

Alternating Projection Approach for Nonlinear Blind Separation of Sparse Sources

Saeed Akhavan, Hamid Soltanian-Zadeh
School of Electrical and Computer Engineering
University of Tehran
Tehran, Iran
s.akhavan@ut.ac.ir, hszadeh@ut.ac.ir

Abstract—In this paper, we propose an iterative method to solve the nonlinear blind source separation (BSS) problem when the sources are sparse. The main idea to solve the problem is approximation of the nonlinear mixture functions with the polynomial functions. Then, using an alternating approach, the sources and the coefficients of polynomial functions are estimated. The proposed approach is similar to one employed in dictionary learning algorithms for sparse representation. In fact, in iterations where the sources are estimated, we cluster the signals, and in iterations where the coefficients of polynomial functions are estimated, we assign a polynomial manifold to each cluster. Experimental results demonstrate the effectiveness of the proposed method relative to state-of-the-art methods.

Index Terms—Blind source separation, nonlinear mixtures, sparse sources, polynomial approximations

I. Introduction

The goal of blind source separation (BSS) is extracting the sources from the recorded signals in the receiver when the mixture functions are unknown and some prior information are available about the sources. In general, BSS can be categorized into linear and nonlinear depending on the mixture functions. The linear BSS has widely been investigated in different scenarios, while the nonlinear BSS has not been explored comprehensively because it is much more complicated than the linear BSS [1]. In this study, the focus is on the nonlinear BSS problem when the sources are sparse signals which can happen in several applications such as system identification tasks. In the following, we summarize the well-known researches which consider this scenario for performing BSS [2]–[5], and then, state our contribution.

In [2], [3], it is assumed that the mixture functions have linear-quadratic (LQ) forms. The proposed strategy is canceling the nonlinear elements of the outputs in the beginning. It is shown that by subtraction of the outputs with some specific weights, which must be estimated, the nonlinear terms are canceled in the considered model. After omitting the nonlinear terms, a linear BSS method is employed to retrieve the sources. The main drawback of this approach is that the structure of the mixture functions is assumed to be known while this information is not available most of the time in practical situations.

Unlike the previous approach, the authors of [4], [5] do not constrain the mixture functions to specific structures, and they obtain the unknown parameters by analyzing the scatter plot of the observations. They propose a two-stage procedure: 1) clustering; and 2) manifold learning for performing BSS. The idea has been taken from the geometrical method employed for performing SCA in the linear model [1]. There are two main drawbacks in this idea. The first drawback is the performance of the proposed method. If the clustering does not perform perfectly, the accuracy of the manifold learning would severely decrease. Hence, there must be an alternation between clustering and manifolds learning to get accurate results similar to the approach used in dictionary learning algorithms for sparse representation. The second drawback is that the implementation of the mentioned idea for data with more than three dimensions is very difficult. The clustering of such data while there may be several overlapped areas is not possible. For this reason, all of the results presented in [4], [5] were obtained by considering two sources and two outputs.

This study tries to solve the above drawbacks and proposes a more general framework for retrieving sparse sources from nonlinear mixtures. We first substitute the mixture functions by their polynomial approximations. Then, using an alternating minimization approach, the coefficients of polynomial functions and sparse sources are alternately estimated until convergence of the parameters. This means that we alternately perform the clustering and manifold learning.

II. Proposed Method

We express the model as:

$$\begin{aligned} y_m &= f_m(s_1, s_2, \dots, s_N) = f_m(\mathbf{s}) \\ m &= 1, 2, \dots, M, \quad \|\mathbf{s}\|_0 = N_0 \ll N \end{aligned} \quad (1)$$

where $f_m : \mathbb{R}^N \rightarrow \mathbb{R}$, $s_n \in \mathbb{R}$, and $y_m \in \mathbb{R}$ respectively show the m^{th} mixture function, the n^{th} source, and the m^{th} observed signal. $\mathbf{s} = [s_1, s_2, \dots, s_N]^T \in \mathbb{R}^N$ and $N_0 \in \mathbb{N}$ represent the vector of sources and the sparsity level, respectively. Without loss of generality, we suppose that: 1) the outputs of the mixture functions are zero when

the inputs are all zero; and 2) the sources are distributed in the convergence region of the Maclaurin series of the mixture functions.

If we approximate the mixture functions by the polynomial functions up to order K and use the matrix notation, we get

$$\mathbf{y} = \mathbf{F} \tilde{\mathbf{s}} \quad (2)$$

where \mathbf{y} shows the observed signals, the entries of \mathbf{F} are corresponding to the coefficients of polynomial approximations, and $\tilde{\mathbf{s}}$ consists of the powers of s_1, s_2, \dots, s_N , and all of their multiplication terms up to order K .

If we respectively stack \mathbf{y} and $\tilde{\mathbf{s}}$ at different time instants in the columns of \mathbf{Y} and $\tilde{\mathbf{S}}$, we get

$$\mathbf{Y} = \mathbf{F} \tilde{\mathbf{S}} \quad (3)$$

Hence, we should factorize \mathbf{Y} into \mathbf{F} and $\tilde{\mathbf{S}}$ in such a way that each column of $\tilde{\mathbf{S}}$ is a block sparse vector with a specific structure. There is a noticeable point regarding the existing scaling ambiguity between \mathbf{F} and $\tilde{\mathbf{S}}$ in the factorization considered in (3). It can be easily shown that by considering the columns of \mathbf{F} which are corresponding to the first power of s_1, s_2, \dots, s_N , as unit norm vectors, the existing scaling ambiguity between \mathbf{F} and $\tilde{\mathbf{S}}$ would be omitted [1].

Based on (3) and the mentioned remarks, we can find that there are similarities between the problem considered here and the dictionary learning problem for sparse representation. Hence, we can tackle the problem in a similar manner. We minimize the representation error $\|\mathbf{Y} - \mathbf{F} \tilde{\mathbf{S}}\|_F^2$ subject to the aforementioned constraints and find the parameters. We use an alternating approach to minimize the representation error. This means that some feasible initial values are considered for \mathbf{F} and $\tilde{\mathbf{S}}$, then, the following two steps, i.e., $\tilde{\mathbf{S}}$ -Update and \mathbf{F} -Update, are alternately performed until convergence of the parameters.

1) $\tilde{\mathbf{S}}$ -Update: Based on the considered objective function, i.e., the representation error, this step must individually be performed for the data at each time instant, i.e.,

$$\begin{aligned} \{\hat{s}_1, \hat{s}_2, \dots, \hat{s}_N\} = \underset{s_1, s_2, \dots, s_N}{\operatorname{argmin}} \quad & \|\mathbf{y} - \mathbf{F} \tilde{\mathbf{s}}\|_2^2 \\ \text{s.t.} \quad & \|\mathbf{s}\|_0 = N_0 \end{aligned} \quad (4)$$

We use the idea of matching pursuit (MP) to perform this step [6]. In fact, we find the active sources one by one. At first, we find the source which has the most contribution in the generation of \mathbf{y} . If we consider the columns of \mathbf{F} which are corresponding to the powers of s_n in $\mathbf{F}^{(n)}$, the following constrained optimization problem must individually be solved for $n = 1, 2, \dots, N$:

$$\hat{s}_{n^*} = \underset{s_n}{\operatorname{argmin}} \|\mathbf{y} - \mathbf{F}^{(n)} [s_n \ s_n^2 \ \dots \ s_n^K]^T\|_2^2 \quad (5)$$

Then, the source leading to the smaller representation error would be considered as the first active source. It

should be noted that (5) can be optimized using many of the developed algorithms exploited for optimization of polynomial functions such as GpoSolver [7]. Then, we remove the contribution of the first estimated source from \mathbf{y} , and find the next source which has the most contribution in the generation of the observed signal.

We repeat the aforementioned procedure until finding N_0 sources, and finally, make the remaining sources equal to zero.

2) \mathbf{F} -Update: In this step, the following constrained optimization problem must be solved:

$$\begin{aligned} \hat{\mathbf{F}} = \underset{\mathbf{F}}{\operatorname{argmin}} \quad & \|\mathbf{Y} - \mathbf{F} \tilde{\mathbf{S}}\|_F^2 \\ \text{s.t.} \quad & \|\mathbf{F}(:, z)\|_2 = 1, \quad z \in \Sigma \end{aligned} \quad (6)$$

where Σ denotes the indices of columns of \mathbf{F} which are corresponding to the first power of s_1, s_2, \dots, s_N . In fact, the constraint mentioned in (6) solves the scaling ambiguity problem. We use the idea of method of optimal directions (MOD) [8] to solve the problem which leads to

$$\hat{\mathbf{F}} = \mathbf{Y} \tilde{\mathbf{S}}^\dagger \quad (7)$$

followed by normalizing the columns of $\hat{\mathbf{F}}$ whose indices exist in Σ . It should be mentioned that † denotes the pseudo inverse.

By performing a few iteration between $\tilde{\mathbf{S}}$ -Update and \mathbf{F} -Update, the unknown parameters are determined.

Before presenting the simulation results, three important points must be discussed at the end of this section.

The first point is regarding the sparsity level (N_0). We assumed that this parameter is known. Usually, some prior information is available about the system and data. Hence, the parameter may exist in the prior knowledge. Moreover, the parameter can be found using the cross validation frameworks. For instance, the system is trained and tested with different sparsity level (N_0). Then, the sparsity level which leads to the minimum training and testing error is considered as the optimum sparsity level.

The second point is about the polynomial approximations. The polynomial bases are not necessarily optimal. However, we need to consider some approximations to be able to track the problem. In this study, we have employed polynomial approximations because they convert the problem to a dictionary learning problem, and hence, the problem becomes tractable.

The last point is about the identifiability issues of the considered nonlinear model. Similar to the linear BSS problem where there are permutation and scaling ambiguities in retrieving the sources, it can be shown that in the nonlinear BSS problem stated in (1), the sources can be retrieved up to a permutation and an invertible function. The first ambiguity means that the order or the arrangement of the sources can not be found which is not important in BSS, while the second ambiguity is very important. The second ambiguity states that if we consider the invertible functions $g_n : \mathbb{R} \rightarrow \mathbb{R}$ for $n = 1, 2, \dots, N$ such

that $s_n = g_n(u_n)$, then, both of s_n and u_n are accepted estimations for the n^{th} source, because

$$y_m = f_m(s_1, s_2, \dots, s_N) = f_m(g_1(u_1), g_2(u_2), \dots, g_N(u_N))$$

$$m = 1, 2, \dots, M \quad (8)$$

This ambiguity makes the objective functions, used to find the sources in the nonlinear model, highly non-convex because there are infinite number of invertible functions. However, in the proposed method, due to the similarity of the factorization proposed in (3) and the considered constraints with the dictionary learning problem, it can be shown that the parameters are uniquely obtained up to a sign or a permutation. The reason that the second ambiguity does not exist in the proposed method is that we restrict the search space of the mixture functions to the polynomial functions with order K which can describe the mixture functions as best as possible. This means that the original sources (s_n for $n = 1, 2, \dots, N$) and the actual mixture functions (f_m for $m = 1, 2, \dots, M$) are not necessarily estimated in the proposed method, however, the sources (u_n for $n = 1, 2, \dots, N$) and the combination of the actual mixture functions (f_m for $m = 1, 2, \dots, M$) with the secondary invertible functions (g_n for $n = 1, 2, \dots, N$) which leads to the best polynomial functions of order K are retrieved in the proposed method.

III. Results

A. First Simulation

We assume that $M = N = 3$ and $N_0 = 2$, and then, generate the signals for 1000 samples using:

$$\begin{cases} y_1 = \alpha_1 \exp(s_1) + \alpha_2 \exp(2s_2) + \alpha_3 \exp(3s_3) + \beta_1 \\ y_2 = \alpha_1 \exp(2s_1) + \alpha_2 \exp(3s_2) + \alpha_3 \exp(s_3) + \beta_2 \\ y_3 = \alpha_1 \exp(3s_1) + \alpha_2 \exp(s_2) + \alpha_3 \exp(2s_3) + \beta_3 \end{cases} \quad (9)$$

where $\alpha_1 = \alpha_2 = \alpha_3 = 0.267$ are some coefficients to make the columns of \mathbf{F} which correspond to the first power of s_1, s_2, s_3 unit norm, and $\beta_1 = \beta_2 = \beta_3 = -0.801$ are used to make the simulated signals zero when the sources are all zero. The positions of the nonzero entries of the sources are randomly selected, and their values are chosen from uniform random variables distributed between $[-1 \ 1]$. The joint distribution of the sources and the generated signals are shown in Fig 1.

We consider $K = 5$ as the order of polynomial approximation, and then, apply the proposed method on the simulated signals. The joint distribution of the sources and the estimated sources are shown in Fig. 2.

As shown, the estimated sources are invertible, or in other words, one-to-one functions of the actual sources which shows that the BSS has been performed perfectly using the proposed method. It should be noted that the

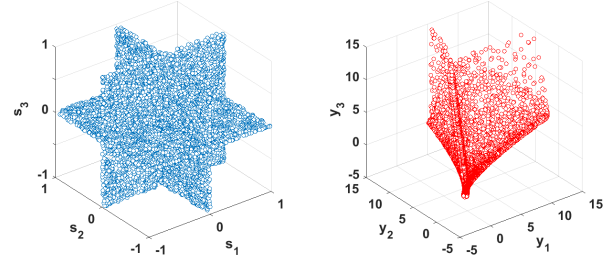


Fig. 1: The scatter plot of the sources (left) and the generated signals (right) for the problem stated in (9).

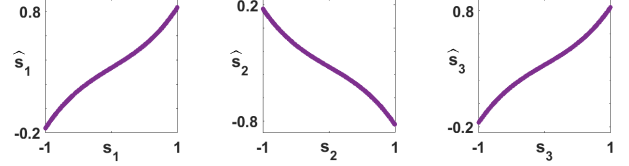


Fig. 2: The scatter plot of the sources and their estimations for the problem stated in (9).

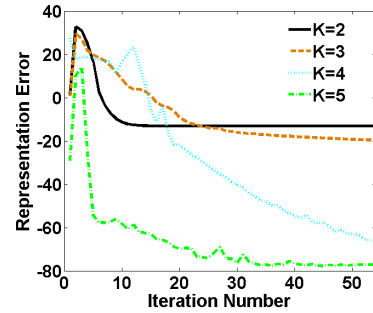


Fig. 3: The representation error (in dB) versus iteration number by considering different order for the polynomial approximations of the exponential mixture functions.

goal of BSS is separating the sources and not reconstructing the sources. Here, the separation of the sources has been performed. For reconstruction of the sources, we need more information about the sources or the structure of mixture functions.

To show the effect of choosing K on the performance of the proposed method, and also, to illustrate the convergence behavior of the proposed method, the representation error, i.e., $\|\mathbf{Y} - \mathbf{F}\tilde{\mathbf{S}}\|_F^2$, versus iteration number is shown in Fig. 3 for different K .

As shown, the representation error decreases when K increases because the exponential functions have Maclaurin series expansion, and they are better estimated using higher order polynomial functions. However, it should be noted that by increasing K , the computational complexity of the proposed method increases because the dimension of matrices \mathbf{F} and $\tilde{\mathbf{S}}$ increases.

TABLE I: MSE (in dB) in different SNR (in dB).

Method \ SNR	10	20	30	100
Method1	-14.3	-51.6	-94.3	-124.6
Proposed Method	-17.4	-53.6	-89.3	-122.5

B. Second Simulation

We respectively consider the proposed methods in [3] and [5] as Method1 and Method2 and compare their performance with the proposed method. It should be emphasized that the number of methods dealing with nonlinear BSS of sparse sources are so limited, and we selected the best methods in the field for comparison.

We individually compare the methods with the proposed method, because the methods cannot be applied on any type of mixture functions and easily generalized to any model orders. For each method, we tried to generate the data similar to the corresponding reference paper as explained in the following.

1) Comparison with Method1: We consider $M = 3$, $N = 3$ and $N_0 = 2$, and generate the outputs for $T = 1000$ samples using a LQ structure using:

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} -0.56 & 0.36 & -0.33 \\ 0.33 & 0.76 & 0.32 \\ 0.75 & 0.53 & -0.88 \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \\ s_3 \end{bmatrix} + \begin{bmatrix} 0.88 & -0.81 & 0.32 \\ -1.14 & -2.94 & -0.75 \\ -1.06 & 1.43 & 1.37 \end{bmatrix} \begin{bmatrix} s_1 s_2 \\ s_1 s_3 \\ s_2 s_3 \end{bmatrix} + \begin{bmatrix} n_1 \\ n_2 \\ n_3 \end{bmatrix} \quad (10)$$

The noises n_1 , n_2 , and n_3 are independent and identically distributed (i.i.d) Gaussian noise with $\mathcal{N}(0, \sigma^2)$ distribution. Since the LQ structure has also a polynomial form, we considered the columns of the first mixture matrix, which are corresponding to the first power of s_1 , s_2 , and s_3 , unit norm to estimate the actual sources. Hence, we use the mean squared error (MSE) criterion to evaluate the performance of the methods. It should be noted that the permutation and sign ambiguities must be omitted before calculation of MSE.

We apply Method1 and the proposed method (with $K = 2$) on the considered nonlinear model in different SNR. The obtained MSE over 100 trials in different SNR are reported in Table I. Moreover, the averages of convergence times over all of the trials and SNR are 5.8 *sec* and 11.3 *sec* for Method1 and proposed method, respectively.

As reported, the methods have similar performance, however, the computational complexity of the proposed method is higher than Method1. The main reason is that no prior information about the mixture functions is considered in the proposed method. When we set $K = 2$ as the order of polynomial approximations in the proposed method, the terms s_1^2 , s_2^2 , and s_3^2 also insert in the proposed factorization. Hence, it takes more time to find the optimum values of matrices \mathbf{F} and $\hat{\mathbf{S}}$.

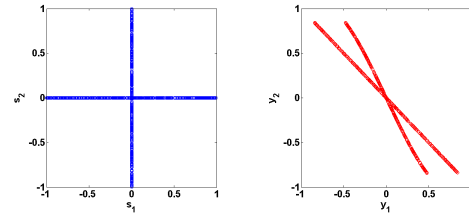


Fig. 4: The scatter plot of the sources (left) and the generated signals (right) when the mixture functions are sinusoidal functions.

TABLE II: Averages of N-ENF (in dB) in different SNR (in dB).

Method \ SNR _{dB}	10	20	30	100
Method2	-3.1	-48.6	-79.3	-99.6
Proposed Method	-16.3	-60.3	-83.1	-102.9

2) Comparison with Method2: We consider $M = 2$, $N = 2$, and $N_0 = 1$, and generate the outputs for $T = 1000$ samples using:

$$y_1 = \sin(s_1 - 0.5s_2) + n_1, \quad y_2 = \sin(s_2 - s_1) + n_2 \quad (11)$$

The noises n_1 and n_2 and the sources are generated similar to the previous sections. The joint distribution of the sources and the generated signals are shown in Fig. 4 when there is no noise.

We apply Method2 and the proposed method (with $K = 5$) on the considered nonlinear model in different SNR. For evaluating the performance of the methods, we first fit a nonlinear curve using smoothing splines to the joint distribution of the estimated sources and actual sources. Then, the average of the error of this fitting in different sources, which is called normalized error of nonlinear fit (N-ENF), is considered as the evaluation criterion [5]. The averages of the obtained N-ENF over 100 trials in different SNR are reported in Table II. Moreover, the averages of convergence times over all of the trials and SNR are 120.7 *sec* and 128.2 *sec* for Method2 and the proposed method, respectively.

As reported, the methods have similar computational complexity, however, the proposed method has better performance than Method2 especially in low SNR. The reason is that the data are first clustered in Method2, and then, some manifolds are fitted to the data, and this procedure does not repeat alternately. Therefore, since the initial clustering cannot be performed perfectly in low SNR, it severely affects on the performance of the manifold learning and the accuracy of the final results. Moreover, the main advantage of the proposed method relative to Method2 is that it can be generalized to higher order nonlinear models, while the implementation of Method2 is very difficult even in the three-dimensional space ($M = 3$). For this reason, all of the results presented in [5] are for two-dimensional data.

TABLE III: Number of components in $\tilde{\mathbf{s}}$ when $N = 15$ and $K = 3$.

Term	s_i	s_i^2	$s_i s_j$	s_i^3	$s_i s_j^2$	$s_i s_j s_k$	Total
Number of terms	15	15	105	15	210	455	815

TABLE IV: Percentage of successful recovery rate in different SNR_{dB} and N_0 .

$N_0 \setminus \text{SNR}_{dB}$	10	20	30	100
3	79.1	85.4	88.3	92.4
4	68.2	81.3	85.6	87.3
5	5.7	74.6	83.2	86.4

C. Third Simulation

In this simulation, we show the efficiency of the proposed method for identification of a nonlinear system when the inputs are sparse signals. We use the successful recovery rate criterion to report the results. Similar to dictionary learning problems, if we consider each column of \mathbf{F} as an atom, the successful recovery rate represents the number of correctly estimated atoms divided by the number of atoms. We say that an atom is correctly estimated if its cross correlation coefficient with the actual atom was above 0.99. In fact, this criterion shows the performance of the proposed method in retrieving the actual nonlinear systems.

We assume that the outputs, which are $M = 10$ dimensional signals, are generated by the nonlinear mixture of $N = 15$ sparse sources. We assume that the nonlinear mixture functions can be expressed using the polynomial approximations up to order $K = 3$. Based on the considered parameters, it can be easily shown that $\tilde{\mathbf{s}}$ has 815 entries as reported in Table III.

We generate the matrix \mathbf{F} by a random matrix of size 10×815 with zero-mean and unit-variance independent and identically distributed (i.i.d.) Gaussian entries, followed by normalizing the columns of \mathbf{F} which are corresponding to the first power of the sources. At each time instant, we randomly consider N_0 sources as the active sources. Then, the values of these sources are chosen from Gaussian random variables with zero-mean and unit variance. The output signals are generated according to (2) for 20000 samples. Finally, the zero-mean white Gaussian noise is linearly added to the outputs signals.

The average percentage of the successful recovery rate over 100 trials in different SNR_{dB} and N_0 is reported in Table IV.

As reported, when SNR_{dB} increases and N_0 decreases, the quality of the estimation of the nonlinear system increases. It is worth mentioning that none of the methods explained in the previous subsection can be employed for solving the considered nonlinear BSS problem due to the dimensions considered for the nonlinear system and sources.

IV. Conclusion

We developed an alternating projection method to solve the nonlinear BSS of sparse sources using the idea employed in dictionary learning problem. We confirmed the efficiency of the proposed method using simulated data in different scenarios. Moreover, we compared the proposed method with state-of-the-art methods in different simulations. The proposed method has three main advantages in comparison with other methods. The first advantage is that no prior information is required about the structure of mixture functions. The second advantage is regarding the performance of the proposed method. Since the clustering and manifold learning are alternately performed in the proposed method, the accuracy of BSS increases. Finally, the last advantage is that the proposed method can be generalized and applied on data with more than three dimensions.

References

- [1] P. Comon and C. Jutten, Handbook of Blind Source Separation: Independent Component Analysis and Applications. Elsevier, 2010.
- [2] L. T. Duarte, R. A. Ando, R. Attux, Y. Deville, and C. Jutten, "Separation of sparse signals in overdetermined linear-quadratic mixtures," in International Conference on Latent Variable Analysis and Signal Separation. Springer, 2012, pp. 239–246.
- [3] Y. Deville, "Sparse component analysis: A general framework for linear and nonlinear blind source separation and mixture identification," in Blind Source Separation. Springer, 2014, pp. 151–196.
- [4] M. Puigt, A. Griffin, and A. Mouchtaris, "Nonlinear blind mixture identification using local source sparsity and functional data clustering," in 2012 IEEE 7th Sensor Array and Multichannel Signal Processing Workshop (SAM), June 2012, pp. 481–484.
- [5] B. Ehsandoust, B. Rivet, C. Jutten, and M. Babaie-Zadeh, "Nonlinear blind source separation for sparse sources," in Signal Processing Conference (EUSIPCO), 2016 24th European. IEEE, 2016, pp. 1583–1587.
- [6] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in Signals, Systems and Computers, 1993. 1993 Conference Record of The Twenty-Seventh Asilomar Conference on. IEEE, 1993, pp. 40–44.
- [7] J. Heller and T. Pajdla, "Gposolver: a matlab/c++ toolbox for global polynomial optimization," Optimization Methods and Software, vol. 31, no. 2, pp. 405–434, 2016.
- [8] K. Engan, S. O. Aase, and J. H. Husoy, "Method of optimal directions for frame design," in Acoustics, Speech, and Signal Processing, 1999. Proceedings., 1999 IEEE International Conference on, vol. 5, 1999, pp. 2443–2446 vol.5.