Blind source separation with the low computational costs for the mobile and portable speech equipment

Kazunobu Kondo

Abstract

In this dissertation, frequency bin selection is proposed as a method to reduce the computational cost of blind source separation (BSS) based on frequency domain independent component analysis (FDICA). Clear voice quality is expected by users when communicating through speech processing equipment such as teleconferencing equipment and mobile telephones. Speech processing equipment is usually manufactured with embedded processors, especially digital signal processors (DSP), because of their low electric power consumption and the need for real-time processing. BSS has been widely investigated for use in speech enhancement applications for the purpose of obtaining higher voice quality, and FDICA is one of the most popular methods used by researchers to perform BSS. This is because acoustic conditions are usually reverberant, and FDICA is an inverse filter method, and thus minimizes reverberation influence. However, the computational cost of FDICA is quite high because FDICA estimates inverse filter coefficients in each frequency bin using an optimization scheme employing higher-order statistics, usually an iterative update algorithm. Current DSPs can achieve high levels of performance, however the appropriate choice of a DSP strongly depends on the specifications of the speech processing equipment in which it is to be used. For mobile devices, due to battery life considerations, low-powered DSPs are normally used. ICA stores long-term observed signals because its optimization scheme employing higher-order statistics, so that the external memory usually consists of dynamic random access memory (DRAM). However, the required wait states of DRAM is another issue, because of the waste of electric power this involves. In addition, the amount of power wasted is not negligible. Therefore, implementing FDICA using embedded processors is difficult, despite the high performance of current DSPs. The proposed method aims at reducing computational cost by reducing the number of frequency bins in

which the ICA algorithm is performed, thus reducing computational cost and increasing the feasibility of implementing FDICA in embedded processors. The proposed frequency bin selection method utilizes spatial correlation; the determinant of the spatial covariance matrix or the magnitude squared coherence (MSC) between two microphones.

Two types of frequency bin selection are proposed for FDICA, one for mobile telephone devices and one for portable speech processing equipment, such as portable teleconferencing systems. For mobile devices, the determinant of the spatial covariance matrix is used to select the frequency bins. The determinant is theoretically analyzed, since its characteristics simultaneously reflect both directional information as well as the power of the source signals. In other words, signal separability can be evaluated using the determinant. In the unselected frequency bins, a Wiener filter is obtained using the tentatively separated signals, which are the output of the null-beamformers. Use of the separated signals results in improved performance, because this cancels out the signal distortion caused for acoustic reasons by microphone array signal processing. Performance, as measured by the segmental signal-to-noise ratio, is experimentally evaluated, and significant improvement is achieved. Cepstral distortion is also employed to evaluate performance, but results show deterioration in performance instead of improvement as the number of frequency bins selected is reduced. The trade-off between these two measures of distortion is used as a criterion to determine the number of frequency bins selected, and it is determined that 64 bins should be selected. Compared to conventional FDICA methods, the proposed method achieves a more than 80 percent reduction in computational cost. Despite this large reduction in computational cost, the segmental signal-to-noise ratio is improved by about 2 dB, while deterioration in cepstral distortion is restrained to only about 1 dB.

For portable speech equipment, a dodecahedral microphone array (DHMA) is used as a small, agglomerative sound capture system. DHMAs have ten faces, with six microphones installed on each face. When performing BSS using DHMAs, FDICA is executed under overdetermined conditions, since the number of the microphones exceeds the number of source signals. The size of the FDICA separation matrix when performing BSS with a DHMA is quite large, making this approach extremely computationally expensive. The

permutation solution requires especially large computational resources because its clustering method involves a large number of similarity calculations. Magnitude squared coherence is utilized to select frequency bins for analysis. The shape of a DHMA is similar to that of a spherical microphone array, but its acoustic characteristics are somewhat different due to its many flat faces. Two arbitrarily chosen microphones are used to calculate magnitude squared coherence, and frequency regions are evaluated on this basis, with lower MSC values indicating the separable frequency region. Frequency bin selection is not uniformly spaced when using the proposed method. The resulting reduction in computational cost exceeds 80 percent, even with only a 68 percent reduction in the number of frequency bins selected. Separation performance, as measured by the signal-to-interference ratio, deteriorates, however, compared to FDICA which frequency bins are uniformly selected, the deterioration is restrained by about 1 dB. As for distortion, the segmental signal-to-noise ratio deteriorates slightly, although cepstral distortion is improved. For these distortion measures, compared to methods in which the frequency bins are uniformly selected, both measures are improved when using the proposed method.

Both frequency bin selection methods calculate the spatial correlation between only two microphones, and these spatial correlations are categorized as second-order statistics. This implies that BSS based on FDICA can become more computationally efficient, while maintaining the signal distortion level, through the combination of second-order statistics. When speech processing equipment is manufactured with DSPs, the required wait states of the DRAM external memory is an important issue to consider when implementing FDICA. FDICA must store long-term observed signals in DRAMs because the high-speed internal memory in the DSPs should not be taken up just to store these signals. By reducing the number of frequency bins selected, the proposed method also reduces required memory consumption, which restrains the amount of memory access required. The computational costs of conventional FDICA are clarified, and a target number of operations for the use of FDICA with internal DSPs are estimated: 200 mega-operations for mobile devices and 160 giga-operations for portable equipment. Calculations are made which show that the proposed method does not exceed the target computational cost. The function responsible for the dominant computational expense changes according to the number of frequency bins selected, indicating to engineers which software function should be optimized. Taking

into consideration objective measures used to evaluate popular speech enhancement applications, about 10 to 20 dB is the allowable value for the absolute signal-to-interference ratio. Even though separation performance deteriorated when using the proposed methods, performance remained in the range allowable for practical commercial applications. On the other hand, measures of distortion remained at equivalent levels, while still achieving a more than 80 percent reduction in computational cost. As a consequence, the proposed method involves a practical trade-off between separation performance and computational cost on the one hand, and a quite advantageous trade-off between signal distortion and computational cost on the other. These findings indicate that the proposed BSS methods are practical, and that they are acceptable for use in speech processing equipment with embedded processors. Future work includes more investigation into the separation method used for the unselected frequency bins, as well as into on-line implementation.

Acknowledgments

I would sincerely like to thank my academic advisor, Professor Kazuya Takeda, for his guidance during my Ph. D. studies, and for his many valuable suggestions and occasional pointed corrections, which helped advance this research. The encouragement he gave me during the entire process will always be appreciated.

I would also like to thank the members of my dissertation committee; Professor Hiromi Nakaiwa, for his careful reading of the first draft of this study and for his helpful comments, and Professor Norihide Kitaoka, for his helpful guidance, encouragement and suggestions during the entire process.

Additionally, I would like to thank Professor Takanori Nishino for his helpful technical suggestions. The encouragement he gave me will always be appreciated.

I extend my deep appreciation to professors, staff, and students of Nagoya University's Takeda Laboratory, for their support and friendship, and for the many fruitful discussions we have had over the years. I would especially like to thank Yusuke Mizuno, a fellow student at the Takeda Laboratory who collaborated with me on the latter part of this research.

I would also like to thank the members of the Research & Development Division of the Yamaha Corporation for the interesting discussions we had regarding this research. I am especially thankful to Dr. Yu Takahashi, a colleague at Yamaha, for our valuable discussions.

I also wish to express my appreciation to Professor Hiroshi Saruwatari of the Nara Institute of Science and Technology for giving me valuable and fruitful suggestions regarding conference and journal paper submissions.

And last, but not least, I am grateful to my family and dear friends, for their encouragement and support, which allowed me to follow my dreams all these years.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Preface

Speech communication and speech recognition systems are now widely in use, generally under reverberant and noisy conditions. Originally, the Internet was intended to mainly be an infrastructure for a text communication, i.e., e-mail and hypertext browsing. However, in the last few decades the Internet has evolved into a multi-media environment. And in the last decade, it has begun to be widely used as a communication infrastructure for Internet telephony of speech and real-time video, using PCs, teleconferencing equipment, or mobile devices. Speech communication equipment can be used under many different acoustic conditions. Teleconferencing equipment, for example, is commonly used in reverberant meeting rooms. Mobile devices, however, are used outside meeting rooms, where the acoustic conditions are not reverberant, but noisy. Noise can be produced by a large variety of sound sources, such as air conditioners, projectors, or the voices of people who are not the target speakers. Both noise and reverberation degrade the quality of speech, but high speech quality is needed to achieve clear speech communication or reliable speech recognition results. The noise from an air conditioner or a projector is called stationary noise, which is noise that is relatively constant in nature in terms of energy, pitch, location, and onset time. Noise reduction methods such as spectral subtraction [3] can reduce stationary noise, and these methods have achieved successful results for the last forty years. However, for non-stationary noise, these noise reduction methods can not work appropriately.

1.2 Blind source separation

At a microphone, target speech and the speech of other speakers are captured simultaneously, resulting in a mixed signal. In such cases, the speech of other speakers is considered to be noise, because the speech of the other speakers interferes with the target speech signal. Such non-stationary noise as speech from other speakers. can not be reduced by noise reduction methods. To cope with this problem, blind source separation (BSS) is considered widely to be used for speech enhancement applications, because it can separate the mixed signal into individual sound sources without any advance information.

1.2.1 Types of blind source separation

BSS can be broadly classified into two categories: time-frequency masking (TFM) and inverse response methods. The TFM method is commonly used to extract sparse signals in under-determined conditions, i.e., when the number of source signals exceeds the number of microphones. Sparse signal representation usually assumes that only one source signal has a time-frequency component, for example, speech signals. One of the most researched TFM methods is the degenerate unmixing estimation technique (DUET) [4]. DUET evaluates the directions of arrival (DOA) of the source signals and their relative amplitudes to make clusters that correspond to individual sources. After clustering the source signals, a binary mask or Wiener filter is used to separate the observed signals in the frequency domain. The DUET concept has been continuously refined, resulting in various methods which have been proposed in recent years. For example, DOA and a time-frequency mask are iteratively estimated using the expectation maximization (EM) algorithm [5]. Another approach is the use of k-means clustering, which involves evaluation of normalized amplitude and phase differences [6] when there are more than three microphones, even if the microphones are positioned non-uniformly. DUET-based techniques achieve significant separation performance, which is further improved by sparse signal representation. Sparse signal representation is effective when acoustic conditions are anechoic. The inverse response method, however, does not assume that the source signals are sparse. When acoustic conditions are reverberant, the temporal envelope of the source signal is ”smeared” by reverberation over neighboring time-frequency components. In fact, acoustic conditions

are usually reverberant, sparse signal representation methods work less effective under reverberant conditions than the inverse response methods. In other words, methods based on inverse response are more appropriate than TFM for sound source separation.

1.2.2 Indepedent component analysis

Prior to 1990, much of the research on BSS was based on nonlinear decorrelation [7–9]. In the 1990s, it was clarified that nonlinear decorrelation was strongly connected with independent component analysis (ICA), and a BSS technique based on ICA was widely researched. ICA assumes that source signals are statistically independent, and this statistical independence is used to separate mixed signals. Statistical independence can be evaluated on the basis of non-Gaussianity [10, 11]. A probability distribution of the sum of independent random variables gradually approaches a Gaussian distribution, according to the central limit theorem. Therefore, non-Gaussianity can be used as a measure of the statistical independence of separated signals.

One measure of non-Gaussianity is kurtosis. Data sets with high kurtosis tend to have heavy tails with its probability distribution, in other words, infrequent observations. Kurtosis can be used for ICA [12–14]. When the probability distribution of a source signal is super Gaussian, the source signal's kurtosis is higher than the kurtosis of a Gaussian distribution. If independent, super Gaussian signals were mixed together, a probability distribution of the mixed signal would approach Gaussianity according to the central limit theorem. In other words, mixing independent signals results in a decrease in the mixed signal's kurtosis. Therefore, maximizing a separated signal's kurtosis corresponds to maximizing its non-Gaussianity, thus mixed signals can be separated into separate source signals using ICA. Kurtosis is calculated using the second and fourth moments. Because kurtosis is very sensitive to outliers when it is being calculated, large errors tend to occur, which is one disadvantage of using kurtosis as a means of evaluating statistical independence.

Statistical independence can also be evaluated using negentropy [15] as another measure of non-Gaussianity. The other measures which do not evaluate non-Gaussianity, such

as the Kullback-Leibler divergence, maximum likelihood estimation (MLE) [16–18], infomax [19] and mutual information [15, 20], can also be used for ICA. The ICA algorithms which use these measures assume specific probability distributions of the source signals. Entropy expresses the degree of statistical randomness. Negentropy measures the difference in entropy between a given distribution and the Gaussian distribution. If a random variable would concentrate at a specific value, entropy would be small and the peakedness of the probability distribution would be high. Low entropy corresponds to high non-Gaussianity, therefore negentropy can be used to evaluate statistical independence. The Kullback-Leibler divergence measures the difference between two probability distributions. In the case of ICA, if the separated signals were statistically independent, the Kullback-Leibler divergence would become zero. Therefore, statistical independence can be evaluated by minimizing the Kullback-Leibler divergence. MLE optimizes separation coefficients by using probability distributions of the source signals. Likelihood in MLE consists of the mixed signals and the separation coefficients. The mixed signals remain constant during the MLE process, therefore the likelihood of the separated signals is only changed by the separation coefficients. Therefore, maximizing likelihood corresponds to evaluating the statistical independence among the separated signals using the separation coefficients. Note that MLE can be viewed theoretically as a process to minimize a Kullback-Leibler divergence between two distributions. Infomax is an optimization principle for neural networks. In the case of ICA, infomax maximizes joint entropy of the separated signals, which corresponds to maximizing statistical independence among the separated signals. Note that infomax for ICA [19] is the same as MLE when the differential of the nonlinear function for infomax equals the probability distribution function of MLE. Kullback-Leibler divergence, MLE and infomax all employ nonlinear correlation among the separated signals as the criterion of statistical independence. These criteria have been utilized in ICA algorithms for the last twenty years.

Gradient and fixed-point algorithms are used for nonlinear optimization, such as when evaluating statistical independence. Note that a natural gradient improves an algorithm's efficiency and stability [21, 22]. Gradient and fixed-point algorithms are iterative update rules. During the iterative update stage, statistical independence is evaluated using matrix

multiplication to calculate nonlinear correlation among the separated signals. The separated signals are calculated using separation coefficients and the mixed signals. These separation coefficients are expressed in a matrix, which is called the separation matrix. The separation matrix changes with every iterative update, so that the separated signals must be recalculated with every update. This leads to a large amount of matrix multiplication. Nonlinear correlation in the ICA algorithms must be calculated using separated signals with long temporal lengths, which means that matrix multiplication must be performed for each time slot. Therefore, during each iterative update, and for each time slot, matrix multiplication in ICA algorithms must be performed. The resulting high computational cost is one disadvantage of ICA.

Acoustic conditions are often reverberant, as mentioned in Section 1.1. Reverberation corresponds to an acoustic transfer function from a sound source to a microphone. An acoustic transfer function is also called a room impulse response (RIR), which consists of direct and reflected sounds. The time delay of the direct sound corresponds to the distance from the sound source to the microphone. Reflected sounds are observed after the direct sound, and this convolution of the source signal and the RIR results in a reverberant signal.

Is is commonly assumed when using ICA that several microphones will be used to separate the sound source signals, which are transfered to the microphones through different acoustic paths. At one microphone, the sound signals are observed and mixed simultaneously. If an acoustic condition is anechoic, the mixing condition is called an instantaneous mixture. If an acoustic condition is reverberant, the mixing condition is called a convolutive mixture. ICA research began focusing on instantaneous mixtures in the 1990s [10, 11]. Regarding convolutive mixtures, in the late 1990s Murata and Smaragdis proposed frequency domain ICA (FDICA) [23, 24]. After 2000, FDICA researchers concentrated on obtaining higher performance [25–30]. Another approach, time domain ICA (TDICA) using a finite impulse response (FIR) filter, has also been studied for use with convolutive mixtures [10, 31–34]. In the case of TDICA, the separation matrix consists of a FIR filter matrix, and the FIR filters correspond to the inverse RIRs. TDICA was compared with FDICA by Nishikawa [35], whose experiments showed that TDICA's separation performance was lower for convolutive mixtures, due to the length of the FIR filters. This was because RIRs usually last hundreds of milliseconds, and FIR filter length can easily exceed

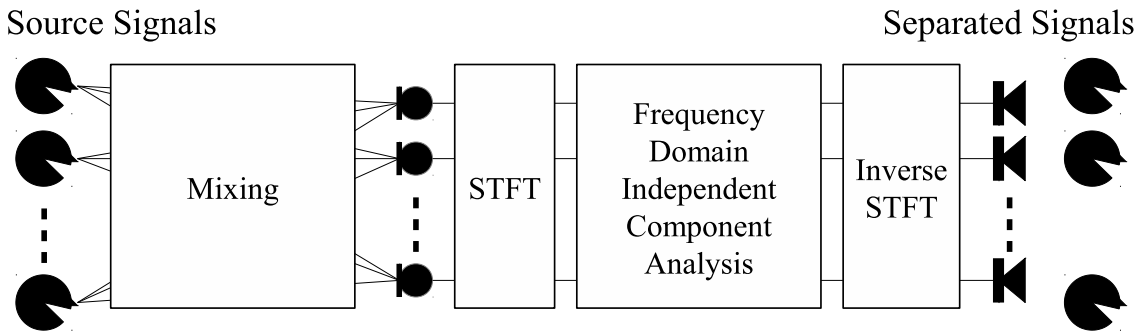

Figure 1.1: A block diagram of BSS using FDICA. Source signals are mixed and observed at a microphone array. FDICA separates them in the frequency domain.

2,000 samples when the sampling frequency is 8 kHz and the length of the RIR is 250 milliseconds, for example. This is a serious drawback for sound source separation, therefore FDICA is a more appropriate BSS method for the convolutive mixtures which occur under reverberant conditions.

ICA assumes statistical independence among the source signals, and FDICA additionally assumes statistical independence among the frequency bins. A block diagram of BSS using FDICA is shown in Figure 1.1. The sound signals are transfered from their source positions to the microphones, and the signals are mixed at the microphones. The observed signals are transformed from the time domain to the frequency domain by Short Time Fourier Transform (STFT). ICA separates the observed signals into individual sound signals in each frequency bin. An Inverse Short Time Fourier Transform (ISTFT) transforms the separated signals back into time domain signals. ICA must overcome two problems; scaling and permutation. The scaling problem refers to amplitude ambiguity among the separated signals, because the amplitude of the separated signals is not necessarily equal to the amplitude of the source signals. If FDICA's scaling problem is not resolved, then the scales of the separation matrix will be different in different frequency bins. The permutation problem refers to ambiguity regarding the order of the separated signals. In the case of FDICA, the order may vary in different frequency bins. If the permutation problem of FDICA is not solved, then a separated signal will include different source signals in different frequency bins. Both problems result in deterioration of separation performance.

Therefore, both problems must be solved, and the solutions lead to additional computational costs. Solving the scaling and permutation problems is discussed in Section 2.2.4.

As a conclusion of the overview of BSS using FDICA, FDICA involves considerable computational costs with respect to the iterative update rule of the ICA algorithms, and due to solving the scaling and permutation problems. Therefore, computational cost becomes a critical issue when FDICA is used. Methods of reducing the computational cost of both matrix multiplication during iterative updating, and of solving the scaling and permutation problems, should be considered.

1.3 Implementation issues of FDICA and review of embedded processors

BSS using FDICA can be a standard method used by speech enhancement applications under reverberant acoustic conditions. From the viewpoint of stress-free communication, as little time delay as possible is important, especially for long-distance conversations using speech communication equipment. The central processing units (CPUs) of modern personal computers are very powerful. But when FDICA is being performed, the time delay is difficult to control because many device drivers and applications operate concurrently. In order for real-time systems to make use of low time delay applications, operating systems are often designed with embedded processors. Embedded processors are dedicated to specific tasks, so design engineers can optimize them to increase reliability and performance. But embedded processors can also restrict electric power consumption, resulting is less computational power.

In addition, speech communication equipment is often portable or mobile. Since mobile devices are designed to be quite small, their batteries are also small, however a battery's capacity drops as its size decreases. This means that battery size becomes another critical issue for mobile devices. In order to achieve longer operating times, electric power consumption must be extremely restrained. Therefore, reducing computational costs would be helpful, especially when embedded processors are used in portable teleconference equipment or mobile devices. When FDICA is being performed by embedded processors to

carry out BSS, the computational cost of FDICA is fairly high, and must be reduced.

Representative computational cost can be defined as the sum of the required number of operations and the required amount of memory consumption. For example, the number of operations can be converted into computation time by taking into account the operating speed of a processor, and this can be helpful to select an appropriate processor. Required memory consumption can determine the balance between system requirements and manufacturing costs, with smaller memory consumption requirements leading to lower manufacturing costs.

Digital signal processors (DSP) are a type of embedded processor widely used for speech signal processing. Most DSPs are designed using Harvard architecture, which means there are physically separated memories, some storing instructions and others storing data, with physically separated buses to transfer them. Another common type of embedded processor are microprocessors such as ARM architecture processors, Qualcomm Snapdragon, nVidia Tegra, etc. These microprocessors are designed using Von Neumann architecture, in which there is only one memory, which stores both instructions and data, and a single bus to transfer them both. Speech processing equipment is usually designed as a real-time system. Because voices vary from moment to moment, speech signals are constantly being processed by the DSP, from input to output, which leads to frequent data access. Harvard architecture is more appropriate for this kind of real-time application, due to its separated memories and buses.

The operating speeds of currently available DSPs and microprocessors have become quite high; 1 GHz, for example. As a result, the processing performance of recently developed embedded processors is quite high. For the microprocessors, this high operating speed seems to make up for the disadvantage of the microprocessor's architecture, even though the architecture of DSPs is more advantageous for real-time systems. This implies that DSPs may no longer be required, and that microprocessors could now be used. On the other hand, dynamic random access memory (DRAM) is widely used as the external memory in computing devices. DRAM is inexpensive and large amounts of memory space are available, but DRAM operating speed is very slow; still around a few hundred MHz, which is about one-fifth the working speed of embedded processors. In addition, DRAM has a disadvantageous data access scheme, involving a number of waiting cycles due to

its architecture, causing data transfer to be delayed between the processor and the external memory. Therefore, when using external memory, slow operating speed and waiting cycles reduce processor performance. The advantages of the Harvard architecture make up for part of this problem, suggesting that DSPs should be used for BSS applications instead of microprocessors. While DSPs are waiting, their calculating units are still operating, and this electric power consumption is wasted due to slow data transfer to and from the external memory. These points suggest that reducing the memory consumption required for FDICA would be very useful.

If we can reduce the required number of operations and the required amount of memory consumption needed to perform FDICA, we may be able to achieving the goal of performing FDICA with DSPs. The computational cost of FDICA is discussed in Section 2.2, and target levels for number of operations and required memory consumption are discussed in Section 2.3.

1.4 Purpose of this dissertation

In this dissertation, a method of performing FDICA with lower computational costs, using frequency bin selection, is proposed. This lower computational cost is helpful for performing BSS using FDICA with embedded processors, since the proposed method is intended for use in portable equipment and mobile devices. Since FDICA estimates the separation matrix in every frequency bin, the proposed method restrains the number of operations by reducing the number of frequency bins. FDICA must also store long temporal signals in order to estimate the separation matrix, and these signals are stored in every frequency bin. Thus, the proposed method also reduces memory consumption required for FDICA. This reduction in memory consumption also reduces the number of slow data transfers to and from the external memory.

The frequency bins are reduced using criteria based on spatial correlation. First, it is proposed that the determinant of the covariance matrix be used to select the frequency bins in cases in which two microphones are used. Using the power spectrum would appear to be the obvious way to select the frequency bins, however the power spectrum does not include any spatial information. The spatial covariance matrix includes spatial cross-correlation in

its off-diagonal elements. The determinant includes the off-diagonal elements, so that the determinant can be used to evaluate the spatial information. In addition, the determinant is a real number, and this fact is assured by the covariance matrix. It is theoretically and experimentally clarified how these properties of the determinant make it useful for frequency bin selection. Through a BSS experiment, using this method is determined that frequency bins can be appropriately selected for FDICA. Second, the use of magnitude squared coherence (MSC) is proposed to assist in selecting the frequency bins. MSC corresponds to the normalized cross-correlation coefficients between two microphones. When the shape of a microphone array is complex, deviation tends to occur between the theoretical model and the actual properties of the microphone array. If this type of deviation occurs, MSC can still be calculated experimentally. Therefore, the proposed method employs experimental MSC to select the frequency bins.

1.5 Structure of this dissertation

In Chapter 2, BSS using FDICA is introduced at first, and for FDICA functions and two types of target speech processing equipment, computational costs are discussed. Section 2.1.1 introduce a signal model and FDICA functions including an update rule of ICA. Section 2.1.2 introduces a dodecahedral microphone array and the conventional BSS method using DHMA. Section 2.2 discusses computational costs for each FDICA function. For an iterative update rule, computational costs are discussed, in addition, faster convergence methods are reviewed. Computational costs of clustering methods are discussed because permutation solution usually consists of a clustering method. Through the discussion in Section 2.2, the total computational cost of conventional FDICA is concluded. Section 2.3 introduces current digital signal processors into which the proposed method can be implemented, and discusses computational costs for the target speech equipment.

In Chapter 3, for two microphones, a computationally efficient BSS method is proposed. Section 3.1 introduces motivation and strategy for this research. Section 3.2 presents a frequency bin selection method to reduce computational costs, and also presents a theoretical analysis of the selection criteria, which is the determinant of a spatial covariance matrix. A frame-wise Wiener filter is proposed in the same section for source separation in unselected

frequency bins. Section 3.3 shows experimental results and estimate of computational costs for the case of two microphones. In Section 3.4, the proposed method is discussed and summarized.

In Chapter 4, for dodecahedral microphone array (DHMA), a computationally efficient BSS method is proposed, which utilizes the magnitude squared coherence. Section 4.2 presents the proposed method. In Section 4.2.1 and 4.2.2, the magnitude squared coherence (MSC) is introduced, and its characteristics are investigated experimentally. Introducing the proposed frequency bin selection in Section 4.2.3, frequency bins are non-uniformly selected using the experimentally evaluated MSCs. From Section 4.2.4 to 4.2.6, the subspace method and ICA in the selected frequency bins are introduced. In Section 4.2.7, the estimated separation matrices are interpolated to generate interpolated separation matrices in unselected frequency bins. Section 4.3 shows the experimental results and estimate of computational costs. In Section 4.4, the proposed method is discussed and summarized for the BSS method using DHMAs.

Chapter 5 discusses characteristics of the proposed methods, important issues to implement the proposed methods, and remaining issues. The spatial correlation represents the characteristics of the proposed method. For considering the implementation issues, the separation performance and the distortion measures are quantitatively discussed to clarify the evaluated performance, and the estimated computational costs from practical points of view. Remaining issues, in other words future recommendations, are discussed from the points which focus on separation matrices in unselected frequency bins and on-line methods.

Finally, Chapter 6 concludes this dissertation.

Chapter 2

Computational cost of frequency domain independent component analysis

BSS using FDICA improves speech quality for both portable and mobile speech communication equipment, which are usually used under reverberant acoustic conditions. This chapter briefly introduces conventional FDICA and discusses its computational cost. An ICA algorithm is an iterative update rule which involves matrix multiplication during every iteration. With FDICA, the ICA algorithm is applied to every frequency bin, with matrix multiplication during every iteration. In addition, the scaling and permutation problems must be solved. The conventional scaling solution involves an inverse operation using the matrix, and the permutation solution employs a clustering method. Therefore, the number of operations required is extremely large, and the computational cost of the iterative update rule, the permutation solution and the scaling solution are discussed. It is assumed that FDICA is implemented within embedded processors, as mentioned in Chapter 1. The processing performance of embedded processors is currently lower than the computational cost which conventional FDICA needs, so the performance of current embedded processors is also discussed in this chapter. A target computational cost is determined by taking into account the estimated computational cost of FDICA and the performance of current embedded processors.

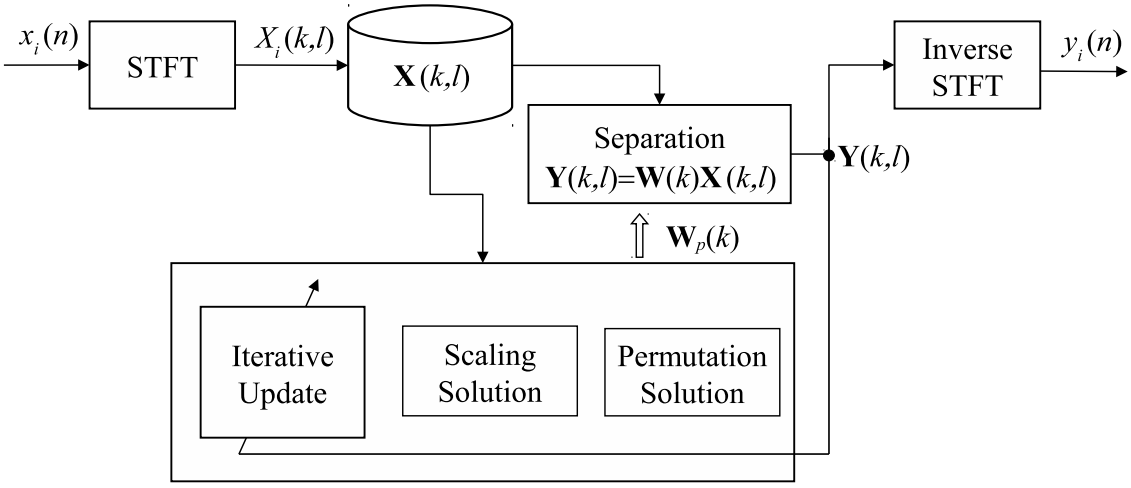

Figure 2.1: Block diagram of conventional FDICA

2.1 Blind source separation using FDICA

2.1.1 Signal processing for FDICA

Figure 2.1 shows a block diagram of FDICA. The observed signals $x_i(n)$ are transformed from the time domain to the frequency domain. n and i represent the sample index and the microphone number, respectively. In Section 1.2.2, convolutive mixtures were introduced. In the frequency domain, a convolutive mixture is formulated as:

$$\mathbf{X}(k,l) = \mathbf{A}(k)\mathbf{S}(k,l), \tag{2.1}$$

where $\mathbf{X}(k,l)$ is an observed signal vector which consists of $X_i(k,l)$. k and l represent the frequency bin index and the frame index, respectively. $\mathbf{S}(k,l)$ is a source signal vector, and $\mathbf{A}(k)$ is the room impulse response in the frequency domain. $\mathbf{A}(k)$ is also called the mixing matrix. Observed signals $X_i(k,l)$ are stored for a designated time period. The stored signals are used to obtain separation matrix $\mathbf{W}(k)$ using the ICA algorithm. In this dissertation, the iterative update rule [23] is used to obtain the separation matrix, which is formulated as:

$$\mathbf{W}_{p+1}(k) = \mathbf{W}_p(k) - \eta \cdot \text{off-diag}\left\{E_l\left[\varphi\left(\mathbf{Y}(k,l)\right)\mathbf{Y}^{\mathrm{H}}(k,l)\right]\right\}\mathbf{W}_p(k), \tag{2.2}$$

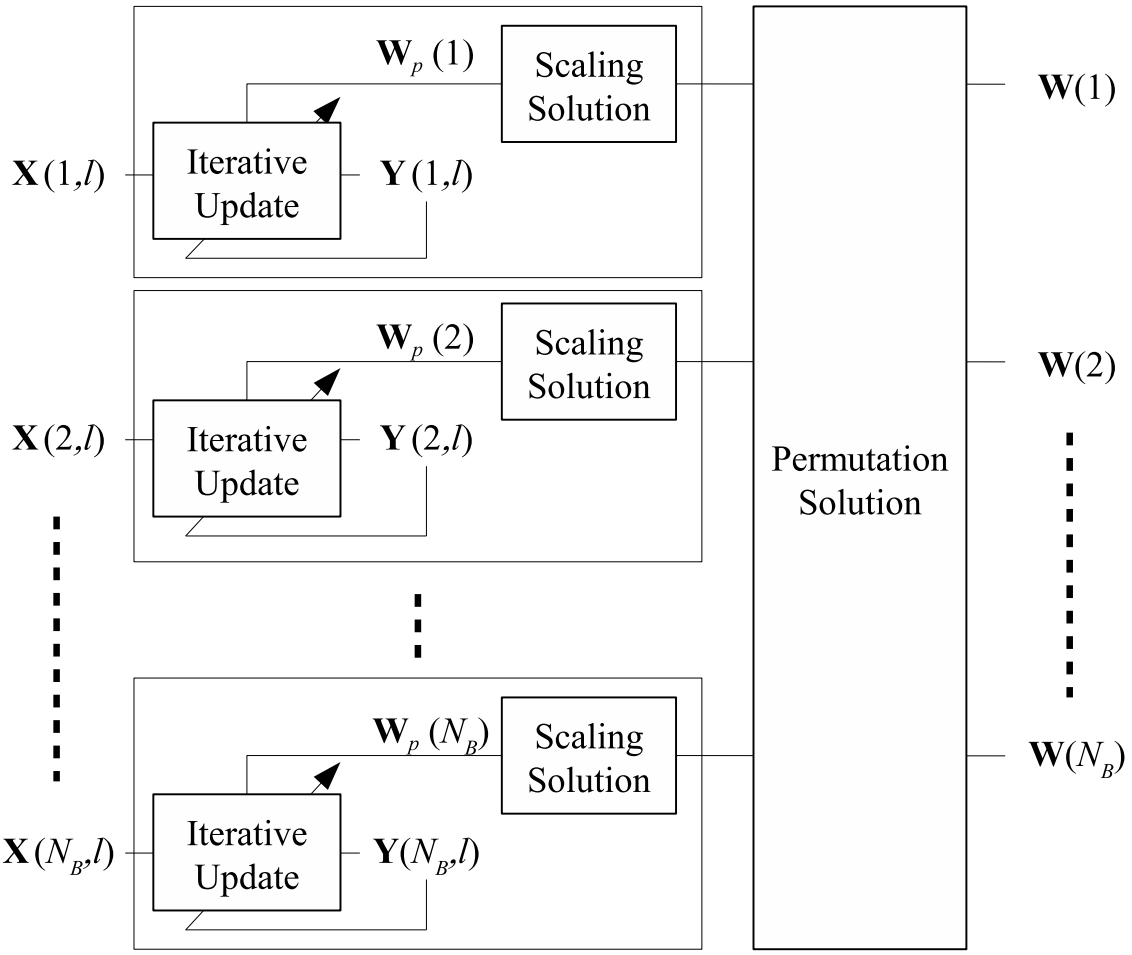

Figure 2.2: Block diagram describing bin-wise process which is the separation matrix estimation of FDICA. An iterative update algorithm is performed, and the scaling problem is solved in each frequency bin $1, 2, \cdots, N_B$. Separation matrices for all frequency bins are input into a permutation solution. $\mathbf{X}(k, l)$ and $\mathbf{Y}(k, l)$ denote observed and separated signals. k and l are the frequency bin and frame indexes. $\mathbf{W}_p(k)$ denotes a separation matrix, and p represents an iteration number. The separation matrix finally obtained is denoted as $\mathbf{W}(k)$.

where p is an iteration number, and η is a step-size. off-diag$(\cdot)$ denotes the operator at which all diagonal elements are set to zero. $\varphi(\cdot)$ denotes a nonlinear function which is described in the next paragraph. The scaling and permutation problems are solved for separation matrix $\mathbf{W}(k)$ after the convergence of the iterative updates. The scaling and permutation solutions are discussed in this section and in 2.2.4. Applying the separation matrix to the observed signals, the separated signals can be represented as:

$$\mathbf{Y}(k,l) = \mathbf{W}(k)\mathbf{X}(k,l), \tag{2.3}$$

where $\mathbf{Y}(k,l)$ is a separated signal vector.

In Eq.(2.2), $\varphi(\cdot)$ denotes a nonlinear function. For sound source separation using FDICA, the nonlinear function is usually a sigmoid function such as a complex hyperbolic tangent in the Cartesian coordinate or the polar coordinate [36]. The sign function sgn$(\cdot)$ is the most computationally efficient sigmoid function, and it extracts the sign of a real number. Because we are trying to develop a BSS method with low computational cost, the sign function is likely to be the most appropriate sigmoid function. The nonlinear function $\varphi(\cdot)$ is defined as:

$$\varphi(Y) \equiv \operatorname{sgn}(\mathrm{Re}\{Y\}) + j \operatorname{sgn}(\mathrm{Im}\{Y\}), \tag{2.4}$$

where Re$\{\cdot\}$ and Im$\{\cdot\}$ denote the real and imaginary parts of a complex number, respectively.

A bin-wise FDICA procedure is shown in Figure 2.2. The scaling problem is solved in every frequency bin. One common scaling solution is the projection method [37]. The scale of the separation matrix is corrected by the calculation of diag$\{\mathbf{W}^{-1}(k)\}\mathbf{W}(k)$ or diag$\{\mathbf{W}^{+}(k)\}\mathbf{W}(k)$ by the projection method. diag$(\cdot)$ is the operator which retains the diagonal elements, but which sets all of the other elements to zero. $(\cdot)^{-1}$ and $(\cdot)^{+}$ represent the inverse matrix and the pseudo-inverse matrix, respectively. The permutation problem can be solved for all the separation matrices in all the frequency bins, and for all the separated signals. Clustering methods, such as generalized Lloyd's algorithms [38], k-means clustering, or hierarchical clustering [2] are common permutation solutions.

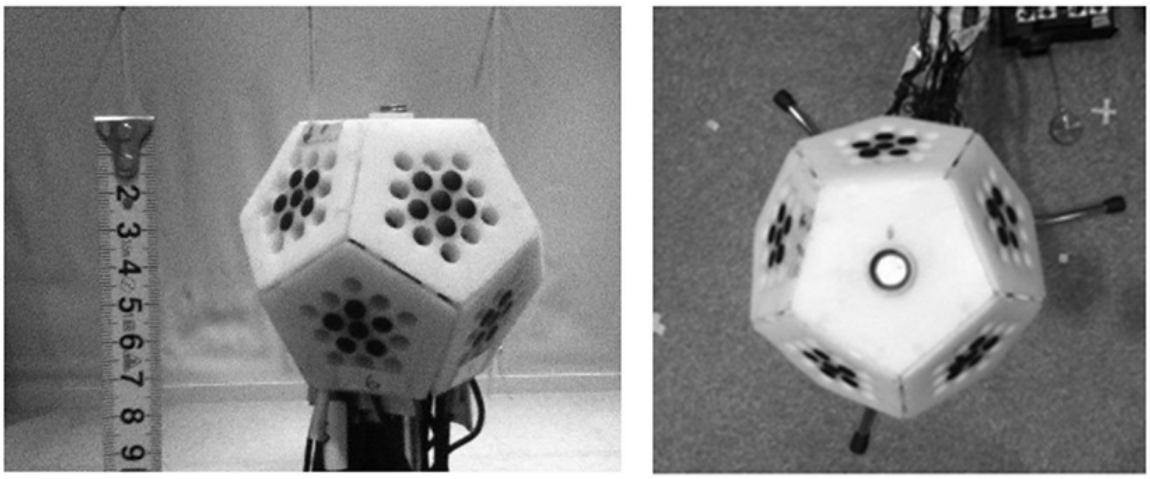

Figure 2.3: Dodecahedral microphone array (DHMA). The diameter of the microphone is about 7 cm. Microphones can be installed on ten faces, excluding the top and bottom. Six microphones were installed for the experimental evaluation.

2.1.2 Portable microphone arrays and blind source separation

Dodecahedral microphone array

Use of a dodecahedral microphone array (DHMA) [1, 2] has been proposed for portable speech communication equipment. In this section, DHMAs are briefly described. Figure 2.3 shows a DHMA. Microphones are installed on ten faces, excluding the top and bottom faces. Sixteen holes appear on each face of the DHMA. Even though DHMAs can be small and portable, up to 160 microphones can be installed. The acoustic properties of DHMAs are different from those of spherical microphone arrays regarding sound pressure levels, arrival times and the influence of diffraction waves, for example. For the experimental evaluation in Chapter 4, six small, omni-directional microphones (SONY ECM-77B) were installed on each face of a DHMA. The total number of microphones used was sixty. Because it is difficult to adjust a large number of microphones, microphone gains were adjusted manually in our BSS using a DHMA experiments.

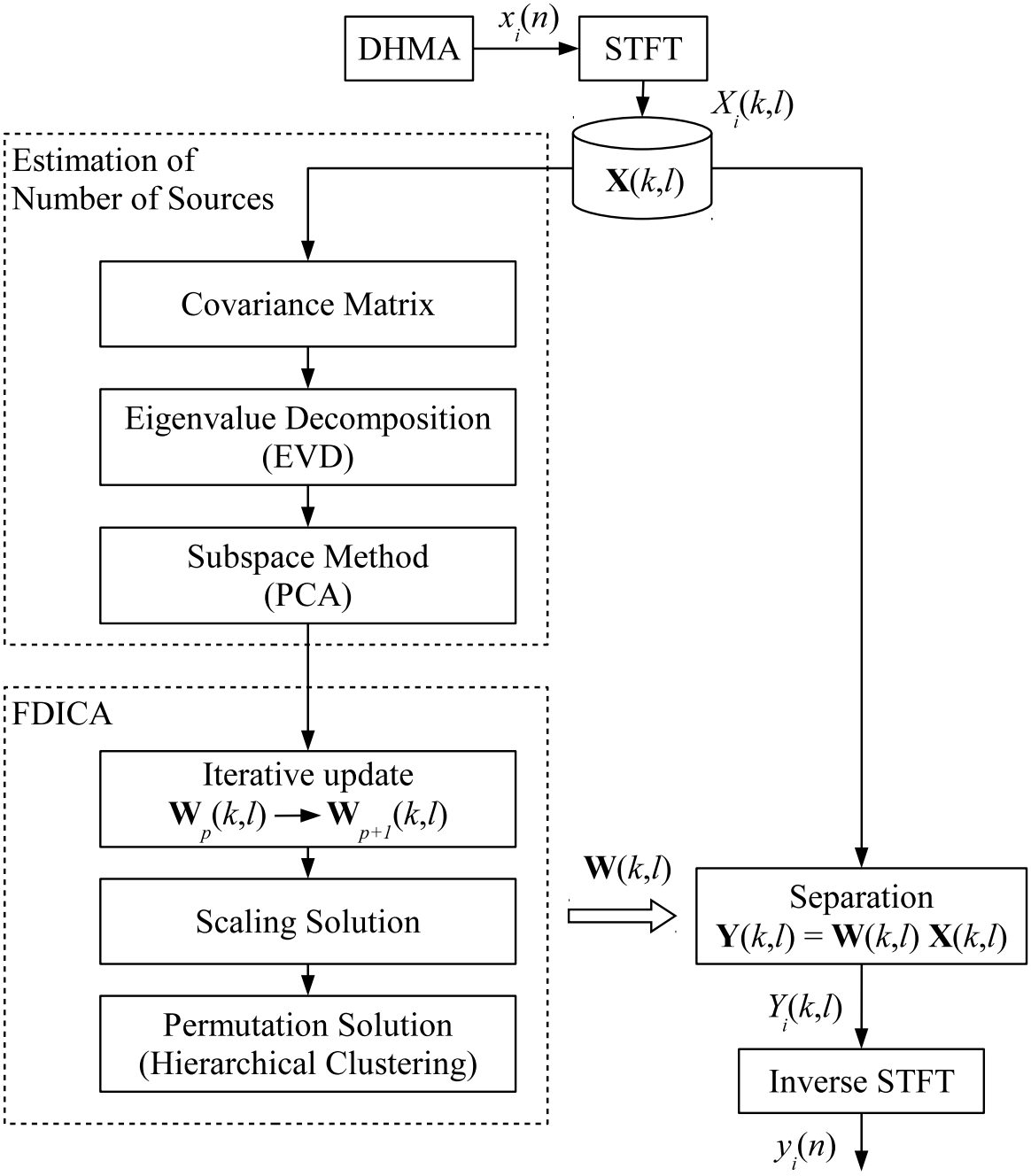

Figure 2.4: Block diagram of previously proposed BSS method using a DHMA [1, 2]. Before estimating the separation matrix, the number of source signals is estimated using eigenvalues, and the subspace matrix is estimated using eigenvectors. Hierarchical clustering corrects the permutation problem by making clusters corresponding to direct and reflected sources.

Blind source separation using a DHMA

FDICA has been proposed for BSS using DHMAs [1, 2]. A block diagram of the process is shown in Figure 2.4. DHMAs are used when there are reverberant acoustic conditions. Room impulse response (RIR) in an enclosed reverberant space is composed of three basic components: the direct sound signal, the early reflections of the signals, and late reverberations of the signals. Direct sound consists of the direction of arrival and the distance between a sound source and a microphone. Reflected sounds can be identified using the image source method, which is mainly influenced by the properties of the walls and the size of the room. In addition, reflected sounds can arrive from any direction. Regarding microphone array techniques, a larger number of microphones increases flexibility, allowing their use under a wider variety of acoustic conditions. Beamforming, also known as spatial filtering, is one microphone array technique. ICA can be categorized an adaptive beamforming technique [39]. The concept of the microphone array technique is similar to the sampling technique for the temporal signal. Microphone distances correspond to sampling intervals, so that shorter distance allows higher spatial frequency. Microphone positions correspond to sampling points, so that larger number of the microphones allows more precise filter shape. When a microphone array uses a large number of microphones, its spatial resolution is high. This means that the beam can become very narrow. In other words, beamforming can capture direct and reflected sounds separately. A common microphone shape for capturing multi-directional sound is a sphere, allowing it to capture sound from any direction. The shape of a DHMA is similar to a sphere, allowing it to also capture sound from any direction. The separation matrix of FDICA corresponds to the inverse mixing matrix, and this separation matrix also includes spatial information. Therefore, BSS using a DHMA can capture sound directionality via the beamforming technique, allowing it to describe the acoustic conditions.

DHMA has two remarkable merits as a microphone array technique: differences in amplitude on its different faces, and spatial aliasing. Generally, larger distances between microphones leads to larger differences in sound pressure. Sound pressure differences for DHMAs are greater than for spherical microphone arrays at frequencies above 6 kHz, as was experimentally confirmed by Ogasawara [1, 2]. The sound pressure on the side of a

DHMA facing a sound source is high, while the pressure on the opposite face is low, and the amplitude difference corresponds to the difference in sound pressure. Therefore, the first merit of DHMAs is that they enlarge amplitude differences. Second, spatial aliasing is unlikely when using a DHMA. The distance between microphones on the same face of a DHMA is very small, which allows for high spatial frequency.

The characteristics of DHMAs and their use in BSS have been fully described in [1,2]. Therefore, BSS using DHMAs is only briefly introduced here and in Chapter 4. BSS methods using DHMAs are assumed to be operating in an overdetermined condition, which means that the number of microphones exceeds the number of source signals. In the frequency domain, the number of sources is estimated using the eigenvalues of the spatial covariance matrix. A threshold is calculated for the eigenvalues in order to project the observed signals onto the subspace signals. The signals in the signal subspace are used for FDICA to obtain the separated signals. The scaling problem is solved by the projection method after obtaining the separation matrix. The permutation problem is solved using hierarchical clustering. The enlarged differences in amplitude of a DHMA are used by the permutation solution, along with direction of arrival (DOA), as the similarity measures to be evaluated by the clustering method. In addition, this method does not require any prior information, for example, the number of sound sources or source locations. Hierarchical clustering involves high computational costs due to the large number of similarity comparisons. When using a DHMA, similarities are calculated for all the transfer functions, the number of which depends on the number of microphones and the number of frequency bins.

2.2 Discussion of computational costs for FDICA

2.2.1 Processing functions for FDICA

In Section 2.1.1, the FDICA algorithm was briefly introduced. FDICA mainly consists of the following functions:

- Time-frequency transformation,

- Iterative update algorithm to obtain the separation matrix,
- Separation in the frequency domain,
- Scaling solution,
- Permutation solution.

In this section, the computational costs of FDICA are estimated to assess its practical feasibility. In the case of mobile devices, two microphones are assumed because this is the smallest possible microphone array. Iterative updating and clustering involve higher computational costs in this situation, than do the other functions. In the case of portable equipment, a DHMA is assumed, and application of the subspace method is an additional function whose computational cost must be evaluated in addition to the usual FDICA functions.

The expressions used when evaluating computational cost are defined as follows:

- N_M: the number of the microphones,
- N_F: size of the fast Fourier transform (FFT),
- N_L: the number of frames,
- N_B: the number of frequency bins,
- N_I: the number of iterations of ICA update rule.

Number of frames N_L represents the length of time the observed signals are stored. N_L depends on the shift size of the short-time Fourier transform (STFT). N_B depends on the size of the FFT N_F. If all the frequency bins are used, $N_B = N_F/2 + 1$.

In the following sections, the computational cost of each function of FDICA are discussed. After discussing each function, the overall computational cost of FDICA is discussed.

2.2.2 STFT and the separation process

The complexity of the FFT is known as $O(n \log_2 n)$, and n represents FFT size. FDICA uses forward and inverse STFTs for each microphone signal, so that the number of operations depends on the values of the variables of:

$$2 \times N_M \times N_L \times N_F \times \log_2 N_F. \tag{2.5}$$

Signal separation requires matrix multiplication, the complexity of which is $O(n^3)$ where n represents the order of a square matrix. Matrix multiplication occurs in each frame and each frequency bin. The separation process depends mainly on matrix multiplication in a frame, so that computational cost depends on the values of the variables of:

$$N_M^3 \times N_B. \tag{2.6}$$

In the frequency domain, all values are complex numbers. Since two real numbers are needed to store a complex number, each multiplication requires the multiplication of four of real numbers.

2.2.3 Iterative update and faster convergence methods

The number of operations required to execute the iterative update rule of FDICA depends on N_B and the values of the other variables of:

$$N_M^3 \times N_B \times N_L \times N_I, \tag{2.7}$$

Iterative update in ICA includes matrix multiplication, the number of multiplication operations of which is N_M^3. Note that each matrix element is a complex number because updating is performed in the frequency domain. As mentioned above, multiplication of complex numbers corresponds to four multiplication operations with real numbers. N_M depends on the system requirements of the speech processing equipment being used, so that N_M cannot be easily changed. N_L affects the estimation accuracy of the separation matrix. Higher values of N_L result in more accurate separation matrices. This means that lower

values for N_L are not an optimum choice for good separation performance. Therefore, reducing N_M and N_L are not good options. In contrast, N_B and N_I can easily be varied to reduce computational cost.

Research proposing the reduction of N_I has already been published, and the number of operations has been successfully restrained by Osako [40], Ema [41], Tachibana [42] and so on. However, reduction of N_B is a task which still remains, as it has not been well researched. Reducing N_B works with reducing N_I simultaneously, so that reducing N_B is also beneficial for the algorithms to reduce N_I. In addition, reducing N_B not only reduces the number of required operations but also reduces the amount of memory consumption required.

For faster convergence of FDICA, Osako proposed that a limited number of frequency bins be used at the beginning of the iterative update stage [40], and that the number of bins be gradually increased in relation to the number of iterations. This method aims at reducing the total number of operations. The frequency bins are selected uniformly, according to which separation matrix is being estimated, and the iterative update is divided into several stages. The number of frequency bins increases uniformly, stage-by-stage, and this increase continues until all the frequency bins are selected. The resulting separation matrix corresponds to the coefficients of an adaptive beamformer [39]. In the selected frequency bins, the directions of arrival (DOA) of the source signals are estimated using the estimated separation matrix. The beamformer coefficients are calculated using the estimated DOAs in the unselected frequency bins, and then used as the initial separation matrix for iterative updates in the next stage. This stage-by-stage processing method involves additional operations, consisting of DOA estimation and calculation of the initial separation matrix, however the total number of operations is restrained.

To achieve faster FDICA convergence, Ema proposed that the number of signals be estimated from the observed signals, and that two different iterative update rules be used [41]. In each frequency bin, DOA was estimated using the normalized phase differences of the observed signals. The number of source signals was then evaluated using a histogram of the estimated DOAs. If the number of source signals is sufficient in one frequency bin, the separation matrix is estimated using the first iterative update rule. For the frequency bins in

which the separation matrix is not estimated, a second iterative update rule is applied to estimate the separation matrix. In those bins, the number of the source signals is insufficient, so that the separation matrix may not converge on an optimal solution by the first rule. The second rule includes the separation matrix from the neighboring frequency bin, which was estimated using the first rule. The number of iterations needed when using the second rule is smaller than when using the first rule, because the included separation matrix works as the suboptimal solution. Therefore, the total number of iterations is restrained, and this means that the total number of operations is restrained.

Tachibana [42] proposed the combination of a closed-form ICA algorithm and a conventional ICA algorithm to achieve faster convergence. The closed-form ICA algorithm is used to obtain a "better" initialization of the separation matrix, while the conventional ICA algorithm consists of higher-order statistics. This means that local optimal solutions exist, in other words, that separation performance depends upon the initialization method. ICA can be solved to be a closed-form expression by using second-order statistics (SOS). Closed-form ICA is a very effective method of obtaining a separation matrix [43], however closed-form ICA is inferior to higher-order ICA with respect to separation performance. The separation matrix created using closed-form ICA can be expected to correspond more closely to the global optimal solution. Therefore, closed-form ICA can be used to obtain the initial matrix for higher-order ICA. This combination achieves faster convergence of the separation matrix, and results in restraining the total number of the iterations, and thus the total number of the operations.

The amount of memory consumption required can be estimated as:

$$N_M \times N_B \times N_L. \tag{2.8}$$

N_M and N_L cannot be easily reduced, for reasons which will be explained in this section, however N_B can be reduced. Note that two real numbers are needed to store a complex number, and that an observed signal is a complex number in the frequency domain. This fact must be kept in mind when estimating the amount of memory space required. N_B tends to be the largest number in Eq. (2.8). For example, the parameters of FDICA might be:

- the length of the observed signals is three seconds for ICA,

- the sampling frequency is 8 kHz,
- the window and FFT size are 1,024,
- the shift size is 256.

In this case, N_L is 94 and N_B is 513. N_M is the number of the microphones, which in the case of mobile devices could be two, while in the case of portable equipment using a DHMA could be up to 160. The size of the FFT should depend on length of the room impulse response (RIR), and RIR length is generally long under reverberant conditions. For example, let's consider a typical meeting situation with a few people in a reverberant room. Assuming the room is less than ten meters long, reverberation time could be up to 500 milliseconds. When FFT size is 1,024, RIR length can be up to 128 milliseconds, corresponding to the length of the FIR filter. In other words, the shortest RIR length is assumed. Sound absorption on the walls change the reverberation time. Usually, the anechoic condition is not preferable to talk because the acoustic feedback helps to control volume of the voice. On the other hand, rich reverberant conditions are difficult to catch the voice from other people in the meeting rooms. 128 milliseconds are almost quater of 500 milliseconds as the reverberation time, and this condition can be considered as the shortest reverberation time to process FDICA. Therefore, N_B is the main factor determining required memory consumption.

2.2.4 Permutation solution and clustering methods

In Section 1.2.2, the issue of the permutation problem was introduced, which refers to ambiguity in the order of the separated signals. For example, assume that FDICA is being applied to two sources, named A and B, and that the correct order is "A, B" in frequency bin k_1. If the permutation problem does not occur, the order of the separated signals will not change. However, the order of the separated signals may switch to "B, A" in frequency bin k_2 due to the permutation problem. If the permutation problem is not solved, separation performance will deteriorate.

Various methods have been proposed to solve the permutation problem, and they can be

broadly classified into two groups: correlation methods and clustering methods. Correlation methods evaluate the correlation between two separated signals in different frequency bins [25, 26, 44, 45]. The two bins must be close to each other because there must be high correlation between the target signals in each bin. Similar temporal changes in two signals implies that the signals were generated by the same source. But if two different signals are mixed, they will exhibit irregular changes over time. Clustering methods place separation coefficients into groups, which are classified by their DOA, or their directivity when using the beamforming technique [39, 46]. The mixing matrix can be obtained using the inverse or pseudo-inverse of the separation matrix. The mixing matrix represents the group of transfer functions from the source positions to the microphone positions. In other words, the coefficients of the separation matrix correspond to the coefficients of the beamformers. A generalized Lloyd's algorithm [47] has been used to make clusters of transfer functions [39]. Note that Lloyd's algorithm is equivalent to the k-means method [48]. Grouping the transfer functions effectively solves the permutation problem by using the k-means method [49]. To achieve a flexible and accurate clustering method, agglomerative hierarchical clustering [50] has also been applied to solve the permutation problem [51]. Hierarchical clustering does not require designating the number of clusters, which is beneficial in situations in which the number of the source signals is unknown. Note that BSS using a DHMA, as described in Section 2.1.2, uses hierarchical clustering as its permutation solution. Various methods have been proposed using the same concept; the coherency of the separation matrices in adjacent frequency bins was used to evaluate source locations by Asano [27]. Source locations are equivalent to DOA. Pham evaluated the continuity of the frequency response of the mixing matrices [52–54]. In adjacent frequency bins, the product of the separation and inverse separation matrices should be nearly diagonal when the permutation problem does not occur. Nesta has proposed using evaluation of the continuity of the phase response as a permutation solution [55–57]. In the time domain, the time difference for a signal is only described by the unitary value, which contains only the time value and does not contain any other values. In the frequency domain, the phase response is expressed by the time difference, but the phase includes the frequency value and varies according to the frequency. The relationship between time difference and phase is linear, so that the phase response has continuity. Combination of the correlation and clustering

methods has also been proposed [58]. Speech signals generally include harmonic components, so Sawada proposed a combination of harmonic correlation and directivity, based on the pseudo-inverse of the separation matrix [59]. The clustering method compares clusters, calculating the similarity or distance between two samples. Similarity must be calculated for all combinations of the two samples, therefore computational cost becomes a serious issue when a clustering method is used.

Another approach, permutation-free ICA, has also been studied [60,61], "free" meaning that a permutation solution is not needed. Permutation-free ICA collects the separation matrices in all the frequency bins, and the collected matrices become elements of one huge matrix. This huge matrix is iteratively updated using an ICA algorithm. The permutation problem of FDICA is caused by the assumption of independence between the different frequency bins, but a single huge matrix avoids this independence assumption. However, the size of the matrix becomes an issue from the viewpoint of computational cost, because ICA algorithms involve matrix multiplication, resulting in an extremely large number of operations due to the huge size of the matrix. In addition, required memory consumption is much larger than for methods in which the separation matrices are not collected.

Another ICA algorithm which does not require a permutation solution, known as TRINICON (Triple-N ICA for convolutive mixtures), uses joint diagonalization of second-order statistics (SOS) [28,62–64]. Discrete Fourier transformation (DFT) is included in the iterative update process of TRINICON, in order to avoid the need for a permutation solution. Inverse and forward DFTs are calculated for the separation matrices every several iterations. Between two transformations, namely in the time domain, aliasing is attenuated for the impulse responses. The inverse of the separation matrix is the same as the group of transfer functions mentioned in this section, and in the time domain these transfer functions are expressed as impulse responses as well. The Fourier transform combines the coefficients in the different frequency bins and in the different time indexes. These processes prevent the complete decoupling of the frequency bins usually caused by the bin-wise independence assumption. This is why a permutation solution is not required when using inverse and forward DFTs during the iterative update process. Note that the same signal processing technique is used in the field of adaptive filter research. Although TRINICON does not

require a permutation solution, using DFTs as TRINICON does is computationally disadvantageous. Skipping several iterations in order to reduce computational cost is a common method, but unsolved permutations may still remain. Ideally, the inverse and forward DFTs must be applied at every iteration. Other methods have been proposed which use DFTs at every iteration [65–68]. If bin-wise independence was assumed when using TRINICON, in other words if DFTs were not used during iterative updates, then a permutation solution for TRINICON would be necessary [63].

The BSS method proposed in this dissertation selects frequency bins for processing in order to reduce computational costs. Selection of specific frequency bins can be done only under the assumption of bin-wise independence. In addition, the correlation method cannot be used because our selection method does not allow signal correlation between different frequency bins. Therefore, using the proposed method, FDICA needs a permutation solution, and only the clustering method can be used.

The complexity of the clustering method depends on the number of elements, clusters, and similarity calculations. The optimal complexity of agglomerative hierarchical clustering is known as $O(n^2)$ [50] where n represents the number of data points. In the case of FDICA, the number of data points depends on the total number of transfer functions in all of the frequency bins. The pseudo-inverse of the separation matrix corresponds to the mixing matrix, and the column vectors represent the transfer functions in the mixing matrix. The inner product between two column vectors can be calculated to evaluate similarity for the clustering method. This means that N_M multiplications are required for a similarity calculation. The number of operations for hierarchical clustering can be calculated as follows:

$$N_M^4 \times N_B^2. \tag{2.9}$$

Note that N_M usually depends on the specification of the speech processing equipment, so reducing N_M is not a good option. Therefore, only the number of frequency bins (N_B) can be reduced in Eq. (2.9). If N_M is large, then the number of operations in the permutation solution also becomes large. Since reducing N_M is not a good option, reducing N_B in the clustering method is another quite feasible method of reducing computational cost.

The k-means method has been discussed in previous paragraphs, and it can also be used as a permutation solution. The disadvantage of using k-means is that the number of clusters must be fixed in advance. However, this method can be applied with mobile devices. Because mobile devices must be small, the number of microphones tends to be small, therefore system requirements can be used to set a fixed number of source signals for mobile devices. The complexity of k-means is $O(n_{ite}n)$ where n_{ite} is the number of the iterations for k-means. The number of the operations can be calculated as:

$$n_{ite} \times N_M^2 \times N_B. \tag{2.10}$$

This is less than the number of operations required for hierarchical clustering, which means that k-means is more appropriate for mobile devices, because computational costs need to be lower.

2.2.5 Scaling solution

The scaling problem refers to ambiguity in the amplitudes of the separated signals. Amplitudes may vary for each source and among different frequency bins, and amplitude scaling ambiguity distorts the results of the ICA algorithm. The scaling problem can be solved using the projection method [37] which employs the pseudo-inverse of the separation matrix. Unexpected amplitude scales are included in the separation matrix just after the ICA algorithm has converged. The projection method is calculated using the separation matrix $\mathbf{W}(k)$ and its pseudo-inverse $\mathbf{W}^+(k)$ as follows:

$$\text{diag}\left\{\mathbf{W}^+(k)\right\}\mathbf{W}(k). \tag{2.11}$$

The pseudo-inverse includes the reciprocal values of the unexpected scales, which is why the projection method can solve the scaling problem. This pseudo-inverse matrix is responsible for most of the computational cost of the projection method.

The computational cost of obtaining an inverse matrix is known to be large, in general. The complexity of the pseudo-inverse can be evaluated using eigenvalue decomposition, because the pseudo-inverse can be calculated with singular value decomposition.

Eigenvalue decomposition can be calculated using the Householder method and the implicit shifted QL method [69]. Singular value decomposition can be calculated using the same method used for eigenvalue decomposition. The Householder method requires $(4/3)n^3$ operations, and the implicit shifted QL method requires $3n^3$ operations. These methods are matrix operations, and n represents the number of elements in a row or column of a square matrix. ICA assumes that the number of source signals equals the number of microphones: n equals N_M. FDICA obtains the $N_M \times N_M$ separation matrix for all of the different frequency bins. Therefore, the number of the operations required for the projection method can be estimated as:

$$\left\{(4/3)\, N_M^3 + 3N_M^3\right\} \times N_B = (13/3)N_M^3 N_B. \tag{2.12}$$

2.2.6 Total computational cost of conventional FDICA

The number of operations required for each component of FDICA have been discussed in previous sections. They are listed as follows:

- STFT: $2 \times N_M \times N_L \times N_F \times \log_2 N_F$,
- Separation: $N_M^3 \times N_F$,
- Iterative update: $N_M^3 \times N_B \times N_L \times N_I$,
- Scaling solution: $(13/3)N_M^3 \times N_B$,
- Permutation solution (k-means): $10 \times N_M^2 \times N_B$

  OR

- Permutation solution (hierarchical clustering): $N_M^4 \times N_B^2$,

where the number of k-means iterations (n_{ite}) is assumed to be ten for the estimation of computational cost. Only STFT does not include a term which is a power of N_M, thus the other functions have higher computational costs than STFT. For conventional FDICA, the number of frequency bins (N_B) equals $N_F/2+1$. Because ICA algorithms optimize higher-order

statistics which can only be accurately calculated using a large amount of data, a meaningful number of frames (N_L) is over a few seconds. N_L and N_I are positive numbers, and they are usually larger than at least ten. For the separation function and the scaling solution, the number of operations can be approximated as $2N_M^3N_B$ and $(13/3)N_M^3N_B$, respectively, thus the iterative update function has a higher computational cost. If the permutation solution used is k-means, iterative update is the dominant function with respect to computational cost, because of $N_LN_I > 10$ and $N_M^3 > N_M^2$. If the permutation solution used is hierarchical clustering, iterative update and permutation solution functions are dominant with respect to the computational cost. The parameters N_B, N_L and N_I determine which function is dominant.

2.3 Properties of current embedded processors and target computational cost

2.3.1 Review of current embedded processors

For speech signal processing, digital signal processors are more appropriate than microprocessors with embedded speech processing functions, for reasons, which are its architecture and real-time speech signal processing, explained in Section 1.3. The operating frequency of the latest DSPs is over 1-GHz, and they have multiple arithmetic units. For example, the Texas Instruments TMS320C6678 has eight cores, and can achieve a peak performance of 160 GFLOPS at an operating speed of 1.25-GHz. Each core can calculate eight multiplications per cycle using a pipeline operation. The TMS320C6678 is very expensive, however, and consumes large amounts of electrical power, about 20 watts at maximum capacity. This means that high-performance DSPs such as the TMS320C6678 are not practical for mobile devices, but can be an appropriate choice for portable equipment. Portable equipment can be taken anywhere due to its small size, and the electric power supply can be an AC power outlet, because this sort of speech communication equipment is usually used in meeting rooms. If many DSPs were used in the same piece of equipment, its size would increase, jeopardizing its portability. If only one high-performance DSP were used in a portable speech device, then it would be small enough to carry anywhere. Therefore,

the target computational cost of portable equipment can be considered to be 160 GFLOPS, based on using a single, high-performance DSP.

On the other hand, low-power DSPs such as the Texas Instruments TMS320C55x are also available. Its electric power consumption is several tens of milliwatts, making it quite appropriate for use in mobile devices. In recent years, battery capacities have increased, with a current capacity of around 2,000 mAh for consumer products. High-performance DSPs operate at 1.5 volts for a core unit, so even with a large capacity battery, they could only operate for about 9 minutes on a mobile device, which is insufficient for teleconferencing. However, in the near future operating time will probably increase, because future high-performance DSPs are expected to consume less electric power. In addition, battery capacity will also probably increase. On the other hand, mobile devices are used to run many rich functions, such as graphical user interfaces, GPS, etc. The power consumption of these functions is not a negligible issue, but these functions contribute to the usefulness and desirability of these devices, so removing them is not an option. Therefore, high-performance DSPs probably cannot be used in mobile devices, due to power consumption issues. For these reasons, use of low-power DSPs in mobile devices is a better option, however signal processing performance is much lower than with high-performance DSPs. The operating speed of low-power DSPs is a few hundred MHz, and the number of multiplication units is only one per cycle, in contrast to the multiple units of high-performance DSPs. Therefore, computational performance of low-power DSPs corresponds to their operating frequency: a few hundred MFLOPS. This can give the target computational cost for mobile devices: 200 MFLOPS.

Most DSPs have fast internal memory, which can be accessed without wait states, however this memory is very small due to its expense. This internal memory can be used to help estimate target required memory consumption. The internal memory size per core of the TMS320C6678 is 512 kBytes. FDICA must store the observed signal to estimate the separation matrix. For example, when the number of microphones (N_M) equals two, over 800 kBytes of memory is required. An example of the calculation of the required memory is as follows:

- observed signal length: 3 seconds,

Table 2.1: Target computational costs estimated by the target equipment and appropriate DSPs

	mobile	portable
Number of operations	160 GFLOPS	200 MFLOPS
Memory consumption	depend on balance of system requirements and algorithm performance	

- number of the microphones N_M: 2,
- 16-bit for one sample,
- 8 kHz sampling frequency.

Thus the required amount of memory exceeds the size of the internal memory, which means that the size of the internal memory is insufficient for storing the observed signals. In addition, internal memory is designed to be used as working memory, not as storage. But the internal memory is sufficient for algorithms requiring high level computation, such as FFT or matrix calculation. Let's look at another example, in which there are a large number of microphones. Required memory consumption grows to over 12 MBytes when performing BSS with a DHMA with sixty microphones. Therefore, an external memory must be used to store the observed signals. However the required wait states of external memories are a serious issue, as mentioned in Section 1.3.

Low memory consumption helps to restrain the total number of required wait states during external memory access. This means that reducing memory consumption improves the feasibility of performing FDICA with embedded processors. If the sampling frequency is higher than 8 kHz, required memory consumption must be increased to store the observed signals. Wait states are a very practical signal processing issue, however this problem tends to be overlooked. While lower memory consumption is preferable, target required memory consumption depends a great deal on the balance between system requirements and algorithm separation performance, so that it is difficult to designate a target value. As a consequence, target computational costs, discussed in this section, are listed in Table 2.1.

Table 2.2: Estimated computational costs of conventional FDICA with two microphones

Method	Complexity
STFT (forward & inverse)	$4N_L N_F \log_2 N_F$
Iterative update of separation matrix	$4N_I N_L N_F$
Scaling solution using projection method	$17N_F$
Permutation solution using k-means clustering	$20N_F$
Total	$[\{4\log_2 N_F + 4N_I\} N_L + 37] N_F$

2.3.2 Examples of computational costs for BSS using conventional FDICA

Mobile devices with two microphones

In this section, we assume the use of a small mobile device with two microphones, which is the smallest possible microphone array. The number of source signals should also be small, so a fixed number of source signals and microphones is assumed. These assumptions are reasonable because the device is small, and the number of users is almost always two. Because mobile devices are usually operated with a small battery, k-means is an appropriate permutation solution, because its computational cost is lower than that of hierarchical clustering, and because the number of the clusters is fixed. Table 2.2 shows an example of the number of operations required. To simplify the numbers in the comparison, N_B and $(13/3)N_M^3/2$ are approximated as $N_F/2$ and 17, respectively, in the estimate. The dominant computational cost can be either STFT or the iterative update. If the number of iterations is larger than the logarithm of FFT size, that is $N_I > \log_2 N_F$, then iterative update becomes the dominant computational cost.

If the values used are as follows:

- FFT size: 1024,
- number of frames: 100,
- number of iterations: 200,

Table 2.3: Examples of computational costs for the functions of conventional FDICA with two microphones, for the different sizes of FFT under the 8–kHz sampling frequency

Function	FFT size			
	512	1024	2048	4096
STFT	13.86	15.40	16.94	18.09
Iterative update	308.02	308.02	308.02	301.47
Scaling solution	0.03	0.07	0.14	0.28
Permutation solution (k-means)	0.04	0.08	0.16	0.33

Note: The unit of computational costs is mega-operations.

then the total number of operations is about 80 mega-operations. Note that about 100 frames corresponds to about three seconds at a sampling rate of 8–kHz, with 256 samples as the shift length of STFT. When the sampling frequency is 8–kHz, $\log_2 N_F$ equals 10, and the iterative update is the dominant cost. If the sampling frequency is 48–kHz, the appropriate FFT size is over 8192: $\log_2 N_F = 13$. Additionally, four multiplication operations are needed per complex number, which means that in this example the total number of operations becomes about 320 mega-operations.

For the different sizes of FFT, the examples of estimated computational costs for the FDICA functions is shown in Table 2.3 and Table 2.4. These examples include the influence of the multiplication of two complex numbers. Table 2.3 is under that the sampling frequency is 8–kHz, and Table 2.4 is under that the sampling frequency is 48–kHz. The sizes of FFT are chosen by considering similar time lengths for two different sampling frequencies.

Figure 2.5 and Figure 2.6 show the ratio of the computational costs between FDICA functions. It is clearly recognized tha the iterative update is always the dominant computational cost of FDICA, for the case of two microphones which assumes the mobile devices. The iterative update includes the numbers, N_I, N_L and N_F in Table 2.2. As discussed in Section 2.2.3, reducing N_I is not preferable from the viewpoint of the estimation accuracy,

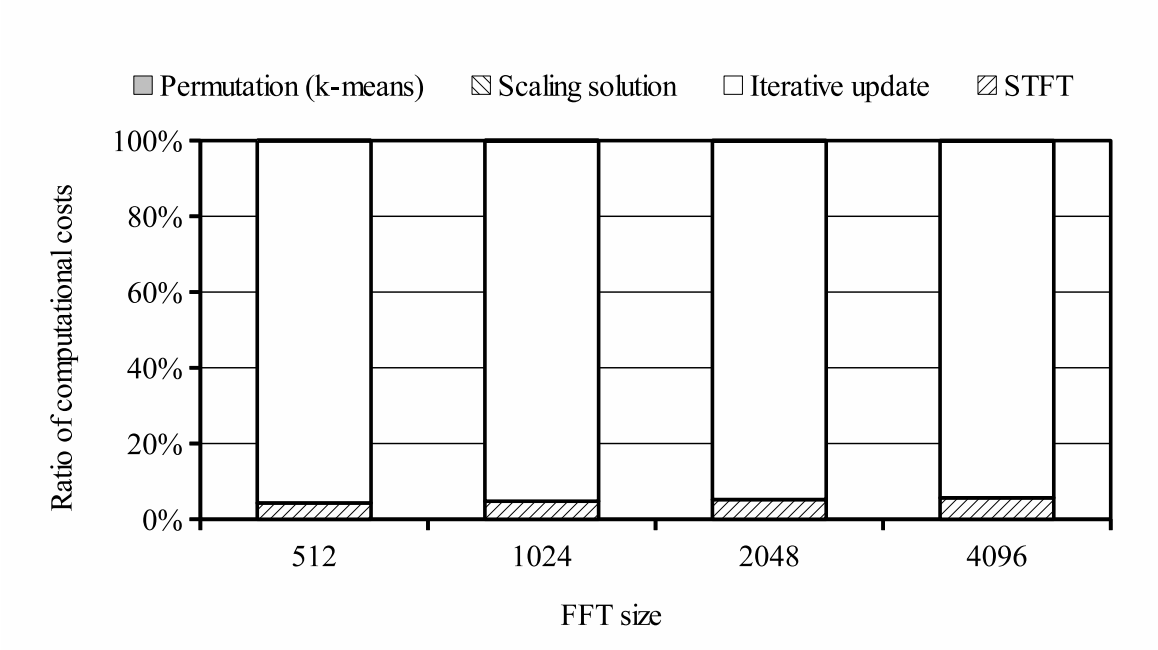

Figure 2.5: The ratios of estimated computational costs for the conventional FDICA with two microphones under the different FFT sizes and that the sampling frequency is 8–kHz. Even though four functions are shown, it is very easy to recognize that the iterative update function is dominant.

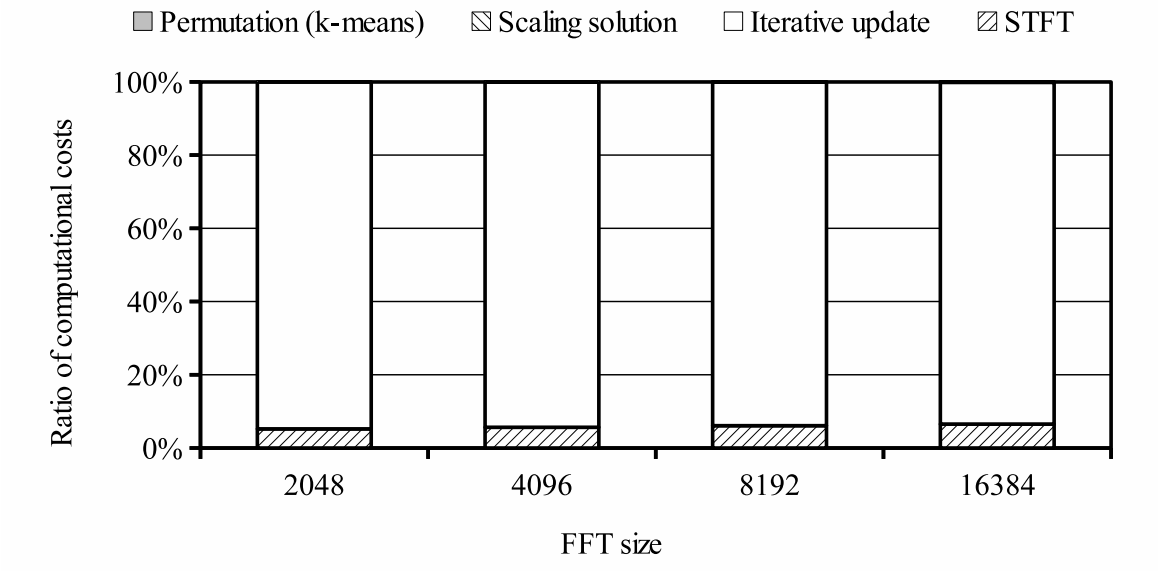

Figure 2.6: The ratios of estimated computational costs for the conventional FDICA with two microphones under the different FFT sizes and that the sampling frequency is 48–kHz. Even though four functions are shown, it is very easy to recognize that the iterative update function is dominant.

Table 2.4: Examples of computational costs for the functions of conventional FDICA with two microphones, for the different sizes of FFT under the 48–kHz sampling frequency

Function	FFT size			
	2048	4096	8192	16384
STFT	101.29	110.89	119.28	128.45
Iterative update	1,841.56	1,848.12	1,835.01	1,835.01
Scaling solution	0.14	0.28	0.56	1.11
Permutation solution (k-means)	0.16	0.33	0.66	1.31

Note: The unit of computational costs is mega-operations.

and the researches for reducing N_I have been reported and work properly. Therefore, the estimation examples give insight again which limiting the number of the frequency bins is a reasonable way for reducing the computational cost, that is discussed in Section 2.2.3 as well.

Portable equipment with DHMAs

When using BSS with a DHMA, an additional step, which utilizes the subspace method, is required to estimate the number of source signals. A block diagram of the process is shown in Figure 2.4 in p. 17. The number of source signals is estimated using a threshold operation on the eigenvalues of the spatial covariance matrix. The subspace matrix consists of the eigenvectors related to the eigenvalues which exceed the threshold. The subspace matrix is applied to the observed signal to extract the subspace signals. Eigenvalue decomposition (EVD) is the dominant computational cost when using the subspace method. EVD can be calculated using the Householder method and the implicit shifted QL (ISQL) method [69]. The number of operations required for the Householder method can be represented as n^3. The number of operations required for the ISQL method can be represented as $(13/3)n^3$. An example estimation of the number of the operations required for BSS using a DHMA is as follows:

Table 2.5: Estimated computational costs of conventional FDICA with DHMAs

Method	Complexity
STFT (forward & inverse)	$120N_L N_F \log_2 N_F$
Covariance matrix	$1800N_L N_F$
Eigenvalue decomposition	$4.64 \times 10^5 N_F$
Subspace method	$600N_L N_F$
Iterative update of separation matrix	$4000N_L N_I N_F$
Scaling solution	$4.64 \times 10^5 N_F$
Permutation solution (hierarchical clustering)	$2.03 \times 10^5 N_F^2$
Total	$\left[\{120 \log_2 N_F + 4000N_I + 2400\} N_L + 2.03 \times 10^5 N_F + 9.28 \times 10^5\right] N_F$

- N_Q: order estimation of the subspace,
- $N_{K(k)}$: the number of separated signals in each frequency bin k,
- $N_{\bar{K}}$: the average number of separated signals.

Reverberant conditions exist when the sound signal consists of both direct and reflected sounds. Reflected sound is also known as the "virtual source" when using the image source method. The number of direct and reflected sounds is greater than the number of direct sounds, i.e., the number of source signals. Therefore, if assuming that reverberant conditions exist, the order of subspace N_Q is less than N_M, and N_Q should be greater than $N_{K(k)}$. The permutation solution depends on the total number of transfer functions, thus, the subspace method influences the permutation solution. $N_{K(k)}$ varies among the frequency bins. Assuming a constant value for $N_{K(k)}$ is very difficult because the number of direct and reflected sounds depends not only on acoustic conditions, but also on the frequency region of the signals. To simplify the computational cost estimation, the average value of $N_{K(k)}$ is used, which is $N_{\bar{K}}$. The k-means method requires that the number of the clusters be

Table 2.6: Examples of computational costs for the functions of the BSS method using DHMAs, for the different sizes of FFT under the 8–kHz sampling frequency

Function	FFT size			
	512	1024	2048	4096
STFT	0.42	0.46	0.51	0.54
Covariance matrix	0.69	0.69	0.69	0.68
Eigenvalue decomposition	0.95	1.90	3.80	7.60
Subspace method	0.23	0.23	0.23	0.23
Iterative update	308.02	308.02	308.02	301.47
Scaling solution	0.95	1.90	3.80	7.60
Permutation solution (hierarchical clustering)	212.86	851.44	3,405.77	13,623.10

Note: The unit of computational costs is giga-operations.

defined in advance, but the hierarchical clustering method does not have this requirement. Therefore, hierarchical clustering is the appropriate permutation solution for BSS using a DHMA.

The number of operations for each process are calculated as follows:

- STFT (forward & inverse) : $2N_M N_L N_F \log_2 N_F$
- Covariance matrix : $N_M^2 N_L N_B$,
- Eigenvalue decomposition : $(13/3)N_M^3$
- Subspace method : $N_Q N_M N_L N_B$,
- Iterative update of separation matrix : $N_Q^3 N_L N_I N_B$
- Scaling solution (hierarchical clustering) : $\{N_M N_{\tilde{K}} N_B\}^2$

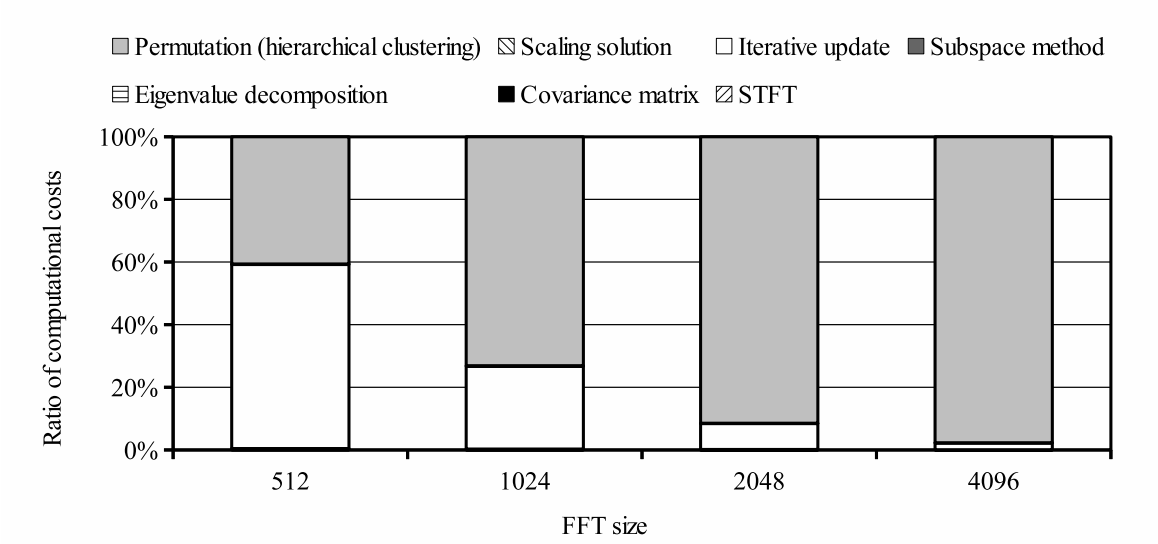

Figure 2.7: The ratios of estimated computational costs for the BSS method using DHMA under the different FFT sizes and that the sampling frequency is 8–kHz. Even though seven functions are shown, the permutation solution is the dominant cost for larger FFT sizes. For smaller FFT size, however, the iterative update is dominant.

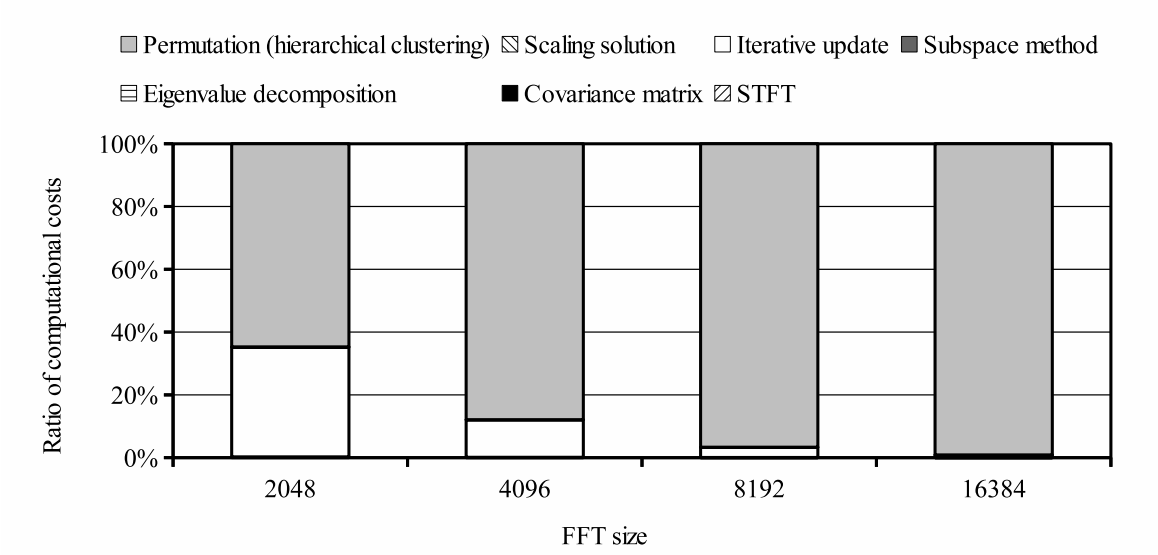

Figure 2.8: The ratios of estimated computational costs for the BSS method using DHMA under the different FFT sizes and that the sampling frequency is 48–kHz. Even though seven functions are shown, the permutation solution is the dominant cost for larger FFT sizes. For smaller FFT size, the permutation solution is still dominant, however the iterative update becomes not negligible.

Table 2.7: Examples of computational costs for the functions of the BSS method using DHMAs, for the different sizes of FFT under the 48–kHz sampling frequency

Function	FFT size			
	2048	4096	8192	16384
STFT	3.04	3.33	3.58	3.85
Covariance matrix	4.14	4.16	4.13	4.13
Eigenvalue decomposition	3.80	7.60	15.20	30.41
Subspace method	1.38	1.38	1.38	1.38
Iterative update	1,841.56	1,848.12	1,835.01	1,835.01
Scaling solution	3.80	7.60	15.20	30.41
Permutation solution (hierarchical clustering)	3,405.77	13,623.10	54,492.40	217,969.59

Note: The unit of computational costs is giga-operations.

N_M is sixty which means six microphones on each face of the DHMA as described in Section 2.1.2. If N_Q and $N_{\tilde{K}}$ are assumed to be one-third and one-fourth of N_M, respectively, then the number of operations shown in Table 2.5 can be revised. The size of the FFT (N_F) should be chosen based on reverberation time, which is usually a few hundred milliseconds in meeting rooms, for example. If the sampling frequency is 8–kHz, the size of the FFT should be larger than 1,024 samples, which corresponds to 128 milliseconds. If $N_F \geq 1024$ is satisfied, hierarchical clustering becomes the predominant computational cost. If $N_F = 1024$ is satisfied, the number of the operations becomes 212 giga-operations, without taking into account the multiplication of complex numbers required for hierarchical clustering. Considering that four multiplication operations using a real number are required for each multiplication by a complex number, the number of required operations becomes about 850 giga-operations for hierarchical clustering.

For the different sizes of FFT, the examples of estimated computational costs for the functions of the BSS method using DHMAs is shown in Table 2.6 and Table 2.7. These examples include the influence of the multiplication of two complex numbers. Table 2.6 is under that the sampling frequency is 8–kHz, and Table 2.7 is under that the sampling frequency is 48–kHz. The sizes of FFT are chosen by considering similar time lengths for two different sampling frequencies.

Figure 2.7 and Figure 2.8 show the ratio of the computational costs between functions for the BSS method using DHMAs. For the case $N_F \geq 1024$, the computational cost of hierarchical clustering is the dominant cost, and it increases when the size of FFT increases. In addition, for the smaller sizes of FFT, the iterative update becomes not negligible. Hierarchical clustering only depends on N_F in Table 2.5. Limiting the number of the frequency bins contributes to reduce the computational cost for the iterative update. These facts give insight again that limiting the number of the frequency bins is a reasonable way, that is discussed in Section 2.2.4 as well.

2.3.3 Discussion of target computational costs

In this section, the target computational cost of BSS with FDICA is discussed, in regards to computational cost sufficiently to allow the use of embedded processors. Two types of applications have been considered: mobile devices and portable equipment. It has been discussed that DRAM involves the slow data transfer characteristics, which leads to delayed signal processing and power wastage in the DSP. The required memory consumption of BSS with FDICA already exceeds the internal memory size of the high performance TMS320C6678 DSP, for example, which means that external memory must therefore also be used. Thus, the influence of DRAM waiting cycles on DSPs must be considered, which is the equivalent of tripling the number of operations estimated previously in this dissertation. Taking into account the effect of slow data access, target computational costs are re-estimated in Table 2.8. Current DSPs have been discussed in Section 2.3.1, and they were categorized into two types: low-power, inexpensive DSPs and high-performance, expensive DSPs. They are listed in Table 2.9.

For mobile devices, it is assumed that BBS using FDICA will be performed using a

Table 2.8: Computational costs of conventional FDICA for the target speech equipment

	Mobile	Portable
Number of operations	900 mega-operations	2.4 tera-operations
Required memory consumption	800 kBytes	12 MBytes
Power supply	Battery	AC power

Table 2.9: DSP categories

	Low-power	High-performance
Operating frequency	Few hundred MHz	1.25–GHz
Peak performance	equals operating frequency	160 GFLOPS
Size of internal memory	64–256 kBytes	512 kBytes

low-power DSP. In this case, only about one-third of the number of operations conventionally required are feasible, with respect to the estimated computational cost for conventional FDICA shown in Tables 2.8 and 2.9. Since mobile devices also need to run rich functions such as user interfaces, GPS, etc., basic functions such as speech processing should not consume large amounts of computational resources. Even one-third the number operations used conventionally is still excessive, because this would consume almost all of the processing capacity of a low-power DSP. At most, one-fourth or one-fifth the number of operations are feasible. Therefore, the target number of operations becomes about 200 mega-operations.

On the other hand, for portable equipment it is assumed that high-performance DSPs can be used for BSS. The Texas Instruments TMS320C6678 DSP has eight cores, making it a good candidate. Note that its peak signal processing performance is 160 GFLOPS. If multiple high-performance DSPs were used in this kind of equipment, they could perform conventional FDICA with a large number of microphones. However, the TMS320C6678 is quite expensive, with a price of over 160 U.S. dollars per DSP. If fifteen TMS320C6678

DSPs were used, portable equipment using conventional BSS with a DHMA would be feasible in terms of computational resources, but might not be feasible in terms of manufacturing cost, which would be over 2,000 U.S. dollars just for the DSPs. At this cost, the market would be extremely limited for manufacturers. In addition, coordination of multiple DSPs requires complex system management, which is also a drawback for manufacturers. Therefore, in this dissertation it is assumed that using just one high-performance DSP is reasonable when designing portable equipment to perform BSS with FDICA. At most one-tenth or one-fifteenth of the number of possible operations is necessary if we can achieve our target computational cost, which in this dissertation is 160 giga-operations.

As a conclusion, the target computational costs for each assumed speech equipment as follows:

- Mobile devices: 200 mega-operations,
- Portable equipment: 160 giga-operations.

Chapter 3

Blind source separation for the mobile devices with two microphones

Low-power DSPs are expected by the reason which its electric power consumption is several tens of milliwatts, making it quite appropriate for use in mobile devices. Reducing computational costs of BSS using FDICA can widen the range of application fields, because computational costs of conventional FDICA have already exceeds the performance of low-power DSPs, as mentioned in Section 2.3. In the case of the mobile devices, two microphones are assumed because this is the smallest possible microphone array. The determinant of the spatial covariance matrix is theoretically analyzed for two microphones, and the determinant can simultaneously evaluate the number of source signals and the relative strength of the signals among different frequencies. This characteristic contributes to select promising frequency bins for source separation, and the proposed method only estimate the ICA separation matrices in the selected frequency bins. In unselected bins, the Wiener filter consists of the tentative separated signals which are outputs of the null-beamformer. The proposed method shows significant improvement with respect to both low computational costs and low distortion in the separated signals, through the computational costs estimation and the experimental evaluation.

3.1 Motivation and strategy

A semi-blind source separation (semi-BSS) method with low computational costs has been introduced in [70]. This semi-BSS method intuitively uses the determinant of a spatial covariance matrix to select frequency bins for reducing computational costs. In unselected frequency bins, the null-beamformer (NBF) is used to separate the observed signal instead of the ICA separation matrix. One of the purposes in this chapter is clarifying the theoretical reason for the bin selection criteria, which is the determinant of the spatial covariance matrix. This is introduced by an analysis of the determinant using the sound propagation and signal processing of the NBF. Another purpose in this chapter is how to improve the performance deterioration that depends on the drawback involving the NBF. The NBF consists of time delay adjustments and a subtraction between two microphone signals in the time domain. A phase difference between two microphones is very small in the low frequency region because of long wave lengths. This is due to, in the low frequency region, that the observed signals between two microphones are very similar even with that the time delay adjustment is applied. This leads to the amplitude attenuation of the NBF in the low frequency region, and results in the signal distortion of the separated signal in the unselected frequency bins. Especially in mobile devices, the separated signals might be extremely distorted because the distance between microphones would be very small. Even though poor separation performance by the NBF in unselected frequency bins, the Wiener filter can still work to improve the separated sound quality, in which the output of the NBF is employed in this dissertation. The experimental results shows improved performance, higher segmental signal-to-noise ratio than the case which ICA works in all the frequency bins. A block diagram of the proposed method is shown in Figure 3.1.

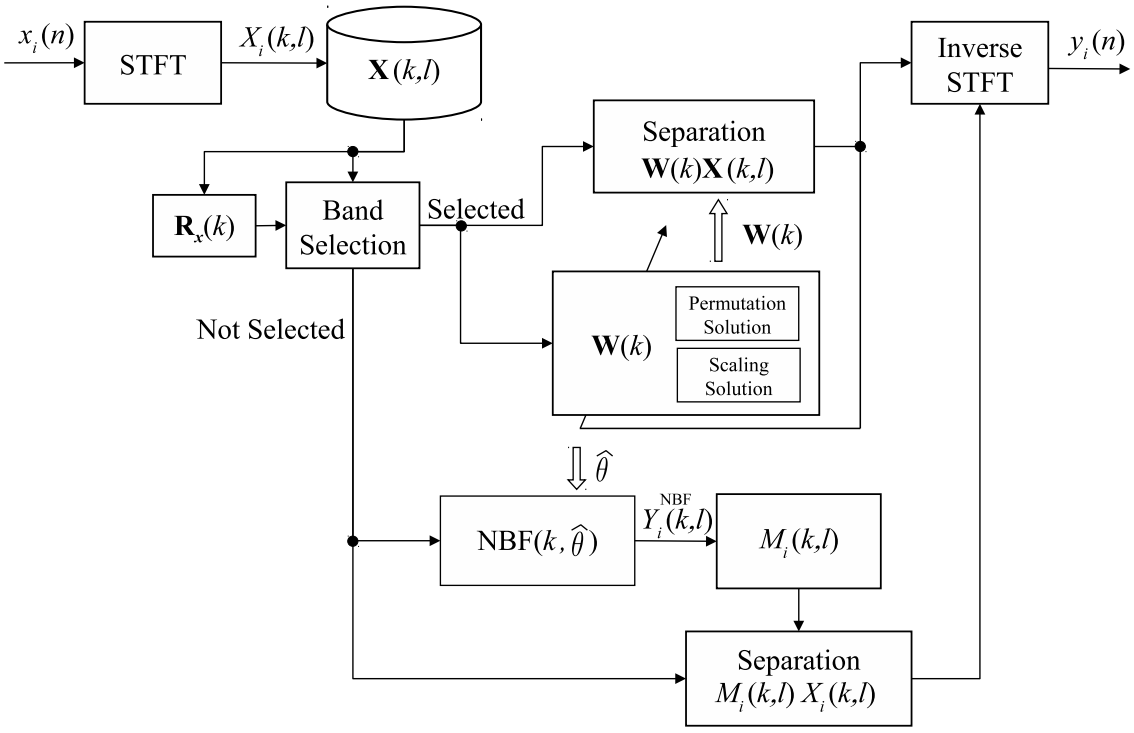

Figure 3.1: Block diagram of proposed method for two microphones. Capital letters show frequency domain signal such as $X_i(k,l)$, small letters show time domain signal such as $x_i(n)$. Separation matrix $\mathbf{W}(k)$ is updated by the iterative update rule, and the separated signals by ICA are obtained. The frame-wise Wiener filter $M_i(k,l)$ is obtained by tentative separated signal by the null-beamformer which consists of estimated source direction. Separated signals, $\mathbf{W}(k)\mathbf{X}(k,l)$ and $M_i(k,l)X_i(k,l)$, are gathered for all the frequency bins, and transformed into the time domain signals by inverse STFT.

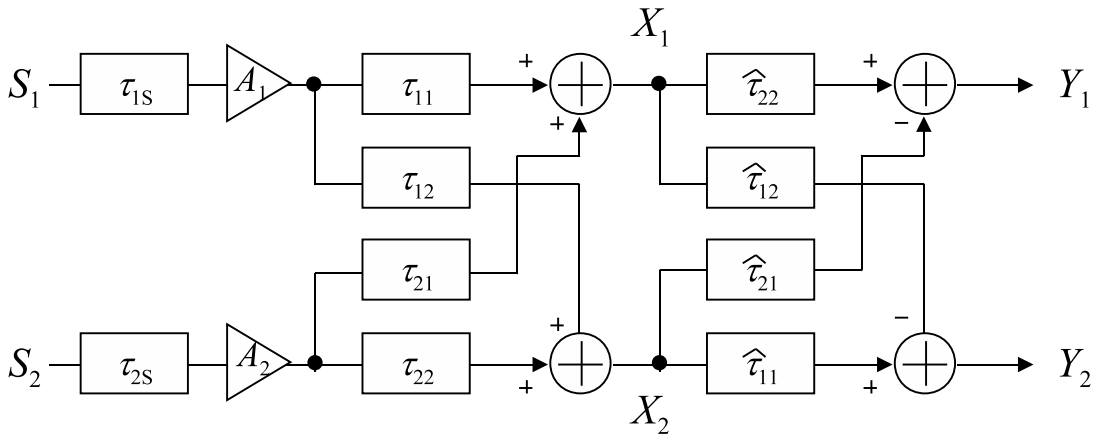

Figure 3.2: Block diagram of mixing and separation based on the signal flow. τ_{iS} and $A_i(k)$ are delay and gain corresponding to distance between source and center positions of microphone array. $\tau_{ij}(k)$ is the delay of each microphone, and $\hat{\tau}_{ij}(k)$ is the estimated delay from an estimated source direction. Left half means mixing procedure, and right half means separation procedure.

3.2 Proposed BSS method using two microphones

3.2.1 Signal model in the case of two microphones

The signal model is formulated by the same manner described in Section 2.1. In this chapter, the case of two microphones is considered; there are two source signals and two microphones. In this case, the source and observed signal vectors are formulated as:

$$\begin{aligned}\mathbf{S}(k,l) &= \left[S_1(k,l), S_2(k,l)\right]^{\mathrm{T}},\\ \mathbf{X}(k,l) &= \left[X_1(k,l), X_2(k,l)\right]^{\mathrm{T}}.\end{aligned} \tag{3.1}$$

where $(\cdot)^{\mathrm{T}}$ denotes the transpose operator. $S_{(\cdot)}(k,l)$ means the source signal in the frequency bin k and the frame l of the channel $(\cdot)$. $X_{(\cdot)}(k,l)$ denote the observed signal described as the same manner of $S_{(\cdot)}(k,l)$.

The left half of Figure 3.2 shows the block diagram of signal propagation, and indexes k and l are omitted to simplify the diagram. In Figure 3.2, the source signals are located

at direction $\theta_i(k)$ (i corresponds to the source number) in each frequency bin k, because in the reverberant condition the estimated source direction deviates in each frequency bin. The reverberation leads convolution among the reflected sound signals at the same time index. Each reflected sounds in a narrow frequency band have different time delays, which correspond to DOAs of each reflected sound. The reflected sounds are attenuated though their propagation, so that the observed signal equals to the sum of the direct sound, and the attenuated and different arrival sounds. Therefore, the observed direction deviates around the direction of the direct sound. τ_{iS} and $A_i(k)$ are a delay and a gain corresponding to the distance between the source and the center position of the microphone array. In addition, $\tau_{ij}(k)$ is the delay of each microphone based on the center position of the microphone array (j corresponds to the microphone number) in the k-th frequency bin.

Therefore, the mixing matrix $\mathbf{A}(k)$ is formulated as follows:

$$\mathbf{A}(k) = \begin{bmatrix} A_1(k)\mathrm{e}^{-j\omega(k)(\tau_{1S}+\tau_{11}(k))} & A_2(k)\mathrm{e}^{-j\omega(k)(\tau_{2S}+\tau_{21}(k))} \\ A_1(k)\mathrm{e}^{-j\omega(k)(\tau_{1S}+\tau_{12}(k))} & A_2(k)\mathrm{e}^{-j\omega(k)(\tau_{2S}+\tau_{22}(k))} \end{bmatrix}, \tag{3.2}$$

where $\omega(k)$ is an angular frequency that equals $2\pi(kF_s/N_F)$, F_s and N_F are the sampling frequency and the size of the FFT, respectively.

3.2.2 Frequency bin selection by the determinant of the spatial covariance matrix for reducing computational costs

In this section, the theoretical analysis of the frequency bin selection is introduced. Intuitively, if there is only one source in a frequency bin, the rank of the covariance matrix is not full and the determinant becomes zero. In contrast, if there are two sources, the rank of the spatial covariance matrix is full and the determinant never becomes zero. In other words, the determinant can describe the number of source signals, only for two microphones. Therefore, in the previously proposed semi-BSS method [70], the determinant of the spatial covariance matrix is used for bin selection criterion, however it was only intuitively used. Hereinafter, the determinant is theoretically analyzed. On the other hand, the power of observed signals is common criterion to describe the degree of spectral influence. A theoretical analysis of the trace of the spatial covariance matrix, corresponding the power

of the observed signals, is also introduced. In addition, experimental comparison is shown to clarify the difference between the determinant and the trace.

Spatial covariance matrix

The spatial covariance matrix $\mathbf{R}_x(k)$, in a frequency bin k, is calculated to evaluate the determinant and the trace as follows:

$$\mathbf{R}_x(k) = E_l\left[\mathbf{X}(k,l)\mathbf{X}^{\mathrm{H}}(k,l)\right], \tag{3.3}$$

where $E_l[\cdot]$ is the expectation operator over frame l and $(\cdot)^{\mathrm{H}}$ is the Hermitian operator.

Analysis of the determinant of the spatial covariance matrix

In this section, the determinant of the spatial covariance matrix is analyzed. The spatial covariance matrix of the observed signals $\mathbf{R}_x(k)$ is transformed using the mixing model in Eq. (2.1) as follows:

$$\begin{aligned}\mathbf{R}_x(k) &= E_l\Big[\mathbf{X}(k,l)\mathbf{X}^{\mathrm{H}}(k,l)\Big] \\ &= \mathbf{A}(k)E_l\Big[\mathbf{S}(k,l)\mathbf{S}^{\mathrm{H}}(k,l)\Big]\mathbf{A}^{\mathrm{H}}(k) \\ &\equiv \mathbf{A}(k)\mathbf{R}_s(k)\mathbf{A}^{\mathrm{H}}(k),\end{aligned} \tag{3.4}$$

where $\mathbf{R}_s(k)$ is the spatial covariance matrix of the source signals. According to the assumptions of FDICA, the source signals are independent each other. Therefore, $\mathbf{R}_s(k)$ becomes a diagonal matrix, and each element is equivalent to the source power $\sigma_i(k)$ in the k-th frequency bin as follows:

$$\mathbf{R}_s(k) = \begin{bmatrix}\sigma_1(k) & 0 \\ 0 & \sigma_2(k)\end{bmatrix}. \tag{3.5}$$

The determinant of the spatial covariance matrix, $\det\mathbf{R}_x(k)$, is written with Eq. (3.4) as

follows:

$$\begin{aligned} \det\mathbf{R}_x(k) &= \det\Big\{\mathbf{A}(k)\mathbf{R}_s(k)\mathbf{A}^{\mathrm{H}}(k)\Big\} \\ &= \det\mathbf{R}_s(k)\cdot\det\Big\{\mathbf{A}^{\mathrm{H}}(k)\mathbf{A}(k)\Big\}. \end{aligned} \tag{3.6}$$

Meanwhile, the determinant can be calculated by the product of the eigenvalues, and in the case of the diagonal matrix, the determinant consists of the product of the diagonal elements. The spatial covariance matrix of the source signals is the diagonal matrix; $\det\mathbf{R}_s(k)$ of Eq. (3.6) can be obtained as the product of $\sigma_i(k)$. Substituting Eq. (3.2) into $\det\{\mathbf{A}^{\mathrm{H}}(k)\mathbf{A}(k)\}$ of Eq. (3.6), the propagation component of the determinant is obtained as follows:

$$\begin{aligned} \det\Big\{\mathbf{A}^{\mathrm{H}}(k)\mathbf{A}(k)\Big\} &= \det\begin{bmatrix} 2A_1^2(k) & \\ A_1(k)A_2(k)\Big\{\mathrm{e}^{j\omega(k)(\tau_{21}(k)-\tau_{11}(k))}+\mathrm{e}^{j\omega(k)(\tau_{22}(k)-\tau_{12}(k))}\Big\} & \\ & A_1(k)A_2(k)\Big\{\mathrm{e}^{j\omega(k)(\tau_{11}(k)-\tau_{21}(k))}+\mathrm{e}^{j\omega(k)(\tau_{12}(k)-\tau_{22}(k))}\Big\} \\ & 2A_2^2(k) \end{bmatrix} \\ &= 2A_1^2(k)A_2^2(k)\Big[1-\cos\Big\{\omega(k)(\tau_{11}(k)-\tau_{21}(k)-\tau_{12}(k)+\tau_{22}(k))\Big\}\Big]. \end{aligned} \tag{3.7}$$

Consequently, the determinant of $\mathbf{R}_x(k)$ is obtained as follows:

$$\begin{aligned} &\det\mathbf{R}_x(k) = \\ &\quad 2A_1^2(k)A_2^2(k)\Big[1-\cos\Big\{\omega(k)(\tau_{11}(k)-\tau_{21}(k)-\tau_{12}(k)+\tau_{22}(k))\Big\}\Big]\sigma_1(k)\sigma_2(k). \end{aligned} \tag{3.8}$$

Analysis of the trace of the spatial covariance matrix

In this section, the trace of the spatial covariance matrix is analyzed. The trace is the sum of the diagonal elements, in addition, the trace corresponds to the sum of the eigenvalues. $\mathbf{R}_s(k)$ is the diagonal matrix, and thus $\mathbf{R}_x(k)$ can be isolated for each source as follows:

$$\mathbf{R}_x(k) = \mathbf{A}_1(k)\mathbf{R}_{s1}(k)\mathbf{A}_1^{\mathrm{H}}(k) + \mathbf{A}_2(k)\mathbf{R}_{s2}(k)\mathbf{A}_2^{\mathrm{H}}(k), \tag{3.9}$$

where $\mathbf{R}_{si}(k)$ has only one element as follows:

$$\begin{aligned}\mathbf{R}_{s1}(k) &= \begin{bmatrix}\sigma_1(k) & 0\\ 0 & 0\end{bmatrix},\\ \mathbf{R}_{s2}(k) &= \begin{bmatrix}0 & 0\\ 0 & \sigma_2(k)\end{bmatrix},\end{aligned} \tag{3.10}$$

and the mixing matrix $\mathbf{A}(k)$ can also be isolated as follows:

$$\begin{aligned}\mathbf{A}_1(k) &= \begin{bmatrix}A_1(k)\mathrm{e}^{-j\omega(k)\tau_{11}(k)} & 0\\ A_1(k)\mathrm{e}^{-j\omega(k)\tau_{12}(k)} & 0\end{bmatrix},\\ \mathbf{A}_2(k) &= \begin{bmatrix}0 & A_2(k)\mathrm{e}^{-j\omega(k)\tau_{21}(k)}\\ 0 & A_2(k)\mathrm{e}^{-j\omega(k)\tau_{22}(k)}\end{bmatrix}.\end{aligned} \tag{3.11}$$

The trace is a linear map, therefore, the trace of the spatial covariance matrix can be considered separately for each source. In addition, the trace is invariant under the cyclic permutations. Accordingly, the trace of the spatial covariance matrix is transformed as follows:

$$\begin{aligned}\mathrm{tr}\mathbf{R}_x(k) &= \mathrm{tr}\Big\{\mathbf{A}_1(k)\mathbf{R}_{s1}(k)\mathbf{A}_1^{\mathrm{H}}(k) + \mathbf{A}_2(k)\mathbf{R}_{s2}(k)\mathbf{A}_2^{\mathrm{H}}(k)\Big\}\\ &= \mathrm{tr}\Big\{\mathbf{R}_{s1}(k)\mathbf{A}_1^{\mathrm{H}}(k)\mathbf{A}_1(k)\Big\} + \mathrm{tr}\Big\{\mathbf{R}_{s2}(k)\mathbf{A}_2^{\mathrm{H}}(k)\mathbf{A}_2(k)\Big\}.\end{aligned} \tag{3.12}$$

Using the isolated covariance matrix Eq. (3.10) and the isolated mixing matrix Eq. (3.11), the term $\mathrm{tr}\{\mathbf{R}_{si}(k)\mathbf{A}_i^{\mathrm{H}}(k)\mathbf{A}_i(k)\}$ is transformed as follows:

$$\mathrm{tr}\Big\{\mathbf{R}_{si}\mathbf{A}_i^{\mathrm{H}}(k)\mathbf{A}_i(k)\Big\} = 2\sigma_i A_i^2(k). \tag{3.13}$$

Consequently, the trace is obtained as follows:

$$\mathrm{tr}\mathbf{R}_x(k) = A_1^2(k)\sigma_1(k) + A_2^2(k)\sigma_2(k). \tag{3.14}$$

Procedure for selecting frequency bins

For each criterion, the determinant or the trace, the same rule is used to select frequency bins. Since each criterion is a real value, the selection is performed according to the largest magnitude of the criterion, until the number of bins selected reaches the designated number. The selection procedure works as follows:

1. Calculate the spatial covariance matrix $\mathbf{R}_x(k)$.

2. Calculate either criterion, $\det\mathbf{R}_x(k)$ or $\mathrm{tr}\mathbf{R}_x(k)$.

3. Sort the criterion in descending order of its magnitude.

4. Select the designated number of bins from the sorted list of criteria.

The designated number of bins can be defined by system requirements or about 10 percent of the total number of bins can be selected; 10 percent means a rough estimate based on the experimental results discussed in a following section.

3.2.3 BSS using ICA in the selected frequency bins

In this section, obtaining separation matrix of ICA is introduced in the selected frequency bins, including the scaling solution and the permutation solution.

Separation matrix in selected frequency bins

Following STFT and selecting the frequency bins, the separation matrix $\mathbf{W}(k)$ is obtained using the general ICA algorithm in the selected bins, the update rule is introduced in Eq. (2.2).

Directions of arrival from the separation matrix and permutation solution

As mentioned in Section 2.2.4, the DOA of the source signals are estimated from the separation matrix, and the permutation is solved using the estimated DOA such as in [39].

For two microphones, the separation matrix is represented as follows:

$$\mathbf{W}(k) \equiv \begin{bmatrix} w_{11}(k) & w_{12}(k) \\ w_{21}(k) & w_{22}(k) \end{bmatrix} = \begin{bmatrix} \mathbf{w}_1(k) \\ \mathbf{w}_2(k) \end{bmatrix}, \tag{3.15}$$

where $\mathbf{w}_i(k) \equiv [w_{i1}(k)\, w_{i2}(k)]$. From the standpoint of array signal processing, the directivity pattern in each frequency bin is calculated from $\mathbf{w}_i(k)$, and the DOA of a source signal is estimated as follows:

$$\hat{\psi}_i(k) = \arg\min_{\psi} \left\{ \mathbf{w}_i^{\mathrm{T}}(k) \kappa(k, \psi) \right\}, \tag{3.16}$$

where $\kappa(k, \psi) = [1, \mathrm{e}^{j\rho(k,\psi)}]$ is a steering vector and $\rho(k, \psi)$ is the time difference corresponding to a steering direction as:

$$\rho(k, \psi) \equiv 2\pi \frac{kF_s}{N_F} \frac{d}{c} \sin(\psi), \tag{3.17}$$

where ψ is a steering direction to search the DOA. d is the distance between two microphones and c is the velocity of sound, respectively. Finding out the the directional null $\hat{\psi}_i(k)$ by Eq. (3.16), this corresponds to the source direction in each frequency bin.

However, the source directions are still not determined. The permutation problem is solved using the same method introduced in [39]. After clustering directional null in all the selected frequency bins, each $\hat{\psi}_i(k)$ belongs to the correct source cluster. This is the permutation solution, then the source direction $\hat{\theta}_i(k)$ is obtained.

Collected source directions $\hat{\theta}_i(k)$ are averaged over the selected bins to obtain the estimated DOA of the source signal, $\hat{\theta}_i$, as follows:

$$\hat{\theta}_i = \frac{1}{N_B} \sum_{k \in \Xi} \hat{\theta}_i(k), \tag{3.18}$$

where N_B is the number of bins and Ξ is a set of the selected bins.

Theoretical analysis of the scaling solution by the projection method for two microphones

The projection method in [37] is applied to solve the scaling ambiguity of the FDICA separation matrix. The separated signal by the projection method corresponds to one of observed signals, and a theoretical analysis of this fact is introduced in this section. Substituting Eq. (3.1) and Eq. (3.2) into Eq. (2.1), when the observed source signal is denoted as $X_{ij}(k,l)$ for the i-th source signal at the j-th microphone, the observed signal is transformed as follows:

$$\begin{aligned}\begin{bmatrix} X_1(k,l) \\ X_2(k,l) \end{bmatrix} &= \begin{bmatrix} X_{11}(k,l) + X_{21}(k,l) \\ X_{12}(k,l) + X_{22}(k,l) \end{bmatrix} \\ &= \begin{bmatrix} \mathrm{e}^{-j\omega(k)\tau_{11}(k)} & \mathrm{e}^{-j\omega(k)\tau_{21}(k)} \\ \mathrm{e}^{-j\omega(k)\tau_{12}(k)} & \mathrm{e}^{-j\omega(k)\tau_{22}(k)} \end{bmatrix} \begin{bmatrix} A_1(k)S_1(k,l) \\ A_2(k)S_2(k,l) \end{bmatrix} \\ &\equiv \mathbf{B}(k)\mathbf{S}'(k,l),\end{aligned} \tag{3.19}$$

where $\mathbf{B}(k)$ only consists of delay factors of the mixing matrix $\mathbf{A}(k)$; in other words, $\mathbf{B}(k)$ corresponds to the mixing matrix with the plane wave assumption. $\mathbf{S}'(k,l)$ is the source signal with the observed amplitude. If the separation matrix $\mathbf{W}(k)$ corresponds to the inverse of the mixing matrix $\mathbf{B}(k)$, Cramer's formula can be applied to transform the separation matrix as follows:

$$\begin{aligned}\mathbf{W}(k) &= \mathbf{D}(k)\mathbf{B}^{-1}(k) \\ &= \mathbf{D}(k)\frac{1}{\mathrm{det}\mathbf{B}(k)}\tilde{\mathbf{B}}(k) \\ &\equiv \begin{bmatrix} \lambda_1(k) & 0 \\ 0 & \lambda_2(k) \end{bmatrix} \begin{bmatrix} \mathrm{e}^{-j\omega(k)\tau_{22}(k)} & -\mathrm{e}^{-j\omega(k)\tau_{21}(k)} \\ -\mathrm{e}^{-j\omega(k)\tau_{12}(k)} & \mathrm{e}^{-j\omega(k)\tau_{11}(k)} \end{bmatrix} \\ &\equiv \mathbf{\Lambda}(k)\tilde{\mathbf{B}}(k),\end{aligned} \tag{3.20}$$

where $\mathbf{D}(k)$ means the matrix of the scaling ambiguity. $\mathbf{\Lambda}(k)$ is the scaling matrix that consists of the scaling ambiguity $\mathbf{D}(k)$ and the reciprocal number of the determinant of the matrix $\mathbf{B}(k)$. $\tilde{\mathbf{B}}(k)$ corresponds to part of the inverse matrix $\mathbf{B}(k)$.

The projection method can solve the scaling ambiguity by calculating $\mathrm{diag}\{\mathbf{W}^{-1}(k)\}\mathbf{W}(k)$.

The operator diag(·) remains diagonal elements of matrix (·) and the other all elements set zero. Therefore the FDICA separated signal $\mathbf{Y}(k,l)$ is obtained as follows:

$$\begin{aligned}
\mathbf{Y}(k,l) &= \mathrm{diag}\left\{\mathbf{W}^{-1}(k)\right\}\mathbf{W}(k)\mathbf{X}(k,l) \\
&= \frac{1}{\det\tilde{\mathbf{B}}(k)}\begin{bmatrix} \mathrm{e}^{-j\omega(k)\tau_{11}(k)} & 0 \\ 0 & \mathrm{e}^{-j\omega(k)\tau_{22}(k)} \end{bmatrix}\mathbf{\Lambda}^{-1}(k)\mathbf{\Lambda}(k)\tilde{\mathbf{B}}(k)\mathbf{B}(k)\mathbf{S}'(k,l) \\
&= \frac{1}{\det\mathbf{B}(k)}\begin{bmatrix} \mathrm{e}^{-j\omega(k)\tau_{11}(k)} & 0 \\ 0 & \mathrm{e}^{-j\omega(k)\tau_{22}(k)} \end{bmatrix}\mathbf{I}\left(\{\det\mathbf{B}(k)\}\,\mathbf{I}\right)\mathbf{S}'(k,l) \\
&= \begin{bmatrix} \mathrm{e}^{-j\omega(k)\tau_{11}(k)} & 0 \\ 0 & \mathrm{e}^{-j\omega(k)\tau_{22}(k)} \end{bmatrix}\mathbf{S}'(k,l) \\
&= \begin{bmatrix} X_{11}(k,l) \\ X_{22}(k,l) \end{bmatrix}.
\end{aligned} \tag{3.21}$$

Consequently, the separated signal by the projection method corresponds to one of the observed source signals, $X_{ii}(k,l)$ for the i-the source signal at the i-th microphone.

Finally, $\mathrm{diag}\{\mathbf{W}^{-1}(k)\}\mathbf{W}(k)$ can be considered as the new separation matrix that the scaling ambiguity is solved by the projection method.

3.2.4 Frame-wise Wiener filter in unselected frequency bins

In this section, a frame-wise Wiener filter method is proposed, which improves separation performance, especially in the low frequency region.

Tentative separation by NBF and its theoretical analysis

The delay value $(d/c)\sin(\hat{\theta}_i)$ is calculated by the estimated DOA $\hat{\theta}_i$, and this is applied to the NBF to obtain tentative separated signals. As mentioned in Section 3.2.1, propagation is assumed to be from the source position to the center position of the microphones. The source direction only depends on the delay between two microphones. Considering the fact that the separated signal by the projection method corresponds to one of the observed source signals. In this case, the propagation delay τ_{iS} can be omitted without loss of generality, if we assume that the distance between the microphones is small enough, such as in

mobile equipments. In addition, it is a reasonable consideration that the distance between the source position and the microphone position can be represented as the gain $A_i(k)$ by the spherical wave assumption. Consequently, the source position is only depends on the source direction which is represented as the delay $\tau_{ij}(k)$. The observed signal $X_j(k,l)$ can be written as follows:

$$X_j(k) = A_1(k)S_1(k,l)\mathrm{e}^{-j\omega(k)\tau_{1j}(k)} + A_2(k)S_2(k,l)\mathrm{e}^{-j\omega(k)\tau_{2j}(k)}. \tag{3.22}$$

The right half of Figure 3.2 shows the block diagram of the tentative separation, and this process represents the null-beamformer. The output signals of the NBF are formulated as follows:

$$\begin{aligned} Y_1^{\mathrm{NBF}}(k,l) &= X_1(k,l)\mathrm{e}^{-j\omega(k)\hat{\tau}_{22}} - X_2(k,l)\mathrm{e}^{-j\omega(k)\hat{\tau}_{21}}, \\ Y_2^{\mathrm{NBF}}(k,l) &= -X_1(k,l)\mathrm{e}^{-j\omega(k)\hat{\tau}_{12}} + X_2(k,l)\mathrm{e}^{-j\omega(k)\hat{\tau}_{11}}, \end{aligned} \tag{3.23}$$

where the estimated delay of each channel is considered as follows:

$$\begin{aligned} \hat{\tau}_{11} &= -(d/c)\sin(\hat{\theta}_1)/2, \\ \hat{\tau}_{12} &= (d/c)\sin(\hat{\theta}_1)/2, \\ \hat{\tau}_{21} &= (d/c)\sin(\hat{\theta}_2)/2, \\ \hat{\tau}_{22} &= -(d/c)\sin(\hat{\theta}_2)/2. \end{aligned} \tag{3.24}$$

Substituting Eq. (3.22) into Eq. (3.23), the relationship between the source signals $S_1(k,l), S_2(k,l)$ and the tentative separated signal $Y_1^{\mathrm{NBF}}(k,l)$ is obtained as follows:

$$\begin{aligned} Y_1^{\mathrm{NBF}}(k,l) &= X_1(k,l)\mathrm{e}^{-j\omega(k)\hat{\tau}_{22}} - X_2(k,l)\mathrm{e}^{-j\omega(k)\hat{\tau}_{21}} \\ &= A_1(k)S_1(k,l)\mathrm{e}^{-j\omega(k)(\tau_{11}(k)+\hat{\tau}_{22})} + A_2(k)S_2(k,l)\mathrm{e}^{-j\omega(k)(\tau_{21}(k)+\hat{\tau}_{22})} \\ &- A_1(k)S_1(k,l)\mathrm{e}^{-j\omega(k)(\tau_{12}(k)+\hat{\tau}_{21})} - A_2(k)S_2(k,l)\mathrm{e}^{-j\omega(k)(\tau_{22}(k)+\hat{\tau}_{21})}. \end{aligned} \tag{3.25}$$

In the reverberant condition, the estimated DOA in each frequency bin is deviated by a mixing of direct and reflected sounds. However, a deviation is quite small, because a direct sound is much stronger than reflected sounds generally. According to this fact, assuming

that the estimated direction $\hat{\theta}_i$ is approximately equivalent to the source direction θ_i; therefore $\tau_{ij}(k) \approx \hat{\tau}_{ij}$ can be an appropriate assumption. $Y_2^{\mathrm{NBF}}(k,l)$ is calculated in the same way, and we obtain the approximate relationship as follows:

$$\begin{aligned} Y_i^{\mathrm{NBF}}(k,l) &\approx A_i(k)S_i(k,l)\left\{\mathrm{e}^{-j\omega(k)(\hat{\tau}_{11}+\hat{\tau}_{22})} - \mathrm{e}^{-j\omega(k)(\hat{\tau}_{12}+\hat{\tau}_{21})}\right\} \\ &\equiv A_i(k)S_i(k,l)C(k). \end{aligned} \tag{3.26}$$

where $C(k)$ is a constant corresponding to the delay term in the frequency bin k. In the low frequency region, $\omega(k)$ is smaller than in the high frequency region, and thus the amplitude of the delay section of Eq. (3.26) takes a small value due to the low frequency. This is the main reason of the degradation which appears in an output signal of the NBF, as mentioned in Section 3.2.

Wiener filter obtained by the tentative separated signals

Considering a cost function to obtain the Wiener filter with regard to the observed source signal as follows:

$$\min_{\alpha_i} E_l\left[\left\{A_i(k)S_i(k,l) - \alpha_i X_i(k,l)\right\}^2\right] \tag{3.27}$$

where E_l is the expectation operator and α_i is a variable. To minimize the cost function in Eq. (3.27), the differentiations of α_i are considered, and the assumption of the independence between each source signal is also considered. When the independence and expectation are utilized to derive the Wiener filter, cross correlation between each source signal become zero. Therefore, we obtain the Wiener filter and it is approximated by the tentative separated signals in Eq. (3.26) as follows:

$$\begin{aligned} \alpha_i &= \frac{E_l\left[A_i^2(k)|S_i(k,l)|^2\right]}{E_l\left[A_1^2(k)|S_1(k,l)|^2\right] + E_l\left[A_2^2(k)|S_2(k,l)|^2\right]} \\ &\approx \frac{E_l\left[|Y_i^{\mathrm{NBF}}(k,l)|^2/C^2(k)\right]}{E_l\left[|Y_1^{\mathrm{NBF}}(k,l)|^2/C^2(k)\right] + E_l\left[|Y_2^{\mathrm{NBF}}(k,l)|^2/C^2(k)\right]}, \\ &= \frac{E_l\left[|Y_i^{\mathrm{NBF}}(k,l)|^2\right]}{E_l\left[|Y_1^{\mathrm{NBF}}(k,l)|^2\right] + E_l\left[|Y_2^{\mathrm{NBF}}(k,l)|^2\right]}. \end{aligned} \tag{3.28}$$

$C(k)$ corresponds to the delay term, and it describes the frequency characteristics of the NBF. The proposed Wiener filter in Eq. (3.28) cancels out the affect of the NBF characteristics $C(k)$, and this means preventing signal attenuation in the low frequency region cased by the NBF.

In this research, the shortest expectation is considered in order to reduce computational costs. The frame-wise Wiener filter $M_i(k,l)$ is obtained as follows:

$$M_i(k,l) = \frac{\left|Y_i^{\mathrm{NBF}}(k,l)\right|^2}{\left|Y_1^{\mathrm{NBF}}(k,l)\right|^2 + \left|Y_2^{\mathrm{NBF}}(k,l)\right|^2}. \tag{3.29}$$

The obtained frame-wise Wiener filter varies frame by frame, and thus it can trace temporal changes in each speech source signal.

Finally, the output separated signals are obtained as follows:

$$\mathbf{Y}(k,l) = \begin{cases} \mathbf{W}(k)\mathbf{X}(k,l) & \text{(FDICA)} \\ \begin{bmatrix} M_1(k,l)X_1(k,l) \\ M_2(k,l)X_2(k,l) \end{bmatrix} & \text{(Wiener filter)} \end{cases}. \tag{3.30}$$

3.3 Evaluation

In this section, computational costs of the proposed method are estimated, and the performance of the proposed method is evaluated by a source separation simulation.

3.3.1 Comparison of selection criteria between determinant and trace of spatial covariance matrix

In this section, performance comparison of frequency bin selection is introduced to evaluate which criteria is more appropriate.

As mentioned in Section 3.2.2, if there is only one source in one of frequency bins, the rank of $\mathbf{R}_x(k)$ is not full and the determinant of $\mathbf{R}_x(k)$ becomes zero. According to Eq. (3.14), if there is only one source in one of frequency bins, the trace never becomes

zero. In addition, Eq. (3.8) contains the term $1 - \cos(\{\omega(k)\})$, and this works as a weighting factor. $1 - \cos(\{\omega(k)\})$ takes smaller values according to the lower frequency; this corresponds to the relative attenuation against the higher frequency. Therefore, the term $1 - \cos(\{\omega(k)\})$ means that bin selection is unlikely to occur in the low frequency region via the determinant criterion. As mentioned in Section 3.1, the phase difference becomes small in the low frequency region, and this is a disadvantage of using microphone array signal processing methods such as FDICA. In light of these considerations, the trace has a tendency to select bins occur in the low frequency region, more so than the determinant. This implies that the determinant is better suite than the trace to select frequency bins.

This tendency is evaluated by a bin selection experiment in which the DOA is estimated from the separation matrix. In this experiment, the performance difference should be confirmed, in the DOA estimation task using the separation matrix. Therefore, the FDICA algorithm and parameters are same as in the other experiments, meanwhile the simulation is performed only for the anechoic condition and with 32 bins selected. This is because, in the reverberant condition, there is a deviation in the estimated DOA, and it is difficult to evaluate the difference in performance from the low frequency bin to the high frequency bin.

The covariance matrix is calculated using recorded speech signals in each frequency bin, then the determinant and trace are calculated respectively. Following the calculation of the criterion, some bins are selected according to the selection process in Section 3.2.2. For the selected bins, the FDICA separation matrix is iteratively estimated using Eq. (2.2), and the DOA of the source signal is estimated via the FDICA separation matrix in each frequency bin. In this evaluation, the number of the bins selected is set at 32 because it is easier to confirm the tendency in which bins are selected. The direction of the source signals is set about {-45,45} degrees; note that this direction is not precise because the loudspeakers are set by hand.

Figure 3.3 shows an example of the determinant and the trace of the covariance matrix in each frequency bin. The horizontal axis shows the frequency. The vertical axis shows the normalized value of the determinant and trace. The determinant and trace are averaged over the different voice types. Normalization is applied to show the characteristics, so each value is divided by the maximum value. Bin selection is performed based on the relative

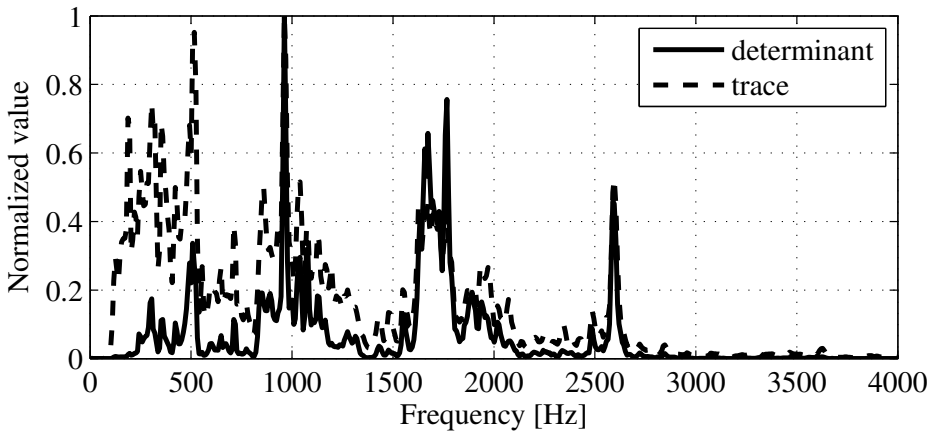

Figure 3.3: Example of determinant/trace of covariance matrix. Solid and dashed lines indicate determinant and trace, respectively. Each value is normalized by maximum value of each criterion.

value of the determinant or the trace over the frequency bins. Therefore, the normalization is appropriate without loss of generality. In Figure 3.3, the values in the low frequency region show significant differences, in particular that the values for the determinant are lower than those for the trace.

Figure 3.4 shows experimental results of the DOA estimation from the FDICA separation matrix. The DOA estimation might be affected by noise signals in the real world, for example, acoustical ambient noise or electric circuit noise. Therefore the performance of the beamformer is degraded in the low frequency region. As shown in Figure 3.4(b), the trace criteria selects more bins in the low frequency region than the determinant criteria. In addition, since this experiment is performed in the anechoic condition, the estimated directions should not vary among the selected bins. However, the estimated directions via the trace criteria are more varied than the determinant criteria in the low frequency region, and this fact implies that noise signals affected the DOA estimation. Consequently, DOA estimation via the determinant criteria is more effective and precise than estimation using the trace criteria; in other words, it is advantageous to use the determinant for bin selection.

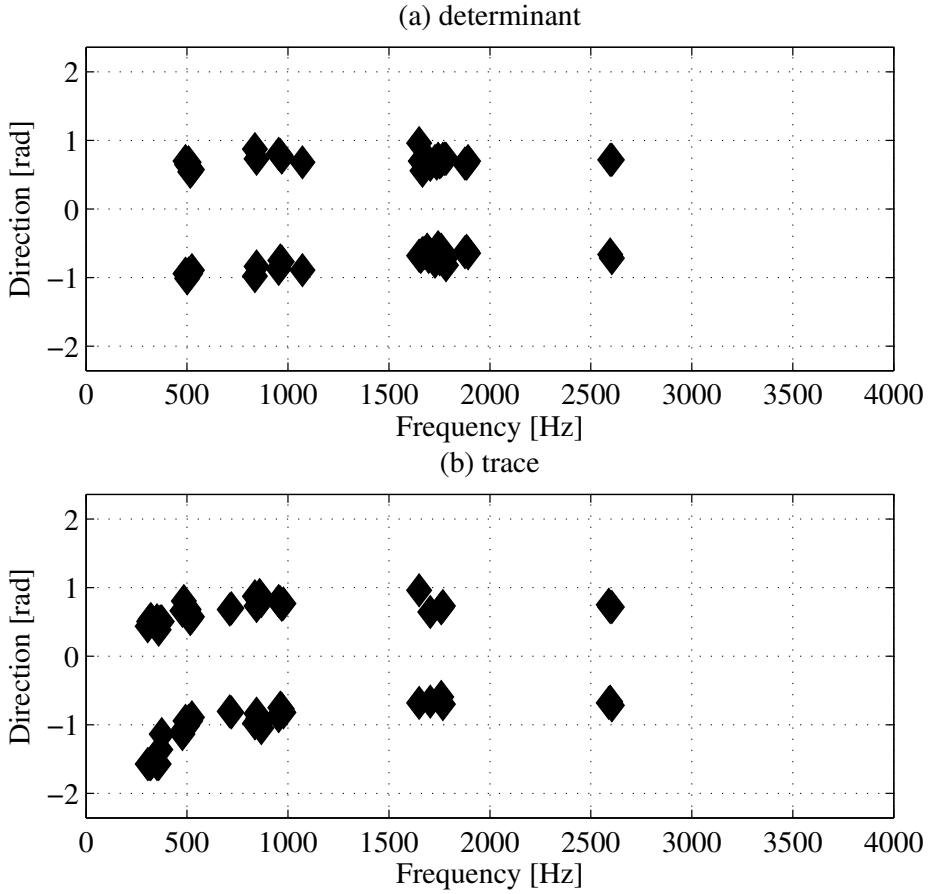

Figure 3.4: Estimated DOAs via determinant/trace selection criteria under anechoic condition. Figs. 3.4 (a) and 3.4 (b) show examples of estimated DOAs in frequency bins selected by each criteria. In Fig. 3.4 (b), trace shows a tendency in low frequency region, in which more lower frequency bins are selected and deviations of estimated DOAs appear.

Table 3.1: FDICA parameters for evaluations with two microphones

FFT Size	1024 samples
FFT Shift	256 samples
Learning Time	3 seconds
Iteration	max.200 times
Step Size	0.01
Initial Matrix	Identity
Permutation Solution	DOA [39]
Scaling Solution	Projection method [37]

3.3.2 Estimate of computational costs in the case of two microphones

Estimating the number of operations (multiplication, addition as floating operations) based on Eq. (2.2) is used to evaluate the computational efficiency of the proposed method. The parameters of FDICA for the computational costs estimation are shown in Table 3.1. The estimated computational costs are shown in Table 3.2. The number of frequency bins for the estimate is 64, which is determined from the experimental results in Section 3.3.3. 'Separation' in Table 3.2 includes operations corresponding to the separation matrix, the NBF and the Wiener filter. The unit 'MOPs' represents the number of mega-operations per second. Each percentage in Table 3.2 means a ratio to conventional FDICA; conventional FDICA means that the all of the frequency bins are used for BSS using ICA.

As shown in Table 3.2, the proposed method achieves an over 80 percent reduction in the level of computational costs compared with conventional FDICA. The computational costs of the proposed method is almost equivalent to previously proposed semi-BSS method. In addition, obtaining the separation matrix using an iterative update rule shows the dominant computational cost, and the additional cost by the proposed method is sufficiently small in Table 3.2. The frequency bin selection method significantly contributes the reduction in the level of computational costs.

Table 3.2: Estimated computational costs and ratios compared to conventional FDICA

	conventional FDICA	conventional semi-BSS		proposed	
FFT	8.5	8.5	100%	8.5	100%
Covariance Matrix	–	1.6	(additional)	1.6	(additional)
Separation Matrix	275	34	12.5%	34	12.5%
Projection Method	0.05	0.006	12.5%	0.006	12.5%
Permutation Solver	2.8	0.35	12.5%	0.35	12.5%
Separation	1	1	100%	3.2	320%
Total	287	46	16%	48	17%

3.3.3 Source separation simulation

Simulation conditions

The speech signals are recorded with two omni-directional microphones (SHURE SM93) and the distance between the microphones is 3.6 cm. The recording conditions are shown in Table 3.3 and Figure 3.5. The voice signals are played back via loudspeakers and recorded individually. The observed signals are mixed in the same energy when the simulation is performed; in other words, a mixing level is 0 dB. The parameters of FDICA for the simulation are shown in Table 3.1; they are same as in the case of the computational costs estimate. The number of bins ranges from 384 to 32 because these numbers are roughly the ratio of integers, $3/4, 1/2, \cdots, 1/16$, for the total number of frequency bins, which is 513.

As mentioned in Chapter 1, FDICA must store the observed signals whose lengths are longer than a few seconds. Additionally, the separated signals to estimate higher-order statistics are different in every iteration, because the separation matrix is updated in every iteration. These are the primary reasons that the computational costs required for FDICA is high. Therefore, the smaller number of selections contributes to lower computational costs,

Table 3.3: Signals for simulation using two microphones

	Anechoic	Reverberant
Samp. Freq.	8 kHz	
Rev. Time	–	500 msec
Voice type	Male(2), Female(2)	
Location pair	{-45,45}, {-90,0}, {-45,0} degrees	

and thus results in smaller memory consumption as well.

Although the classical ICA update rule is used in this research (see Eq. (2.2) and Section 3.2.3), it can be replaced with any state-of-the-art ICA update rule. Therefore, the proposed method, which consists of bin selection and the use of a frame-wise Wiener filter for the unselected bins, is an improvement of the FDICA method, without loss of generality.

Evaluation measure

The performance of the proposed method is evaluated using the segmental signal-to-noise ratio (SNR_{seg}) [71] and cepstral distortion (CD) [72]. SNR_{seg} is a very common measure for evaluating noise suppression, for example, in high-efficiency coding such as MP3, etc. In general, SNR_{seg} is known to have a better correlation with the perception of noisy speech by humans than entire interval SNR [71]. The proposed method is based on the Wiener filter, in other words time-frequency masking method, and thus the degradation of the separated signals can be estimated. Therefore, SNR_{seg} is appropriate for evaluating the proposed method. CD is another measure of the degree of distortion in the cepstrum domain, and this can evaluate distortion of a spectral envelope.

The projection method [37] is used to solve the scaling ambiguity of FDICA, as mentioned in Table 3.1. This means that each separated signal corresponds to one of the observed source signals at the microphones. Therefore, the objective measures are calculated between one of the observed source signals and the separated signal. Each term of the right side in Eq. (3.22) corresponds to one of the observed source signals. The observed signal

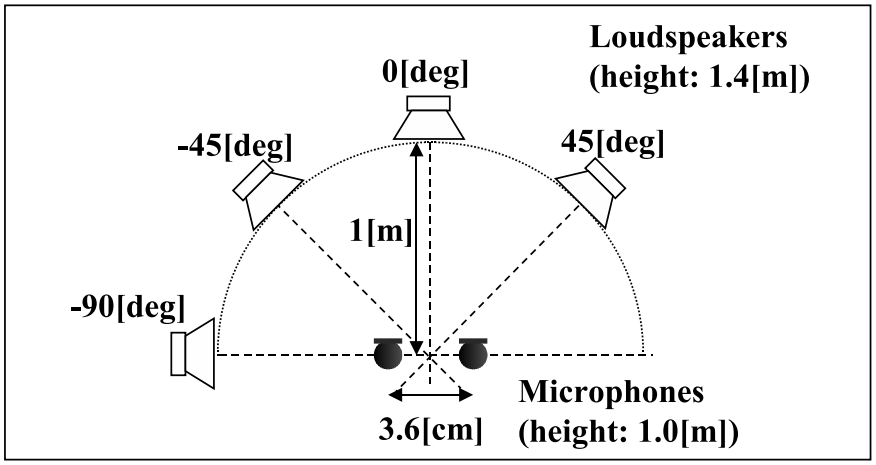

Figure 3.5: Recording setup of microphones and loudspeakers for two-microphones BSS. This loudspeaker position is used for evaluations: {-45,45}, {-90,0}, {-45,0} degrees. Hight and distance of omni-directional microphones are 1 meter and 3.6 cm, respectively. Loudspeakers at 1.4 meter individually play recorded speech signals.

at the j-th microphone in the time domain $x_j(n)$ is formulated as follows:

$$\begin{aligned} x_1(n) &= x_{11}(n) + x_{21}(n), \\ x_2(n) &= x_{12}(n) + x_{22}(n), \end{aligned} \tag{3.31}$$

where i is the source number. $x_{ij}(n)$ corresponds to the time domain signal corresponding to $X_{ij}(k,l)$ in Eq. (3.19). In this case, $\mathrm{SNR}_{\mathrm{seg}}$ is defined as follows:

$$\mathrm{SNR}_{\mathrm{seg}} \equiv \frac{1}{2}\sum_i \frac{1}{N_{l_s}} \sum_{l_s} 10\log_{10} \frac{\sum_m x_{ii}^2(m,l_s)}{\sum_m \{x_{ii}(m,l_s) - y_i(m,l_s)\}^2}, \tag{3.32}$$

where $x_{ij}(m,l)$ and $y_i(m,l)$ are the observed source signal and the separated signal in the time domain of frame l_s and time m in the frame. N_{l_s} is the number of frames, and the obtained $\mathrm{SNR}_{\mathrm{seg}}$ for each channel are averaged. CD is calculated only during the speech

Table 3.4: Experimental results: segmental SNR

Number of selected bins	Computational costs	PREV,A [dB]	PROP,A [dB]	PREV,R [dB]	PROP,R [dB]
513	1.0	5.02	4.80	0.96	1.06
384	0.76	3.42	5.23	0.56	1.13
256	0.53	1.34	5.70	-0.41	1.54
192	0.41	0.34	5.89	-0.89	1.79
128	0.29	-0.65	6.07	-1.61	2.30
64	0.17	-1.59	6.86	-2.27	2.84
32	0.10	-1.90	7.04	-2.74	2.91

components, in other words only during utterance intervals, and it is defined as follows:

$$\mathrm{CD} \equiv \frac{1}{2}\sum_{i}\frac{20}{N_{l_c}\log 10}\left\{\sum_{l_c}\sqrt{\sum_{\nu=1}^{B}2\left(C_{x_{ii}(m,l_c)}(\nu,l_c)-C_{y_i(m,l_c)}(\nu,l_c)\right)^2}\right\}, \tag{3.33}$$

where $C_{(\cdot)}(\nu, l_c)$ is the ν-th cepstral coefficient of the signal $(\cdot)$ in frame l_c, and N_{l_c} is the number of frames. The obtained CD values for each channel are averaged; note that '$1/2$' means an average for two channels. B is the number of dimensions of the cepstrum used in the evaluation; we set $B = 20$. A small CD value indicates that the sound quality of the separated signal is high.

Simulation results of source separation

Table 3.4 and 3.5 and Figure 3.6 and 3.7 show the performance of the proposed method. 'PREV' denotes the previously proposed semi-BSS method and 'PROP' denotes the proposed method. 'A' and 'R' denote anechoic and reverberant conditions. The computational costs corresponding to the number of bins selected are estimated by the same rule in Section 3.3.2. The x-axis shows the computational costs, and the y-axis shows $\mathrm{SNR}_{\mathrm{seg}}$ and

Table 3.5: Experimental results: cepstral distortion

Number of selected bins	Computational costs	PREV,A [dB]	PROP,A [dB]	PREV,R [dB]	PROP,R [dB]
513	1.0	0.42	0.44	1.47	1.46
384	0.76	1.01	0.60	2.90	1.63
256	0.53	1.80	0.87	4.58	2.03
192	0.41	2.18	1.00	5.29	2.26
128	0.29	2.73	1.19	6.20	2.59
64	0.17	3.30	1.22	7.48	2.73
32	0.10	3.75	1.40	8.76	2.91

CD, respectively. The y-axis of CD has been flipped because a lower CD value indicates better sound quality. Figure 3.6 and Figure 3.7 show that the proposed method is significantly better than the conventional method. According to that the number of frequency bins is reduced, SNR_{seg} is improved and CD is deteriorated by the proposed method, respectively. This means that the proposed method shows a trade-off, and that 64 might be the best number of bins to select in this experiment.

Figure 3.8 shows the performance comparison among other methods. The x-axis shows SNR_{seg}, and the y-axis shows CD, respectively. Again, the y-axis is turned over because CD improves with a smaller value. 'PROP', 'FDICA' and 'DUET' denote the proposed method, conventional FDICA and DUET, respectively, while 'A' and 'R' denote anechoic and reverberant conditions. In this comparison, the number of bins selected is 64. For performing DUET, only the time difference is used as the similarity criteria to classify the source signals, because the power of a source signal is same as each other in this experiment. In Figure 3.8, the upper right corner shows better performance, and the lower left corner shows worse performance. Therefore, the sound quality of the proposed method is better than conventional FDICA for SNR_{seg}, and better than DUET for CD. This tendency

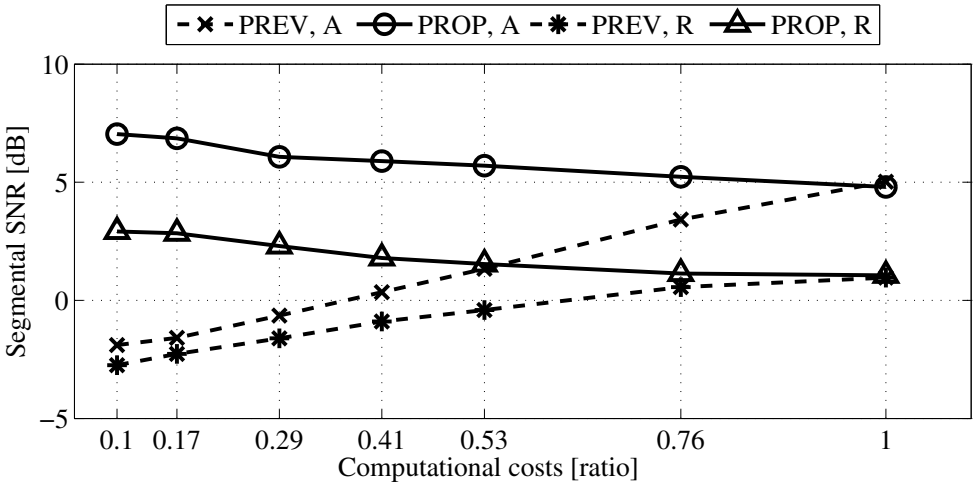

Figure 3.6: Segmental signal-to-noise ratio. Solid and dashed lines correspond to proposed and previously proposed semi-BSS methods, respectively. 'A' means anechoic condition, and 'cross' and 'circle' indicate the same condition. 'R' means reverberant condition, and 'asterisk' and 'triangle' indicate the same condition.

is shown in both anechoic and reverberant conditions.

3.4 Discussion

As shown in Table 3.2, the proposed method significantly reduces computational costs. In Section 3.3.3, DUET is compared with the proposed method with regard to the separation performance. DUET is a clustering method, therefore its computational cost strongly depends on a length of the observed signal. We consider that k-means clustering is used as the clustering method to obtain the separated signals for DUET. Under the same condition in Table 3.1 for FFT and other parameters, the computational cost of DUET for 3, 10 and 30 seconds observed signals are estimated as 60, 200 and 600 MOPs. If the positions of the source signals would not be changed so much during the sound capture, for example every speaker is sitting a chair in a meeting room, FDICA store the shorter length of observed signals than the total length of the observed signals. From these considerations and

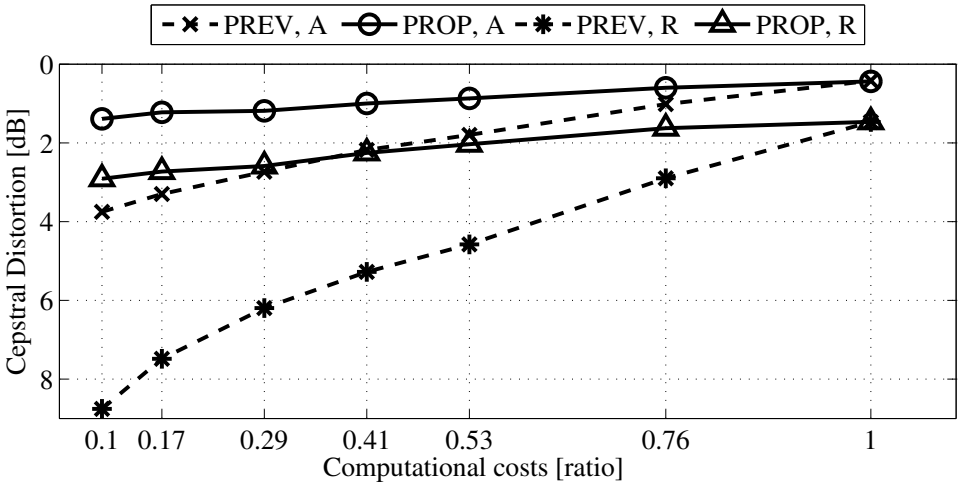

Figure 3.7: Cepstral distortion. Solid and dashed lines correspond to proposed and previously proposed semi-BSS methods, respectively. 'A' means anechoic condition, and 'cross' and 'circle' indicate the same condition. 'R' means reverberant condition, and 'asterisk' and 'triangle' indicate the same condition. Y-axis is turned over because it improves with a smaller value.

the estimate result in Table 3.2, the proposed method is more reliable than DUET from the viewpoint of computational costs.

In Section 1.3, DSPs have been discussed to be appropriate devices for the speech processing equipment because of its architecture, and for mobile devices, the low-power DSPs have been discussed and assumed to be implemented BSS using FDICA into, as mentioned in Section 2.3.1. The required wait states of DRAM external memory are an important issue, so that the estimated computational costs include this issue by tripling the number of required operations. The computational cost of conventional FDICA is almost 900 MOPs. The power consumption is much lower than the high-performance DSPs, however, the operating frequency of the low-power DSPs is around 200 MHz. 144 MOPs are assumed as the computational cost of the proposed method with tripling computational costs in Table 3.2. The estimation shows that the proposed method can be implemented into the low-power DSPs.

Note that, such microprocessors as ARM architecture processors work at high speed,

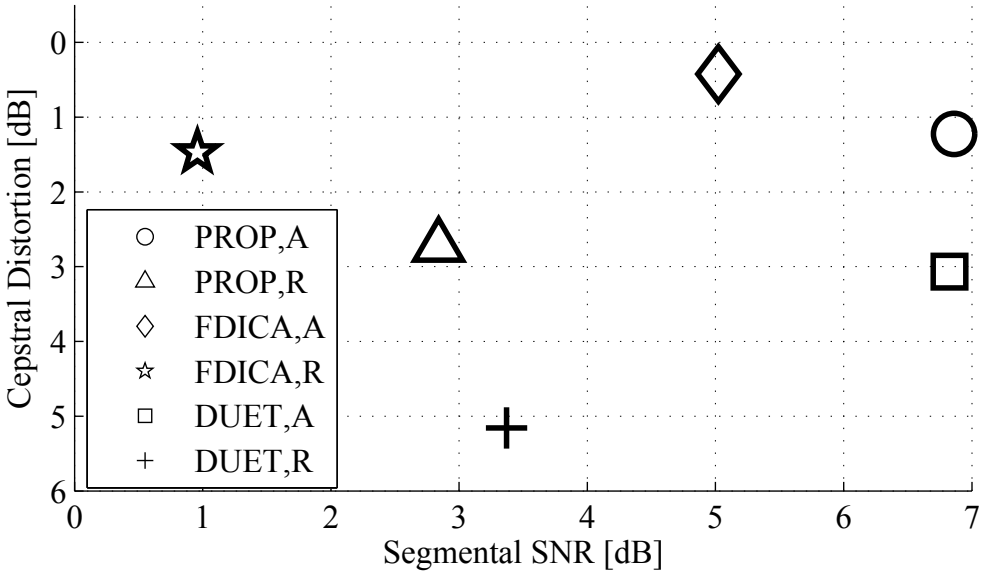

Figure 3.8: Performance comparison with conventional FDICA, proposed method, and DUET. X-axis shows segmental signal-to-noise ratio, and y-axis indicates cepstral distortion. 'Diamond' and 'star' indicate conventional FDICA under anechoic and reverberant conditions, respectively. 'Circle' and 'triangle' indicate proposed method under the same conditions. 'Square' and 'plus' indicate DUET under the same condition.

over 1 GHz. However 900 MOPs corresponds to 900 MHz and BSS concurrently works with many rich functions, so consuming full of the processor performance is a difficult way to realize the mobile speech equipment.

For such the RISC processors as the ARM architecture processors, the proposed method consumes around one-fifth of the all processing performance. The other four-fifth of the all processing performance can be consumed to perform the functions, for example the user interface, besides the basic speech function. Therefore, it can be possible that the proposed method in RISC processors can work the other functions cuncurrently, and in other words, the proposed method can be implemented even in the present RISC processors. Therefore, it can be said that the proposed method shows a practical performance for mobile equipment.

Figure 3.6 shows that the proposed method's separation performance exceeds not only the previously proposed semi-BSS method but also conventional FDICA. The proposed method combines the separation matrix and the Wiener filter; linear and non-linear processing in the frequency domain. Since the FDICA separation matrix corresponds to the coefficients of the beamformer [39], especially in the low frequency region, the separated signal is attenuated by a property of the NBF. On the other hand, the proposed method cancels out the NBF drawback in Eq. (3.28). This shows that the combination of linear and non-linear processing in the frequency domain has never failed the performance, on the contrary, the non-linear processing can cover the degradation of the linear processing. The proposed method is somewhat better than both conventional FDICA and DUET in Figure 3.8. In addition, the efficiency of the proposed method is evaluated via a computational estimate. Therefore, from these results, the proposed method shows both the effectiveness and efficiency, concurrently.

Chapter 4

Blind source separation for the portable equipment with DHMAs

In Chapter 2, the dodecahedral microphone array (DHMA) and its BSS method using DHMAs are briefly introduced. DHMA is advantageous to solve the permutation problem with hierarchical clustering because its dodecahedral shape reflects acoustic characteristics more than the spherical microphone array. Therefore, the spatial correlation of DHMAs can be expected to reduce computational costs like the case of two microphones, because the spatial correlation depends on its shape. The magnitude squared coherence is one of common measures of the spatial correlation, therefore this feature can be utilized for reducing computational costs by the frequency bin selection method. Separation performance of the proposed method is improved against the case which frequency bins are uniformly selected, under the condition which computational costs are significantly reduced.

4.1 Motivation and strategy

In Chapter 3, computational costs are significantly reduced by the proposed frequency bin selection method, and the distortion of the separated signals is also restrained, for the BSS method using FDICA with the frequency bin selection method which the spatial correlation is used as the selection criterion. From the similar point of view, the spatial correlation of DHMAs can be utilized because the shape of DHMAs has significant property to evaluate the sound propagation, for example the permutation solution in [1, 2]. However, only a spatial amplitude distribution is used in the BSS method using DHMAs. Such magnitude square coherence (MSC) as the spatial correlation still remains to be utilized. The shape of DHMAs is too complex to analyze theoretically, therefore the amplitude characteristics of DHMAs has been evaluated experimentally to compare a spherical microphone array in the conventional method. The theoretical model of MSC in the microphone array research field [73] are valuable to consider acoustic characteristics. In this chapter, an experimental MSC of DHMAs is introduced, and comparison with the theoretical model of MSC shows the strategy that MSC can be utilized for the frequency bin selection method to reduce computational costs of the BSS method using DHMAs.

4.2 Proposed BSS method using DHMA

A block diagram of the proposed method is shown in Figure 4.1. Note that, as mentioned in Section 3.3.3, an iterative update rule can be replaced with any state-of-the-art ICA with the permutation solution. In other words, the frequency bin selection method is not constrained by the update rule of FDICA, so there is no loss of generality.

4.2.1 Magnitude squared coherence

During estimating the separation matrix of FDICA, it is very important that the iterative update of the separation matrix is performed on highly separable frequency bins when using bin selection method. In this chapter, MSC is considered as a method of selecting the separable frequency bins. MSC corresponds to a measure of the interference between two

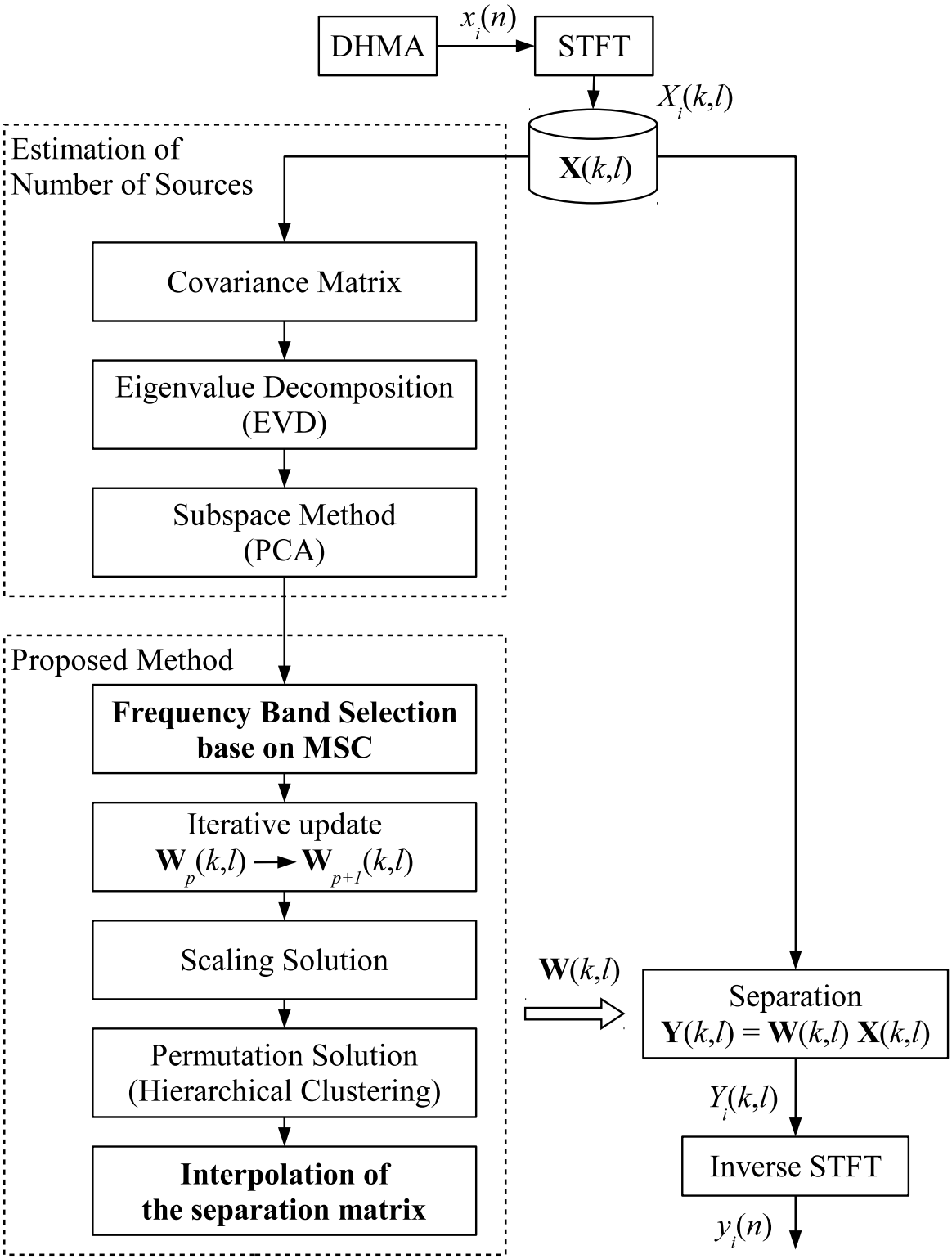

Figure 4.1: Block diagram of proposed method with DHMAs. Procedure is basically identical as conventional method proposed by Ogasawara. Proposed frequency bin selection method stays between subspace method and FDICA.

signals, and is formulated as follows:

$$C_{X_1X_2}(k) = \frac{E_l\left[|P_{X_1X_2}(k,l)|^2\right]}{E_l\left[|P_{X_1X_1}(k,l)|\right] E_l\left[|P_{X_2X_2}(k,l)|\right]}, \tag{4.1}$$

where $P_{X_1X_1}(k,l)$ and $P_{X_2X_2}(k,l)$ are the power spectra of signals X_1 and X_2, respectively, and $P_{X_1X_2}(k,l)$ is a cross spectrum. k and l denote frequency bin index and frame index respectively. $E_l[\cdot]$ is the expectation operator over frame l. $C_{X_1X_2}(k)$ is MSC between two signals X_1 and X_2 in the frequency bin k. The formulation of MSC is the normalized cross spectrum in each frequency bin, and thus the range of MSC shows $0 \leq C_{X_1X_2}(k) \leq 1$. In the case of a diffused noise field, MSC is formulated as follows:

$$C_{X_1X_2}(k) = \mathrm{sinc}\left(\frac{2\pi k F_s}{N_F} d_{mic} c^{-1}\right)^2, \tag{4.2}$$

where $\mathrm{sinc}(\cdot)$ means the sinc function $(\sin(x)/x)$, F_s is a sampling frequency, N_F is the FFT size, d_{mic} is a microphone distance and c is the velocity of sound, respectively. A theoretical formulation of the diffused noise field is used to evaluate characteristics of the noise field condition in [74], or to model the noise field for the post-filtering of the microphone array processing in [75]. BSS conducted under the condition of multiple sources that can be considered as a diffused noise field. A diffused noise field means that sound waves are randomly arriving from every possible direction, so that observed signals at two microphones have a variety of phase differences according to the directions of the sound sources. In the high frequency region, larger phase differences can be observed than in the low frequency region. The phase differences vary more widely in the high frequency region. Therefore, weaker coherence characteristics are observed in the high frequency region, which results in MSC assuming smaller values. Consequently, MSC can evaluate the phase difference between two signals, and MSC can contribute to increasing separation performance by selecting more number of the frequency bins with small MSC values.

Table 4.1: Simulation conditions for the BSS method using DHMAs

Sampling Frequency	40 kHz
Source Signal	Speech (6 Males, 6 Females), 4 seconds
Target Frequency Region	0–8 kHz
Number of Sources	12
Velocity of Sound	340 meter/second
Reverberation Time	138 milliseconds
Window Function	Hann
Window Length	1024 samples
Shift Length	256 samples
FFT Length (N_F)	1024 samples

4.2.2 Characteristics of MSC for DHMAs

In this section, the effectiveness of using MSC for DHMAs is experimentally evaluated. As mentioned in Section 2.1.2, the acoustic pressure distribution of DHMAs is different from that of spherical microphone arrays. In [1, 2], this comparison was made experimentally because the shape of a DHMA makes it too complicated to evaluate this characteristic theoretically. For the same reason, in the current study, the experimental evaluation is also used. Two microphones are arbitrarily selected from the 60 microphones of the DHMA (6 microphones are installed on each face), and the MSC of these two microphones are calculated using the measured impulse responses. The conditions used to evaluate the experimental MSC are shown in Table 4.1, in addition, the source signals are mixed in the same energy. The position of loudspeakers and the DHMA are shown in Figure 4.2.

Figure 4.3 shows the MSC between two microphones on the same face of the DHMA (microphone distance d_{mic} = 7 millimeter). A dashed line shows an experimental characteristic from the measured impulse response, and a solid line shows a theoretical characteristic which is formulated using the sinc function. Figure 4.4 shows MSC on different faces of the

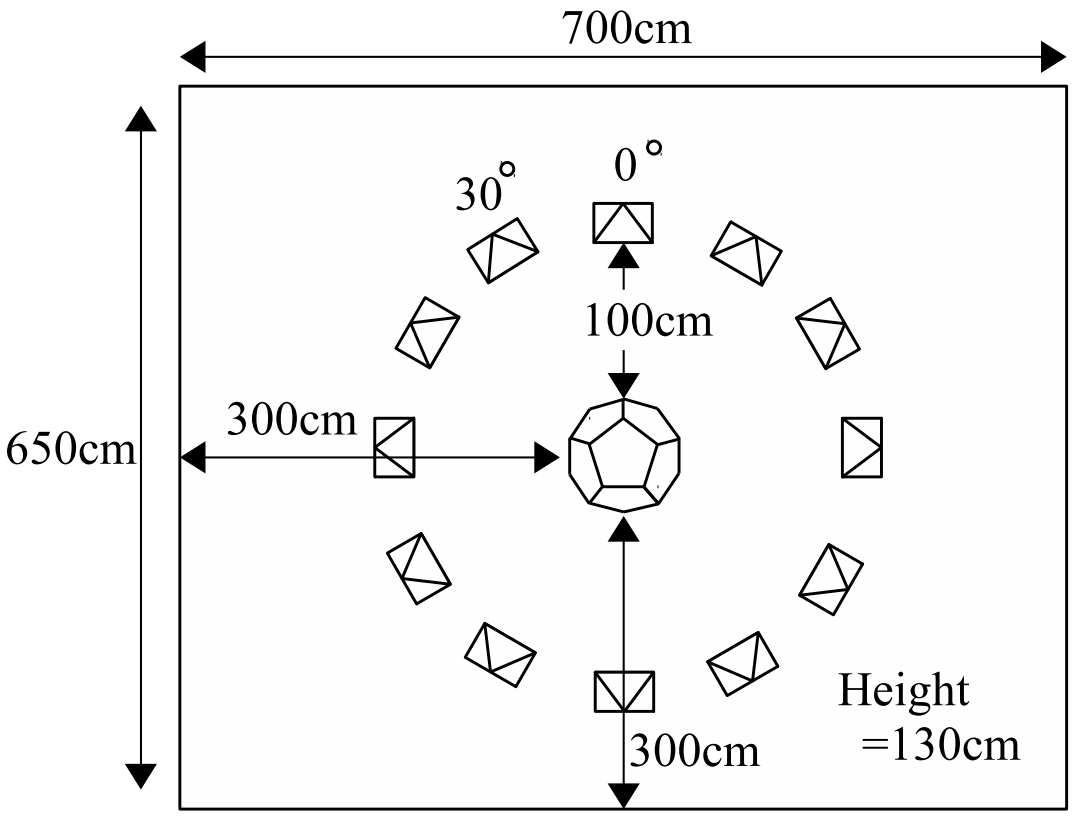

Figure 4.2: Source and loudspeaker positions for DHMA evaluations. Height of DHMA and loudspeakers is 130 cm. Reverberation time of room is 138 msec.

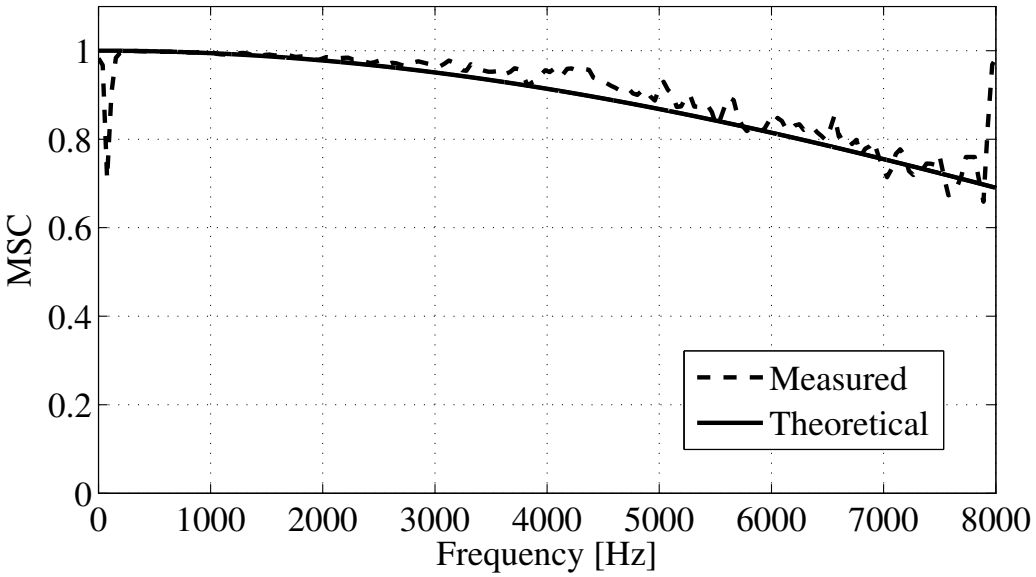

Figure 4.3: Example of MSC: Same face

DHMA (microphone distance d_{mic} = 42 millimeter). According to Figure 4.3, when both microphones are on the same face, the experimental MSC is equivalent to the theoretical MSC. On the other hand, when the microphones are on different faces, the experimental MSC is different from the theoretical value. The experimental MSC in Figure 4.4 in the low frequency region is smaller than the theoretical MSC, and this fact is important that the phase difference of the DHMA is larger than the theoretical characteristic. In other words, the DHMA shows the separable property even in the low frequency region. This is due to the shape of the DHMA. This fact leads the bin selection concept which is employed in the next section. In the high frequency region, the spatial aliasing resulting from the large distance between microphones. From the viewpoint of the source separation, microphones on the same face might be mainly used in the middle and high frequency region, in contrast microphones on the different faces might be mainly used in the low frequency region.

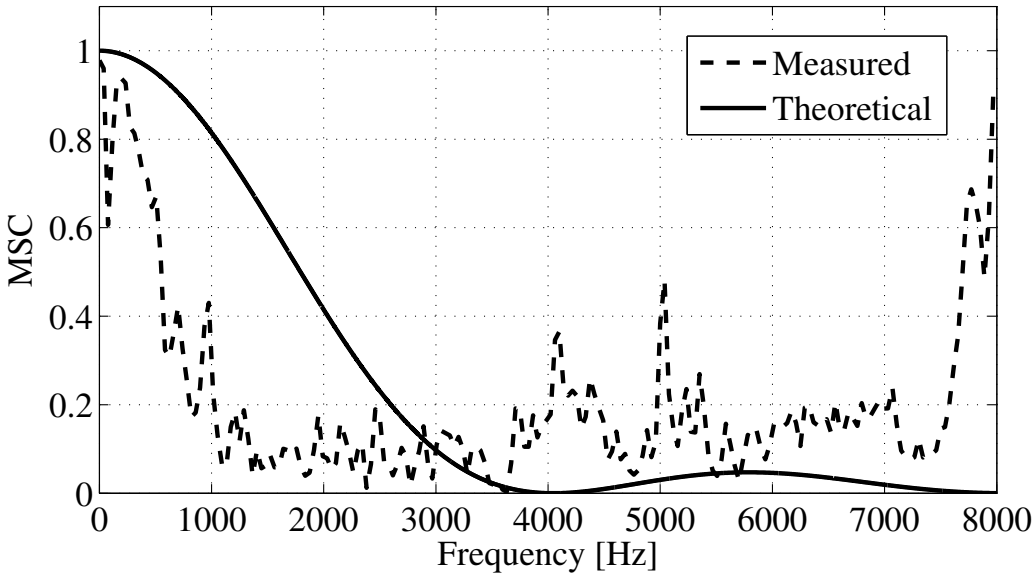

Figure 4.4: Example of MSC: Different faces

4.2.3 Frequency bin selection using averaged experimental MSC for reducing computational costs

Figure 4.5 shows the averaged experimental MSC (AEMSC) on the same face of the microphone, and on the different faces, respectively. In this research, the frequency bin selection is based on the AEMSC. Small MSC values correspond to large phase differences, and thus selection should occur mainly in regions with small MSC values. In order to prevent a bias in which bins are selected, we consider three frequency regions based on MSC, which leads to the selection of a large number of bins with small MSC values. Figure 4.6 shows frequency regions B_1, B_2, B_3. Boundary frequencies correspond to a mean of the AEMSC shown in Figure 4.6. f_b is determined by the cross point between the AEMSC for the same face and its mean, and f_a is determined in the same manner as the AEMSC of the different face. The values of f_a and f_b in this research are 1016 Hz and 5040 Hz, respectively. As mentioned in Section 4.2.1, the small MSC contributes to select the separable frequency bins. The larger number of the frequency bins should be selected in the small MSC region than the large MSC region. A mean indicates a threshold on which MSC values are small

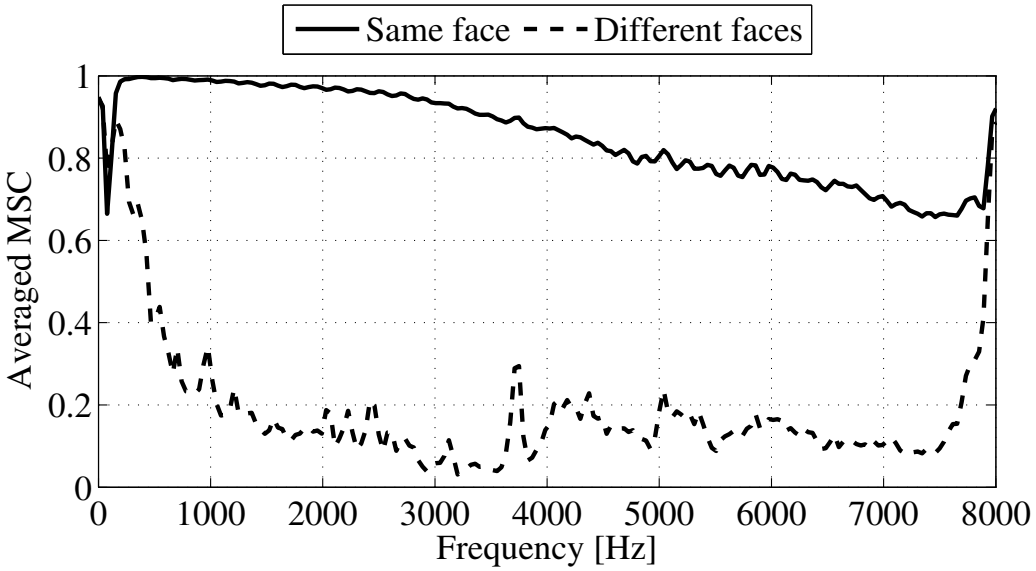

Figure 4.5: Averaged experimental MSC (AEMSC). MSCs observed on same and different faces are averaged. Solid line means AEMSC on same faces, and dashed line means AEMSC on different faces.

or large. In other words, the mean of the AEMSC divide the frequency region into the smaller and larger MSC regions. When the frequency becomes high, the trend of the both AEMSCs becomes small. This fact is appeared in Figure 4.5. Therefore, the number of the selected frequency bins in the high frequency region should be larger than in the low frequency region. In the highest frequency region, the microphones in a same face might be chosen by the BSS method using DHMA, so that f_b should be as low as possible to give that the high frequency region becomes broad, and also be determined by the AEMSC for the same face. Therefore, f_b is determined as the cross point between the AEMSC for the same face and its mean. In the middle frequency region, the AEMSC for the different faces is lower than its mean which is shown as the thick dashed line in Figure 4.6. However, in the low frequency region, the AEMSC for the different faces is still high, thus f_a should be a low frequency value to give that the middle frequency region becomes broad. Therefore, f_a is determined by the cross point between the AEMSC for the different faces and its mean, in addition to the lowest cross point. Consequently, the mean of the AEMSC provide the

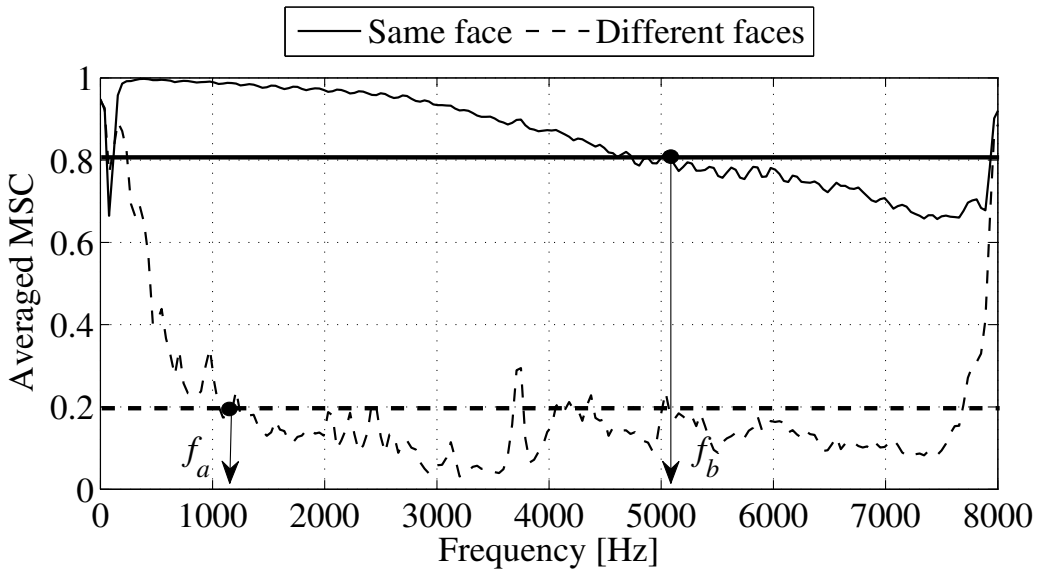

Figure 4.6: Region of bin selection. f_a is determined by a cross point between AEMSC on the different faces (thin dashed line) and its mean (thick dashed line). f_b is determined by a cross point between AEMSC on the same faces (thin solid line) and its mean (thick solid line). Values of f_a and f_b are 1016 Hz and 5040 Hz, respectively.

appropriate thresholds for dividing the separable frequency regions.

In Section 4.3, the number of bins in each frequency region is used as a parameter to evaluate the proposed method. As mentioned in Section 4.2.1, in the high frequency region, weaker coherence characteristics are observed; this means that the higher frequency region is more separable than the lower frequency region. Therefore, the number of bins in the higher frequency region should be larger than the lower frequency region. For example, the bins $1/5$, $1/3$ and $1/2$ are selected in regions B_1, B_2 and B_3, respectively. The total number of frequency bins is 77, reduced from 200 bins, which correspond to the target frequency region 0–8 kHz, with a size of FFT 1024.

4.2.4 Subspace method for reducing number of observed signals under overdetermined condition

The signal model in this chapter is same as in Section 2.1. The source signals are transferred from their locations to the DHMA, and a mixing matrix is described in frequency domain $\mathbf{A}(k)$ which is an $N_S \times N_M$ matrix where N_S and N_M are the number of sources and microphones, respectively. The observed signal is represented by $\mathbf{X}(k,l) = \mathbf{A}(k)\mathbf{S}(k,l)$, which is same as in Eq. (2.1) for frequency bin index k, and frame index l. $\mathbf{X}(k,l)$ and $\mathbf{S}(k,l)$ represent the observed and source signal vectors, respectively.

The BSS method using a DHMA is conducted under an overdetermined condition which means $N_M > N_S$, because a DHMA can include up to 160 microphones. FDICA assumes that the number of source signals is equal to the number of the microphones, and thus an estimation of the number of source signals and a reduction in the number of observed signals are needed to perform FDICA. In addition, the reduced dimensions must exceed the number of actual sound sources, because not only actual sound source signals but also reflection waves are used in FDICA. The spatial covariance matrix is calculated by an expectation of the observed signal $\mathbf{X}(k,l)$, and it is decomposed into eigenvalues. The number of virtual sound sources N_Q that include direct sound sources and early-reflected sources is estimated from eigenvalue diagonal matrix $\Lambda(k)$ as follows:

$$\Lambda(k) = \mathrm{diag}\left[\lambda_1, \cdots, \lambda_{N_M}\right], \tag{4.3}$$

where diag denotes the operator at which all of matrix elements except diagonal elements are set to zero, and λ_i is the i-th eigenvalue. This corresponds to the estimation of the number of sound sources with directivity, because such sound sources are spatially correlated. Normalization, whose summation of the eigenvalues is 1, is calculated at each frequency as follows:

$$\lambda_m \leftarrow \frac{\lambda_m}{\sum_{i=1}^{N_M} \lambda_i}, \quad m = 1, \cdots, N_M. \tag{4.4}$$

The threshold for the normalized eigenvalues evaluates the number of virtual sound

sources in each frequency bin, and the maximum estimated value in all frequencies is assumed to be N_Q, because the number of estimated sound sources is different in each frequency bin. Following the estimation of the number of virtual sources, eigenvectors, which are estimated with eigenvalues, are employed to reduce the number of observed signals using the subspace method. A more detailed description of the process explained above is given in [1, 2].

4.2.5 FDICA for subspace signals in selected frequency bins

The observed signal is reduced to the N_Q dimension using the subspace method, following estimation of the number of virtual sound sources N_Q. FDICA estimates separation matrix $\mathbf{U}(k)$ for the reduced number of the observed signals, but only in the selected bins, via an update rule with iteration [76] based on the principle presented in [19]. Separation matrix $\mathbf{W}(k)$ for actual separation is obtained through a combination of a subspace matrix and separation matrix $\mathbf{U}(k)$. The separated signal $\mathbf{Y}(k, l)$ is obtained as follows:

$$\mathbf{Y}(k, l) = \mathbf{W}(k)\mathbf{X}(k, l). \tag{4.5}$$

The scaling problem is solved using the projection method [37]. Next, the number of dominant source signals in each frequency bin $N_{K(k)}$ is calculated from the separated signals via a threshold operation.

4.2.6 Permutation solution using characteristics of DHMAs

Solution of the permutation problem affects separation performance significantly, and Ogasawara has proposed a method which combines the acoustic pressure distribution and the relative phase distance [1, 2]. For solving the permutation problem, one of the representative methods is the correlation method. Not only the direct sound source but also the early-reflected sources, however, should be considered as the source signals, for the BSS method using a DHMA. This implies that similar time differences from different locations are included in the separated signals. These signals might be very similar in regard to the power envelope, so that it is difficult to distinguish them by the correlation in between different

and close frequency bins, for solving the permutation based on the power envelopes of the separated signals Another representative permutation solution is the clustering method. For the BSS method using a DHMA, the clustering method, therefore, might be more appropriate. The method proposed in [1, 2] significantly improves ability to solve the permutation problem for a DHMA, however it uses transfer function clustering, which leads to a high level of computational costs. In this section, the permutation solution for the BSS method using DHMAs is briefly explained, and estimate of computational costs is introduced when the frequency bin selection method is applied.

Acoustic transfer function clustering

The Moore-Penrose pseudo-inverse of the separation matrix corresponds to the mixing matrix; in other words, it corresponds to the transfer functions between the source signals and the observed signals. The transfer function is estimated as the q-th column vector $\mathbf{w}_q^+(k)$ of pseudo-inverse $\mathbf{W}^+(k)$.

Two similarities are considered, acoustic pressure distribution $p(\mathbf{w}_q^+(k))$ for amplitude similarity D_a, and relative phase distance $\phi(\mathbf{w}_q^+(k))$ for phase similarity D_p. As mentioned in Section 2.1.2, a DHMA has an acoustic pressure difference between each face, and the acoustic pressure distribution shows the characteristics of the directions of the sources. The acoustic pressure distribution is formulated as follows:

$$p(\mathbf{w}_q^+(k)) = \left[\frac{1}{|M(1)|} \sum_{m \in M(1)} \left| w_{q,m}^+(k) \right|, \cdots, \frac{1}{|M(10)|} \sum_{m \in M(10)} \left| w_{q,m}^+(k) \right| \right], \tag{4.6}$$

$$p\left(\mathbf{w}_q^+(k)\right) \leftarrow \frac{p\left(\mathbf{w}_q^+(k)\right)}{\sum_q p\left(\mathbf{w}_q^+(k)\right)}, \tag{4.7}$$

where $M(\mu)$ represents the set of microphones on the μ-th face and $w_{q,m}^+(k)$ is the transfer function between source q and the m-th microphone on face μ. Amplitude similarity D_a is formulated using acoustic pressure distribution $p(\mathbf{w}_q^+(k))$ with the ν-th centroid $\mathbf{c}_\nu$ as follows:

$$D_a\left(\mathbf{w}_q^+(k), \mathbf{c}_\nu\right) = \left\| p(\mathbf{w}_q^+(k)) - p(\mathbf{c}_\nu) \right\|^2. \tag{4.8}$$

Phase similarity is calculated using the normalized phase difference $\phi(\mathbf{w}_q^+(k))$ between two microphones, calculated from normalized time difference $\tau_{q,m}(k)$ as follows:

$$\phi(\mathbf{w}_q^+(k)) = \left[\exp\left(j\tau_{q,1}(k)\right), \cdots, \exp\left(j\tau_{q,N_M}(k)\right)\right], \tag{4.9}$$

$$\tau_{q,m} = \gamma \frac{\angle w_{q,m}^+(k)}{F_s k / N_F}, \tag{4.10}$$

where γ is a normalization constant and $\angle$ means the operator which obtain an argument of a complex number. Consequently, the phase similarity D_p is formulated as follows:

$$D_p\left(\mathbf{w}_q^+(k), \mathbf{c}_\nu\right) = \sum_{l=1}^{10} \left| \sum_{m \in M(\mu)} \phi\left(w_{q,m}^+(k)\right)^* \phi\left(c_{\nu,m}\right) \right| \tag{4.11}$$

where $(\cdot)^*$ represents the complex conjugate and $c_{\nu,m}$ represents the ν-th centroid of the m-th microphone. After the normalization of D_a and D_p by their mean and variance respectively, the combined similarity $\mathcal{J}(\cdot)$ in each frequency bin for hierarchical clustering is calculated as follows:

$$\mathcal{J}\left(\mathbf{w}_{q1}^+(k_\alpha), \mathbf{w}_{q2}^+(k_\beta)\right) = \left\{a(k_\alpha) + a(k_\beta)\right\} D_a + \left\{b(k_\alpha) + b(k_\beta)\right\} D_p, \tag{4.12}$$

$$a(k) = \left\{\frac{k/I}{N_F/2}\right\}^\rho, \; b(k) = 1 - a(k), \tag{4.13}$$

where I is a parameter to adjust the phase similarity weighting, and ρ is a parameter to adjust the boundary frequency between the amplitude and phase similarities.

The similarity described in Eq. (4.12) is calculated with hierarchical clustering as the permutation correction of the BSS method using DHMAs.

4.2.7 Interpolated separation matrices in unselected frequency bins

As mentioned previously, high computational costs are one of the drawbacks to performing hierarchical clustering for the BSS method using DHMAs. The frequency bin selection contributes to reduce computational costs, not only during estimation of the separation matrix, but also during hierarchical clustering for the permutation solution. After permutation

correction, the separation matrix in the unselected bins is obtained as the linearly interpolated matrix from the separation matrix in the neighboring frequency bins, which has already been estimated. If the mixing matrices would be obtained and could be considered as linear phase FIR filter, the frequency phase response should be linear. A spatial characteristic of a mixing filter is described as a time difference combination, and the time difference is equivalent to the phase difference in the frequency domain. Therefore, it can be said that an interpolation of linear phase mixing matrices, which could be estimated from the inverse matrix of the separation matrix, should be appropriate. However, the inverse matrix, which consumes order $O(n^3)$ complexity (n means a n-by-n square matrix) [69], leads to additional computational costs. The aim of this research is reducing computational costs, so that avoiding additional computational costs is preferable. Therefore, in the unselected bins, the interpolation of the separation matrix in the neighboring frequency bins are used as the separation matrix.

4.3 Evaluation

4.3.1 Estimate of computational costs in the case of DHMAs

Evaluation measure for computational costs

In Chapter 3, the number of floating operations of multiplication and addition is counted precisely for each function to evaluate computational costs. From a practical viewpoint, this criterion is valuable for estimating the possibility of implementation with embedded processors, and in addition it is useful to estimate system requirements for manufacturers. On the other hand, in particular when there are a large number of microphones, it is difficult to estimate computational costs precisely using the same method because of multiple numerical calculations. Therefore, in this chapter, only the number of multiplication is evaluated. This is also valuable for estimation of computational cost because the multiplication operator in the embedded processors is one of the most expensive units and can represents the processor performance.

Table 4.2: Computational costs for the BSS method using DHMAs

Method	Complexity
STFT(Forward&Inverse)	$2N_M N_L N_F \log_2 N_F$
Covariance Matrix	$N_M^2 N_L N_B$
Eigenvalue Decomposition	$(13/3)\, N_M^3 N_B$
Subspace Method	$N_Q N_M N_L N_B$
Separation Matrix	$N_Q^3 N_L N_I N_B$
Projection Method	$(13/3)\, N_M^3 N_B$
Hierarchical Clustering	$(N_M N_{K(k)} N_B)^2$

Estimate of the order of computational complexity

In general, eigenvalue decomposition (EVD) is solved by the Householder method (HHM) and the implicit shifted QL method (ISQL) [69]. In [69], the numerical calculation method for singular value decomposition (SVD) is also introduced and consists of HHM and ISQL. Therefore, in our estimate, the computational complexity of the EVD and pseudo-inverse by SVD can be estimated by HHM and ISQL. Computational costs of HHM and ISQL are introduced as $(4/3)n^3$ and $3n^3$ respectively in [69]. For hierarchical clustering, an efficient algorithm is introduced in [50] and computational cost is n^2.

Estimated computational costs are shown in Table 4.2 where:

- N_M: Number of microphones,
- N_L: Number of frames,
- N_F: FFT size,
- N_B: Number of bins selected, in the case of all bins selected, this number correspond to the Nyquist frequency of speech signals,
- N_Q: Number of subspaces,

- $N_{K(k)}$: Number of separated source signals in each frequency bin,
- N_I: Number of iterations.

An example estimate is shown in Table 4.3 which is calculated from the computational costs in Table 4.2 using concrete numbers. N_M is 60 as described in Section 1.4; six microphones are installed on each face, and the DHMA has ten faces for microphone arrays. In our experiments described in Section 4.3, some numbers are given in Table 4.1

- N_L is 625, which corresponds to the length of the source signals as 4 sec,
- N_F is 1024,
- N_B is
  - 200 in the case of the full bin for the DHMA (0–8 kHz),
  - 80 in the case of the proposed method (for example estimate),
- N_Q is 25 (determined by preliminary experiment),
- $N_{K(k)}$ is 15 (determined by preliminary experiment),
- N_I is 200.

Note that, to simplify estimate, $N_{K(k)}$ is not varied in every frequency bin. Actual iteration is terminated by a convergence test, however the number of the iteration is constant to simplify the estimate. As shown in Table 4.3, the total estimated computational cost has the same order of the computational cost as hierarchical clustering. The proposed method results in an 84% reduction in computational costs as a result of reducing number of frequency bins from 200 to 80; however, the number of frequency bins has only been reduced 60%. Agglomerative hierarchical clustering is based on a bottom-up algorithm, and this is the reason for the large reduction in computational costs. Hierarchical clustering needs to calculate similarities between all of the transfer functions, and thus computational costs depend on the number of initial elements. The reduced computational cost of the hierarchical clustering process results in a power of two reduction in computational costs compared to reduction of the number of frequency bins.

As mentioned in Section 2.3, the influence of the slow access of the external memory is considered as three times of the computational costs estimated. The architecture of the high-performance DSPs consists of several cores such as the latest CPU. One of the high-performance DSPs shows 160 GFLOPS with 8 cores as mentioned in Section 2.3. As shown in Table 4.3, the amount of calculation of the proposed method is around 80 giga-operations. When the influence of the slow access of the external memory is applied to this estimate, the amount of calculation becomes around 240 giga-operations. In contrast, the amount of calculation of the previously proposed BSS method using DHMA shows over 500 giga-operations; this corresponds to 1500 giga-operations with the external memory influence. If some number of the high-performance DSPs are connected and work concurrently, the BSS method using DHMA can be implemented; however, about ten DSPs are needed to perform. This fact leads to the big size of the equipments. The proposed method reduces the size of equipment and lowers manufacturing costs.

Table 4.3: Estimated computational costs of the BSS method using DHMAs

Method	Computational Costs (Previous)	Computational Costs (Proposed)	Ratio (Reduction [%])	Computational Costs (TRINICON)	Ratio (Reduction [%])
STFT(Forward&Inverse)	7.68E8	7.68E8	1.0 (0[%])	3.69E9	4.8
Covariance Matrix	1.80E9	7.20E8	0.4 (60[%])	4.61E10	25.6
Eigenvalue Decomposition	7.49E8	3.00E8	0.4 (60[%])	7.67E9	10.3
Subspace Method	7.50E8	3.00E8	0.4 (60[%])	4.01E10	53.5
Separation Matrix	2.50E9	1.00E9	0.4 (60[%])	3.13E11	156.6
Projection Method	7.49E8	3.00E8	0.4 (60[%])	7.67E9	10.3
Hierarchical Clustering	5.18E11	8.29E10	0.16 (84[%])	-	-
TOTAL	5.26E11	8.63E10	0.16 (84[%])	4.19E11	0.80 (20[%])

Comparison to ICA via second-order statistics

TRINICON [63, 64] is one of the most common BSS methods using ICA. It uses joint diagonalization using second-order statistics (SOS), in addition it is known as its computational efficiency. As mentioned in Section 2.2.4, when inverse and forward DFTs are calculated for the separation matrices every several iterations, then these processes prevent the complete decoupling of the frequency bins usually caused by the bin-wise independence assumption, in other words, the permutation solution is not required. Involving DFTs in the update equation, Eq. (53) in [64], is a general way. However, using DFTs as TRINICON is computationally disadvantageous. Even though the speech signal is limited up to 8–kHz, using all of the frequency bins is necessary to estimate the separation matrix. In contrast, the proposed method allows circular convolution for restraining computational costs. Eq. (67) in [64] is the simplest update rule of TRINICON with some approximations. Under the same configuration for the estimate of computational costs in Section 4.3.1, additional parameters for TRINICON must be considered that FFT size is four times N_F, the number of bins for the separation matrix is $4N_F/2+1$, number of blocks for the joint diagonalization is $N_J = 20$, and the number of iterations $N_{I'} = 40$. The computational cost of the update rule Eq. (67) in [64] is $\left\{3N_Q^3(4N_F/2 + 1)N_J\right\} N_{I'}$, and it is dominant computational cost of TRINICON as shown in Table 4.3. No permutation solution certainly reduces total computational costs, however considering the separation filter in the time domain increases the computational costs of the iterative update.

When bin-wise independence is assumed during using TRINICON, a permutation solution is necessary [63]. In addition, as mentioned in [64], the approximation applied to the update equation disturbs the perfect permutation correction. In other words, this means that the separation performance is deteriorated by these approximations, so that involving DFTs in the update rule is the essential characteristic of TRINICON, from the point of view that the permutation solution is not required. The proposed method is compared with TRINICON involving DFTs in the update rule. The proposed method is also compared with the previously proposed BSS method using DHMA because it is important to evaluate feasibility of the proposed method, which might represent a balance between separation performance and computational costs.

4.3.2 Source separation simulation

Simulation conditions and evaluation measures

The experimental conditions are the same as in Figure 4.2 and Table 4.1. Separation performance is evaluated by improvement in the signal-to-interference ratio (SIR_{imp}).

$$\mathrm{SIR}_{imp}^{(\xi)} = \mathrm{SIR}_{out}^{(\xi)} - \mathrm{SIR}_{in}^{(\xi)}, \tag{4.14}$$

$$\mathrm{SIR}_{in}^{(\xi)} = 10\log_{10}\left[\frac{\sum_t \left\{x_{J\xi}(t)\right\}^2}{\sum_t \sum_{(J'\neq\xi)} \left\{x_{JJ'}(t)\right\}^2}\right], \tag{4.15}$$

$$\mathrm{SIR}_{out}^{(\xi)} = 10\log_{10}\left[\frac{\sum_t \left\{y_{\xi\xi}(t)\right\}^2}{\sum_t \sum_{(J'\neq\xi)} \left\{y_{\xi J'}(t)\right\}^2}\right], \tag{4.16}$$

where $x_{J\xi}(t)$ represents the observed source signal ξ on the J-th microphone and $y_{\xi\xi}(t)$ represents the output signal which corresponds to source signal ξ. $\mathrm{SIR}_{imp}^{(\xi)}$ is averaged over all of the source signals. The proposed frequency bin selection method might cause a deterioration in separation performance as a result of the limited number of frequency bins. Therefore, the other important factor is the quality of the separated sound, and signal distortion should be evaluated. As mentioned in Chapter 3, segmental signal-to-noise ratio ($\mathrm{SNR}_{\mathrm{seg}}$) is a very common measure for evaluating noise suppression, and $\mathrm{SNR}_{\mathrm{seg}}$ is known to have a better correlation with the perception of noisy speech by humans than entire interval SNR [71]. Cepstral distortion (CD) [72] is another measure of the degree of distortion via the cepstrum domain, and this can evaluate distortion of a spectral envelope. $\mathrm{SNR}_{\mathrm{seg}}{}^{(\xi)}$ is formulated as follows:

$$\mathrm{SNR}_{\mathrm{seg}}{}^{(\xi)} = \frac{1}{N_{l_s}} \sum_{l_s} 10\log_{10} \frac{\sum_t \left\{x_{J\xi}(t,l_s)\right\}^2}{\sum_t \left\{x_{J\xi}(t,l_s) - y_{\xi\xi}(t,l_s)\right\}^2}, \tag{4.17}$$

where l_s is a frame number and N_{l_s} is the number of frames used to evaluate $\mathrm{SNR}_{\mathrm{seg}}$. CD is calculated from speech components, and it is defined as follows:

$$\mathrm{CD}^{(\xi)} = \frac{20}{N_{l_c} \ln 10} \sum_{t} \sqrt{\sum_{\kappa=1}^{L} 2 \left\{ C_{x_{J\xi}(t,l_c)}(\kappa, l_c) - C_{y_{\xi\xi}(t,l_c)}(\kappa, l_c) \right\}^2}, \tag{4.18}$$

where l_c is a frame number, κ is the index of the cepstrum coefficient and N_{l_c} is the number of frames for CD. $C_{(\cdot)}(\cdot)$ is the cepstrum coefficient and L is the number of dimensions of the cepstrum used in the evaluation; we set L=20. $\mathrm{SNR}_{\mathrm{seg}}{}^{(\xi)}$ and $\mathrm{CD}^{(\xi)}$ are also averaged over all of the source signals.

Simulation results

Results for each objective measure are shown in Table 4.4. The first column shows the ratio of selected bins in each frequency region, for example, '(1/5,1/3,1/2)' means one-fifth for lowest frequency region B_1, one-third for middle frequency region B_2 and one-half for highest frequency region B_3. Each region is divided into 1016 Hz and 5040 Hz, respectively. The number of selected bins means the total number of selected frequency bins. Computational costs in this table equal the ratio compared to using the full bin, and the method of estimate of computational costs is described in Section 4.3.1. Figure 4.7–4.9 shows the performance of the proposed method. The x-axis of each figure represents the ratio of computational costs, and the y-axis represents the objective measure.

4.4 Discussion

The configuration '(1,1,1)' in Table 4.4 corresponds to the previously proposed BSS method using DHMA, and these results are shown at ratio 1.0 (10^0) point on the x-axis of Figure 4.7–4.9. SIR_{imp} shows the contribution to separation performance of a large number of selected bins. This result is assumed before the experiment. SIR_{imp} deteriorates as the number of selected bins decreases, and $\mathrm{SNR}_{\mathrm{seg}}$ and CD show different characteristics of this deterioration. $\mathrm{SNR}_{\mathrm{seg}}$ shows that only a small deterioration occurs under 1 dB; in other words, $\mathrm{SNR}_{\mathrm{seg}}$ shows almost equivalent performance using limited number of frequency

Table 4.4: Experimental results for the proposed BSS method using DHMAs

(B_1,B_2,B_3)	Number of selected bins	Computational Costs	SIR_{imp} [dB]	SNR_{seg} [dB]	CD [dB]
(1,1,1) (previous method)	200	1.0	24.4	7.85	2.65
(1/3,1/2,1)	134	0.45	22.3	7.80	2.51
(1/3,1/2,1/2)	97	0.24	21.8	7.74	2.48
(1/5,1/3,1/2)	77	0.15	20.6	7.52	2.30
(1/5,1/3,1/3)	64	0.1	20.8	7.45	2.30
uniformly spaced	64	0.1	19.9	6.18	2.58

bins. Even though SIR_{imp} and SNR_{seg} deteriorate, CD is still slightly improved because the lower value of CD shows the better performance. The proposed method is focused on a *non-uniformly* spaced selection of frequency bins. In addition, the aim of this non-uniformly spaced selection is to utilize the characteristics of the dodecahedral shape of the microphone array. When 64 bins were selected, SIR_{imp} of the proposed method shows almost 1 dB higher than in the case of uniformly spaced bin selection. In particular, SNR_{seg} shows significant improvement. CD is slightly improved to uniformly spaced bin selection. The magnitude squared coherence, theoretically and experimentally described in Section 4.2.2 and 4.2.3, reflects this characteristic. The experimental results indicate that lower values of MSC contribute to separation performance, particularly in the high frequency region that the number of selected bins is larger than the others. Even though computational cost is reduced by around 90 percent as compared to using the full bin, the acoustic characteristics of the proposed method contribute to improving not only the separation performance, but also the quality of the separated sound.

The proposed frequency bin selection method for the frequency domain BSS method

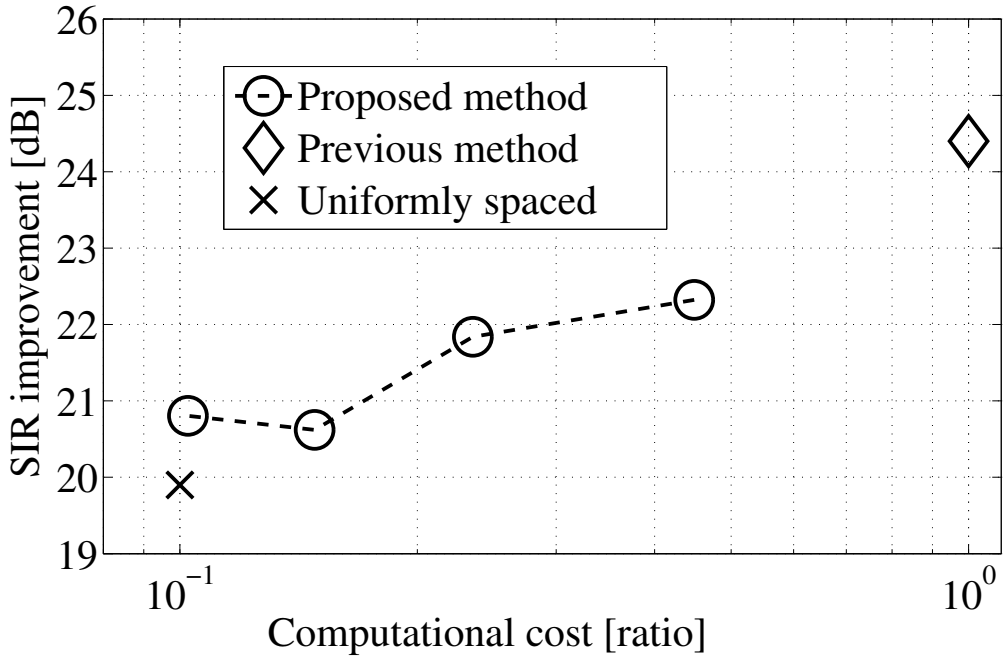

Figure 4.7: SIR improvement. 'Diamond' means previously proposed method that uses all frequency bins, and this case corresponds to no computational cost reduction, shown as a ratio that equals one (10^0). 'Circle' means proposed method. 'Cross' means uniformly spaced selection case for comparison with proposed method that uses a non-uniformly spaced selection.

is advantageous to achieve the trade-off between the separation performance with the significantly low degradation of the sound quality and computational costs. However, the deterioration of the separation performance shows a disadvantage of the proposed method compared to the method which uses all the frequency bins. On the other hand, such the joint diagonalization using SOS as TRINICON involves DFTs in its update rule. This results in no permutation solution, so that TRINICON shows less computational costs, around 20 percent reduction, than the BSS method using the permutation solution. When the number of microphones is small or the voice terminals have a high computation power, it is easy to perform TRINICON. However, the constraint of the linear convolution does not allow to reduce further computational costs. The voice terminals are generally implemented in

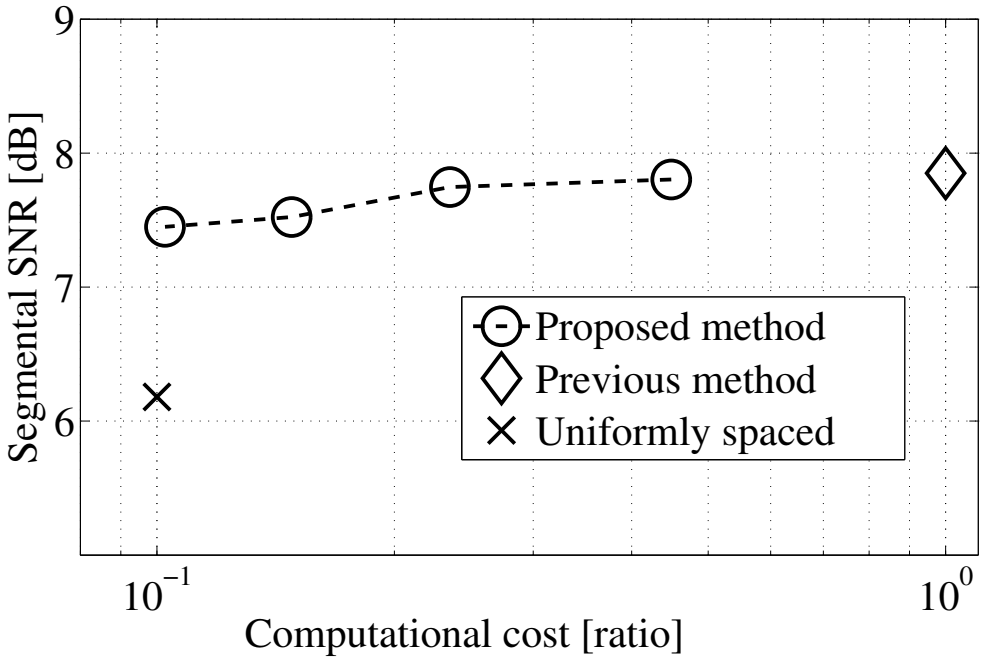

Figure 4.8: Segmental SNR. 'Diamond', 'circle' and 'cross' mean previously proposed method, proposed method, and uniformly spaced selection case.

the embedded systems as mentioned in Section 1, the lower computational costs are required to perform the BSS method. The proposed bin selection method can reduce further computational costs, and it is easy to achieve over 50 percent reduction.

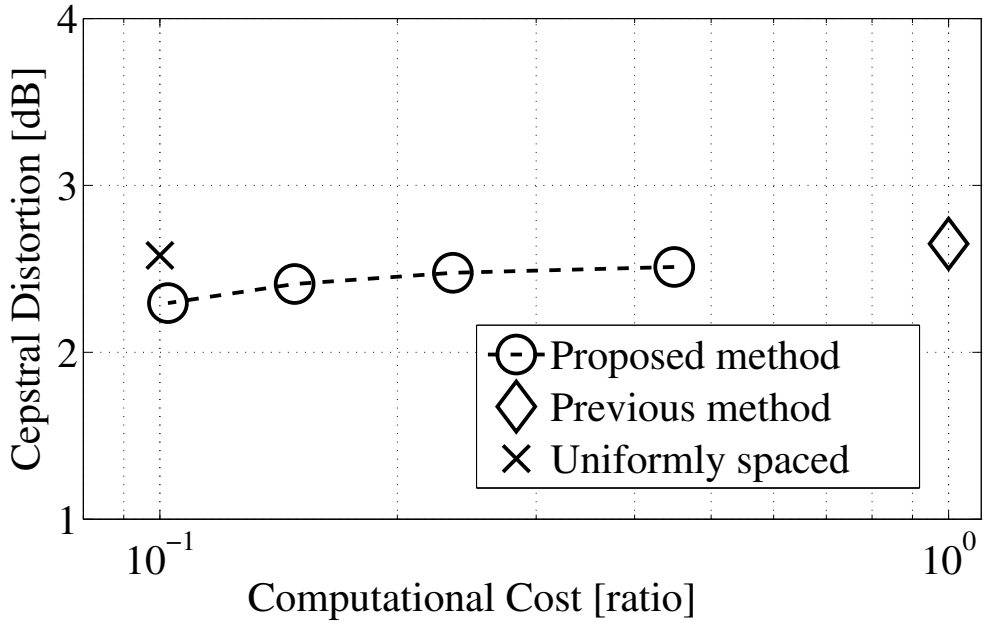

Figure 4.9: Cepstral Distortion. 'Diamond', 'circle' and 'cross' mean previously proposed method, proposed method, and uniformly spaced selection case.

Chapter 5

Overall discussion

First, this chapter discusses the proposed method's use of spatial correlation between two microphones to select frequency bins for source separation. Second, important implementation issues are discussed. Finally, remaining issues and possible future work are discussed.

5.1 Spatial correlation between two microphones

The proposed method uses spatial correlation to select promising frequency bins for source separation. When using two microphones, the spatial covariance matrix is equivalent to spatial correlation, and its determinant is used as the criterion to select the frequency bins. The determinant can simultaneously evaluate the number of source signals and the relative strength among frequencies. When using a DHMA, magnitude squared coherence (MSC) is used as the criterion to designate the sets of the frequency bins. The proposed method uses the arbitrary combination of the impulse responses at two microphones to obtain the averaged MSC. Since MSC can evaluate phase differences between microphones, it can be used as a criterion for the separable frequency region. Of course if there are three or more microphones, spatial information can be calculated, so the spatial covariance matrix can also be calculated for three or more microphones. For frequency bin selection, the real valued and scalar criteria are preferable especially for the computational inexpensiveness. Phase difference in the frequency domain is described as a complex number, thus the spatial information includes a complex number. The proposed bin selection criteria, the determinant of the covariance matrix and the MSC, are real numbers. Both criteria are scalar values in a frequency bin. The proposed method uses only two microphones to calculate these criteria. System requirements define the specifications of speech processing equipment and the number of microphones used, and these requirements differ widely because such equipment is designed and manufactured for different applications. Therefore, equipment specifications and the number of microphones used may vary widely. Experimental results using the proposed method in this dissertation show that only two microphones are sufficient to evaluate spatial information, even if the number of microphones available is much larger. This calculation is much simpler than if all of the spatial information available is used.

From another technical viewpoint, the off-diagonal elements of the spatial covariance matrix (SCM) simultaneously include the directions and signal power of the source signals. In contrast, the SCM's diagonal elements do not include this spatial information. These diagonal elements correspond to the power spectrum of the observed signals, namely the

Table 5.1: Comparison of the number of the operations for the target speech equipment

Assumed system (DSP)	Mobile (Low-power)	Portable (High-performance)
Target	200 mega-operations	160 giga-operations
Proposed method	144 mega-operations	240 giga-operations

mixed signals. The off-diagonal elements of the covariance matrix express the spatial cross-correlation spectrum between microphones. From a statistical point of view, the spatial correlation is categorized into second-order statistics. In this dissertation, only second-order statistics are considered in order to reduce the number of the frequency bins. In other words, this is another opportunity to restrain the computational cost of FDICA. In addition, the physical characteristics of the microphone array can also be utilized in combination with second-order statistics, such as the experimental MSC.

5.2 Important issues for FDICA implementation

5.2.1 Target and estimated computational costs

As discussed in Chapters 1 and 2, when implementing FDICA within embedded processors, one of the most important issues is the required wait states of DRAM external memory. This is because storing large amounts of observed signals is necessary for statistical signal processing, especially when using optimization schemes via higher-order statistics, such as ICA. Embedded processors have some internal high-speed memory, however the amount of memory space is very limited, so this internal memory should be reserved for computationally expensive algorithms, such as FFT or matrix calculation. This implies that DRAM external memory must be used to store long-term observed signals, however the required wait states of such external memory lead to DSP waiting cycles, which waste electric power. This is a critical issue for small speech processing devices, such as the portable speech equipment and mobile phones discussed in this dissertation.

Target numbers for DSP operations are discussed in Section 2.3, and are listed in Table 5.1. The estimated number of operations required when using the proposed method are also listed in Table 5.1. The effect of DSP waiting cycles is estimated to be equivalent to tripling the number of required operations, thus the numbers in Table 5.1 have already been increased to reflect the effect of using DRAM with wait states.

In the case of mobile devices, which are assumed to have two microphones, the estimated number of required operations is 144 mega-operations. Comparing the target and estimated numbers of operations, it seems feasible to implement the proposed method using a low-power DSP. In the case of portable equipment with a DHMA, the estimated number of operations is 240 giga-operations, which exceeds the target number. Because the proposed method is a more efficient form of BSS than conventional BSS with FDICA, the signal processing algorithm has been formulated for the highest level of program optimization, in other words the design level optimization. The program optimization includes elements such as overall design, source code, build, compile, and so on. If the level of optimization of these elements is lowered, source code, build and compile, this should significantly reduce the number of operations required when using the proposed method. Generally, the number of operations can easily be reduced by 50 percent in these lower level optimizations, but achieving a 90 percent reduction would be challenging. Reducing the estimated number of operations by 50 percent results in a new estimate of 120 giga-operations, which would make it feasible to implement the proposed method using a single, high-performance DSP. Note that the BSS method with a DHMA [1,2] requires 1500 giga-operations, in part due to the required wait states of DRAMs, and that this is more than ten times our target number of operations.

Even though the effects of the required wait states of DRAMs are taken into account in the estimated computational costs shown in Table 5.1, total required memory consumption should be discussed further, as this concrete number is valuable for defining system requirements. The proposed method designates the number of frequency bins to be selected, and the number of bins selected is assumed to be proportional to required memory consumption. Therefore, selecting a smaller number of bins results in lower required memory consumption, and thus to more efficient implementation. If the number of designated frequency bins is 64 and FFT size is 1024, the memory reduction ratio would be about 88 percent, resulting

in a required memory consumption level of about 96 kBytes for mobile devices. Note that the selection of 64 designated frequency bins was determined on the basis of experimental results. A required memory consumption level of 96 kBytes makes it feasible to use a static random access memory (SRAM) as the external memory. In this case, the required wait states of DRAMs do not need to be taken into consideration, greatly reducing the required number of operations, however SRAM is much more expensive than DRAM. In the case of portable equipment, required memory consumption drops to about 1.44 MBytes as a result of algorithm modification and bin selection, which means that use of SRAM is not feasible because this exceeds the memory capacity of SRAM. For conventional BSS using FDICA, the required memory consumption figures for mobile devices and portable equipment are 800 kBytes and 12 MBytes, respectively, as shown in Table 2.8. These estimated memory consumption levels are also too large to use SRAM. As this discussion makes clear, estimation of required memory consumption is an important consideration when designing speech equipment systems, one which also impacts manufacturing costs.

The number of the microphones used (N_M) also has an effect on the computational cost of FDICA functions. In Section 2.2, N_M was discussed in relation to determining which functions are the dominant factors in computational cost, and how these dominant functions can differ according to the number of microphones used. For example, when N_M is small, the iterative update process is the dominant computational cost. In contrast, when N_M is large, the permutation solution, consisting of the clustering method, becomes the dominant computational cost. This implies that optimization should be focused on functions whose costs are largely determined by N_M. When implementing the proposed method, this approach can indicate the direction of which important functions need to be optimized. For industry, this approach can help restrain software development costs.

5.2.2 Separation performance

In general, lower computational costs widen the range of application fields a proposed method can be applied in. Experimental results have shown that separation performance deteriorates as the number of ICA frequency bins decreases. This tendency has been observed for both mobile devices and portable equipment, however this deterioration is not

proportional to the reduction in the number of the frequency bins. In other words, there is a trade-off between separation performance and the number of the frequency bins selected, but performance drops more slowly than the rate of bin reduction, an observation on which our proposed separation method is based. The separation performance of conventional FDICA is about 20 to 30 dB of improvement in the signal-to-interference ratio (SIR_{imp}) [1, 2]. 30 dB of SIR_{imp} means 0.001 times the power of the interference signal. If twelve source signals were captured simultaneously, and the power of each signal was equal, the signal-to-interference ratio (SIR) of the mixed signal would be about -10 dB at the microphones. This value can be calculated using the following equation: $10\log_{10}(1/11)$. Thus, 30 dB of SIR_{imp} corresponds to 20 dB of absolute SIR. On the other hand, to achieve practical speech enhancement, the signal-to-noise ratio (SNR) should be about 10 to 20 dB. For example, a suppression of 15 dB means 0.03 times the power of the suppressed noise signal. In the example above, of FDICA with twelve sources, 15 dB of absolute SIR corresponds to 25 dB of SIR_{imp}. Therefore, a SIR_{imp} of 20 dB can be adopted as minimum separation performance target as possible, which corresponds to 10 dB of absolute SIR. Separation performance of the proposed BSS method using a DHMA exceeds this allowable minimum, with separation performance exceeding 20 dB of SIR_{imp}, while achieving a 90 percent reduction in the level of computational cost. For the previously proposed BSS method using a DHMA, experimental results of 24 dB of SIR_{imp} are reported, which corresponds to 14 dB of absolute SIR. This level of separation performance is equivalent to 0.04 times the power of the interference signal in the separated signal, under conditions in which the signal power of all the source signals are equal to one another. Using the proposed method, interference signal power is 0.1 times the power of the suppressed signal, which is only about twice the interference signal power when using the previously proposed BSS method with a DHMA. As explained in this paragraph, the proposed method satisfies the allowable minimum separation performance while sharply reducing the number of selected frequency bins, and thus computational cost.

On the other hand, SIR is not an appropriate measure of signal distortion. Signal-to-noise ratio (SNR) and cepstral distortion (CD) can be used to evaluate the degree of mismatch between unprocessed and processed signals, with the mismatch equaling signal distortion. SNR is a time domain measure and CD evaluates the comparison between

the spectral envelopes in the cepstrum domain, so these measures can be used to evaluate distortion from different perspectives. When only two microphones are used, the segmental signal-to-noise ratio (SNR_{seg}) is significantly improved. The FDICA separation matrix works as the adaptive beamformer, so that the separated signal tends to be attenuated in the low frequency region. This attenuation is caused by the acoustical properties of a two microphone array. The proposed method uses a Wiener filter in the non-selected frequency bins, and the filter is used more in the lower than in the higher frequency region. The Wiener filter cancels out the attenuation caused by the beamformer, as discussed in Section 3.4. This is due to the improvement in SNR_{seg} when using two microphones. In contrast, when using two microphones CD deteriorates, however this deterioration is less than 1 dB, which is a good trade off due to the large reduction in computational cost which results, accounting for over 80 percent of the overall reduction in computational cost. Therefore speech signal distortion is acceptably restrained by the proposed method, despite the large reduction in computational cost. For BSS with a DHMA, SNR_{seg} deteriorates, but this deterioration is slight, at less than 1 dB. When the frequency bins are uniformly selected, SNR_{seg} deterioration exceeds 1 dB. Even though SNR_{seg} deteriorates slightly when using the proposed method, CD improves because low cepstral distortion results in better performance, as shown in Figure 4.9. From the effect on both SNR_{seg} and CD when using the proposed method with a DHMA, significantly less distortion is achieved, even though there is an over 80 percent reduction in computational cost. In comparison, DUET [4], which is a time-frequency masking method, causes more distortion than the proposed method. In fact, since the proposed method uses a Wiener filter, it can also be categorized as a time-frequency masking method. Note that combining FDICA with a time-frequency masking method may be another avenue for improving BSS performance. In the discussion of SNR_{seg} and CD, the proposed method indicates how it has a tendency to cause almost equivalent amounts of distortion in SNR_{seg} and CD compared to the BSS methods which have been reported previously.

The proposed method illustrates a quite favorable trade-off between reduced computational cost and signal distortion. Regarding the SIR, the proposed method also shows the practical trade-off between separation performance and reduced computational cost. System developers and manufacturers can choose appropriate configurations with respect to the

balance between separation performance, signal distortion, system requirements and manufacturing costs. For speech communication, users communicate with each other through speech transmitting equipment, and these users prefer more natural sound rather than highly enhanced but distorted speech. The proposed method is suitable for this purpose. Even though separation performance deteriorates when using the proposed method, the amount of distortion is equivalent to that which occurs when using methods that employ all of the frequency bins. 80 percent reduction in overall computational cost, which is experimentally evaluated, can be achieved using the proposed method. Separation performance is in the acceptable range, as the SIR_{imp} is over 20 dB, and $\mathrm{SNR}_{\mathrm{seg}}$ and CD remain almost equivalent to, or are better than, results achieved when using conventional methods. Consequently, the proposed method is a practical BSS method which can be utilized in speech processing equipment with embedded processors.

5.3 Remaining issues

5.3.1 Separation matrix in unselected frequency bins

The proposed frequency bin selection method has an issue regarding separation of the observed signals in the unselected bins. Conventional FDICA cannot avoid the acoustic restrictions of microphone arrays because of linear signal processing. In the case of two microphones however, use of a Wiener filter for the unselected bins works well, especially in the low frequency region. This countermeasure to the drawback of microphone arrays suggests a method to restrain the signal distortion caused by microphone array processing. For the BSS method using a DHMA, in the unselected frequency bins only the interpolated separation matrix separates the observed signal. This interpolation can be applied to the inverse or pseudo-inverse of the estimated separation matrix. This corresponds to the interpolation between estimated mixing matrices, which did not work well, however, in our preliminary experiments. An interpolated separation matrix is practical, however, because separation using the interpolated separation matrix can be considered as a tentative separation, similar to the use of two microphones. At least in the low frequency region, signal

distortion may be improved using this approach, because the distortion is caused for acoustical reasons related to microphone array processing. On the other hand, if the number of source signals is large, the signal power of the separated signals might become an issue, because a Wiener filter corresponds to a dividing point in observed signal power. Therefore, additional investigation is required to determine how to obtain the separated signals in the unselected bins, especially when there are a large number of source signals.

5.3.2 On-line implementation

The proposed method was evaluated using batch processing, however an on-line method might be more practical. When portable speech equipment is being used, the positions of speakers and microphones do not usually change. In most cases, the speakers are sitting around a table, and the microphones are placed on the table. For mobile devices, however, the situation is more variable. Acoustic conditions may vary widely. For example, a speaker may bring the device from a small room into a large room, or the speaker may put the device on a table in the middle of a conversation. In order to deal with varying acoustic conditions, a block-wise on-line method of FDICA has been proposed [77]. Batch processing in one block estimates the separation matrix, and the separation matrices are then estimated again, block-by-block, using the observed signals from only one block. The block used for estimation includes some frames of Short Time Fourier Transform, and the block is shifted from time to time. In the proposed method, the selected frequency bins are fixed while estimating the separation matrix. Re-selecting the frequency bins is probably necessary when the block-wise on-line method is used with the proposed method. The following ideas and issues should be investigated in future work:

- approximated covariance matrix with a forgetting factor to obtain a selection criterion,
- block segmentation parameters (block length, overlap, etc.),
- management scheme for keeping the separation matrix in the selected frequency bins, because a selected bin may not be selected in the next block,
- management scheme for storing the observed signal based on block changes.

To manage storage of the observed signals, one possible method is to make block length longer than signal length for storage. ICA estimates the separation matrix using higher-order statistics, so signal length should be relatively long for the purpose of estimation, as mentioned in Section 2.1. Signal length should be at least a few seconds, and it should be noted that longer signal length is expected to improve separation performance. For the block-wise on-line method, block length should not be too long, because the goal is to adapt to changes in acoustic conditions. When block length is one second and the length of the observed signal is same as the block length, separation performance may deteriorate. These parameters should be determined by the system requirements of the speech processing equipment. The memory management scheme, however, should be carefully designed, and its influence on separation performance should be experimentally evaluated, especially regarding the distortion measures.

Chapter 6

Conclusion

The method of blind source separation (BSS) proposed in this dissertation uses frequency domain independent component analysis (FDICA), but limits the number of frequency bins selected in order to reduce computational costs by up to more than 80 percent. Experimental results show that separation performance deteriorates as a result of limiting the number of frequency bins, however the amount of distortion which results when using the proposed method remains at levels roughly equivalent to those of conventional methods using unlimited numbers of frequency bins. And when using two microphones, the segmental signal-to-noise ratio (SNR_{seg}) exceeds that of methods using conventional FDICA. When using a dodecahedral microphone array (DHMA), deterioration in SNR_{seg} is restrained to under 1 dB, and cepstral distortion is slightly improved.

BSS methods such as FDICA can be categorized as speech processing functions, similar to noise reduction methods or acoustic echo cancelers. When manufacturing mobile or portable speech processing equipment, using a single embedded processor is the most practical approach, even if several speech processing functions will be operating concurrently using the same processor. Computational costs for each function of FDICA were clarified, confirming that use of conventional FDICA in embedded processors concurrently with other speech processing functions is not realistic. In order to reduce computational cost, the proposed method selects a limited number of frequency bins using spatial correlation, which is a type of second-order statistics. Note that second-order statistics can be

combined to reduce FDICA's computational cost, even though ICA algorithms use higher-order statistics for mathematical optimization. By reducing computational costs by a ratio of 80 percent, the proposed FDICA method becomes quite feasible for use in embedded processors. It was also shown how the function responsible for the dominant computational cost can change, depending on the number of microphones used, allowing system design and optimization strategies to take into account dominant costs when choosing the number of microphones to be used. This insight enables system designers to better balance separation performance and manufacturing costs.

The following are notable features of the proposed method.

- Frequency bins for signal separation by ICA are selected using spatial correlation:
  - Spatial correlation provides directional information of the source signals.
  - When using two microphones, the determinant of the spatial covariance matrix can simultaneously evaluate the number of source signals and their respective strengths.
  - The trace of the covariance matrix is not appropriate for frequency bin selection. The trace equals the power of the observed signal, and never includes spatial information.
  - When using a DHMA, the magnitude squared coherence reflects the acoustical and spatial characteristics of the DHMA's shape which contributes to determine separable frequency regions.
- A significant reduction in computational costs is achieved:
  - An 80 percent reduction in cost is achieved, while maintaining acceptable performance.
  - The proposed method works with any ICA iterative update algorithm, including state-of-the-art FDICA with a permutation solution.
- Distortion levels are comparable to methods in which all of the frequency bins are used for signal separation by ICA:

  - The proposed frequency bin selection process is superior to uniform selection because of higher separation performance.
  - When using two microphones, the Wiener filter cancels out the acoustical disadvantage of the null-beamformer, even though tentative separation shows deterioration in performance.
  - The combination of the Wiener filter and FDICA improves the segmental signal-to-noise ratio, especially in the low frequency region.
  - When using a DHMA, deterioration of the segmental signal-to-noise ratio is significantly restrained.
  - Cepstral distortion is slightly improved.

- The critical functions regarding computational cost are clarified for FDICA:
  - When using two microphones, the iterative update process is responsible for the largest share of computational costs.
  - When using a DHMA, the permutation solution has the greatest computational cost.
  - The effect of increasing the number of microphones suggests that the clustering method is more critical than the numerical operation of the matrix calculation in regards to computational cost.

Future work includes developing a separation method for the unselected frequency bins, as well as an on-line frequency bin selection method. These refinements are expected to improve separation performance and adaptability to changes in acoustic conditions, respectively.

Bibliography

[1] M. Ogasawara, T. Nishino, and K. Takeda. A small dodecahedral microphone array for blind source separation. In *Proc. ICASSP*, pages 229–232, Mar. 2010.

[2] M. Ogasawara, T. Nishino, and K. Takeda. Blind source separation using dodecahedral microphone array under reverberant conditions. *IEICE transactions on fundamentals of electronics, communications and computer sciences*, E94-A(3):897–906, Mar. 2011.

[3] S. Boll. Suppression of acoustic noise in speech using spectral subtraction. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 27(2):113–120, Apr. 1979.

[4] O. Yilmaz and S. Rickard. Blind separation of speech mixtures via time-frequency masking. *IEEE Transactions on Signal Processing*, 52(7):1830–1847, Jul. 2004.

[5] Y. Izumi, N. Ono, and S. Sagayama. Sparseness-based 2ch bss using the em algorithm in reverberant environment. In *Proc. WASPAA*, pages 147–150, Oct. 2007.

[6] S. Araki, H. Sawada, R. Mukai, and S. Makino. Underdetermined blind source sparse source separation for arbitrarily arranged multiple sensors. *Signal processing*, 87(8):1833–1847, Aug. 2007.

[7] J. Herault and B. Ans. Circuits neuronaux à synapses modifiables: décodage de messages composites par apprentissage non supervisé. *Comptes Rendus de l'Académie des Sciences*, 299(III-13):525–528, 1984.

[8] J. Herault, C. Jutten, and B. Ans. Détection de grandeurs primitives dans un message composite par une architecture de calcul neuromimétique en apprentissage non supervisé. In *Actus du Xéme colloque GRESTI*, pages 1017–1022, 1985.

[9] C. Jutten. *Calcul neuromimétique et traitement du signal, analyse en composantes indépendentes*. PhD thesis, PhD thesis, INPG, Univ. Grenoble, 1987.

[10] T. W. Lee. *Independent Component Analysis - Theory and Applications*. Kluwer, Norwell, MA, 1998.

[11] A. Hyvärinen, J. Karhunen, and E. Oja. *Independent Component Analysis*. Wiley, New York, 2001.

[12] N. Delfosse and P. Loubaton. Adaptive blind separation of independent sources: a deflation approach. *Signal processing*, 45(1):59–83, 1995.

[13] A. Hyvärinen and E. Oja. A fast fixed-point algorithm for independent component analysis. *Neural computation*, 9(7):1483–1492, 1997.

[14] Z. Malouche and O. Macchi. Adaptive unsupervised extraction of one component of a linear mixture with a single neuron. *IEEE Transactions on Neural Networks*, 9(1):123–138, 1998.

[15] A. Hyvärinen. Fast and robust fixed-point algorithms for independent component analysis. *IEEE Transactions on Neural Networks*, 10(3):626–634, 1999.

[16] M. Gaeta and J.-L. Lacoume. Source separation without prior knowledge: the maximum likelihood solution. In *Proc. EUSIPCO*, pages 621–624, Sep. 1990.

[17] D.-T. Pham. Blind separation of instantaneous mixture of sources via an independent component analysis. *IEEE Transactions on Signal Processing*, 44(11):2768–2779, 1996.

[18] D.-T. Pham and P. Garat. Blind separation of mixture of independent sources through a quasi-maximum likelihood approach. *IEEE Transactions on Signal Processing*, 45(7):1712–1725, 1997.

[19] A. J. Bell and T. J. Sejnowski. An information-maximization approach to blind separation and blind deconvolution. *Neural compututation*, 7:1129–1159, Nov. 1995.

[20] P. Comon. Independent component analysis, a new concept? *Signal processing*, 36(3):287–314, 1994.

[21] S. Amari, A. Cichocki, and H. H. Yang. A new learning algorithm for blind signal separation. *Advances in neural information processing systems*, pages 757–763, 1996.

[22] S. Amari. Neural learning in structured parameter spaces-natural riemannian gradient. *Advances in neural information processing systems*, pages 127–133, 1997.

[23] N. Murata and S. Ikeda. An on-line algorithm for blind source separation on speech signals. In *Proc. NOLTA98*, pages 923–926, Sep. 1998.

[24] P. Smaragdis. Blind separation of convolved mixtures in the frequency domain. *Neurocomputing*, 22:21–34, Nov. 1998.

[25] J. Anemüller and B. Kollmeier. Amplitude modulation decorrelation for convolutive blind source separation. In *Proc. ICA*, pages 215–220, Jun. 2000.

[26] N. Murata, S. Ikeda, and A. Ziehe. An approach to blind source separation based on temporal structure of speech signals. *Neurocomputing*, 41(1-4):1–24, Oct. 2001.

[27] F. Asano, S. Ikeda, M. Ogawa, H. Asoh, and N. Kitawaki. A combined approach of array processing and independent component analysis for blind separation of acoustic signals. In *Proc. ICASSP*, pages 2729–2732, May 2001.

[28] L. Shobben and W. Sommen. A frequency domain blind signal separation method based on decorrelation. *IEEE Transactions on Signal Processing*, 50(8):1855–1865, Aug. 2002.

[29] M. Z. Ikram and D. R. Morgan. A beamforming approach to permutation alignment for multichannel frequency-domain blind speech separation. In *Proc. ICASSP*, volume I, pages 881–884, May 2002.

[30] S. Araki, S. Makino, R. Mukai, and H. Saruwatari. Equivalence between frequency domain blind source separation and frequency domain adaptive null beamformers. In *Proc. Eurospeech*, pages 2595–2598, Sep. 2001.

[31] S. Amari, S. C. Douglas, A. Cichocki, and H. H. Yang. Multichannel blind deconvolution and equalization using the natural gradient. In *Signal Processing Advances in Wireless Communications, First IEEE Signal Processing Workshop on*, pages 101–104, Apr. 1997.

[32] M. Kawamoto, K. Matsuoka, and N. Ohnishi. A method of blind separation for convolved non-stationary signals. In *Neurocomputing*, volume 22, pages 157–171, Nov. 1998.

[33] K. Matsuoka and S. Nakashima. Minimal distortion principle for blind source separation. In *Proc. ICA*, pages 722–727, Dec. 2001.

[34] S. C. Douglas and X. Sun. Convolutive blind separation of speech mixtures using the natural gradient. *Speech communication*, 39:65–78, Jan. 2003.

[35] T. Nishikawa, H. Saruwatari, and K. Shikano. Comparison of time-domain ica, frequency-domain ica and multistage ica for blind source seapration. In *Proc. EUSIPCO*, volume 2, pages 15–18, Sep. 2002.

[36] H. Sawada, R. Mukai, S. Araki, and S. Makino. Polar coordinate based nonlinear function for frequency domain blind source separation. In *Proc. ICASSP*, volume I, pages 1001–1004, May 2002.

[37] S. Ikeda and N. Murata. A method of ica in time-frequency domain. In *Proc. ICA*, pages 365–371, Jan. 1999.

[38] T. Nishikawa, H. Saruwatari, and K. Shikano. Blind separation of more than two sources based on high-convergence algorithm combining ica and beamforming. In *Proc. EUSIPCO*, Sep. 2005.

[39] S. Kurita, H. Saruwatari, S. Kajita, K. Takeda, and F. Itakura. Evaluation of blind signal separation method using directivity pattern under reverberant conditions. In *Proc. ICASSP*, volume 5, pages 3140–3143, Jun. 2000.

[40] K. Osako, Y. Mori, Y. Takahashi, H. Saruwatari, and K. Shikano. Fast convergence blind source separation using frequency subband interpolation by null beamforming. *IEICE TRANSACTIONS on Fundamentals of Electronics, Communications and Computer*, E91-A:1357–1361, Jun. 2009.

[41] T. Ema and N. Hamada. Fdica using time frequency cell selection for blind source separation. In *NCSP'05*, pages 471–474, Mar. 2005.

[42] K. Tachibana, H. Saruwatari, Y. Mori, S. Miyabe, K. Shikano, and A. Tanaka. Efficient blind source separation combining closed-form second-order ica and non closed-form higher-order ica. In *Proc. ICASSP*, pages 45–48, Apr. 2007.

[43] A. Tanaka, H. Imai, and M. Miyakoshi. Theoretical foundations of second-order-statics-based blind source separation for non-stationary sources. In *Proc. ICASSP*, pages 600–603, May 2006.

[44] S. Ikeda and N. Murata. An approach to blind source separation of speech signals. In *Proc. International Conference on Artificial Neural Networks*, pages 761–766, Sep. 1998.

[45] K. Rahbar and J.P. Reilly. A frequency domain method for blind source separation of convolutive audio mixtures. *IEEE Transactions on Speech and Audio Processing*, 13(5):832–844, Sep. 2005.

[46] M. Z. Ikram and D. R. Morgan. Permutation inconsistency in blind speech separation; investigation and solutions. *IEEE Transactions on Speech and Audio Processing*, 13(1):1–13, Jan. 2005.

[47] A. Gersho and R. M. Gray. *Vector Quantization and Signal Compression*. Kluwer Academic Publishers, Norwell, MA, 1998.

[48] D. MacKay. *Information Theory, Inference, and Learning Algorithms*, chapter 20, pages 284–292. Cambridge University Press, Cambridge, 2003.

[49] H. Sawada, S. Araki, R. Mukai, and S. Makino. Blind extraction of dominant target sources using ica and time-frequency masking. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(6):2165–2173, Oct. 2006.

[50] C.F. Olson. Parallel algorithms for hierarchical clustering. *Parallel computing*, 21:1313–1325, Aug. 1995.

[51] S. Winter, W. Kellermann, H. Sawada, and S. Makino. Map-based underdetermined blind source separation of convolutive mixtures by hierarchical clustering and l1-norm minimization. *EURASIP Journal on Applied Signal Processing*, 2007(1):1–12, Jan. 2007.

[52] D.-T. Pham, Ch. Servière, and H. Boumaraf. Blind separation of speech mixtures based on nonstationarity. In *Signal Processing and Its Applications, 2003. Proceedings. Seventh International Symposium on*, volume 2, pages 73–76. IEEE, Jul. 2003.

[53] D.-T. Pham, Ch. Servière, and H. Boumaraf. Blind separation of convolutive audio mixtures using nonstationarity. In *Proc. ICA*, pages 981–986, Apr. 2003.

[54] Ch. Servière and D.-T. Pham. Permutation correction in the frequency domain in blind separation of speech mixtures. *EURASIP Journal on Applied Signal Processing*, 2006(1):177–177, Jan. 2006.

[55] F. Nesta, M. Omologo, and P. Svaizer. A novel robust solution to the permutation problem based on a joint multiple tdoa estimation. In *Proc. IWAENC*, Sep. 2008.

[56] F. Nesta, M. Omologo, and P.Svaizer. Multiple tdoa estimation by using a state coherence transform for solving the permutation problem in frequency-domain bss. In *Proc. MLSP*, pages 43–48. IEEE, Oct. 2008.

[57] F. Nesta, T. S. Wada, and B.-H. Juang. Coherent spectral estimation for a robust solution of the permutation problem. In *Proc. WASPAA*, pages 105–108. IEEE, Oct. 2009.

[58] W. Wang, J. A. Chambers, and S. Sanei. A novel hybrid approach to the permutation problem of frequency domain blind source separation. In *Proc. ICA*, pages 532–539, Sep. 2004.

[59] H. Sawada, R. Mukai, S. Araki, and S. Makino. A robust and precise method for solving the permutation problem of frequency-domain blind source separation. *IEEE Transactions on Speech and Audio Processing*, 12(5):530–538, Sep. 2004.

[60] A. Hiroe. Solution of permutation problem in frequency domain ica, using multivariate probability density functions. In *Proc. ICA*, pages 601–608, 2006.

[61] T. Kim, H. T. Attias, S.-Y. Lee, and T.-W. Lee. Blind source separation exploiting higher-order frequency dependencies. *IEEE Transactions on Speech and Audio Processing*, 15(1):70–79, 2007.

[62] L. Parra and C. Spence. Convolutive blind separation of non-stationary sources. *IEEE Transactions on Speech and Audio Processing*, 8:320–327, May. 2000.

[63] H. Buchner, R. Aichner, and W. Kellermann. A generalization of a class of blind source separation algorithms for convolutive mixtures. In *Proc. ICA*, pages 945–950, Apr. 2003.

[64] H. Buchner, R Aichner, and W. Kellermann. A generalization of blind source separation algorithms for convolutive mixtures based on second order statistics. *IEEE Transactions on Speech and Audio Processing*, 13(1):120–134, 2005.

[65] A. D. Back and A. C. Tsoi. Blind deconvolution of signals using a complex recurrent network. In *Neural Networks for Signal Processing [1994] IV. Proceedings of the 1994 IEEE Workshop*, pages 565–574, Sep. 1994.

[66] R. H. Lambert and A. J. Bell. Blind separation of multiple speakers in a multipath environment. In *Proc. ICASSP*, pages 423–426, Apr. 1997.

[67] T. W. Lee, A. J. Bell, and R. Orglmeister. Blind source separation of real world signals. In *Neural Networks, 1997., International Conference on*, pages 2129–2135, Jun. 1997.

[68] M. Joho and P. Schniter. Frequency domain realization of a multi-channel blind deconvolution algorithm based on the natural gradient. In *Proc. ICA*, pages 543–548, Apr. 2003.

[69] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical recipes in C*. Cambridge University Press, 1998.

[70] K. Kondo, M. Yamada, and H. Kenmochi. A semi-blind source separation method with a less amount of computation suitable for tiny dsp modules. In *Proc. of Interspeech*, pages 1339–1342, Sep. 2009.

[71] J.R. Deller, J.G. Proakis, and J.H.L. Hansen. *Discrete-Time Processing of Speech Signals*. Macmillan, New York, 1993.

[72] L. Rabiner and B. Juang. *Fundamentals of Speech Recognition*. Prentice-Hall, New Jersey, 1993.

[73] M. Brandstein and D. Ward. *Microphone Arrays: Signal Processing Techniques and Applications*. Springer-Verlag, New York, 2001.

[74] J. Meyer and K. U. K. Uwe Simmer. Multi-channel speech enhacnement in a car envirionment using wiener filtering and spectral subtraction. In *Proc. ICASSP*, pages 1167–1170, Apr. 1997.

[75] I. A. McCowan and H. Bourlard. Microphone array post-filter based on noise field coherence. *IEEE Transactions on Speech and Audio Processing*, 11:709–716, 2003.

[76] S. Choi, S. Amari, A. Cichocki, and R. Liu. Natural gradient learning with a nonholonomic constraint for blind deconvolution of multiple channels. In *First International Workshop on Independent Component Analysis and Signal Separation*, pages 371–376, Jan. 1999.

[77] Y. Takahashi, T. Takatani, K. Osako, H. Saruwatari, and K. Shikano. Blind spatial subtraction array for speech enhancement in noisy environment. *IEEE Transactions on Audio, Speech, and Language Processing*, 17(4):650–664, May 2009.

Publications

Journal papers

1. K. Kondo, Y. Takahashi, S. Hashimoto, H. Saruwatari, T. Nishino, and K. Takeda. Improved method of blind speech separation with low computational complexity. *Journal of Advances in Acoustics and Vibrations*, July 2011. doi:10.1155/2011/765429.

2. K. Kondo, Y. Mizuno, T. Nishino, and K. Takeda. Practically efficient blind speech separation using frequency band selection based on magnitude squared coherence and a small dodecahedral microphone array. *Journal of Electrical and Computer Engineering*, Aug. 2012. doi:10.1155/2012/324398.

3. Y. Takahashi, H. Saruwatari, K. Shikano, and K. Kondo. Musical noise analysis in methods of integrating microphone array and spectral subtraction based on higher-order statistics. *EURASIP Journal on Advances in Signal Processing*, Apr. 2010. doi:10.1155/2010/431347.

4. H. Saruwatari, Y. Ishikawa, Y. Takahashi, T. Inoue, K. Shikano, and K. Kondo. Musical noise controllable algorithm of channelwise spectral subtraction and adaptive beamforming based on higher-order statistics. *IEEE Trans. ASLP*, 19(6):1457–1466, Aug. 2010. doi:10.1109/TASL.2010.2091636.

5. T. Inoue, H. Saruwatari, Y. Takahashi, T. Inoue, K. Shikano, and K. Kondo. Theoretical analysis of musical noise in generalized spectral subtraction based on higher-order statistics. *IEEE Trans. ASLP*, 19(6):1770–1779, Aug. 2011. doi:10.1109/

TASL.2010.2098871.

6. R. Miyazaki, H. Saruwatari, T. Inoue, Y. Takahashi, T. Inoue, K. Shikano, and K. Kondo. Musical-noise-free speech enhancement based on optimized iterative spectral subtraction. *IEEE Trans. ASLP*, 20(7):2080–2094, Sep. 2012. doi:10.1109/TASL.2012.2196513.

7. M. Yamada, G. Wichern, K. Kondo, M. Sugiyama, and H. Sawada. Noise adaptive optimization of matrix initialization for frequency-domain independent component analysis. *Digital Signal Processing*, 23(1):1–8, Jan. 2013. doi:10.1016/j.dsp.2012.08.010.

International conferences

1. K. Kondo, M. Yamada, and H. Kenmochi. A semi-blind source separation method with a less amount of computation suitable for tiny DSP modules. In *Proc. of Interspeech 2009*, pages 1339–1342, Sep. 2009.

2. K. Kondo, Y. Takahashi, S. Hashimoto, H. Saruwatari, T. Nishino, and K. Takeda. Efficient blind speech separation suitable for embedded devices. In *Proc. of EUSIPCO 2011*, pages 2319–2323, Aug. 2011.

3. Y. Mizuno, K. Kondo, T. Nishino, N. Kitaoka, and K. Takeda. Fast source separation based on selection of effective temporal frames. In *Proc. of EUSIPCO 2012*, pages 914–918, Aug. 2012.

4. J. J. Bosch, K. Kondo, R. Marxer, and J. Janer. Score-informed and timbre independent lead instrument separation in real-world scenarios. In *Proc. of EUSIPCO 2012*, pages 2417–2421, Aug. 2012.

5. Y. Uemura, Y. Takahashi, H. Saruwatari, K. Shikano, and K. Kondo. Automatic optimization of spectral subtraction based on musical noise assessment via higher-order statistics. In *Proc. of IWAENC 2008*, Sep. 2008.

6. Y. Uemura, Y. Takahashi, H. Saruwatari, K. Shikano, and K. Kondo. Musical noise generation analysis for noise reduction methods based on spectral subtraction and mmse-stsa estimation. In *Proc. of ICASSP 2009*, pages 4433–4436, Apr. 2009.

7. Y. Takahashi, Y. Uemura, H. Saruwatari, K. Shikano, and K. Kondo. Musical noise analysis based on higher order statistics for micropohne array and nonlinear signal processing. In *Proc. of ICASSP 2009*, pages 229–232, Apr. 2009.

8. Y. Takahashi, Y. Uemura, H. Saruwatari, K. Shikano, and K. Kondo. Structure selection algorithm for less musical-noise generation in integration systems of beamforming and spectral subtraction. In *Proc. of SSP 2009*, pages 701–704, Aug. 2009.

9. Y. Takahashi, H. Saruwatari, K. Shikano, and K. Kondo. Theoretical musical-noise analysis and its generalization for methods of integrating beamforming and spectral subtraction based on higher-order statistics. In *Proc. of ICASSP 2010*, pages 93–96, Mar. 2010.

10. Y. Ishikawa, H. Saruwatari, Y. Takahashi, K. Shikano, and K. Kondo. Musical noise controllable algorithm of channelwise spectral subtraction and beamforming based on higher-order statistics criterion. In *Proc. of CIP 2010*, pages 81–86, June 2010.

11. T. Inoue, Y. Takahashi, H. Saruwatari, K. Shikano, and K. Kondo. Theoretical analysis of musical noise in generalized spectral subtraction: why should not use power/amplitude subtraction? In *Proc. of EUSIPCO 2010*, pages 994–998, Aug. 2010.

12. T. Inoue, H. Saruwatari, Y. Takahashi, K. Shikano, and K. Kondo. Theoretical analysis of iterative weak spectral subtraction via higher-order statistics. In *Proc. of MLSP 2010*, pages 220–225, Sep. 2010.

13. H. Saruwatari, Y. Takahashi, K. Shikano, and K. Kondo. Blind speech extraction combining ICA-based noise estimation and less-musical-noise nonlinear post processing. In *Proc. of Asilomar Conf. 2010*, pages 1415–1419, Nov. 2010. (Invited Talk).

14. T. Inoue, H. Saruwatari, K. Shikano, and K. Kondo. Theoretical analysis of musical noise in Wiener filter via higher-order statistics. In *Proc. of APSIPA 2010*, pages 121–124, Dec. 2010.

15. T. Inoue, H. Saruwatari, K. Shikano, and K. Kondo. Theoretical analysis of musical noise in Wiener filtering family via higherorder statistics. In *Proc. of ICASSP 2011*, pages 5076–5079, May 2011.

16. K. Yagi, Y. Takahashi, H. Saruwatari, K. Shikano, and K. Kondo. Music sginal separation by orthogonality and maximum-distance constrained nonnegative matrix factorization with target signal information. In *Proc. of AES45th*, pages 2–5, Mar. 2012.

17. R. Miyazaki, H. Saruwatari, T. Inoue, K. Shikano, and K. Kondo. Musical-noise-free speech enhancement: Theory and evaluation. In *Proc. of ICASSP 2012*, pages 4565–4568, Mar. 2012.

18. R. Miyazaki, H. Saruwatari, K. Shikano, and K. Kondo. Musical-noise-free blind speech extraction using ica-based noise estimation and iterative spectral subtraction. In *Proc. of ISSPA 2012*, pages 322–327, July 2012.

19. R. Miyazaki, H. Saruwatari, K. Shikano, and K. Kondo. Musical-noise-free blind speech extraction using ica-based noise estimation with channel selection. In *Proc. of IWAENC 2012*, Sep. 2012.

20. S. Kanehara, H. Saruwatari, R. Miyazaki, K. Shikano, and K. Kondo. Theoretical analysis of musical noise generation in noise reduction methods with decision-directed a priori SNR estimator. In *Proc. of IWAENC 2012*, Sep. 2012.

21. Y. Takahashi, R. Miyazaki, H. Saruwatari, and K. Kondo. Theoretical analysis of musical noise in nonlinear noise reduction based on higher-order statistics. In *Proc. of APSIPA 2012*, Dec. 2012. (Invited Talk).

22. S. Kanehara, H. Saruwatari, R. Miyazaki, K. Shikano, and K. Kondo. Comparative study on various noise reduction methods with decision-directed a priori SNR estimator via higher-order statistics. In *Proc. of APSIPA 2012*, Dec. 2012.

23. R. Miyazaki, H. Saruwatari, K. Shikano, and K. Kondo. Musical-noise-free speech enhancement based on iterative Wiener filtering. In *Proc. of APSIPA 2012*, Dec. 2012.

24. Y. Iwao, H. Saruwatari, N. Kamado, K. Shikano, K. Kondo, and Y. Takahashi. Stereo music signal separation combining directional clustering and nonnegative matrix factorization. In *Proc. of ISSPIT 2012*, Dec. 2012.

Technical meeting

1. K. Kondo, Y. Takahashi, S. Hashimoto, T. Nishino, and K. Takeda. Tiny-setup blind source separation via time-varying softmask based on alternative separation matrix. In *IEICE Technical Report EA2010-126 110(471)*, pages 1–6, Mar. 2011. (in Japanese).

2. Y. Mizuno, S. Esaki, K. Kondo, T. Nishino, N. Kitaoka, and K. Takeda. Reducing computational complexity of ICA source separation based on coherence between observed signals. In *IEICE Technical Report EA2011-33 111(89)*, pages 19–24, June 2011. (in Japanese).

3. Y. Mizuno, K. Kondo, T. Nishino, N. Kitaoka, and K. Takeda. Reducing computational complexity of FDICA source separation based on source number evaluation. In *IEICE Technical Report EA2012-110 112(347)*, pages 5–10, Dec. 2012. (in Japanese).

4. Y. Uemura, Y. Takahashi, H. Saruwatari, K. Shikano, and K. Kondo. Relationship between logarithmic kurtosis ratio and degree of musical noise generation on spectral subtraction. In *IEICE Technical Report EA2008-44 108(143)*, pages 43–48, July 2008.

5. Y. Uemura, Y. Takahashi, H. Saruwatari, K. Shikano, and K. Kondo. Objective sound quality comparison based on higher-order statistics for nonlinear noise reduction methods. In *IEICE Technical Report EA2008-127 108(411)*, pages 68–72, Jan. 2009.

6. Y. Takahashi, Y. Uemura, H. Saruwatari, K. Shikano, and K. Kondo. Objective sound quality evaluation for combination method of beamforming and spectral subtraction. In *IEICE Technical Report EA2008-128 108(411)*, pages 73–78, Jan. 2009.

7. 上村 益永, 高橋 祐, 猿渡 洋, 鹿野 清宏, 近藤 多伸. 高次統計量によるミュージカルノイズ尺度に基づくスペクトル減算法の短時間区間自動最適化. **第 *23* 回 信号処理シンポジウム**, pages 241–246, Nov. 2008.

8. Y. Ishikawa, Y. Takahashi, H. Saruwatari, K. Shikano, and K. Kondo. Musical-noise regulation for combination method of channel-domain nonlinear speech enhancement and adaptive array signal processing. In *IEICE Technical Report EA2009-3 109(55)*, pages 11–16, May 2009.

9. T. Inoue, Y. Takahashi, H. Saruwatari, K. Shikano, and K. Kondo. Mathematical metric of musical noise in arbitrary exponent domain ss. In *IEICE Technical Report EA2010-7 110(54)*, pages 37–42, May 2010.

10. T. Inoue, H. Saruwatari, Y. Takahashi, K. Shikano, and K. Kondo. Mathematical metric of amount of musical noise in recursive spectral subtraction. In *IEICE Technical Report EA2010-30 110(71)*, pages 47–52, June 2010.

11. T. Inoue, H. Saruwatari, K. Shikano, and K. Kondo. Mathematical metric of musical noise in Wiener filtering. In *IEICE Technical Report EA2010-105 110(401)*, pages 13–18, Jan. 2011.

12. K. Yagi, H. Saruwatari, K. Shikano, K. Kondo, and Y. Takahashi. Instrumental signal separation based on non-negative matrix factorization with target signal information. In *IEICE Technical Report EA2011-46 111(136)*, pages 21–26, July 2011.

13. 宮崎 亮一, 猿渡 洋, 鹿野 清宏, 近藤 多伸. ミュージカルノイズフリー雑音抑圧の一般化理論とその信号抽出への応用. **第 *26* 回信号処理シンポジウム**, pages 368–373, Nov. 2011.

14. R. Miyazaki, H. Saruwatari, K. Shikano, and K. Kondo. Evaluation of musical-noise-free noise reduction under real acoustic environments. In *IEICE Technical Report EA2011-108 111(402)*, pages 25–30, Jan. 2012.

15. R. Miyazaki, H. Saruwatari, K. Shikano, and K. Kondo. Iterative blind spatial subtraction array for musical-noise-free speech enhancement in diffuse noise. In *IEICE Technical Report EA2011-125 111(490)*, pages 31–36, Mar. 2012.

16. S. Kanehara, R. Miyazaki, H. Saruwatari, K. Shikano, and K. Kondo. Mathematical metric of musical noise for various nonlinear speech enhancement algorithms. In *IEICE Technical Report EA2012-44 112(76)*, pages 67–72, June 2012.

17. 金原 涼美, 猿渡 洋, 宮崎 亮一, 鹿野 清宏, 近藤 多伸. 様々な非線形音声強調法における近似モデルを用いたミュージカルノイズ発生量解析. 第 *27* 回信号処理シンポジウム, pages 430–435, Nov. 2012.

18. 宮崎 亮一, 猿渡 洋, 鹿野 清宏, 近藤 多伸. 様々な動的雑音推定器に基づくミュージカルノイズフリー雑音抑圧処理の評価. 第 *27* 回信号処理シンポジウム, pages 436–441, Nov. 2012.

19. Y. Takahashi, K. Kondo, and S. Hashimoto. Harmonic and nonharmonic signal decomposition for musical signal using cepstral domain median filtering. In *IEICE Technical Report EA2012-39 112(76)*, pages 37–42, June 2012. (in Japanese).

Annual meetings

1. K. Kondo and M. Yamada. Frequency domain blind source separation based on learning bands selection. In *Proc. of ASJ2009 Spring*, pages 815–816, Mar. 2009. (in Japanese).

2. K. Kondo and M. Yamada. An effects for the source separation between different learning band selection methods. In *Proc. of ASJ2009 Autumn*, pages 755–756, Sep. 2009. (in Japanese).

3. K. Kondo and M. Yamada. An effects for the source separation between different source powers on learning band selection methods. In *Proc. of ASJ2010 Spring*, pages 807–808, Mar. 2010. (in Japanese).

4. K. Kondo, Y. Takahashi, S. Hashimoto, T. Nishino, and K. Takeda. Performance improvement in the lower frequency region for tiny-setup ICA and instantaneous softmask. In *Proc. of ASJ2011 Spring*, pages 799–800, Mar. 2011. (in Japanese).

5. K. Kondo, Y. Takahashi, T. Komatsu, T. Nishino, and K. Takeda. Complementary Wiener filter for speech dereverberation and theoretical derivation. In *Proc. of ASJ2012 Autumn*, pages 801–802, Sep. 2012. (in Japanese).

6. M. Yamada and K. Kondo. Semi-blind source separation with covariance fitting and higher-order ICA. In *Proc. of ASJ2009 Spring*, pages 813–814, Mar. 2009. (in Japanese).

7. M. Yamada, G. Wichern, M. Sugiyama, and K. Kondo. Semi-blind source separation under ambient noise condition change. In *Proc. of ASJ2009 Autumn*, pages 751–754, Sep. 2009.

8. M. Yamada, Y. Takahashi, K. Kondo, and H. Saruwatari. Bootstrap aggregating spectral subtraction for musical noise reduction. In *Proc. of ASJ2010 Spring*, pages 851–852, Mar. 2010. (in Japanese).

9. Y. Mizuno, S. Esaki, K. Kondo, T. Nishino, N. Kitaoka, and K. Takeda. Reducing computational complexity of ICA source separation based on magnitude squared coherence. In *Proc. of ASJ2011 Autumn*, pages 721–722, Sep. 2011. (in Japanese).

10. Y. Mizuno, K. Kondo, T. Nishino, N. Kitaoka, and K. Takeda. Reducing computational complexity of ICA source separation based on observed power spectrum. In *Proc. of ASJ2012 Spring*, pages 863–866, Mar. 2012. (in Japanese).

11. T. Komatsu, K. Kondo, T. Nishino, N. Kitaoka, and K. Takeda. Experimental evaluation of complementary Wiener filter for speech dereverberation. In *Proc. of ASJ2012 Autumn*, pages 799–800, Sep. 2012. (in Japanese).

12. Y. Uemura, Y. Takahashi, H. Saruwatari, K. Shikano, and K. Kondo. Automatic optimization scheme in spectral subtraction with higher-order statistics-based musical

noise assessment. In *Proc. of ASJ2008 Autumn*, pages 691–694, Sep. 2008. (in Japanese).

13. Y. Uemura, Y. Takahashi, H. Saruwatari, K. Shikano, and K. Kondo. Analytic study on generated musical noise in nonlinear noise suppression processes. In *Proc. of ASJ2009 Spring*, pages 723–726, Mar. 2009. (in Japanese).

14. Y. Takahashi, H. Saruwatari, K. Shikano, and K. Kondo. Automatic structure selection method in integration of beamforming and spectral subtraction based on metric for the amount of musical-noise generation. In *Proc. of ASJ2009 Autumn*, pages 635–638, Sep. 2009. (in Japanese).

15. Y. Ishikawa, Y. Takahashi, H. Saruwatari, K. Shikano, and K. Kondo. Subjective evaluation of musical noise controlable array signal processing. In *Proc. of ASJ2009 Autumn*, pages 639–642, Sep. 2009. (in Japanese).

16. T. Inoue, Y. Takahashi, Y. Ishikawa, H. Saruwatari, K. Shikano, and K. Kondo. Mathematical analysis of musical noise for generalized spectral subtraction method. In *Proc. of ASJ2010 Spring*, pages 759–762, Mar. 2010. (in Japanese).

17. Y. Takahashi, H. Saruwatari, K. Shikano, and K. Kondo. Generalization of theoretical musical-noise analysis in methods of integrating beamforming and spectral subtraction based on higher-order statistics. In *Proc. of ASJ2010 Spring*, pages 759–762, Mar. 2010. (in Japanese).

18. Y. Ishikawa, Y. Takahashi, H. Saruwatari, K. Shikano, and K. Kondo. Evaluation of musical noise controllable array signal processing in real environment. In *Proc. of ASJ2010 Spring*, pages 767–768, Mar. 2010. (in Japanese).

19. T. Inoue, Y. Takahashi, H. Saruwatari, K. Shikano, and K. Kondo. Mathematical analysis and evaluation of musical noise for iterative spectral subtraction method. In *Proc. of ASJ2010 Autumn*, pages 635–638, Sep. 2010.

20. T. Inoue, H. Saruwatari, K. Shikano, and K. Kondo. Mathematical analysis of musical noise for nonlinear singal processing. In *Proc. of ASJ2011 Spring*, pages 671–674, Mar. 2011. (in Japanese).

21. R. Miyazaki, H. Saruwatari, T. Inoue, K. Shikano, and K. Kondo. Musical-noise-free speech enhancement: Its theory and evaluation. In *Proc. of ASJ2011 Autumn*, pages 601–604, Sep. 2011. (in Japanese).

22. K. Yagi, H. Saruwatari, K. Shikano, K. Kondo, and Y. Takahashi. Evaluation of instrumental signal separation based on constrained non-negative matrix factorization with target signal information. In *Proc. of ASJ2011 Autumn*, pages 905–908, Sep. 2011. (in Japanese).

23. R. Miyazaki, H. Saruwatari, K. Shikano, and K. Kondo. Evaluation of speech distortion in musical-noise-free noise reduction method. In *Proc. of ASJ2012 Spring*, pages 805–808, Mar. 2012. (in Japanese).

24. K. Yagi, H. Saruwatari, K. Shikano, K. Kondo, and Y. Takahashi. Evaluation of instrumental signal separation based on supervised and constrained non-negative matrix factorization with basis-divergence maximization. In *Proc. of ASJ2012 Spring*, pages 1031–1034, Mar. 2012. (in Japanese).

25. S. Kanehara, H. Saruwatari, R. Miyazaki, K. Shikano, and K. Kondo. Mathematical analysis of musical noise for speech enhancement algorithms with decision-directed a priori SNR estimator. In *Proc. of ASJ2012 Autumn*, pages 633–636, Sep. 2012. (in Japanese).

26. R. Miyazaki, H. Saruwatari, K. Shikano, and K. Kondo. Musical-noise-free blind speech extraction using ica-based noise estimation with channel selection. In *Proc. of ASJ2012 Autumn*, pages 691–694, Sep. 2012. (in Japanese).

27. Y. Iwao, H. Saruwatari, N. Kamado, K. Shikano, K. Kondo, and Y. Takahashi. Evaluation of music signal separation combining directional clustering and nonnegative matrix factorization. In *Proc. of ASJ2012 Autumn*, pages 947–950, Sep. 2012. (in Japanese).