-divergence as cost function and $\rho = 2.5$.

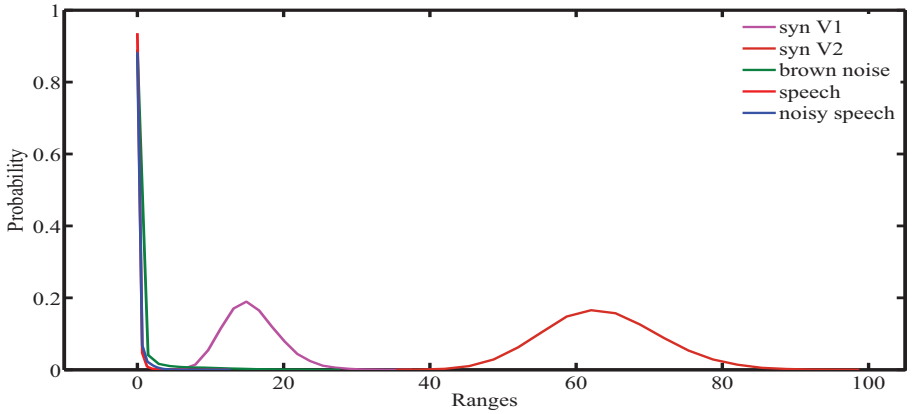

Fig. 5. Comparison of probability distribution density of different signals including real spectrum data and synthetic data.

accuracy performance on the synthetic data. Thus, we compare the distribution of different signals and analyse the correlation of distribution of signals, the cost function, and the accuracy of the ADMM algorithms to identify the issue. We consider two synthetic data sizes $(M, K, N) = (513, 25, 185)$(V1) and $(200, 100, 1000)$(V2). We chose the magnitude spectrum of speech, brown noise, and noisy speech with size $V = 513 * 185$. As shown in Fig.4, the distribution of synthetic data approximates the normal distribution while the real spectrum data approximates the Laplace distribution. For data with a normal distribution, while applying to the cost functions like KL-divergence, it can turn the KL-divergence into a Euclidean distance which coincides with the distribution of the signals. However, since the real spectrum data is non-stationary signal, it is desired to develop sparse NMF using ADMM with the cost function of KL-divergence to improve further the accuracy performance of algorithms.

VII. Conclusions

In this paper, the instability and low accuracy of Universal ADMM have been shown through the experiments. The analysis found that they are caused by the ill-condition of linear system on real data sets. To solve the ill-condition problem, we proposed two stable methods: Algorithm 1 and Algorithm 2. Algorithm 1 aims at adapting the step size to guarantee the convergence. Algorithm 2 attempts to solve the ill-condition problem using a pivot element weighting iterative method. The experiments showed that our proposed algorithms are more stable than Universal ADMM, and in particular, Algorithm 2 performs better than the other two algorithms in terms of stability and accuracy. But the accuracy performance of all ADMM algorithms on real non-negative data sets is still far from that on synthetic data sets. In order to understand this issue, we compare the distribution of different signals and analyse the correlation of the distribution of signals, the cost function, and the accuracy of ADMM algorithms. In order to further improve the performance of the ADMM algorithms for NMF on non-stationary signals, we shall develop sparse NMF using ADMM with the cost function of KL-divergence in the near future.

Acknowledgment

We thank Dennis L. Sun from Stanford University for valuable discussions. This work was supported by the National Natural Science Foundation of China (Grant No. 61175018, 61571363) and the Seed Foundation of Innovation and Creation for Graduate Students in Northwestern Polytechnical University.

REFERENCES

[1] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, pp. 788–791, 1999.

[2] P. Smaragdis and J. C. Brown, "Non-negative matrix factorization for polyphonic music transcription," *in IEEE Workshop on Applications of Signal Processing to Audio and Acoustics(WASPAA03)*, pp. 177–180, 2003.

[3] D. FitzGerald, M. Cranitch, and E. Coyle, "On the use of the beta divergence for musical source separation," in *Proc. Irish Signals and Systems Conference*, 2009.

[4] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," *Advances in neural information processing systems*, vol. 13, pp. 556–562, 2001.

[5] E. F. Gonzalez and Y. Zhang, "Accelerating the lee-seung algorithm for non-negative matrix factorization," *Dept. Comput. & Appl. Math., Rice Univ., Houston, TX, Tech. Rep. TR-05-02*, 2005.

[6] C.-J. Lin, "Projected gradient methods for nonnegative matrix factorization," *Neural computation*, vol. 19, no. 10, pp. 2756–2779, 2007.

[7] J. Kim and H. Park, "Fast nonnegative matrix factorization: An active-set-like method and comparisons," *SIAM Journal on Scientific Computing*, vol. 33, no. 6, pp. 3261–3281, 2011.

[8] D. L. Sun, "Alternating direction method of multipliers for non-negative matrix factorization with the beta-divergence," *IEEE International Conference on Acoustic, Speech and Signal Processing (ICASSP)*, pp. 6242–6246, 2014.

[9] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends® in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.

[10] M. Fortin and R. Glowinski, "Augmented lagrangian method," *North-Holland, Amsterdam, New York*, 1983.

[11] A. Cichocki and S. Amari, "Families of alpha-beta-and gamma-divergences: Flexible and robust measures of similarities," *Entropy*, vol. 12, no. 6, pp. 1532–1568, 2010.

[12] M. Shi and L. Gu, *Scientific and Engineering Calculation Bases*. Tsinghua University Press Ltd, 1999.

[13] L. Tang and P. Li, "A pivot element weighting iterative method for solving ill-conditioned linear equations," *Science Technology and Engineering*, vol. 12, pp. 381–383, 2012.

[14] M. E. Davies and C. J. James, "Source separation using single channel ica," *Signal Processing*, vol. 87, no. 8, pp. 1819–1832, 2007.

[15] F. Asano, Y. Motomura, H. Asoh, and T. Matsui, "Effect of pca filter in blind source separation," in *Proc. ICA*, 2000, pp. 57–62.

[16] A. Zinovyev, U. Kairov, T. Karpenyuk, and E. Ramanculov, "Blind source separation methods for deconvolution of complex signals in cancer biology," *Biochemical and biophysical research communications*, vol. 430, no. 3, pp. 1182–1187, 2013.

[17] B. King and L. Atlas, "Single-channel source separation using simplified-training complex matrix factorization," in *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*. IEEE, 2010, pp. 4206–4209.

[18] H. Kameoka, N. Ono, K. Kashino, and S. Sagayama, "Complex nmf: A new sparse representation for acoustic signals," in *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*. IEEE, 2009, pp. 3437–3440.

[19] E. Vincent, R. Gribonval, and C. Févotte, "Performance measurement in blind audio source separation," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 14, no. 4, pp. 1462–1469, 2006.