University of Louisville
ThinkIR: The University of Louisville's Institutional Repository

Electronic Theses and Dissertations

12-2007

Eigenvector-based multidimensional frequency estimation : identifiability, performance, and applications.

Jun Liu
University of Louisville

Follow this and additional works at: https://ir.library.louisville.edu/etd

Recommended Citation
Liu, Jun, "Eigenvector-based multidimensional frequency estimation : identifiability, performance, and applications." (2007). *Electronic Theses and Dissertations.* Paper 842.
https://doi.org/10.18297/etd/842

This Doctoral Dissertation is brought to you for free and open access by ThinkIR: The University of Louisville's Institutional Repository. It has been accepted for inclusion in Electronic Theses and Dissertations by an authorized administrator of ThinkIR: The University of Louisville's Institutional Repository. This title appears here courtesy of the author, who has retained all other copyrights. For more information, please contact thinkir@louisville.edu.

EIGENVECTOR-BASED MULTIDIMENSIONAL FREQUENCY ESTIMATION: IDENTIFIABILITY, PERFORMANCE, AND APPLICATIONS

By

Jun Liu
M.S., EE, Huazhong University of Science and Technology, 2001

A Dissertation
Submitted to the Faculty of the
Graduate School of the University of Louisville
in Partial Fulfillment of the Requirements
for the Degree of

Doctor of Philosophy

Department of Electrical and Computer Engineering
University of Louisville
Louisville, Kentucky

December 2007

EIGENVECTOR-BASED MULTIDIMENSIONAL FREQUENCY ESTIMATION: IDENTIFIABILITY, PERFORMANCE, AND APPLICATIONS

By

Jun Liu
M.S., EE, Huazhong University of Science and Technology, 2001

A Dissertation Approved on

August 7, 2007
Date

by the following Dissertation Committee:

Xiangqian Liu, Committee Chair

Jacek. M Zurada

John H. Lilly

Prasanna Sahoo

Xiaoli Ma

ACKNOWLEDGEMENTS

First of all, I would like to thank my advisor, Dr. Xiangqian Liu, for his valuable advice, guidance and help throughout this academic experience. Without his critical feedback on my research and technique writing, it would not be possible for me to accomplish this work. I also thank Dr. Xiaoli Ma for her valuable comments for my research work and for serving on my doctoral committee. I appreciate Prof. Prasanna Sahoo for his lessons in mathematical statistics and algebra and for serving on my committee. I also thank Prof. Jacek M. Zurada and Prof. John H. Lilly for serving on my committee and for their constructive feedback.

I am grateful to the Dept. of ECE for the financial support and the U.S. Army Research Laboratory and the U.S. Army Research Office for their funding support under grant number W911NF-05-1-0485. I thank Ms. Lisa Bell, the program assistant senior, for her help.

I thank my labmates and classmates, Ms. Jingli Li, Mr. Gang Zhao, and Mr. Dongqing Chen, and all the other friends of mine in Louisville, for their kind help during these memorable years. Without their support, I could not go through those difficult moments and achieve this goal.

Finally, I would like to thank my family for their unconditional love and support throughout my life. Especially, I want to thank my wife Hong, for her encouragement and support. As my nearest colleague and dearest friend, she accompanies me all along.

ABSTRACT

EIGENVECTOR-BASED MULTIDIMENSIONAL FREQUENCY ESTIMATION: IDENTIFIABILITY, PERFORMANCE, AND APPLICATIONS

Jun Liu

August 7, 2007

Multidimensional frequency estimation is a classic signal processing problem that has versatile applications in sensor array processing and wireless communications. Eigenvalue-based two-dimensional (2-D) and N-dimensional (N-D) frequency estimation algorithms have been well documented, however, these algorithms suffer from limited identifiability and demanding computations. This dissertation develops a framework on eigenvector-based N-D frequency estimation, which contains several novel algorithms that estimate a structural matrix from eigenvectors and then resolve the N-D frequencies by dividing the elements of the structural matrix. Compared to the existing eigenvalue-based algorithms, these eigenvector-based algorithms can achieve automatic pairing without an extra frequency pairing step, and thus the computational complexity is reduced. The identifiability, performance, and complexity of these algorithms are also systematically studied. Based on this study, the most relaxed identifiability condition for the N-D frequency estimation problem is given and an effective approach using optimized weighting factors to improve the performance of frequency estimation is developed. These results are applied in wireless communication for time-varying multipath channel estimation and prediction, as well as for joint 2-D Direction-of-arrival (DOA) tracking of multiple moving targets.

TABLE OF CONTENTS

Page

ACKNOWLEDGEMENTS iii
ABSTRACT iv
LIST OF TABLES viii
LIST OF FIGURES ix

CHAPTER

1 INTRODUCTION 1
- 1.1 Applications of the Multidimensional Frequency Estimation . . 2
- 1.2 State of the Art 3
- 1.3 Identifiability Bound 5
- 1.4 Performance Analysis 6
- 1.5 Main Contributions 7
- 1.6 Problem Statement of Multidimensional Frequency Estimation . 8
  - 1.6.1 2-D Frequency Estimation from one snapshot 8
  - 1.6.2 N-D Frequency estimation from a single snapshot 9
  - 1.6.3 N-D Frequency Estimation from multiple snapshots . . . 10

2 THE IMDF ALGORITHM 11
- 2.1 The IMDF algorithm 11
  - 2.1.1 Some Properties of the Khatri-Rao Products 11
  - 2.1.2 The 2-D IMDF Algorithm 12
  - 2.1.3 The N-D IMDF Algorithm 17
  - 2.1.4 The IMDF algorithm In Multiple Snapshots Case 20
- 2.2 Statistical Identifiability 21

    2.2.1 Statistical Identifiability of the 2-D IMDF Algorithm . . 21
    2.2.2 Statistical Identifiability of the N-D IMDF Algorithm . . 25
    2.2.3 Statistical Identifiability in the Multiple Snapshot Case . 27
  2.3 Computational Complexity 30
  2.4 Performance Analysis 30
  2.5 Simulation Results 34

3 EIGENVECTOR-BASED ESTIMATION WITH WEIGHTING FACTORS 38
  3.1 The Sufficient Snapshot Case 39
    3.1.1 Eigenvector-based algorithms using weighting factors . . 39
    3.1.2 Performance Analysis 41
    3.1.3 Simulation Results 42
  3.2 The Finite Snapshot Case 44
  3.3 Perturbation Analysis in the Finite Snapshot Case 48
    3.3.1 The Perturbation of Singular Vectors in SVD 48
    3.3.2 The Perturbation of $\boldsymbol{U}_s$ 55
    3.3.3 The Perturbation of $\boldsymbol{T}$ 57
    3.3.4 The Perturbation of $e^{\omega_{f,n}}$ 59
    3.3.5 Simplification of (3.77) and (3.78) 60
    3.3.6 The Theoretic Mean and MSE of the Estimates 61

4 OPTIMIZATION OF WEIGHTING FACTORS 63
  4.1 Optimizing Weighting Factors to Minimize Variances 63
  4.2 Simulation Results 69
    4.2.1 2-D Identical Frequency Estimation from Single Snapshot 69
    4.2.2 2-D Close Frequency Estimation from Multiple Snapshots 71
    4.2.3 3-D Identical Frequency Estimation from Multiple Snapshots 72
    4.2.4 Complexity Comparison 72

5 APPLICATION IN TIME-VARYING CHANNEL ESTIMATION 74

5.1 System Model 75

5.1.1 Adaptive OFDM System 75

5.1.2 Time-Varying Channel Models 77

5.1.3 Pilot Pattern 79

5.2 Channel Identification and Prediction 79

5.2.1 Estimation of the Model Order and Signal Subspace . . . 81

5.2.2 Estimation of the Transformation Matrix $\boldsymbol{T}$ 82

5.2.3 Estimation of Model Parameters 84

5.2.4 Channel Prediction 85

5.3 Simulation Results 85

6 APPLICATION IN TRACKING OF MULTIPLE MOVING TARGETS 90

6.1 Data Model for 2-D DOA Tracking 91

6.2 Subspace Tracking Based on the LOAFR1 Algorithm 93

6.3 Estimation of the Transformation Matrix $\boldsymbol{T}(t)$ 93

6.4 Estimation of DOAs 94

6.5 Update of the Weighting Factors 95

6.6 Simulation Results 98

7 CONCLUSION 102

REFERENCES 105

CURRICULUM VITAE 116

LIST OF TABLES

TABLE **Page**

2.1 The 2-D IMDF algorithm 17

2.2 Computational complexity of the 2-D IMDF algorithm when $T = 1$. 31

3.1 An eigenvector-based algorithm for N-D frequency estimation using weighting factors 41

4.1 An improved eigenvector-based algorithm using optimal weighting factors 67

5.1 OFDM channel identification using joint 2-D frequency estimation . . 86

6.1 The complexity order of the 2-D DOA tracking algorithm 97

7.1 Algebraic algorithms for N-D frequency estimation 103

LIST OF FIGURES

FIGURE **Page**

2.1 Comparison of ID bounds for 2-D frequency estimation (the two dimensions have equal size: $M_1 = M_2$). 25

2.2 Identifiability (ID) bound of 2-D frequency estimation: (a) ID bound versus M_1 when $M_1 = M_2$; (b) ID bounds versus the number of snapshots where $M_1 = 10$ and $M_2 = 6$. 29

2.3 Standard deviation (STD) of 2-D frequency estimation error when $F = 1$: (a) STD of $\Delta\omega$; (b) STD of $\Delta\nu$; (c) Average STD of $\Delta\omega$ and $\Delta\nu$; (d) STD as a function of K_1 when $K_2 = 14$; (e) STD as a function of K_2 when $K_1 = 6$; (f) STD of eigenvector-based and eigenvalue-based estimation, with RMSE of the 2-D IMDF estimation (eigenvector-based) and the corresponding CRB, when $K_1 = 5$ and $K_2 = 6$. 33

2.4 RMSE of 2-D frequency estimation. (a) Case 1: Moderately close frequencies in both dimensions; (b) Case 2: Very close frequencies in both dimensions; (c) Very close frequencies in the first dimension, well separated frequencies in the second dimension; (d) Very close frequencies in the second dimension, well separated frequencies in the first dimension. 37

3.1 RMSE of 3-D frequency estimation with multiple snapshots: (a) frequency tuples are fixed in all realizations; (b) frequency tuples are randomly generated in each realization. 44

4.1 The average efficiency as a function of α_1 and α_2 ($|\alpha_1| = \sqrt{\frac{1}{3}}$, $|\alpha_2| = \sqrt{\frac{1}{3}}$, and $\alpha_3 = \sqrt{\frac{1}{3}}$). 66

4.2 (a) RMSE of different optimization methods versus SNR; (b) RMSE of different optimization methods versus the number of iterations of Steps 3-4. 69

4.3 (a) Comparison of different algorithms for 2-D frequency estimation from single snapshot; (b) Comparison of optimized $\boldsymbol{\alpha}$ and randomly chosen $\boldsymbol{\alpha}$. 70

4.4 (a) Comparison of different algorithms for 2-D frequency estimation from multiple snapshots; (b) Comparison of optimized $\boldsymbol{\alpha}$ and randomly chosen $\boldsymbol{\alpha}$. 71

4.5 (a) Comparison of different algorithms for 3-D frequency estimation from single snapshot; (b) Comparison of optimized $\boldsymbol{\alpha}$ and randomly chosen $\boldsymbol{\alpha}$. 72

4.6 The number of floating point operations versus M_1 when: (a) $F = 3$; (b) $F = 30$. 73

5.1 Adaptive OFDM system. 77

5.2 Channel trace of the first tap at SNR = 15 dB. 88

5.3 Snapshot of predicted channel at 610-th symbol at SNR = 15 dB. . . 88

5.4 Channel NMSE versus (a) time at SNR=15 dB and (b) SNR when the 610-th symbol is received. 89

6.1 The trajectory of the true and estimated 2-D DOAs for three moving targets (SNR = 2dB) 99

6.2 The true and estimated elevation angles for three moving targets (SNR = 2dB) 100

6.3 The true and estimated azimuth angles for three moving targets (SNR = 2dB) 100

6.4 The NMSE versus SNR 101

CHAPTER 1

INTRODUCTION

Two-dimensional (2-D) and N-dimensional (N-D) frequency estimation has a variety of practical applications, including radar signal processing, wireless communication, and image processing. In this chapter, Section 1.1 and Section 1.2 review some applications and the state of the art of multidimensional frequency estimation. Then, Section 1.3 and Section 1.4 consider two theoretical issues associated with the N-D frequency estimation: the identifiability condition in the noiseless case and the statistical performance in the noisy case.

The two-dimensional frequency estimation problem can be stated as follows. Given an $M_1 \times M_2$ measurement data set $\boldsymbol{X}$ with typical element

$$x_{m_1,m_2} = \sum_{f=1}^{F} c_f e^{j(m_1-1)\omega_{f,1}} e^{j(m_2-1)\omega_{f,2}} + w_{m_1,m_2}, \tag{1.1}$$

for $m_1 = 1, \ldots, M_1$, and $m_2 = 1, \ldots, M_2$, how to estimate the parameter triples $(\omega_{f,1}, \omega_{f,2}, c_f)$ for $f = 1, \cdots, F$, where the frequencies $\omega_{f,1}, \omega_{f,2} \in \Pi$, and $\Pi := [-\pi, \pi]$. w_{m_1,m_2} is observation noise. The frequency pair $(\omega_{f,1}, \omega_{f,2})$ is a 2-D frequency component, and there are F such components to be estimated. The data model (1.1) can be generalized to N-D and multiple snapshot case. For example, a single snapshot N-D frequency mixture can be modeled as an N-D array $\underline{\boldsymbol{X}}$ with typical element

$$x_{m_1,m_2,\cdots,m_N} = \sum_{f=1}^{F} c_f \prod_{n=1}^{N} e^{j\omega_{f,n}(m_n-1)} + w_{m_1,m_2,\cdots,m_N}, \tag{1.2}$$

where $m_n = 1, \ldots, M_n$, for $n = 1, \ldots, N$, and M_n is the dimension size of the n-th dimension. The total dimension size is $M := \prod_{n=1}^{N} M_n$. In (1.2), the frequencies

$\omega_{f,n} \in \Pi$, for $f = 1, \ldots, F$, $n = 1, \ldots, N$, and $w_{m_1,m_2,\cdots,m_N}$ is observation noise. Similarly T snapshots of N-D frequency data mixtures may be modeled as T N-D arrays, $\underline{\boldsymbol{X}}(t)$, with typical element

$$x_{m_1,m_2,\cdots,m_N}(t) = \sum_{f=1}^{F} c_f(t) \prod_{n=1}^{N} e^{j\omega_{f,n}(m_n-1)} + w_{m_1,m_2,\cdots,m_N}(t), \quad t = 1, \cdots, T, \quad (1.3)$$

where t is the snapshot index, which can be a time index, or trial index in case of multiple trials of experiments. $T = 1$ corresponds to the single snapshot case. The objective of N-D frequency estimation is to estimate $\{\omega_{f,n}\}_{n=1}^{N}$, for $f = 1, \ldots, F$, from given $\underline{\boldsymbol{X}}(t)$, $t = 1, \cdots, T$.

1.1 Applications of the Multidimensional Frequency Estimation

Earlier applications of N-D frequency estimation in signal processing include the Direction-of-Arrival (DOA) estimation in radar signal processing using an antenna array. When the antenna array is a Uniform Linear Array (ULA), to estimate the DOA is a problem of 1-D frequency estimation [1, 2, 3]. When the antenna array is a Uniform Rectangular Array (URA), the problem becomes 2-D frequency estimation [4, 5, 6]. The two dimensions are azimuth and elevation angles. If the samples of a sinusoidal signal are collected in the time dimension, 3-D frequency estimation can be applied to jointly estimate the 2-D angles and signal carrier frequencies as in [7]. If the signal frequencies are hopping, 3-D frequency estimation algorithms are applicable when combined with other tracking techniques as in [8].

The N-D frequency estimation is also critical to the estimation of the parameters in the deterministic multipath channel models in wireless communication [9, 10, 11]. The JADE algorithm in [9] is a 2-D frequency estimator to resolve the delay and DOA parameters in wireless multipath channels. The most general deterministic multipath channel is the so-called Double Direction Channel Model (DDCM) [12], which is used in wireless channel sounding extensively. The N-D frequency estimation is an important tool to estimate the model parameters such as DOA, Direction-of-Depart (DOD), delay, doppler shift in each path of the model [13, 14, 15].

Other applications of N-D frequency estimation were also found in recent years. For example, in motion estimation of video compression, the estimation of the 3-D motion vector field can be formulated as a 3-D frequency estimation problem [16]. In nuclear magnetic resonance (NMR) spectroscopy, the measured 2-D resonance data is obtained by exciting a molecular system with a 2-D radio-frequency pulse sequence and can be modeled as sum of 2-D damped harmonics [17, 18].

Since N-D frequency estimation is so important in practice, the development of reliable and efficient algorithms with both theoretical support and application value become a necessity.

1.2 State of the Art

Both non-parametric methods, which are based on Fast Fourier Transform (FFT), and parametric algorithms, which are based on parametric data modelling, have been developed for 1-D as well as N-D frequency estimation. In the 1-D case, the performance of FFT-based estimation is bounded by the resolution limit of Fourier transform ($2\pi/M$), which is decided by the sample size (M). The parametric algorithms, by exploiting data structures, can offer superior performance and better resolution over non-parametric methods. The estimation error of these methods can be arbitrarily small as long as the number of snapshots is sufficiently large or the noise power is sufficiently small, which is not shared by FFT-based methods. Therefore, these algorithms are often called high resolution methods. The MUSIC algorithm [19] and the ESPRIT algorithm [3] are two well-known examples of such high resolution algorithms for 1-D frequency estimation. Although these algorithms are effective for 1-D frequency estimation, extensions to 2-D and N-D cases are nontrivial.

The main challenge of N-D frequency estimation is N-D frequency association or pairing. In practice, the frequencies are easy to be resolved separately in each dimension, but it is difficult to effectively associate the N-D frequencies of the same component. If N-D MUSIC [20] is implemented by N-D grid search, no frequency association step is needed, but N-D grid search is very complex. Alternatively, 1-D frequency estimation can be implemented in individual dimensions of the N-D mixture followed by a frequency pairing step [21, 22, 14]. For 2-D frequency estimation, there are some methods to pair the two sets of the frequencies such as comparing ratios or sums of the two sets of frequencies [22]. However, when the problem is generalized to the N-D case, the frequencies in all the other dimensions are often paired with those of a single dimension, which is clearly not optimal. For example, in the RARE algorithm [14], all the frequencies are paired with the first dimension, whose frequencies are assumed to be well separated.

A better method to pair N-D frequencies is to resort to joint diagonalization techniques. This method attempts to simultaneously diagonalize N matrices, whose eigenvalues are the estimates of the N-D frequencies. Various subspace algorithms [5, 9, 13, 23] can be classified into this category. For example, the JADE algorithm [9] and the N-D Unitary ESPRIT algorithm [13] use an iterative Jacobi-type approach to implement joint diagonalization. Some other techniques can also be considered as variants of the joint diagonalization technique. For example, the 2-D Unitary ESPRIT [5] diagonalizes a complex matrix to obtain the eigenvalues of two real matrices constructed from a unitary transformation. The 2-D ESPRIT algorithm [23] attempts to diagonalize a weighted matrix constructed from two matrices using randomly generated weighting factors.

Since most existing subspace-based N-D frequency estimation algorithms attempt to resolve the N-D frequencies from eigenvalues, a frequency pairing step or joint diagonalization process is necessary to associate the N-D frequencies of the same components. However, the computational complexity of this step is often high. For example, the computational order the frequency pairing step of the Matrix Enhancement and Matrix Pencil (MEMP) algorithm [21] is $\mathcal{O}(MF^3)$. Therefore, other approaches that have lower computational complexity are desired. In [24], an algebraic algorithm, namely the *Multidimensional Folding* (MDF) algorithm, is proposed to estimate the N-D frequencies from eigenvectors. Compared to the eigenvalue-based algorithms, the MDF algorithm achieves automatic pairing once an eigenvalue decomposition (EVD) is solved, therefore it does not require an extra frequency pairing step. This dissertation develops several novel eigenvector-based N-D frequency estimation algorithms and study their identifiability and performance in a systematic way.

1.3 Identifiability Bound

The identifiability (ID) condition of an algorithm is the sufficient condition under which the algorithm gives a unique solution to an estimation problem in the noiseless case. The algorithm identifiability may be different from the model identifiability. For N-D frequency estimation, the maximum number of resolvable frequencies for a given sample size in the noiseless case is referred to as the ID bound. The higher the ID bound is, the more relaxed the sufficient condition is. The ID condition of N-D frequency estimation is of interest in both theoretical exploration and engineering practice. In practice, the ID bound is of importance in situations where data samples come at a premium, e.g., in spatial sampling for direction estimation using an antenna array, where the number of samples in space domain has a high cost, and in time sampling for time-varying parameter estimation in wireless channel sounding, where only a few samples may be obtained in a time segment with constant parameters.

In a deterministic sense, for example, for 2-D frequency estimation, the ID bounds in [25] and [26] are constrained by $\min(M_1, M_2)$, where M_1 and M_2 are the sample sizes along the two dimensions. It is shown in [27] that the maximum number of identifiable frequencies is proportional to $M_1 + M_2$. Furthermore, if it is assumed that the N-D frequencies are drawn from a continuous distribution, it is shown in [24] that approximately $M_1 M_2/4$ 2-D frequencies are uniquely resolvable almost surely (i.e. with probability one), which is the most relaxed ID bound before the work in this dissertation. The MDF algorithm forms the constructive proof of this statistical

ID result [24].

One key technique to improve identifiability and performance is data smoothing. Several similar N-D smoothing techniques are employed in different frequency estimation algorithms. For example, the MEMP algorithm [21] constructs an enhanced data matrix using 2-D smoothing technique. The Unitary ESPRIT algorithm [13] employs an N-D smoothing preprocess step to construct a large smoothed data matrix. The 2-D MDF algorithm has a multidimensional smoothing step, which can be visualized as first expanding the 2-D data to a higher-dimensional structure, then folding back this structure into a matrix [24].

1.4 Performance Analysis

In the noisy case, the performance of an algorithm is its ability to combat against noise (i.e. performance demonstrates how accurate the algorithm can estimate). The performance is often measured by the mean square error (MSE) of the estimates. The Cramér Rao bound (CRB) indicates the performance bound of all unbiased estimators. Although the performance of an N-D frequency estimation algorithm can be evaluated by comparing MSE to CRB using Monte Carlo simulations, it is desirable to analyze its performance theoretically since analytical results are more rigorous and general.

There are extensive performance analysis results for 1-D ESPRIT [28] and MUSIC [19], but performance analysis for N-D frequency estimation algorithms is relatively limited in the literature. The MEMP algorithm is analyzed in [29]. However, since the frequency pairing step is difficult to analyze in closed forms, the performance analysis in [29] assumes that perfect pairing is always achieved, which is not the case in practice. Although the performance of a joint diagonalization process can be approximately analyzed, there are few analytic results for the Unitary ESPRIT algorithm using simultaneous Schur decomposition.

Performance analysis is also desirable to develop theoretically robust algorithms. Firstly, performance analysis is necessary to understand some behaviors of an algorithm. For example, although the MDF algorithm demonstrate superior performance in Monte Carlo simulations in most cases, it is found that sometimes its performance is not satisfying for certain distributions of the N-D frequencies. Secondly, performance analysis is required to optimize parameter configuration in some algorithms. For example, the 2-D ESPRIT algorithm [23] uses weighting factors to diagonalize two matrices simultaneously, but it is unknown how to choose the weighting factors so as to improve the performance of that algorithm. The RARE algorithm [14] also use weighting factors to construct a mixed polynomial in order to associate N-D frequencies. However, the impact of these factors on the performance is only analyzed using computer simulations [14], and no analytic results about the estimation errors are given.

1.5 Main Contributions

This dissertation develops a framework on eigenvector-based N-D frequency estimation, which includes several novel N-D frequency estimation algorithms. These algorithms are shown to have more relaxed ID bound and better performance than existing subspace algorithms. The main contributions of this dissertation are listed as follows.

(1) The ID condition of proposed eigenvector-based 2-D and N-D frequency estimation algorithms is given. Those algorithms have the most relaxed statistical ID bound up to date. This result is summarized in Chapter 2.

(2) Using matrix perturbation theory, Chapter 3 analyzes the performance of the proposed eigenvector-based algorithms and validates the theoretical results by computer experiments. In sufficient snapshot case, it is proved that the eigenvector-based frequency estimation algorithms are asymptotically consistent estimators with respect to the number of snapshots and noise power [30].

(3) Chapter 4 proposes to use weighting factors to obtain a better estimate

of the eigenvector matrix. By introducing adaptive weighting factors, this method attempts to perform EVD to an optimal weighted matrix so that the dispersiveness of the eigenvalues in the EVD is as large as possible. The simulation results demonstrate that the N-D frequency estimation algorithms based on weighted diagonalization can offer superior performance over existing algorithms [31, 32].

(4) Chapter 5 applies the aforementioned results in wireless communication to estimate and predict the time-varying multipath channel in adaptive OFDM systems. Furthermore, Chapter 6 proposes a fast 2-D frequency estimation algorithm and apply it in joint 2-D DOA tracking of multiple moving targets.

1.6 Problem Statement of Multidimensional Frequency Estimation

In this section, the multidimensional frequency estimation problem is stated again in matrix forms. The introduction is started from the 2-D case, then extended to the N-D case. The notations and abbreviations used in this dissertation are defined in Appendix D.

1.6.1 2-D Frequency Estimation from one snapshot

Given (1.1), one can define two Vandermonde matrices: $\boldsymbol{A} \in \mathbb{C}^{M_1 \times F}$ and $\boldsymbol{B} \in \mathbb{C}^{M_2 \times F}$, with generators $e^{j\omega_{f,1}}$ and $e^{j\omega_{f,2}}$, $f = 1, \ldots, F$, respectively, then the 2-D mixture in (1.1) can be written in matrix form as

$$\boldsymbol{X} = \boldsymbol{A}\boldsymbol{D}(\boldsymbol{c})\boldsymbol{B}^T + \boldsymbol{W}, \tag{1.4}$$

where $\boldsymbol{c} = [c_1, c_2, \ldots, c_F]^T$ and $\boldsymbol{W}$ is the corresponding noise matrix. Eqn. (1.4) can also be written in vector form. For example, let

$$\boldsymbol{x} = \left[x_{1,1} \;\; x_{1,2} \;\; \cdots \;\; x_{1,M_2} \;\; x_{2,1} \;\; \cdots \;\; x_{M_1,M_2}\right]^T,$$

then it can be verified that

$$\boldsymbol{x} = (\boldsymbol{A} \odot \boldsymbol{B})\boldsymbol{c} + \boldsymbol{w}, \tag{1.5}$$

where $\boldsymbol{w}$ is a noise vector, which is assumed to be complex Gaussian with zero mean and a covariance matrix $\sigma^2 \boldsymbol{I}_{M_1 M_2}$. Here $\odot$ stands for the Khatri-Rao product of two matrices, which is defined as the column-wise Kronecker product such that

$$[\boldsymbol{a}_1\ \boldsymbol{a}_2\ \cdots\ \boldsymbol{a}_F] \odot [\boldsymbol{b}_1\ \boldsymbol{b}_2\ \cdots\ \boldsymbol{b}_F] = [\boldsymbol{a}_1 \otimes \boldsymbol{b}_1\ \boldsymbol{a}_2 \otimes \boldsymbol{b}_2\ \cdots\ \boldsymbol{a}_F \otimes \boldsymbol{b}_F].$$

In the noiseless case, $\boldsymbol{w} = \boldsymbol{0}$, then

$$\boldsymbol{x} = (\boldsymbol{A} \odot \boldsymbol{B})\boldsymbol{c}. \tag{1.6}$$

The identifiability conditions of proposed algorithms are investigated in the noiseless case and their performance is analyzed in the noisy case.

1.6.2 N-D Frequency estimation from a single snapshot

The data model in scalar form is given in (1.2). Given (1.2), define the sample vector $\boldsymbol{x}$ for the t-th snapshot as

$$\boldsymbol{x} = \left[x_{1,1,\cdots,1}\ x_{1,1,\ldots,2}\ \cdots\ x_{1,1,\ldots,M_N}\ x_{1,1,\ldots,2,1}\ \cdots\ x_{M_1,M_2,\ldots,M_N}\right]^T. \tag{1.7}$$

Furthermore, define N Vandermonde matrices $\boldsymbol{A}_n \in \mathbb{C}^{M_n \times F}$ with generators $\{e^{j\omega_{f,n}}\}_{f=1}^{F}$ such that

$$\boldsymbol{A}_n := [\boldsymbol{a}_{1,n}\ \boldsymbol{a}_{2,n}\ \cdots\ \boldsymbol{a}_{F,n}], \text{ where } \boldsymbol{a}_{f,n} = \left[1\ e^{j\omega_{f,n}}\ \cdots\ e^{j(M_n-1)\omega_{f,n}}\right]^T, \quad n = 1, \ldots, N.$$

It can be verified that

$$\boldsymbol{x} = \boldsymbol{A}\boldsymbol{c} + \boldsymbol{w}, \tag{1.8}$$

where $\boldsymbol{w}$ is the noise vector, and

$$\boldsymbol{A} := \boldsymbol{A}_1 \odot \boldsymbol{A}_2 \odot \cdots \odot \boldsymbol{A}_N, \qquad \boldsymbol{c} := \left[c_1\ c_2\ \cdots\ c_F\right]^T.$$

In the noiseless case, $\boldsymbol{w} = \boldsymbol{0}$, the model becomes

$$\boldsymbol{x} = \boldsymbol{A}\boldsymbol{c}. \tag{1.9}$$

1.6.3 N-D Frequency Estimation from multiple snapshots

The data model in scalar form is given in (1.3). Given (1.3), define the sample vector $\boldsymbol{x}(t)$ for the t-th snapshot as

$$\boldsymbol{x}(t) = \left[x_{1,1,\cdots,1}(t)\ x_{1,1,\ldots,2}(t)\ \cdots\ x_{1,1,\ldots,M_N}(t)\ x_{1,1,\ldots,2,1}(t)\ \cdots\ x_{M_1,M_2,\ldots,M_N}(t)\right]^T. \tag{1.10}$$

It can be verified that

$$\boldsymbol{x}(t) = \boldsymbol{A}\boldsymbol{c}(t) + \boldsymbol{w}(t), \quad t = 1, \ldots, T, \tag{1.11}$$

where $\boldsymbol{w}(t)$ is the noise vector, and $\boldsymbol{c}(t) := \left[c_1(t)\ c_2(t)\ \cdots\ c_F(t)\right]^T$. Define

$$\begin{aligned}\boldsymbol{X} &:= \left[\boldsymbol{x}(1)\ \boldsymbol{x}(2)\ \cdots\ \boldsymbol{x}(T)\right] \in \mathbb{C}^{M \times T},\\ \boldsymbol{C} &:= \left[\boldsymbol{c}(1)\ \boldsymbol{c}(2)\ \cdots\ \boldsymbol{c}(T)\right] \in \mathbb{C}^{F \times T},\end{aligned}$$

then the data model in (1.11) can be rewritten in matrix form as

$$\boldsymbol{X} = \boldsymbol{A}\boldsymbol{C} + \boldsymbol{W}, \tag{1.12}$$

where $\boldsymbol{W}$ is the corresponding noise matrix. In the noiseless case, $\boldsymbol{W} = \boldsymbol{0}$. Notice that the same data models for the case of multiple snapshots have been used in [13, 14].

CHAPTER 2

THE IMDF ALGORITHM

This chapter presents the Improved Multi-Dimensional Folding (IMDF) algorithm for multidimensional frequency estimation. It offers the most relaxed identifiability up to date. The introduction is started from the 2-D single snapshot case. Then, the result is extended to the N-D multiple snapshot case.

2.1 The IMDF algorithm

For convenience, the IMDF algorithm is developed with a noiseless data model. The IMDF algorithm gives the most relax identifiability bound for the multidimensional frequency estimation problem. Its identifiability bound is discussed in Section 2.2.

2.1.1 Some Properties of the Khatri-Rao Products

It can be seen from the matrix models of multidimensional frequency estimation in (1.5) and (1.8) that the Khatri-Rao product of Vandermonde matrices plays a key role in the modelling of multidimensional frequency mixture. In the following, some relevant properties of the Khatri-Rao product are reviewed briefly.

p1) The Khatri-Rao product of a matrix $\boldsymbol{A}$ and a row vector $\boldsymbol{b}^T$ is interchangeable,

$$\boldsymbol{A} \odot \boldsymbol{b}^T = \boldsymbol{b}^T \odot \boldsymbol{A} = \boldsymbol{A}\boldsymbol{D}(\boldsymbol{b}).$$

p2) Another property, which can be found in [34], is $(\boldsymbol{A}\otimes\boldsymbol{B})(\boldsymbol{C}\odot\boldsymbol{D}) = (\boldsymbol{A}\boldsymbol{C})\odot(\boldsymbol{B}\boldsymbol{D})$.

It can be extended into multiple products:

$$(\boldsymbol{A}_1 \otimes \cdots \otimes \boldsymbol{A}_n)(\boldsymbol{B}_1 \cdots \odot \boldsymbol{B}_n) = (\boldsymbol{A}_1\boldsymbol{B}_1) \odot \cdots \odot (\boldsymbol{A}_n\boldsymbol{B}_n).$$

p3) The following property establishes the almost sure full rank of the Khatri-Rao product of multiple Vandermonde matrices, and a proof can be found in [35]. Given a set of Vandermonde matrices $\boldsymbol{A}_n \in \mathbb{C}^{M_n \times F}$, $n = 1, \ldots, N \geq 2$, let

$$\boldsymbol{A} = \boldsymbol{A}_1 \odot \boldsymbol{A}_2 \odot \cdots \odot \boldsymbol{A}_N, \qquad \text{and } M = \prod_{n=1}^{n=N} M_n$$

then $\boldsymbol{A}$ is almost surely (a.s.) full rank, i.e., $r_{\mathbf{A}} = \min(M, F)$, provided that the NF generators of $\boldsymbol{A}_n$, $n = 1, \ldots, N$, are drawn from a distribution that is continuous with respect to the Lebesgue measure in $\mathbb{C}^{NF}$.

2.1.2 The 2-D IMDF Algorithm

In the noiseless case, the data model takes the forms of (1.6). The objective here is to retrieve (ω_f, ν_f, c_f) for $f = 1, \cdots, F$ from $\boldsymbol{X}$. Without loss of generality, one can assume that $\boldsymbol{X}$ in (1.4) is a tall or square matrix, i.e., $M_1 \geq M_2$. Otherwise one can take the transpose of $\boldsymbol{X}$ to switch M_1 and M_2. The following lemma is needed later.

Lemma 1: Given M_1 and M_2, define the following selection matrices

$$\boldsymbol{J}_{l_1,l_2} := \left[\mathbf{0}_{K_1 \times (l_1-1)}\ \boldsymbol{I}_{K_1}\ \mathbf{0}_{K_1 \times (L_1-l_1)}\right] \otimes \left[\mathbf{0}_{K_2 \times (l_2-1)}\ \boldsymbol{I}_{K_2}\ \mathbf{0}_{K_2 \times (L_2-l_2)}\right] \tag{2.1}$$

for $1 \leq l_1 \leq L_1$, $1 \leq l_2 \leq L_2$, where $\boldsymbol{J}_{l_1,l_2}$ is of size $K_1K_2 \times M_1M_2$, and K_i and L_i, for $i = 1, 2$, are positive integers such that

$$K_1 + L_1 = M_1 + 1, \quad K_2 + L_2 = M_2 + 1. \tag{2.2}$$

Further define a 2-D smoothing operator $\mathcal{S}$ for the measurement vector $\boldsymbol{x}$ in (1.5) as

$$\mathcal{S}(\boldsymbol{x}) := \left[\boldsymbol{J}_{1,1}\boldsymbol{x}\ \cdots\ \boldsymbol{J}_{1,L_2}\boldsymbol{x}\ \boldsymbol{J}_{2,1}\boldsymbol{x}\ \cdots\ \boldsymbol{J}_{2,L_2}\boldsymbol{x}\ \cdots\ \boldsymbol{J}_{L_1,L_2}\boldsymbol{x}\right]$$

then it can be verified that

$$\mathcal{S}(\boldsymbol{x}) = \left(\boldsymbol{A}^{(K_1)} \odot \boldsymbol{B}^{(K_2)}\right) \boldsymbol{D}(\boldsymbol{c}) \left(\boldsymbol{A}^{(L_1)} \odot \boldsymbol{B}^{(L_2)}\right)^T. \tag{2.3}$$

Proof: See Appendix A. ■

Suppose that $\boldsymbol{x}$ is a snapshot of 2-D observed data in (1.11), then the smoothed data matrix $\mathcal{S}(\boldsymbol{x})$ is equivalent to

$$\mathcal{S}(\boldsymbol{x}) = \begin{bmatrix} \boldsymbol{X}_1 & \boldsymbol{X}_2 & \cdots & \boldsymbol{X}_{L_1} \\ \boldsymbol{X}_2 & \boldsymbol{X}_3 & \cdots & \boldsymbol{X}_{L_1+1} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{X}_{K_1} & \boldsymbol{X}_{K_1+1} & \cdots & \boldsymbol{X}_{M_1} \end{bmatrix}, \text{ where } \boldsymbol{X}_k = \begin{bmatrix} x_{k,1} & x_{k,2} & \cdots & x_{k,L_2} \\ x_{k,2} & x_{k,3} & \cdots & x_{k,L_2+1} \\ \vdots & \vdots & \ddots & \vdots \\ x_{k,K_2} & x_{k,K_2+1} & \cdots & x_{k,M_2} \end{bmatrix}.$$

Lemma 1 claims that $\mathcal{S}(\boldsymbol{x})=\left(\boldsymbol{A}^{(K_1)} \odot \boldsymbol{B}^{(K_2)}\right) \boldsymbol{D}(\boldsymbol{c}) \left(\boldsymbol{A}^{(L_1)} \odot \boldsymbol{B}^{(L_2)}\right)^T$.

Now one can begin the development the 2-D IMDF algorithm. The idea is to first use data smoothing extensively to enlarge the sample matrix size, and then exploit eigenvector structure for frequency estimation. One can define two selection matrices

$$\boldsymbol{J}_1 = \left[\boldsymbol{I}_{M_1-1}\ \boldsymbol{0}_{(M_1-1)\times 1}\right] \tag{2.4}$$

$$\boldsymbol{J}_2 = \left[\boldsymbol{0}_{(M_1-1)\times 1}\ \boldsymbol{I}_{M_1-1}\right]. \tag{2.5}$$

Due to the shift invariance property of Vandermonde matrices, one can have

$$\boldsymbol{x}_1 := (\boldsymbol{J}_1 \otimes \boldsymbol{I}_{M_2})\,\boldsymbol{x} = \left(\boldsymbol{A}^{(M_1-1)} \odot \boldsymbol{B}\right) \boldsymbol{c} \tag{2.6}$$

$$\boldsymbol{x}_2 := (\boldsymbol{J}_2 \otimes \boldsymbol{I}_{M_2})\,\boldsymbol{x} = \left(\boldsymbol{A}^{(M_1-1)} \odot \boldsymbol{B}\right) \boldsymbol{D}(\boldsymbol{\omega})\boldsymbol{c} \tag{2.7}$$

where $\boldsymbol{x}$ is given in (1.5), and $\boldsymbol{\omega} := \left[e^{j\omega_1},\ e^{j\omega_2},\ \cdots,\ e^{j\omega_F}\right]^T$. One can apply the 2-D smoothing operator $\mathcal{S}$ defined in Lemma 1 to $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$. Since the sizes of both $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ are $(M_1-1)M_2 \times 1$, here the integers in (2.2) should be chosen such that

$$K_1 + L_1 = M_1, \quad K_2 + L_2 = M_2 + 1. \tag{2.8}$$

Applying the 2-D smoothing operator, one can obtain

$$\boldsymbol{X}_{1,\mathcal{S}} := \mathcal{S}(\boldsymbol{x}_1) = \boldsymbol{G}\boldsymbol{D}(\boldsymbol{c})\boldsymbol{H}^T \tag{2.9}$$

$$\boldsymbol{X}_{2,\mathcal{S}} := \mathcal{S}(\boldsymbol{x}_2) = \boldsymbol{G}\boldsymbol{D}(\boldsymbol{c})\boldsymbol{D}(\boldsymbol{\omega})\boldsymbol{H}^T \tag{2.10}$$

where

$$\boldsymbol{G} := \boldsymbol{A}^{(K_1)} \odot \boldsymbol{B}^{(K_2)}, \qquad \boldsymbol{H} := \boldsymbol{A}^{(L_1)} \odot \boldsymbol{B}^{(L_2)}. \tag{2.11}$$

To further explore the data structure, one can perform the backward smoothing on the data vector $\boldsymbol{x}$ in (1.5). Define

$$\boldsymbol{y} := \boldsymbol{\Pi}\boldsymbol{x}^* = (\boldsymbol{A} \odot \boldsymbol{B})\widetilde{\boldsymbol{c}} \tag{2.12}$$

where $\boldsymbol{\Pi}$ is an $M_1M_2 \times M_1M_2$ backward permutation matrix with ones on its antidiagonal, and $\widetilde{\boldsymbol{c}} = [\widetilde{c}_1, \widetilde{c}_2, \cdots, \widetilde{c}_F]^T$, with $\widetilde{c}_f = c_f^* e^{-j(M_1-1)\omega_f - j(M_2-1)\nu_f}$. Applying the same technique to $\boldsymbol{y}$ that is used to obtain $\boldsymbol{X}_{1,\mathcal{S}}$ and $\boldsymbol{X}_{2,\mathcal{S}}$ from $\boldsymbol{x}$, one can obtain

$$\boldsymbol{y}_1 := (\boldsymbol{J}_1 \otimes \boldsymbol{I}_{M_2})\boldsymbol{y} = \left(\boldsymbol{A}^{(M_1-1)} \odot \boldsymbol{B}\right)\widetilde{\boldsymbol{c}} \tag{2.13}$$

$$\boldsymbol{y}_2 := (\boldsymbol{J}_2 \otimes \boldsymbol{I}_{M_2})\boldsymbol{y} = \left(\boldsymbol{A}^{(M_1-1)} \odot \boldsymbol{B}\right)\boldsymbol{D}(\boldsymbol{\omega})\widetilde{\boldsymbol{c}} \tag{2.14}$$

$$\boldsymbol{Y}_{1,\mathcal{S}} := \mathcal{S}(\boldsymbol{y}_1) = \boldsymbol{G}\boldsymbol{D}(\widetilde{\boldsymbol{c}})\boldsymbol{H}^T \tag{2.15}$$

$$\boldsymbol{Y}_{2,\mathcal{S}} := \mathcal{S}(\boldsymbol{y}_2) = \boldsymbol{G}\boldsymbol{D}(\widetilde{\boldsymbol{c}})\boldsymbol{D}(\boldsymbol{\omega})\boldsymbol{H}^T. \tag{2.16}$$

If one defines the following matrices

$$\boldsymbol{Z}_1 := \begin{bmatrix} \boldsymbol{X}_{1,\mathcal{S}} \\ \boldsymbol{Y}_{1,\mathcal{S}} \end{bmatrix}, \quad \boldsymbol{Z}_2 := \begin{bmatrix} \boldsymbol{X}_{2,\mathcal{S}} \\ \boldsymbol{Y}_{2,\mathcal{S}} \end{bmatrix}, \tag{2.17}$$

then he has

$$\boldsymbol{Z} := \begin{bmatrix} \boldsymbol{Z}_1 \\ \boldsymbol{Z}_2 \end{bmatrix} = \begin{bmatrix} \boldsymbol{P}_1 \\ \boldsymbol{P}_1\boldsymbol{D}(\boldsymbol{\omega}) \end{bmatrix} \boldsymbol{H}^T = \boldsymbol{P}\boldsymbol{H}^T, \tag{2.18}$$

where the size of $\boldsymbol{Z}$ is $4K_1K_2 \times L_1L_2$, and $\boldsymbol{P}_1$ and $\boldsymbol{P}$ are defined as

$$\boldsymbol{P}_1 := \begin{bmatrix} \boldsymbol{c}^T \\ \widetilde{\boldsymbol{c}}^T \end{bmatrix} \odot \boldsymbol{G}, \quad \boldsymbol{P} := \begin{bmatrix} \boldsymbol{P}_1 \\ \boldsymbol{P}_1\boldsymbol{D}(\boldsymbol{\omega}) \end{bmatrix}. \tag{2.19}$$

Since both $\boldsymbol{P}_1$ and $\boldsymbol{H}$ are Khatri-Rao products of Vandermonde matrices, invoking Property p3), when $2K_1K_2 \geq F$ and $L_1L_2 \geq F$, $\boldsymbol{P}_1$ and $\boldsymbol{H}$ are almost surely full column rank. Hence $\boldsymbol{Z}_1$ and $\boldsymbol{Z}_2$ are of rank F almost surely, and the singular value decomposition of $\boldsymbol{Z}$ yields

$$\boldsymbol{Z} = \boldsymbol{U}_s \boldsymbol{\Sigma}_s \boldsymbol{V}_s^H \tag{2.20}$$

where $\boldsymbol{U}_s$ has F columns which together span the column space of $\boldsymbol{Z}$. Since the same space is spanned by the columns of $\boldsymbol{P}$, then there exists an $F \times F$ nonsingular matrix $\boldsymbol{T}^{-1}$ such that

$$\boldsymbol{U}_s = \begin{bmatrix} \boldsymbol{U}_1 \\ \boldsymbol{U}_2 \end{bmatrix} = \boldsymbol{P}\boldsymbol{T}^{-1} \tag{2.21}$$

where $\boldsymbol{U}_s$ is divided into two equal-size sub-matrices $\boldsymbol{U}_1$ and $\boldsymbol{U}_2$. Therefore $\boldsymbol{U}_1^{\dagger}\boldsymbol{U}_2 = \boldsymbol{T}\boldsymbol{D}(\boldsymbol{\omega})\boldsymbol{T}^{-1}$. Assuming that the elements of $\boldsymbol{\omega}$ are distinct, $\boldsymbol{T}$ can be obtained from the eigenvalue decomposition of $\boldsymbol{U}_1^{\dagger}\boldsymbol{U}_2$ up to column scaling and permutation ambiguity, i.e., the eigenvalue decomposition of $\boldsymbol{U}_1^{\dagger}\boldsymbol{U}_2$ gives

$$\boldsymbol{U}_1^{\dagger}\boldsymbol{U}_2 = \boldsymbol{T}_{sp}\boldsymbol{D}(\boldsymbol{\omega})\boldsymbol{T}_{sp}^{-1} \tag{2.22}$$

where $\boldsymbol{T}_{sp} = \boldsymbol{T}\boldsymbol{\Lambda}\boldsymbol{\Delta}$, and $\boldsymbol{\Lambda}$ is a nonsingular diagonal column scaling matrix and $\boldsymbol{\Delta}$ is a permutation matrix. Once obtain $\boldsymbol{T}_{sp}$ is obtained, one can retrieve $\boldsymbol{P}$ and $\boldsymbol{H}$ up to column scaling and permutation using (c.f., (2.21) and (2.18))

$$\boldsymbol{P}_{sp} = \boldsymbol{U}_s\boldsymbol{T}_{sp} = \boldsymbol{P}\boldsymbol{\Lambda}\boldsymbol{\Delta} \tag{2.23}$$

$$\boldsymbol{H}_{sp} = \left(\boldsymbol{T}_{sp}^{-1}\boldsymbol{\Sigma}_s\boldsymbol{V}_s{}^H\right)^T = \boldsymbol{H}\boldsymbol{\Lambda}^{-1}\boldsymbol{\Delta}. \tag{2.24}$$

In (2.24), notice that $\boldsymbol{\Delta}^{-1} = \boldsymbol{\Delta}^T$. One can retrieve ω_f and ν_f from the columns of $\boldsymbol{P}_{sp}$ and $\boldsymbol{H}_{sp}$ by exploiting the structure of the two matrices as explained in the following. The permutation is not an issue here, however, because ω_f and ν_f appear in the same column of $\boldsymbol{P}_{sp}$, as well as in $\boldsymbol{H}_{sp}$, thus are automatically paired; and arbitrary nonzero column scaling is immaterial, because the frequencies can be obtained by dividing suitably chosen elements of the aforementioned column. Therefore the column scaling and permutation will not have material effect on the algorithm, for notation simplicity in the following one can drop the subscript $_{sp}$ of $\boldsymbol{T}_{sp}$, $\boldsymbol{P}_{sp}$, and $\boldsymbol{H}_{sp}$ as long as it is clear from the context that there may exist column-scaling and permutation. Suppose $e^{j\omega_f}$ and $e^{j\nu_f}$ appear in the f-th column of $\boldsymbol{P}$, then $e^{j\omega_f}$ can be retrieved by any one of the following equations in the noiseless case:

$$e^{j\omega_f} = \frac{p_{n,f}}{p_{n-K_2,f}}, \qquad \text{mod}\ (n-1, K_1K_2) \geq K_2 \tag{2.25}$$

similarly, $e^{j\nu_f}$ can be retrieved by any one of the following

$$e^{j\nu_f} = \frac{p_{n,f}}{p_{n-1,f}}, \qquad \text{mod}\ (n-1, K_2) \geq 1. \tag{2.26}$$

Notice that $e^{j\omega_f}$ and $e^{j\nu_f}$ are automatically paired since they are obtained from the same column of $\boldsymbol{P}$.

If the data observations are noisy, the observation vector becomes (1.5), and $\boldsymbol{x}$ in the above algorithm should be replaced by $\widehat{\boldsymbol{x}}$. The estimates of $\boldsymbol{P}$ and $\boldsymbol{H}$ are denoted as $\widehat{\boldsymbol{P}}$ and $\widehat{\boldsymbol{H}}$ respectively. In this case, one can take advantage of the rich structure of the Khatri-Rao product of Vandermonde matrices, and use average to obtain the frequency estimates, thus reduce the variance of estimation error. In fact, $e^{j\omega_f}$ can be estimated by

$$\widehat{e^{j\omega_f}} = \frac{1}{\gamma_1}\left[\sum_{\substack{n=K_2+1\\ \text{mod}(n-1,K_1K_2)\geq K_2}}^{4K_1K_2} \frac{\widehat{p}_{n,f}}{\widehat{p}_{n-K_2,f}} + \sum_{n=L_2+1}^{L_1L_2} \frac{\widehat{h}_{n,f}}{\widehat{h}_{n-K_2,f}}\right] \tag{2.27}$$

where $\gamma_1 = 4(K_1-1)K_2 + (L_1-1)L_2$. Similarly, $e^{j\nu_f}$ can be estimated by

$$\widehat{e^{j\nu_f}} = \frac{1}{\gamma_2}\left[\sum_{\substack{n=2\\ \text{mod}(n-1,K_2)\geq 1}}^{4K_1K_2} \frac{\widehat{p}_{n,f}}{\widehat{p}_{n-1,f}} + \sum_{\substack{n=2\\ \text{mod}(n-1,L_2)\geq 1}}^{L_1L_2} \frac{\widehat{h}_{n,f}}{\widehat{h}_{n-1,f}}\right] \tag{2.28}$$

where $\gamma_2 = 4K_1(K_2-1) + L_1(L_2-1)$. The 2-D frequency estimates can be obtained by

$$\widehat{\omega}_f = \mathcal{I}\left(\log \widehat{e^{j\omega_f}}\right), \qquad \widehat{\nu}_f = \mathcal{I}\left(\log \widehat{e^{j\nu_f}}\right). \tag{2.29}$$

TABLE 2.1

The 2-D IMDF algorithm

1. Form $\widehat{\boldsymbol{x}}_1$, $\widehat{\boldsymbol{x}}_2$ from $\widehat{\boldsymbol{x}}$ using (2.6) and (2.7), and form $\widehat{\boldsymbol{X}}_{1,\mathcal{S}}$ and $\widehat{\boldsymbol{X}}_{2,\mathcal{S}}$ using (2.9) and (2.10). From $\widehat{\boldsymbol{y}}$ from $\widehat{\boldsymbol{x}}$ using (2.12), and construct $\widehat{\boldsymbol{Y}}_{1,\mathcal{S}}$ and $\widehat{\boldsymbol{Y}}_{2,\mathcal{S}}$ using (2.15) and (2.16). Form $\widehat{\boldsymbol{Z}}$ using (2.18).
2. Compute the SVD of $\widehat{\boldsymbol{Z}}$. Partition the F principle left singular vectors (i.e. $\widehat{\boldsymbol{U}}$) into two equal-sized matrices $\widehat{\boldsymbol{U}}_1$ and $\widehat{\boldsymbol{U}}_2$ according to (2.21).
3. Compute the set of eigenvectors (i.e., $\widehat{\boldsymbol{T}}$) of $\widehat{\boldsymbol{U}}_1^{\dagger}\widehat{\boldsymbol{U}}_2$, and obtain estimates of $\widehat{\boldsymbol{P}}$ and $\widehat{\boldsymbol{H}}$ using (2.23).
4. Use (2.27) and (2.28) to calculate $\widehat{e^{j\omega_f}}$ and $\widehat{e^{j\nu_f}}$. The frequency estimates are then obtained by (2.29).

Notice that the method used to compute the frequency through average in (2.27) and (2.28) is in fact the circular mean method in directional statistics [36]. After the frequency estimates are obtained, the amplitude $\boldsymbol{c}$ can be obtained by solving (1.5) using a Least Squares approach. The 2-D IMDF algorithm for frequency estimation from noisy data is summarized in Table 2.1.

2.1.3 The N-D IMDF Algorithm

The 2-D IMDF algorithm can be extended to N-D frequency estimation. In the N-D case, the data model is given in (1.2) or equivalently (1.11). Without loss of generality, one can assume that $M_1 \geq M_n$, for $n \neq 1$, otherwise a simple permutation on (1.11) would make this true. Corresponding to the 2-D smoothing operator in Lemma 1, one can define an N-D smoothing operator[1] as follows.

---

[1] Notice that a similar smoothing operator is used in [13] but its connection to the Khatri-Rao products is not explored.

Lemma 2: Given M_n, $n = 1, \ldots, N$, one can define a set of selection matrices

$$\boldsymbol{J}_{l_n}^{K_n} := \begin{bmatrix}\boldsymbol{0}_{K_n \times (l_n - 1)} & \boldsymbol{I}_{K_n} & \boldsymbol{0}_{K_n \times (L_n - l_n)}\end{bmatrix} \quad : K_n \times M_n$$

$$\boldsymbol{J}_{l_1, l_2, \cdots, l_N} := \boldsymbol{J}_{l_1}^{K_1} \otimes \boldsymbol{J}_{l_2}^{K_2} \cdots \otimes \boldsymbol{J}_{l_N}^{K_N} \quad : \prod_{n=1}^{N} K_n \times \prod_{n=1}^{N} M_n$$

where $l_n = 1, \ldots, L_n$, and K_n and L_n are positive integers satisfying $K_n + L_n - 1 = M_n$, for $n = 1, \ldots, N$. An N-D smoothing operator for the measurement vector $\boldsymbol{x}$ in (1.11) is defined as

$$\mathcal{S}(\boldsymbol{x}) := \begin{bmatrix}\boldsymbol{J}_{1,1,\cdots,1}\boldsymbol{x} & \boldsymbol{J}_{1,1,\ldots,2}\boldsymbol{x} & \cdots & \boldsymbol{J}_{1,1,\ldots,L_N}\boldsymbol{x} & \boldsymbol{J}_{1,1,\ldots,2,1}\boldsymbol{x} & \cdots & \boldsymbol{J}_{L_1,L_2,\ldots,L_N-1}\boldsymbol{x} & \boldsymbol{J}_{L_1,L_2,\ldots,L_N}\boldsymbol{x}\end{bmatrix}$$

then it can be verified that

$$\boldsymbol{X}_{\mathcal{S}} = \mathcal{S}(\boldsymbol{x}) = \boldsymbol{G}\boldsymbol{D}(\boldsymbol{c})\boldsymbol{H}^T \tag{2.30}$$

where

$$\boldsymbol{G} := \boldsymbol{A}_1^{(K_1)} \odot \boldsymbol{A}_2^{(K_2)} \odot \cdots \odot \boldsymbol{A}_N^{(K_N)}, \quad \boldsymbol{H} := \boldsymbol{A}_1^{(L_1)} \odot \boldsymbol{A}_2^{(L_2)} \odot \cdots \odot \boldsymbol{A}_N^{(L_N)}.$$

Proof: See Appendix A. ■

Now one can begin the development of the N-D IMDF algorithm. Using the selection matrices defined in (2.4) and (2.5), one can construct $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ from $\boldsymbol{x}$ of (1.11) as

$$\boldsymbol{x}_1 := (\boldsymbol{J}_1 \otimes \boldsymbol{I}_{M_2} \cdots \otimes \boldsymbol{I}_{M_N})\,\boldsymbol{x} = \left(\boldsymbol{A}_1^{(M_1-1)} \odot \boldsymbol{A}_2 \odot \cdots \odot \boldsymbol{A}_N\right)\boldsymbol{c} \tag{2.31}$$

$$\boldsymbol{x}_2 := (\boldsymbol{J}_2 \otimes \boldsymbol{I}_{M_2} \cdots \otimes \boldsymbol{I}_{M_N})\,\boldsymbol{x} = \left(\boldsymbol{A}_1^{(M_1-1)} \odot \boldsymbol{A}_2 \odot \cdots \odot \boldsymbol{A}_N\right)\boldsymbol{D}(\boldsymbol{\omega}_1)\boldsymbol{c} \tag{2.32}$$

where

$$\boldsymbol{\omega}_1 = [e^{j\omega_{1,1}} \; \cdots \; e^{j\omega_{F,1}}]^T. \tag{2.33}$$

Then one can apply the N-D smoothing operator defined in Lemma 2 to (2.31) and (2.32) to construct the following matrices

$$\boldsymbol{X}_{1,\mathcal{S}} := \mathcal{S}(\boldsymbol{x}_1) = \boldsymbol{G}\boldsymbol{D}(\boldsymbol{c})\boldsymbol{H}^T \tag{2.34}$$

$$\boldsymbol{X}_{2,\mathcal{S}} := \mathcal{S}(\boldsymbol{x}_2) = \boldsymbol{G}\boldsymbol{D}(\boldsymbol{c})\boldsymbol{D}(\boldsymbol{\omega}_1)\boldsymbol{H}^T \tag{2.35}$$

where

$$\boldsymbol{G} = \boldsymbol{A}_1^{(K_1)} \odot \boldsymbol{A}_2^{(K_2)} \odot \cdots \odot \boldsymbol{A}_N^{(K_N)}, \quad \boldsymbol{H} = \boldsymbol{A}_1^{(L_1)} \odot \boldsymbol{A}_2^{(L_2)} \odot \cdots \odot \boldsymbol{A}_N^{(L_N)} \tag{2.36}$$

and K_n and L_n are positive integers subject to

$$K_1 + L_1 = M_1, \quad K_n + L_n = M_n + 1, \ n = 2, \cdots, N. \tag{2.37}$$

Furthermore one can perform forward-backward (FB) smoothing on the single-snapshot vector $\boldsymbol{x}(t)$ in (1.11). Define

$$\boldsymbol{y} := \boldsymbol{\Pi}_M \boldsymbol{x}^* = (\boldsymbol{A}_1 \odot \boldsymbol{A}_2 \cdots \odot \boldsymbol{A}_N)\tilde{\boldsymbol{c}}(t), \tag{2.38}$$

where $\boldsymbol{\Pi}_M$ is the backward permutation matrix of size $M \times M$, and $\tilde{\boldsymbol{c}}(t) = [\tilde{c}_1(t), \cdots, \tilde{c}_F(t)]^T$, with $\tilde{c}_f(t) = c_f^*(t)e^{-j\phi_f}$. Here ϕ_f is defined as

$$\phi_f = \sum_{n=1}^{N} \omega_{f,n}(M_n - 1). \tag{2.39}$$

Applying the same technique to $\boldsymbol{y}$ that is used to obtain $\boldsymbol{X}_{1,\mathcal{S}}$ and $\boldsymbol{X}_{2,\mathcal{S}}$ from $\boldsymbol{x}$, one can obtain

$$\boldsymbol{y}_1 := (\boldsymbol{J}_1 \otimes \boldsymbol{I}_{M_2} \cdots \otimes \boldsymbol{I}_{M_N})\boldsymbol{y} = (\boldsymbol{A}_1^{(M_1-1)} \odot \boldsymbol{A}_2^{(M_2)} \odot \cdots \odot \boldsymbol{A}_N^{(M_N)})\tilde{\boldsymbol{c}}, \tag{2.40}$$

$$\boldsymbol{y}_2 := (\boldsymbol{J}_2 \otimes \boldsymbol{I}_{M_2} \cdots \otimes \boldsymbol{I}_{M_N})\boldsymbol{y} = (\boldsymbol{A}_1^{(M_1-1)} \odot \boldsymbol{A}_2^{(M_2)} \odot \cdots \odot \boldsymbol{A}_N^{(M_N)})\boldsymbol{D}(\boldsymbol{\omega}_1)\tilde{\boldsymbol{c}}, \tag{2.41}$$

$$\boldsymbol{Y}_{1,\mathcal{S}} := \mathcal{S}(\boldsymbol{y}_1) = \boldsymbol{G}\boldsymbol{D}(\tilde{\boldsymbol{c}})\boldsymbol{H}^T, \tag{2.42}$$

$$\boldsymbol{Y}_{2,\mathcal{S}} := \mathcal{S}(\boldsymbol{y}_2) = \boldsymbol{G}\boldsymbol{D}(\tilde{\boldsymbol{c}})\boldsymbol{D}(\boldsymbol{\omega}_1)\boldsymbol{H}^T \tag{2.43}$$

The rest of the N-D IMDF algorithm is the same as (2.17)–(2.24) except for different dimension sizes, which are clear from the N-D context. After $\boldsymbol{P}$ and $\boldsymbol{H}$ are retrieved, N-D frequency estimates can be obtained by simple division and average operation similar to (2.25) for the noiseless case or (2.27) for the noisy case. Notice that individual frequencies of an N-D frequency component $\widehat{\omega}_{f,n}$, $n = 1, \ldots, N$, are automatically paired.

2.1.4 The IMDF algorithm In Multiple Snapshots Case

The data model with multiple snapshot is given in (1.11). The algorithm proposed in Section 2.1.3 can be generalized to multiple snapshot case. The following lemma is necessary for the argument. It claims that the Khatri-Rao product of a random matrix after FB smoothing and multiple Vandermonde matrices has full column rank.

Lemma 3: In (1.11), given N Vandermonde matrices $A_n \in \mathbb{C}^{M_n \times F}$, with generator $\{e^{j\omega_{1,n}}, \cdots, e^{j\omega_{F,n}}\}$, for $n = 1, \cdots, N$, and one complex random matrix $\boldsymbol{C} \in \mathbb{C}^{F \times T}$ with $\boldsymbol{C}_{f,t} = c_f(t)$, if one define

$$\boldsymbol{Q} := \begin{bmatrix} \boldsymbol{C}^T \\ \boldsymbol{\Pi}_T \boldsymbol{C}^H \boldsymbol{D}(\boldsymbol{\phi}) \end{bmatrix}, \tag{2.44}$$

where $\boldsymbol{\Pi}_T$ is a $T \times T$ permutation matrix with ones on its anti-diagonal, and $\boldsymbol{\phi} = [e^{-j\phi_1} \;\; e^{-j\phi_2} \;\; \cdots \;\; e^{-j\phi_F}]^T$, with ϕ_f is defined in (2.39), then the rank of matrix:

$$\boldsymbol{Q} \odot \boldsymbol{A}_1^{(K_1)} \odot \boldsymbol{A}_2^{(K_2)} \cdots \odot \boldsymbol{A}_N^{(K_N)}$$

is $\min\{2T \prod_{n=1}^{N} K_n, F\}$ almost surely, provided that the NF frequencies $(\omega_{f,1}, \cdots, \omega_{f,N})$, $f = 1, \cdots, F$, and amplitudes $(c_f(1), \cdots, c_f(T))$, $f = 1, \ldots, F$, are drawn according to a distribution that is continuous with respect to the Lebesgue measure in Φ^{NF} and $\mathbb{C}^{TF}$ respectively.

Proof: See Appendix B. ■

When multiple snapshots are available, for each snapshot $\boldsymbol{x}(t)$, one can perform the operations in (2.31)-(2.35) and (2.40)-(2.43). Then one can stack these smoothed data matrices to form $\boldsymbol{Z}_1$ and $\boldsymbol{Z}_2$ as follows.

$$\boldsymbol{Z}_1 := \left[\boldsymbol{X}_{1,S}^T(1) \; \boldsymbol{X}_{1,S}^T(2) \; \cdots \; \boldsymbol{X}_{1,S}^T(T) \; \boldsymbol{Y}_{1,S}^T(T) \; \boldsymbol{Y}_{1,S}^T(T-1) \; \cdots \; \boldsymbol{Y}_{1,S}^T(1)\right]^T, \tag{2.45}$$

$$\boldsymbol{Z}_2 := \left[\boldsymbol{X}_{2,S}^T(1) \; \boldsymbol{X}_{2,S}^T(2) \; \cdots \; \boldsymbol{X}_{2,S}^T(T) \; \boldsymbol{Y}_{2,S}^T(T) \; \boldsymbol{Y}_{2,S}^T(T-1) \; \cdots \; \boldsymbol{Y}_{2,S}^T(1)\right]^T. \tag{2.46}$$

It can be verified that

$$\boldsymbol{Z} := \begin{bmatrix} \boldsymbol{Z}_1 \\ \boldsymbol{Z}_2 \end{bmatrix} = \begin{bmatrix} \boldsymbol{P}_1 \\ \boldsymbol{P}_1\boldsymbol{D}(\boldsymbol{\omega}_1) \end{bmatrix} \boldsymbol{H}^T = \boldsymbol{P}\boldsymbol{H}^T, \tag{2.47}$$

where $\boldsymbol{P} := [\boldsymbol{P}_1{}^T\ (\boldsymbol{P}_1\boldsymbol{D}(\boldsymbol{\omega}_1))^T]^T$ and $\boldsymbol{P}_1 := \boldsymbol{Q} \odot \boldsymbol{G}$. Here $\boldsymbol{Q}$ is defined in (2.44). Since $\boldsymbol{H}$ is the Khatri-Rao product of Vandermode matrices, according to Property p3), if $\prod_{n=1}^{N} L_n \geq F$, $\boldsymbol{H}$ is almost surely full column rank. According to Lemma 3, if $2T\prod_{n=1}^{N} K_n \geq F$, $\boldsymbol{P}_1$ is almost surely full column rank. Hence $\boldsymbol{Z}_1$ and $\boldsymbol{Z}_2$ are of rank F, the singular value decomposition of $\boldsymbol{Z}$ yields $\boldsymbol{U}_s$. All the remaining steps to estimate $\boldsymbol{T}$ and N-D frequencies from $\boldsymbol{U}_s$ are similar to those given (2.20)-(2.25) in Section 2.1.3.

2.2 Statistical Identifiability

In this section, the statistical identifiability condition of the IMDF algorithm is derived. The maximum number of uniquely resolvable frequencies in the absence of noise is referred to as the ID bound here. This section shows that the IMDF algorithm offers the most relaxed ID bounds among existing algebraic approaches for 2-D and N-D frequency estimation.

2.2.1 Statistical Identifiability of the 2-D IMDF Algorithm

The identifiability result of the IMDF algorithm for 2-D frequency estimation from single snapshot is summarized in the following theorem.

Theorem 1: Given a sum of F 2-D exponentials as in (1.1) and without loss of generality, assume $M_1 \geq M_2$, the parameter set (ω_f, ν_f, c_f), $f = 1, \cdots, F$, can be uniquely retrieved by the 2-D IMDF algorithm almost surely, if

$$F \leq \max_{\substack{K_1+L_1=M_1 \\ K_2+L_2=M_2+1}} \min(2K_1K_2,\ L_1L_2) \tag{2.48}$$

where $(\omega_f,\ \nu_f)$, $f = 1, \ldots, F$, are assumed to be drawn from a distribution that is continuous with respect to the Lebesgue measure in Π^{2F}, and c_f is also drawn from a continuous distribution on $\mathbb{C}$.

Proof: The 2-D IMDF algorithm described in Section 2.1.2 can be used as a constructive proof of Theorem 1. Suppose that $(\omega_f,\ \nu_f)$, $f = 1, \ldots, F$, are drawn from a distribution that is continuous with respect to the Lebesgue measure in Π^{2F}, and c_f is also drawn from a continuous distribution on $\mathbb{C}$. Given (1.1), it is shown in Section 2.1.2 that the 2-D IMDF algorithm can uniquely retrieve (ω_f, ν_f, c_f), $f = 1, \ldots, F$, with probability one, provided that (see (2.20))

$$F \leq 2K_1K_2, \quad F \leq L_1L_2$$

where K_1, K_2, L_1, and L_2 are positive integers subject to (see (2.8))

$$K_1 + L_1 = M_1, \quad K_2 + L_2 = M_2 + 1. \tag{2.49}$$

Therefore Theorem 1 is proved. ■

To calculate the ID bound (2.48) for the 2-D IMDF algorithm, one can define the following function

$$\Gamma(M_1, M_2) = \max_{\substack{K_1+L_1=M_1+1 \\ K_2+L_2=M_2+1}} \min\left(2K_1K_2,\ L_1L_2\right) \tag{2.50}$$

where K_i and L_i, for $i = 1, 2$, are positive integers. It is clear that the statistical ID bound of the 2-D IMDF algorithm is given by $\Gamma(M_1 - 1, M_2)$. It is difficult to obtain the exact solution of the integer optimization problem $\Gamma(M_1 - 1, M_2)$. But one can give its lower and upper bounds as follows.

Lemma 4: $\Gamma(M_1 - 1, M_2)$ is bounded by

$$\begin{aligned}
&\min\left\{2\left[(\sqrt{2}-1)M_1\right]\left[(\sqrt{2}-1)(M_2+1)\right], \left[(2-\sqrt{2})M_1\right]\left[(2-\sqrt{2})(M_2+1)\right]\right\} \\
&\leq\ \Gamma(M_1 - 1, M_2)\ \leq\ \left\lfloor 0.34M_1(M_2+1)\right\rfloor.
\end{aligned}$$

Proof: The upper bound is proved first. Denote $\lambda_1 = K_1/M_1$, $\lambda_2 = K_2/(M_2+1)$. Clearly $0 < \lambda_1, \lambda_2 < 1$. If K_1, K_2, L_1 and L_2 are real numbers, then

$$\begin{aligned}
\min(2K_1K_2, L_1L_2) &= \min(2M_1(M_2+1)\lambda_1\lambda_2, M_1(M_2+1)(1-\lambda_1)(1-\lambda_2)) \\
&= M_1(M_2+1)\min(2\lambda_1\lambda_2, (1-\lambda_1)(1-\lambda_2)) \\
&\leq M_1(M_2+1)\min\left(\frac{(\lambda_1+\lambda_2)^2}{2}, \frac{(2-\lambda_1-\lambda_2)^2}{4}\right) \\
&= \begin{cases} M_1(M_2+1)\frac{(\lambda_1+\lambda_2)^2}{2}, & 0<\lambda_1+\lambda_2\leq(2\sqrt{2}-2) \\ M_1(M_2+1)\frac{(2-\lambda_1-\lambda_2)^2}{4}, & (2\sqrt{2}-2)\leq\lambda_1+\lambda_2<2 \end{cases} \\
&\leq 0.34M_1(M_2+1).
\end{aligned}$$

The maximum is achieved when $K_1 = M_1(\sqrt{2}-1)$ and $K_2 = (M_2+1)(\sqrt{2}-1)$. In the context, K_1, K_2, and $\Gamma(M_1-1, M_2)$ are all integers, therefore $\Gamma(M_1-1, M_2)\leq \lfloor 0.34M_1(M_2+1)\rfloor$.

Next the lower bound is proved. Let $K_1 = [\sqrt{2}M_1]-M_1$ and $K_2 = [\sqrt{2}(M_2+1)]-(M_2+1)$, where $[\cdot]$ stands for rounding towards nearest integer, then one can obtain a lower bound of $\Gamma(M_1-1, M_2)$ as given in Lemma 4. ■

Remark 1: In Section 2.1.2, the first step of the 2-D IMDF algorithm is to apply smoothing to the column dimension of $\boldsymbol{X}$ to obtain $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ (see, (2.6) and (2.7)). Instead, if smoothing is applied to the row dimension of $\boldsymbol{X}$, then one has

$$\begin{aligned}
\boldsymbol{x}_1 &= \left(\boldsymbol{A}\odot\boldsymbol{B}^{(M_2-1)}\right)\boldsymbol{c} \\
\boldsymbol{x}_2 &= \left(\boldsymbol{A}\odot\boldsymbol{B}^{(M_2-1)}\right)\boldsymbol{D}(\boldsymbol{\nu})\boldsymbol{c}
\end{aligned}$$

where $\boldsymbol{\nu} = \left[e^{j\nu_1}, \cdots, e^{j\nu_F}\right]^T$. The rest of the algorithm is the same as in Section 2.1.2. It is clear that the identifiability condition in this case is $F \leq \Gamma(M_1, M_2-1)$. The following lemma is useful.

Lemma 5: If $M_1 \geq M_2$, then $\Gamma(M_1-1, M_2) \geq \Gamma(M_1, M_2-1)$.

Proof: Suppose $M_1 \geq M_2$. Let

$$(K_1^*, K_2^*, L_1^*, L_2^*) = \arg\max \min_{\substack{K_1+L_1=M_1+1 \\ K_2+L_2=M_2}} (2K_1K_2,\ L_1L_2)$$

then either $K_1^* \geq K_2^* + 1$ or $L_1^* \geq L_2^* + 1$, otherwise

$$M_1 + 1 = K_1^* + L_1^* \leq K_2^* + L_2^* = M_2$$

which contradicts the assumption that $M_1 \geq M_2$.

If $K_1^* \geq K_2^* + 1$, then $2(K_1^* - 1)(K_2^* + 1) \geq 2K_1^* K_2^*$, and one has

$$\begin{aligned} \Gamma(M_1 - 1, M_2) \;&\geq\; \left(\min_{\substack{K_1+L_1=M_1 \\ K_2+L_2=M_2+1}} (2K_1K_2,\ L_1L_2) \right) \Bigg|_{\substack{K_1=K_1^*-1,\ L_1=L_1^* \\ K_2=K_2^*+1,\ L_2=L_2^*}} \\ &\geq\; \Gamma(M_1, M_2 - 1). \end{aligned}$$

If $L_1^* \geq L_2^* + 1$, then $(L_1^* - 1)(L_2^* + 1) \geq L_1^* L_2^*$, and one has

$$\begin{aligned} \Gamma(M_1 - 1, M_2) \;&\geq\; \left(\min_{\substack{K_1+L_1=M_1 \\ K_2+L_2=M_2+1}} (2K_1K_2,\ L_1L_2) \right) \Bigg|_{\substack{K_1=K_1^*,\ L_1=L_1^*-1 \\ K_2=K_2^*,\ L_2=L_2^*+1}} \\ &\geq\; \Gamma(M_1, M_2 - 1). \end{aligned}$$

Therefore if $M_1 \geq M_2$, $\Gamma(M_1 - 1, M_2) \geq \Gamma(M_1, M_2 - 1)$. ■

From Lemma 5 one can conclude that from the identifiability perspective, it is advantageous to first perform smoothing along the dimension with a larger sample size.

Theorem 1 shows that the IMDF algorithm offers a significantly improved ID bound for 2-D frequency estimation over existing algebraic algorithms. Previously the most relaxed statistical ID bound is achieved by the MDF algorithm as shown in [24]. Using Property p3), one is able to obtain the statistical ID bounds of the MEMP algorithm [21]. Fig. 2.1 plots a comparison of the statistical ID bounds of different algebraic algorithms for 2-D frequency estimation in the absence of noise, where $M_1 = M_2$. The MDF algorithm have the ID bound as

$$F \leq \lceil M_1/2 \rceil \lceil M_2/2 \rceil .$$

The statistical ID bound of the MEMP algorithm is slightly smaller than those of the MDF algorithm because no FB smoothing is used in the MEMP algorithm. Note

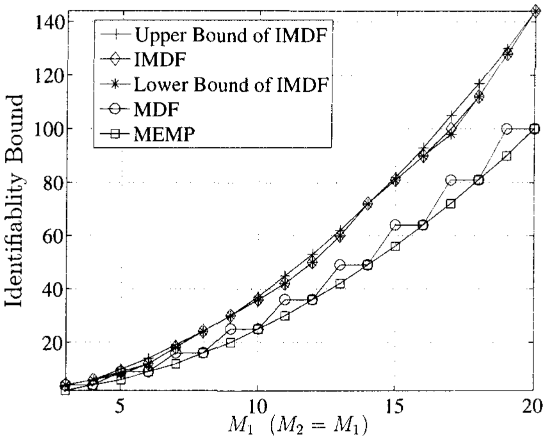


Figure 2.1. Comparison of ID bounds for 2-D frequency estimation (the two dimensions have equal size: $M_1 = M_2$).

that the deterministic ID bound of the MEMP algorithm is $F \leq \min(M_1/2, M_2/2)$ as shown in [21] and [26]. It is clear that the IMDF algorithm outperforms these approaches in terms of the maximum number of resolvable 2-D frequencies. From Fig. 2.1, one can also observe that the upper and lower bounds given in Lemma 4 are very close to the integer solution of $\Gamma(M_1 - 1, M_2)$.

2.2.2 Statistical Identifiability of the N-D IMDF Algorithm

Similar to the 2-D case, one can establish the following ID result for the N-D IMDF algorithm.

Theorem 2: Given a sum of F N-D exponentials in (1.1), without loss of generality, assuming that $M_1 = \max\{M_n, n = 1, \ldots, N\}$, if

$$F \leq \max_{\substack{K_1+L_1=M_1 \\ K_n+L_n=M_n+1 \\ 2\leq n\leq N}} \min\left(2\prod_{n=1}^{N} K_n, \prod_{n=1}^{N} L_n\right) \tag{2.51}$$

and the distributions used to draw the NF frequencies and F amplitudes are continuous with respect to the Lebesgue measure in Π^{NF} and $\mathbb{C}$, respectively, then the parameter $(N+1)$-tuples $(\omega_{f,1}, \ldots, \omega_{f,N}, c_f)$, $f = 1, \ldots, F$, can be uniquely resolved almost surely by the N-D IMDF algorithm.

Proof: In the N-D case, the parameter $(N+1)$-tuples $(\omega_{f,1},\ldots,\omega_{f,N},c_f)$, $f=1,\ldots,F$, are uniquely identifiable by the N-D IMDF algorithm, provided that the corresponding matrices $\boldsymbol{P}_1$ and $\boldsymbol{H}$ are full column rank. Suppose the distributions used to draw the NF frequencies and F amplitudes are continuous with respect to the Lebesgue measure in Π^{NF} and $\mathbb{C}$, respectively, then according to Property p3), $\boldsymbol{P}_1$ and $\boldsymbol{H}$ are almost surely full column rank if

$$F \le 2\prod_{n=1}^{N} K_n, \qquad F \le \prod_{n=1}^{N} L_n$$

where $K_1 + L_1 = M_1$, and $K_n + L_n = M_n + 1$ for $n \ne 1$. Therefore Theorem 2 is proved. ■

Similar to 2-D case, one can define following integer function

$$\Gamma(M_1,\cdots,M_N) = \max_{\substack{K_n+L_n=M_n+1\\ 1\le n\le N}} \min\left(2\prod_{n=1}^{N} K_n,\ \prod_{n=1}^{N} L_n\right) \tag{2.52}$$

then the RHS of (2.51) is given by $\Gamma(M_1-1,M_2,\cdots,M_N)$.

Lemma 6: An upper bound of $\Gamma(M_1-1,M_2,\cdots,M_n)$ is given by

$$\Gamma(M_1-1,M_2,\cdots,M_n) \le \left\lfloor \frac{2M_1\prod_{n=2}^{N}(M_n+1)}{(1+\sqrt[N]{2})^N} \right\rfloor. \tag{2.53}$$

Proof: This lemma is a special case of Lemma 8 when $T=1$. Please refer to the proof of Lemma 8. ■

Let $K_1 = \left[\frac{M_1}{(1+\sqrt[N]{2})^N}\right]$ and $K_n = \left[\frac{M_n+1}{(1+\sqrt[N]{2})^N}\right]$, $1 \le n \le N$, one can obtain a lower bound for $\Gamma(M_1-1,\cdots,M_2)$. Lemma 2 can also be extended to the N-D case, the result is given as follows.

Lemma 7: If $M_1 \ge M_n$, $n \ne 1$, then

$$\Gamma(M_1-1,M_2,\cdots,M_N) \ge \Gamma(M_1,\cdots,M_n-1,\cdots,M_N).$$

Theorem 2 also shows significant improvement over previously available statistical ID bound for N-D frequency estimation, for example, the ID bound given in [24] is $F \le \prod_{n=1}^{N} \lceil M_n/2 \rceil$.

2.2.3 Statistical Identifiability in the Multiple Snapshot Case

For the identifiability bound in multiple snapshots case, the following result is available.

Theorem 3: Given T snapshots of sums of F N-D exponentials as in (1.3) or (1.11), without loss of generality, assume that $M_1 = \max\{M_n, n = 1, \ldots, N\}$, then the parameter set $(\{\omega_{f,n}\}_{n=1}^{N}, \{c_f(t)\}_{t=1}^{T})$, $f = 1, \ldots, F$, are almost surely unique, provided that

$$F \leq \max_{\substack{K_1+L_1=M_1 \\ K_n+L_n=M_n+1 \\ 2\leq n\leq N}} \min\left(2T\prod_{n=1}^{N} K_n, \prod_{n=1}^{N} L_n\right), \tag{2.54}$$

where the NF frequencies $(\omega_{f,1}, \cdots, \omega_{f,N})$, $f = 1, \cdots, F$, and amplitudes $(c_f(1), \cdots, c_f(T))$, $f = 1, \ldots, F$,are assumed to be drawn according to a distribution that is continuous with respect to the Lebesgue measure in Φ^{NF} and $\mathbb{C}^{TF}$ respectively.

Proof: The proof of this theorem follows the same argument as that in Theorem 2. The proof is omitted here. Notice that the proof requires Lemma 3. ■

Define the function

$$\Gamma_T(M_1, \cdots, M_N) = \max_{\substack{K_n+L_n=M_n+1 \\ 1\leq n\leq N}} \min\left(2T\prod_{n=1}^{N} K_n, \prod_{n=1}^{N} L_n\right),$$

then the RHS of (2.54) is given by $\Gamma_T(M_1 - 1, M_2, \cdots, M_N)$.

Lemma 8: If $T \leq (M_1 - 1)\prod_{n=2}^{N} M_n/2$, then

$$\Gamma_T(M_1 - 1, M_2, \cdots, M_n) \leq \left\lfloor \frac{2TM_1\prod_{n=2}^{N}(M_n + 1)}{(1 + \sqrt[N]{2T})^N} \right\rfloor, \tag{2.55}$$

If $T \geq (M_1 - 1)\prod_{n=2}^{N} M_n/2$, then

$$\Gamma_T(M_1 - 1, M_2, \cdots, M_n) = (M_1 - 1)\prod_{n=2}^{N} M_n. \tag{2.56}$$

Proof: Define $\lambda_1 = K_1/M_1$, $\lambda_n = K_n/(M_n+1)$, $n = 2,\cdots,N$, and $\lambda := (\sum_{n=1}^{N}\lambda_n)/N \in [0,1]$. If K_1 and K_n, $n = 2,\cdots,N$ are all real numbers,

$$\begin{aligned}\min\left(2T\prod_{n=1}^{N}K_n, \prod_{n=1}^{N}L_n\right) &= M_1\prod_{n=2}^{N}(M_n+1)\min\left(2T\prod_{n=1}^{N}\lambda_n, \prod_{n=1}^{N}(1-\lambda_n)\right)\\ &\leq M_1\prod_{n=2}^{N}(M_n+1)\min\left(2T\lambda^N, (1-\lambda)^N\right)\\ &= \begin{cases} M_1\prod_{n=2}^{N}(M_n+1)2T\lambda^N, & 0\leq\lambda\leq\frac{1}{1+\sqrt[N]{2T}},\\ M_1\prod_{n=2}^{N}(M_n+1)(1-\lambda)^N, & \frac{1}{1+\sqrt[N]{2T}}\leq\lambda\leq 1\end{cases}\\ &\leq\frac{2TM_1\prod_{n=2}^{N}(M_n+1)}{(1+\sqrt[N]{2T})^N}.\end{aligned}$$

Since the maximum is achieved when $K_1 = \frac{M_1}{(1+\sqrt[N]{2T})^N}$ and $K_n = \frac{M_n+1}{(1+\sqrt[N]{2T})^N}$, $n = 2,\cdots,N$. But K_1 and K_n, $n = 2,\cdots,N$ are all integers, $\Gamma(M_1-1,\cdots,M_N)$ can not equal to the maximum value, therefore

$$\Gamma_T(M_1-1,\cdots,M_N) < \frac{2TM_1\prod_{n=2}^{N}(M_n+1)}{(1+\sqrt[N]{2T})^N}.$$

Because $\Gamma_T(M_1-1,\cdots,M_N)$ is an integer,

$$\Gamma_T(M_1-1,\cdots,M_N)\leq\left\lfloor\frac{2TM_1\prod_{n=2}^{N}(M_n+1)}{(1+\sqrt[N]{2T})^N}\right\rfloor.$$

Notice that since $T \leq (M_1-1)\prod_{n=2}^{N}M_n/2$, it is possible that the equation achieves when K_n, $n = 1,\ldots,N$, equal integers. When $T \geq (M_1-1)\prod_{n=2}^{N}M_n/2$, since $K_n \geq 1$, $M_n \geq L_n$ for $n = 1,\ldots,N$, one can obtain

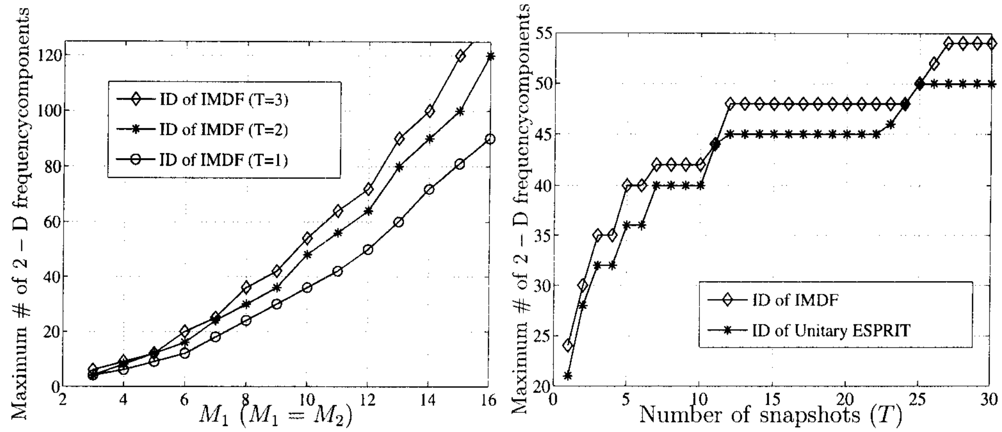
$$2T\prod_{n=1}^{N}K_n \geq (M_1-1)\prod_{n=2}^{N}M_n\prod_{n=1}^{N}K_n \geq (M_1-1)\prod_{n=2}^{N}M_n \geq (L_1-1)\prod_{n=2}^{N}L_n, \quad (2.57)$$

therefore in this case $\Gamma_T(M_1-1,M_2,\cdots,M_n) = (M_1-1)\prod_{n=2}^{N}M_n$. ■

Similar to Lemma 5 and Lemma 7, one can obtain

Lemma 9: If $M_1 \geq M_n$, then

$$\Gamma_T(M_1-1,\cdots,M_n,\cdots,M_N)\geq\Gamma_T(M_1,\cdots,M_n-1,\cdots,M_N).$$



(a) (b)

Figure 2.2. Identifiability (ID) bound of 2-D frequency estimation: (a) ID bound versus M_1 when $M_1 = M_2$; (b) ID bounds versus the number of snapshots where $M_1 = 10$ and $M_2 = 6$.

As a special case, consider $N = 2$. The identifiability condition (2.54) becomes

$$F \leq \max_{\substack{K_1+L_1=M_1+1\\K_2+L_2=M_2+1}} \min\{(K_1-1)K_2,\ 2TL_1L_2\}. \tag{2.58}$$

The identifiability condition of the Unitary ESPRIT algorithm [13] is

$$F \leq \max_{\substack{K_1+L_1=M_1+1\\K_2+L_2=M_2+1}} \min\{(K_1-1)K_2,\ K_1(K_2-1),\ 2TL_1L_2\}. \tag{2.59}$$

In Fig. 3.1 (a), the identifiability bound in (3.25) is plotted as a function of the dimension size with $M_1 = M_2$. It can be seen that the identifiability bound generally increases with the increase of snapshot size and the number of snapshots. Fig. 3.1 (b) plots the identifiability bounds in (2.58) and (2.59) versus the number of snapshots, respectively. It can be seen that when the number of snapshots T is less than a threshold (approximately $M/2$), as T increases, the bounds increase like stairs. When T is greater than the threshold, the bounds become approximately M. For example, for the identifiability condition given in (2.58), when $T \geq (M_1-1)M_2/2$, the bound becomes $(M_1-1)M_2$.

2.3 Computational Complexity

The section gives an estimate of the computation order in term of real multiplications required by the 2-D IMDF algorithm in single snapshot case described in Table 2.1. For the multiple snapshot case, the complexity can be analyzed similarly. It is obvious that the computation required for data smoothing is trivial compared to that of the rest of the algorithm. Therefore the computation order of steps (2)-(4) in Table 2.1 is considered. Table 2.2 lists the number of flops required in each step of the 2-D IMDF algorithm in estimating F 2-D frequencies from an $M_1 \times M_2$ data set, where SVD is assumed to be performed by the Golub-Reinsch SVD algorithm [37]. The computational complexity of the 2-D IMDF algorithm is $\mathcal{O}(M_1^3 M_2^3)$.

Many existing algebraic approaches for 2-D frequency estimation resort to SVD to obtain the signal subspace, and 2-D data smoothing is also commonly used, e.g., in Unitary ESPRIT, MEMP. For those algorithms, the computation complexity to solve an eigenvalue problem is comparable to that of the 2-D IMDF algorithm, which is proportional to $\mathcal{O}(M_1^3 M_2^3)$.

However, one advantage of the IMDF algorithm is that the frequency parameters corresponding to the individual dimensions of a given multidimensional frequency component are associated automatically once the eigenvalue problem is solved. Other algebraic multidimensional frequency estimation algorithms generally require an extra association step, which can be computationally complex. For example, multidimensional Unitary ESPRIT [13] entails a simultaneous Schur decomposition (joint diagonalization) step to attain automatic pairing. While this step improves performance, it also comes with higher computational complexity.

2.4 Performance Analysis

This section provides a first order perturbation analysis for the 2-D IMDF algorithm outlined in Table 2.1. Recall that in the presence of noise, the observation

TABLE 2.2. Computational complexity of the 2-D IMDF algorithm when $T = 1$

Operation	**Dimension Size**	**Required Flops**
SVD of $\boldsymbol{Z}$	$4K_1K_2 \times L_1L_2$	$4(4K_1K_2)^2L_1L_2 + 32K_1K_2(L_1L_2)^2 + 9(L_1L_2)^3$
SVD to compute $\boldsymbol{U}_1^\dagger$	$2K_1K_2 \times F$	$4(2K_1K_2)^2F + 16K_1K_2F^2 + 9F^3$
Multiplication $\boldsymbol{U}_1^\dagger \boldsymbol{U}_2$	$F \times F$	$2F^2K_1K_2$
QR EVD of $\boldsymbol{U}_1^\dagger \boldsymbol{U}_2$	$F \times F$	$25F^3$
Multiplication $\boldsymbol{U}_s\boldsymbol{T}$	$4K_1K_2 \times F$	$4K_1K_2F^2$
Estimate frequencies from $\boldsymbol{P\Lambda\Delta}$	$4K_1K_2 \times F$	$4F(2K_2K_1 - K_1 - K_2)$
Multiplication $(\boldsymbol{T}^{-1}\boldsymbol{\Sigma}_s{\boldsymbol{V}_s}^H)^T$	$F \times L_1L_2$	$F^2L_1L_2$
Estimate Frequencies from $\boldsymbol{H\Lambda}^{-1}\boldsymbol{\Delta}$	$L_1L_2 \times F$	$(2L_1L_2 - L_1 - L_2)F$
Total flops required $\sim \mathcal{O}(M_1^3M_2^3)$		$32K_1K_2(L_1L_2)^2 + (4K_1K_2)^2L_1L_2 + 9(L_1L_2)^3 + 34F^3 + (22K_1K_2 + L_1L_2)F^2 + 4(2K_1K_2)^2F + (8K_1K_2 + 2L_1L_2)F$

vector is (c.f., (1.5))

$$\widehat{\boldsymbol{x}} = (\boldsymbol{A} \odot \boldsymbol{B})\boldsymbol{c} + \boldsymbol{w} \tag{2.60}$$

where $\boldsymbol{w}$ is the perturbation of the data, assumed to be complex Gaussian distributed with zero mean and variance $\sigma^2\boldsymbol{I}_{M_1M_2}$. Suppose the 2-D frequency estimates given by the IMDF algorithm are $\widehat{\omega}_f = \omega_f + \Delta\omega_f$ and $\widehat{\nu}_f = \nu_f + \Delta\nu_f$, for $f = 1, \ldots, F$. The closed form expressions of the expectation and variance of $\Delta\omega_f$ and $\Delta\nu_f$ are derived in [38].

When $F = 1$, the variances of $\Delta\omega_f$ can be reduced to

$$E\left[(\Delta\omega_f)^2\right] = \frac{\sigma^2}{3\gamma_1^2c^2}(3K_2L_2 - p_2^2 + 1)p_2f(K_1, K_2) \tag{2.61}$$

where $p_2 = \min\{K_2, L_2\}$, and $f(K_1, K_2)$ is defined as

$$f(K_1,K_2) = \begin{cases} \left(\frac{1}{L_1L_2} + \frac{1}{4K_1K_2}\right)^2(16K_1 - 28) + 4\left(\frac{1}{L_1L_2} + \frac{1}{2K_1K_2}\right)^2 + 4\left(\frac{1}{4K_1K_2}\right)^2, & 2 \leq K_1 < (M_1 - 1)/2 \\ \left(\frac{1}{L_1L_2} + \frac{1}{4K_1K_2}\right)^2(16K_1 - 28) + 4\left(\frac{1}{L_1L_2} + \frac{1}{2K_1K_2}\right)^2, & K_1 = (M_1 - 1)/2 \\ \left(\frac{1}{L_1L_2} + \frac{1}{4K_1K_2}\right)^2(16K_1 - 24), & K_1 = M_1/2 \\ \left(\frac{1}{L_1L_2} + \frac{1}{4K_1K_2}\right)^2(16L_1 - 28) + 4\left(\frac{2}{L_1L_2} + \frac{1}{4K_1K_2}\right)^2, & K_1 = (M_1 + 1)/2 \\ \left(\frac{1}{L_1L_2} + \frac{1}{4K_1K_2}\right)^2(16L_1 - 28) + 4\left(\frac{2}{L_1L_2} + \frac{1}{4K_1K_2}\right)^2 + 4\left(\frac{1}{L_1L_2}\right)^2. & \frac{M_1+1}{2} < K_1 < M_1 \end{cases} \tag{2.62}$$

The variances of $\Delta\nu_f$ are reduced to

$$E\left[(\Delta\nu_f)^2\right] = \frac{\sigma^2}{3\gamma_2^2c^2}2p_1(6K_1L_1 - 2p_1^2 - 1)g(K_1, K_2) \tag{2.63}$$

where $p_1 = \min\{K_1, L_1\}$, and $g(K_1, K_2)$ is defined as

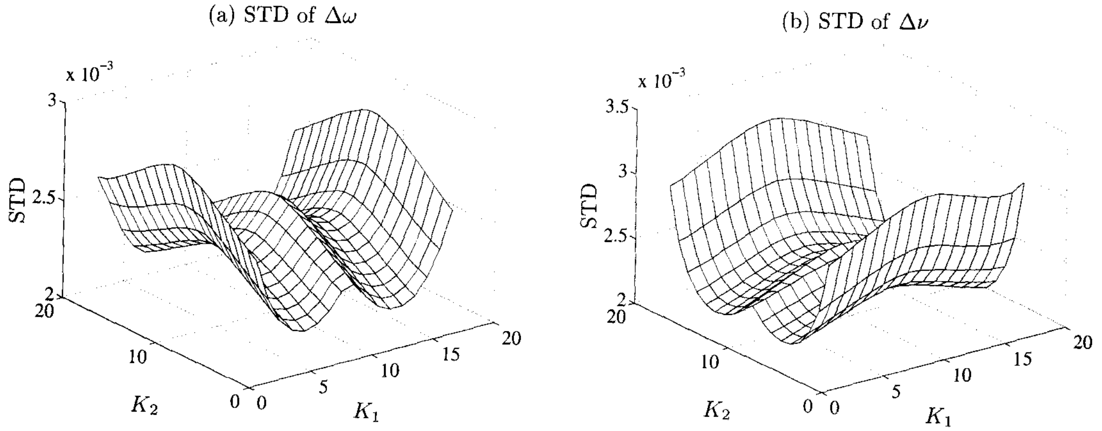
$$g(K_1,K_2) = \begin{cases} \left(\frac{1}{L_1L_2} + \frac{1}{4K_1K_2}\right)^2 (4K_2 - 4) + 4\left(\frac{1}{4K_1K_2}\right)^2, & 2 \leq K_2 < (M_2+1)/2 \\ \left(\frac{1}{L_1L_2} + \frac{1}{4K_1K_2}\right)^2 (4K_2 - 4), & K_2 = (M_2+1)/2 \\ \left(\frac{1}{L_1L_2} + \frac{1}{4K_1K_2}\right)^2 (4L_2 - 4) + 4\left(\frac{1}{L_1L_2}\right)^2. & (M_2+1)/2 < K_2 < M_2 + 1 \end{cases} \tag{2.64}$$

Fig. 2.3 illustrates how the selection of smoothing parameters K_1 and K_2 affects the estimation variance of the 2-D IMDF algorithm, where the standard deviation (STD) of the estimate of a single 2-D frequency pair from a 20×20 data set with $\sigma^2 = 1$ is plotted in various ways as explained as follows.

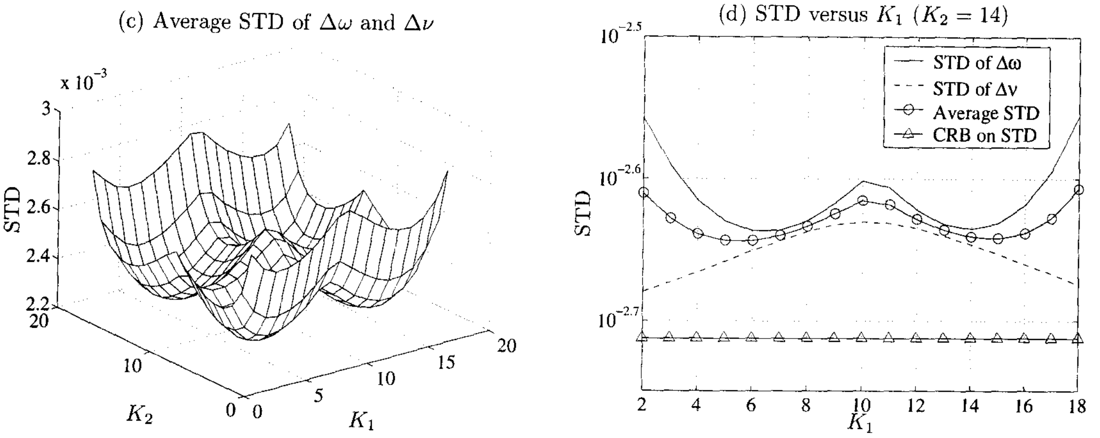
Fig. 2.3 (a) and (b) are the plots of the STDs of $\Delta\omega$ and $\Delta\nu$, respectively. Fig. 2.3 (c) is the plot of the average STD of $\Delta\omega$ and $\Delta\nu$. The smoothing parameter K_1 ranges from 2 to 18, while K_2 varies from 2 to 19, according to (2.8). It is clear that as K_1 and K_2 change, the STD changes, but not significantly. Fig. 2.3 (c) shows that there are four areas where the average estimation variance is minimized. Fig. 2.3 (d) plots the STDs of the frequency estimation error by fixing $K_2 = 14$ while changing K_1. Similarly, Fig. 2.3 (e) plots the STDs of the frequency estimation error by fixing $K_1 = 6$ while changing K_2. Notice that it is chosen that $K_1 = 6$ and $K_2 = 14$ according to the observation from Fig. 2.3 (c). The corresponding Cramér Rao Bound (CRB) is also plotted in Fig. 2.3 (d) and (e). For $\Delta\omega$, the minimum STD is achieved when $K_1 \approx M_1/3$ or $2M_1/3$ and $K_2 \approx 2$ or $M_2 - 1$. The results resemble those in [29]. On the other hand, the minimum STD of $\Delta\nu$ is achieved when $K_1 \approx 2$ or $M_1 - 2$ and $K_2 \approx M_2/3 + 1$ or $2M_2/3 + 1$. From the closed forms of variances in (2.61) and (2.63), one can obtain that the minimal average estimation variance is achieved when

$$\begin{aligned} K_1 &\approx 0.29M_1 \quad \text{or} \quad 0.65M_1 \\ K_2 &\approx 0.29(M_2+1) \quad \text{or} \quad 0.65(M_2+1). \end{aligned}$$

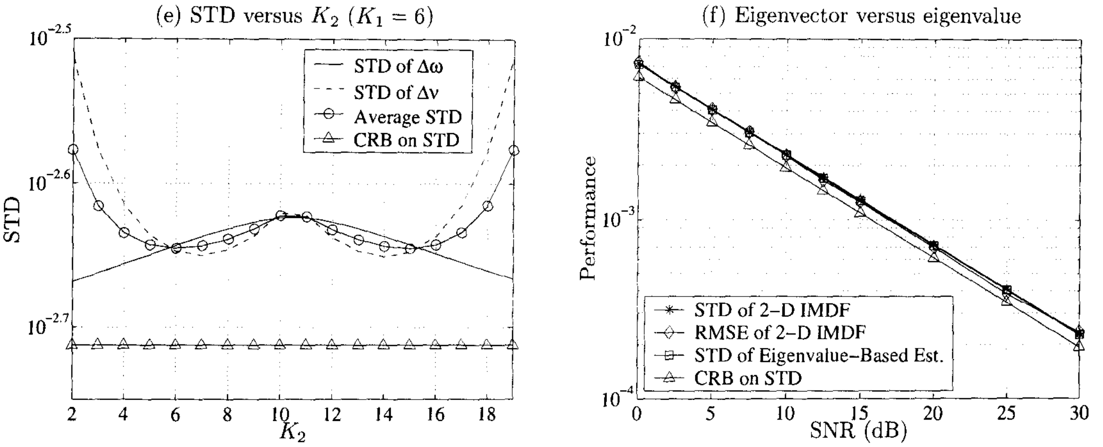
As mentioned earlier, one of the major differences of the IMDF algorithm and existing algebraic algorithms is that the IMDF algorithm estimates frequencies


(a) STD of $\Delta\omega$

(b) STD of $\Delta\nu$


(c) Average STD of $\Delta\omega$ and $\Delta\nu$

(d) STD versus K_1 ($K_2 = 14$)


(e) STD versus K_2 ($K_1 = 6$)

(f) Eigenvector versus eigenvalue

Figure 2.3. Standard deviation (STD) of 2-D frequency estimation error when $F = 1$: (a) STD of $\Delta\omega$; (b) STD of $\Delta\nu$; (c) Average STD of $\Delta\omega$ and $\Delta\nu$; (d) STD as a function of K_1 when $K_2 = 14$; (e) STD as a function of K_2 when $K_1 = 6$; (f) STD of eigenvector-based and eigenvalue-based estimation, with RMSE of the 2-D IMDF estimation (eigenvector-based) and the corresponding CRB, when $K_1 = 5$ and $K_2 = 6$.

from eigenvectors instead of eigenvalues. An interesting question is whether this will result in different estimation variance. Fig. 2.3 (f) studies this issue. Notice that without considering frequency paring, ω_f can be estimated from the eigenvalues of $\boldsymbol{U}_1^{\dagger}\boldsymbol{U}_2$ directly (see, (2.22)). The STD of the eigenvalue-based estimates can then be obtained. The STD of the eigenvalue-based and eigenvector-based estimates are plotted in Fig. 2.3 (f), where a single frequency pair $(\omega, \nu) = (0.41\pi, 0.61\pi)$ is to be estimated from a 20×20 data set. It can be seen that the difference of the two STDs are negligible. This demonstrates that the eigenvector-based estimator (i.e., the IMDF algorithm) is as efficient as those based on eigenvalues. Note that because an extra pairing step is usually required for eigenvalue-based approaches, which could be fallible, so the actual estimation variance of eigenvalue-based approaches may be higher. Fig. 2.3 (f) also plots the CRB, and the root mean square error (RMSE) of the IMDF algorithm obtained by Monte Carlo simulations.

2.5 Simulation Results

This section presents the Monte Carlo simulation results to demonstrate the performance of the proposed 2-D IMDF algorithm. The RMSE of the IMDF algorithm is compared to those of several existing algebraic approaches for 2-D frequency estimation, as well as to the corresponding CRB. The CRB for N-D frequency estimation is given in Appendix C. In all simulations, the amplitude c_f for $f = 1, \ldots, F$, is set to one because the main purpose is frequency estimation.

In the experiments, the IMDF, MDF [24], MEMP [21], Unitary ESPRIT [13] and 2-D ESPRIT [23] algorithms are applied to estimate three 2-D frequencies from 1000 realizations of 20×20 noisy data sets in four cases as follows .

Case 1: Moderately close frequencies in both dimensions:

$$(\omega_1, \nu_1) = (0.41\pi, 0.61\pi), \quad (\omega_2, \nu_2) = (0.45\pi, 0.53\pi), \quad (\omega_3, \nu_3) = (0.37\pi, 0.57\pi).$$

Case 2: Very close frequencies in both dimensions:

$$(\omega_1, \nu_1) = (0.85\pi, 0.57\pi), \quad (\omega_2, \nu_2) = (0.87\pi, 0.62\pi), \quad (\omega_3, \nu_3) = (0.86\pi, 0.68\pi).$$

Case 3: Very close frequencies in the first dimension, well separated in the second dimension:

$$(\omega_1, \nu_1) = (0.85\pi, 0.20\pi), \quad (\omega_2, \nu_2) = (0.87\pi, 0.62\pi), \quad (\omega_3, \nu_3) = (0.86\pi, 0.85\pi).$$

Case 4: Very close frequencies in the second dimension, well separated in the first dimensions:

$$(\omega_1, \nu_1) = (0.20\pi, 0.85\pi), \quad (\omega_2, \nu_2) = (0.62\pi, 0.87\pi), \quad (\omega_3, \nu_3) = (0.85\pi, 0.86\pi).$$

Fig. 2.4 plots the averaged RMSE over the two dimensions for the aforementioned algorithms. All the legends are the same as in Fig. 2.4 (a). The associated CRB, and the STD derived from perturbation analysis of the 2-D IMDF algorithm are also plotted. SNR is defined as (see (1.5))

$$\mathrm{SNR} = 10\log_{10}\frac{\|\boldsymbol{A}\odot\boldsymbol{B}\boldsymbol{c}\|^2}{M_1 M_2 \sigma^2}.$$

For the IMDF algorithm, it is chosen that $K_1 = 5$ and $K_2 = 6$. The smoothing parameters for other algorithms are chosen as follows (notations are from the corresponding references): (i) for the MDF algorithm, $K_1 = L_1 = 10$ according to [24]; (ii) for the Unitary ESPRIT algorithm, $M_{\mathrm{sub1}} = 10$, $M_{\mathrm{sub2}} = 10$ according to [13]; (iii) for the MEMP and 2-D ESPRIT algorithms, $K = 6$ and $L = 6$ according to [21]. Automatic paring of frequency estimates in the two dimensions is achieved by the IMDF, MDF, and 2-D ESPRIT algorithms. Exhaustive search is used for paring in the MEMP algorithm, and simultaneous Schur decomposition is used for frequency pairing for the Unitary ESPRIT algorithm.

As shown in Fig. 2.4, when frequencies are very closely located along the first dimension but well separated in the second dimension, the IMDF algorithm suffers slight performance degradation when compared to other algorithms (Fig. 2.4 (c)), because the eigenvectors obtained using (2.22) may not be independent in this case. However, as expected, the performance of the IMDF algorithm is not affected when frequencies are well separated along the first dimension but very closely located in the second dimension (Fig. 2.4 (d)). In addition, if frequencies are very closely located in both dimensions, IMDF still provide competitive performance over eigenvalue-based approaches (Fig. 2.4 (b)). In all cases, the RMSE of the IMDF algorithm matches the results of the perturbation analysis very well at moderate to high SNRs.

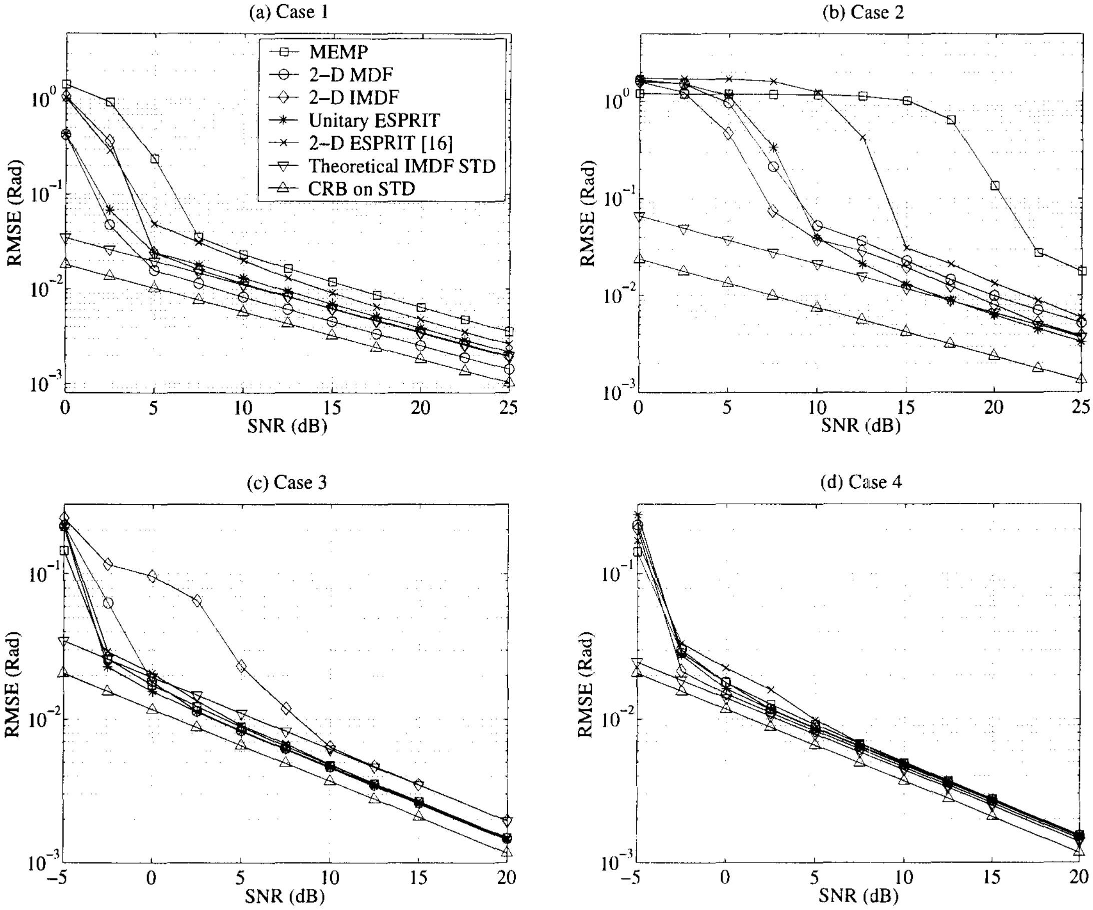


Figure 2.4. RMSE of 2-D frequency estimation. (a) Case 1: Moderately close frequencies in both dimensions; (b) Case 2: Very close frequencies in both dimensions; (c) Very close frequencies in the first dimension, well separated frequencies in the second dimension; (d) Very close frequencies in the second dimension, well separated frequencies in the first dimension.

CHAPTER 3

EIGENVECTOR-BASED ESTIMATION WITH WEIGHTING FACTORS

The IMDF algorithm discussed in the previous chapter has a drawback. In the noiseless case, the IMDF algorithm is not applicable when there are identical frequencies in the dimension along which the first step of smoothing is performed (i.e., (2.6) and (2.7) in the 2-D case, or (2.31) and (2.32) in the N-D case), because the frequencies are estimated from eigenvectors. For example, in the 2-D case, if there are identical frequencies in $\boldsymbol{\omega}$, then the corresponding columns of $\boldsymbol{T}_{sp}$ will be linear combinations of those in $\boldsymbol{T}$. Although the probability of the event that there exist identical frequencies in that dimension is almost zero in practice, the performance of the algorithm in noise case is degraded when there are very close frequencies in the dimension. This is because the estimation of eigenvectors are not accurate when there exist very close eigenvalues. This phenomenon is also illustrated in the simulation results in Section 2.5 of Chapter 2. This chapter proposes to use weighting factors to solve this problem.

Considering the data model of N-D frequency estimation from multiple snapshots (1.11), one can differentiate two situations: the sufficient snapshot case and finite snapshot case. In the former case, data smoothing is not necessary, while in the latter, data smoothing is necessary. Since the sufficient snapshot case is simpler, it is described at first, then the finite snapshot case is detailed. A thorough performance analysis of the proposed eigenvector-based algorithm in finite snapshot case is given because the optimization in Chapter 4 requires such study.

3.1 The Sufficient Snapshot Case

3.1.1 Eigenvector-based algorithms using weighting factors

According to Lemma 8, smoothing technique can not improve the identifiability. In this case, the sample covariance matrix of $\boldsymbol{x}$ in (1.11) should be exploited to estimate the N-D frequencies. This section presents an eigenvector-based frequency estimation algorithm using weighting factors when T is sufficiently large.

The sample covariance matrix $\widehat{\boldsymbol{R}}_x$ of $\boldsymbol{x}(t)$ in (1.11) is

$$\widehat{\boldsymbol{R}}_x := \tfrac{1}{T}\textstyle\sum_{t=1}^{T}\boldsymbol{x}(t)\boldsymbol{x}^H(t) = \tfrac{1}{T}\boldsymbol{X}\boldsymbol{X}^H. \tag{3.1}$$

One can also use following backward and forward smoothing technique to double the number of available snapshots: $\widetilde{\boldsymbol{X}} = [\boldsymbol{X} \;\; \boldsymbol{\Pi}_M\boldsymbol{X}^*]$, where $\boldsymbol{\Pi}_M$ is the backward permutation matrix of size $M \times M$. Then, one can use $\widetilde{\boldsymbol{X}}$ to replace the $\boldsymbol{X}$, and $2T$ to replace T in (3.1). In the noiseless case, it can be verified that $\boldsymbol{R}_x = \boldsymbol{A}\boldsymbol{R}_c\boldsymbol{A}^H$, where $\boldsymbol{R}_c$ is the sample covariance matrix of $\boldsymbol{c}(t)$

$$\boldsymbol{R}_c := \frac{1}{T}\sum_{t=1}^{T}\boldsymbol{c}(t)\boldsymbol{c}^H(t).$$

Since $\boldsymbol{A}$ is the Khatri-Rao product of multiple Vandermonde matrices, according to Property p3), if $M > F$, $\boldsymbol{A}$ is almost surely full column rank. If $\boldsymbol{R}_c$ is nonsingular and $T > F$, then $\boldsymbol{R}_x$ is of rank F. The EVD of $\boldsymbol{R}_x$ yields

$$\boldsymbol{R}_x = \boldsymbol{U}_s\Sigma_s\boldsymbol{U}_s^H,$$

where $\boldsymbol{U}_s$ has F columns that together span the column space of $\boldsymbol{R}_x$. The same space is also spanned by the columns of $\boldsymbol{A}$, therefore there exists a nonsingular matrix $\boldsymbol{T}^{-1}$ satisfying $\boldsymbol{U}_s = \boldsymbol{A}\boldsymbol{T}^{-1}$.

Similar to the IMDF algorithm [38], once the signal subspace $\boldsymbol{U}_s$ is obtained, one can construct two matrices from $\boldsymbol{U}_s$ whose general eigenvalues are the exponentials of the first dimension, and then use the general eigenvectors to estimate N-D frequencies. However, as mentioned before, the IMDF algorithm fails when there exist identical frequencies in the first dimension since the eigenvectors are not linearly independent anymore. Furthermore, it has been shown in [38] that the performance of the IMDF algorithm degrades if there are close frequencies in the first dimension. To address this problem, in the following, a method is presented to construct two matrices whose general eigenvalues are weighted sums of the N-D exponentials. The N-D frequencies are still resolved from the general eigenvectors. Although the form is quite different, the basic idea to avoid repeated eigenvalues is similar to that in [23].

One can define two selection matrices $\boldsymbol{J}_1$ and $\boldsymbol{J}_2$ as

$$\boldsymbol{J}_1 := \boldsymbol{J}_{1,1} \otimes \boldsymbol{J}_{1,2} \cdots \otimes \boldsymbol{J}_{1,N}, \quad \boldsymbol{J}_2 := \sum_{n=1}^{N} \alpha_n \widetilde{\boldsymbol{J}}_{2,n}, \tag{3.2}$$

where $\boldsymbol{J}_{1,n} = \left[\boldsymbol{I}_{M_n-1}\ \boldsymbol{0}_{(M_n-1)\times 1}\right]$, $\boldsymbol{J}_{2,n} = \left[\boldsymbol{0}_{(M_n-1)\times 1}\ \boldsymbol{I}_{M_n-1}\right]$, and

$$\widetilde{\boldsymbol{J}}_{2,n} = \left(\boldsymbol{J}_{1,1} \otimes \cdots \otimes \boldsymbol{J}_{1,n-1}\right) \otimes \boldsymbol{J}_{2,n} \otimes \left(\boldsymbol{J}_{1,n+1} \otimes \cdots \otimes \boldsymbol{J}_{1,N}\right).$$

Here $\{\alpha_n\}_{n=1}^{N}$ are complex weighting factors, which can be randomly chosen initially. As will be shown in Section 3.3, the MSEs of the frequency estimates are affected by these weighting factors in the noisy case. Next, one can obtain two equal-sized matrices $\boldsymbol{U}_1$ and $\boldsymbol{U}_2$ by

$$\boldsymbol{U}_1 := \boldsymbol{J}_1 \boldsymbol{U}_s = \boldsymbol{P}\boldsymbol{T}^{-1}, \tag{3.3}$$

$$\boldsymbol{U}_2 := \boldsymbol{J}_2 \boldsymbol{U}_s = \boldsymbol{P}\boldsymbol{D}(\boldsymbol{\zeta})\boldsymbol{T}^{-1}, \tag{3.4}$$

where $\boldsymbol{P} = \boldsymbol{A}_1^{(M_1-1)} \odot \boldsymbol{A}_2^{(M_2-1)} \odot \cdots \odot \boldsymbol{A}_N^{(M_N-1)}$. It is clear that $\boldsymbol{P}$ has full column rank almost surely if $F \leq \prod_{n=1}^{N}(M_n - 1)$. In (3.18), $\boldsymbol{\zeta} := [\zeta_1, \zeta_2, \cdots, \zeta_F]^T$, and $\zeta_f = \sum_{n=1}^{N} \alpha_n e^{j\omega_{f,n}}$. Then it can be verified that

$$\boldsymbol{U}_1^{\dagger}\boldsymbol{U}_2 = \boldsymbol{T}\boldsymbol{D}(\boldsymbol{\zeta})\boldsymbol{T}^{-1}.$$

Therefore $\boldsymbol{T}$ can be estimated by the eigenvector matrix of $\boldsymbol{U}_1^{\dagger}\boldsymbol{U}_2$. The matrix $\boldsymbol{P}$ can be estimated by

$$\boldsymbol{P} = \boldsymbol{U}_1 \boldsymbol{T}. \tag{3.5}$$

TABLE 3.1

An eigenvector-based algorithm for N-D frequency estimation using weighting factors

---

1. Given (1.11), compute the sample covariance matrix $\boldsymbol{R}$ according to (3.1), then compute the EVD of $\boldsymbol{R}$ to obtain the signal subspace $\boldsymbol{U}_s$.
2. Randomly select weighting factor $\{\alpha_n\}_{n=1}^N$, construct matrices $\boldsymbol{U}_1$ and $\boldsymbol{U}_2$ according to (3.3), and compute the EVD of $\boldsymbol{U}_1^\dagger \boldsymbol{U}_2$ to obtain $\boldsymbol{T}$.
3. Compute $\boldsymbol{P}$ by (3.5) and obtain the N-D frequency estimates by (2.27) and then (2.29).

Finally, the N-D frequencies can be estimated by dividing the elements in the obtained structural matrix $\boldsymbol{P}$ similar to (2.27). the algorithm is summarized in Table 3.1.

3.1.2 Performance Analysis

The performance of this algorithm in sufficient snapshot case is analyzed in [30]. It can be shown that $E\left[\Delta\omega_{f,n}\right] = 0$, and

$$E\left[(\Delta\omega_{f,n})^2\right] = \frac{\sigma^2}{4T\mu_n^2}\left\|\boldsymbol{\eta}_{f,n}\right\|^2 \sum_{i=1}^{F} \frac{\left|[\boldsymbol{T}]_{i,f}\right|^2 \lambda_i}{(\sigma^2-\lambda_i)^2}, \tag{3.6}$$

where $\boldsymbol{\eta}_{f,n}^T := \boldsymbol{\tau}_{f,n}^T\big(\boldsymbol{J}_1 - \boldsymbol{P}_{\text{sp}}\boldsymbol{D}_f\boldsymbol{P}_{\text{sp}}^\dagger(\boldsymbol{J}_1\zeta_f - \boldsymbol{J}_2)\big)$, $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_F$ are the F principle eigenvalues of $\boldsymbol{R}$, σ^2 is the noise variance, $\mu_n := (M_n-2)\prod_{\substack{i=1\\ i\neq n}}^N (M_i-1)$, and $\boldsymbol{\tau}_{f,n}$ are

$$\begin{aligned}
\boldsymbol{\tau}_{f,n}^T &:= [\boldsymbol{P}_{\text{sp}}]_{1,f}^{-1}\left(\boldsymbol{\phi}_{f,1}\otimes\cdots\otimes\boldsymbol{\phi}_{f,n-1}\right)\otimes\boldsymbol{\varphi}_{f,n}\otimes\left(\boldsymbol{\phi}_{f,n+1}\otimes\cdots\otimes\boldsymbol{\phi}_{f,N}\right),\\
\boldsymbol{\phi}_{f,n} &:= \begin{bmatrix}1 \; e^{-j\omega_{f,n}} \;\ldots\; e^{-j(M_n-2)\omega_{f,n}}\end{bmatrix}^T : (M_n-1)\times 1,\\
\boldsymbol{\varphi}_{f,n} &:= \begin{bmatrix}-1 \; 0 \;\cdots\; 0 \; e^{-j(M_n-2)\omega_{f,n}}\end{bmatrix}^T : (M_n-1)\times 1.
\end{aligned}$$

Notice if one uses forward and backward smoothing, the number of snapshots T should be replace by $2T$ in (3.6).

When $T \to \infty$, $E[\Delta\omega_{f,n}^2] \to 0$, therefore the proposed algorithm is consistent in terms of the number of snapshots. When $\sigma \to 0$, $E[\Delta\omega_{f,n}^2] \to 0$, therefore the algorithm is also consistent in terms of SNR.

When $\sigma \to 0$, $\boldsymbol{r}_{f,n}^T$, λ_i and $\boldsymbol{t}_f$ will converge to their noiseless counterparts, which are mainly determined by $\omega_{f,n}$, $f = 1, \cdots, F$ and α_n, $n = 1, \cdots, N$. Therefore $E[\Delta\omega_{f,n}^2]$ can be considered as an asymptotically linear function of σ^2. Based on the 1-D results in [19], one can obtain the CRB for N-D frequency estimation as follows:

$$\mathrm{CRB} = \mathrm{Diag}\left\{ \frac{\sigma^2}{2T} \left[\mathcal{R}(\boldsymbol{D}^H (\boldsymbol{I} - \boldsymbol{A}\boldsymbol{A}^\dagger)\boldsymbol{D}) \circ (\boldsymbol{K} \otimes \boldsymbol{R}_c)^T \right]^{-1} \right\}. \tag{3.7}$$

where $\boldsymbol{K}$ is an $N \times N$ matrix with all the elements equal 1, $\circ$ is the Hadamard product, and

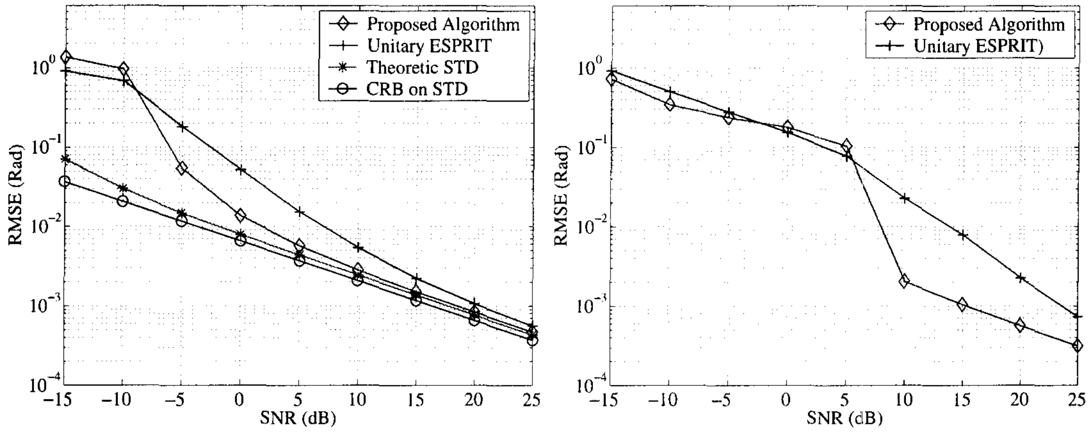
$$\begin{array}{rll}
\boldsymbol{D} = & \begin{bmatrix} \boldsymbol{d}_{1,1} & \boldsymbol{d}_{2,1} & \cdots & \boldsymbol{d}_{F,1} & \boldsymbol{d}_{1,2} & \cdots & \boldsymbol{d}_{F,N} \end{bmatrix} & : M \times NF, \\
\boldsymbol{d}_{f,n} = & (\boldsymbol{\theta}_{f,1} \otimes \cdots \otimes \boldsymbol{\theta}_{f,n-1}) \otimes (\boldsymbol{\Omega}_n \boldsymbol{\theta}_{f,n}) \otimes (\boldsymbol{\theta}_{f,n+1} \otimes \cdots \otimes \boldsymbol{\theta}_{f,N}) & : M \times 1, \\
\boldsymbol{\theta}_{f,n} = & \begin{bmatrix} 1 & e^{j\omega_{f,n}} & \cdots & e^{j(M_n - 1)\omega_{f,n}} \end{bmatrix}^T & : M_n \times 1, \\
\boldsymbol{\Omega}_n = & \boldsymbol{D}(\boldsymbol{\varphi}_n), \text{ with } \boldsymbol{\varphi}_n = \begin{bmatrix} 0 & j & \cdots & j(M_n - 1) \end{bmatrix}^T & : M_n \times M_n.
\end{array}$$

With the decrease of noise power σ^2, the ratio of the error variance and the CRB converges to a constant only determined by the distribution of the N-D frequencies and $\{\alpha_n\}$. If one randomly selects the parameters α_n, simulation results show that the ratio ranges approximately from 1.1 to 4. Therefore the proposed algorithm is relatively efficient. However, in order to further guarantee the performance in noisy case, one should optimize the parameters $\{\alpha_n\}_{n=1}^N$. An optimization strategy to select the parameters is proposed in [31].

3.1.3 Simulation Results

Thanks to the introduction of the random weighting factors, eigenvectors of $\boldsymbol{U}_1^\dagger \boldsymbol{U}_2$ are almost surely distinct even if there exist identical frequencies in one or more dimensions. In practice, identical frequencies are commonly encountered, for example, in joint delay and angle estimation, signals may come from a common direction with different delays due to a common reflector. As another example, in frequency hopping systems, frequency channels are pre-assigned and frequencies may collide. However, note that it is possible that the algorithm may also yields some poorer frequency estimates if both frequencies and weighting factors are randomly generated. This section uses simulation results to show that even in this case, the average performance the proposed algorithm is better than eigenvalue-based algorithms due to the fact that eigenvalues are almost surely distinct when frequencies are randomly generated with or without random weighting factors.

The eigenvector-based algorithm described in Section 3.1.1 is applied to estimate three sets of 3-D frequencies from noisy data with $T = 217$ snapshots, each of size 6×6×6, i.e., $F = 3$, $N = 3$, and $M_1 = M_2 = M_3 = 6$. For comparison, the Unitary ESPRIT [13] is also applied in the experiments. In the first experiment, the three frequency sets are fixed: $(\omega_{1,1}, \omega_{1,2}, \omega_{1,3}) = (0.59\pi, 0.54\pi, 0.60\pi)$, $(\omega_{2,1}, \omega_{2,2}, \omega_{2,3}) = (0.61\pi, 0.44\pi, 0.60\pi)$, and $(\omega_{3,1}, \omega_{3,2}, \omega_{3,3}) = (0.60\pi, 0.34\pi, 0.34\pi)$. The frequencies are closely located in the first dimension and there exist identical frequencies in the third dimension. $\{c_f(t)\}_{f=1}^F$, $t = 1, \ldots, T$, are drawn from complex Gaussian distributions. The RMSEs of the two algorithms over 1000 trials are plotted. The RMSE is averaged over all frequencies in the three dimensions. The weighting factors are $\alpha_1 = 0.3 + 0.9j, \alpha_2 = -0.5 + 0.9j, \alpha_3 = 0.7 + 0.7j$. The associated theoretic standard deviation (STD) of the proposed algorithm and the CRB on STD are also plotted. SNR is defined as (c.f., (1.11)): $\mathrm{SNR} = 10\log_{10}(\sum_{t=1}^T \|\boldsymbol{A}\boldsymbol{c}(t)\|_2^2)/(MT\sigma^2)$. It can be seen from the Fig. 3.1 (a) that the RMSE of the proposed algorithm matches the theoretic STD in high SNR range, and it outperforms the Unitary ESPRIT algorithm for a wide SNR range. In the second experiment, all frequencies, amplitudes, and weighting factors are random generated in each of the 1000 trials. Fig. 3.1 (b) plots the RMSE results. It can be seen that the performance of the proposed algorithm is still better than Unitary ESPRIT in for a wide SNR range, but the threshold



(a) (b)

Figure 3.1. RMSE of 3-D frequency estimation with multiple snapshots: (a) frequency tuples are fixed in all realizations; (b) frequency tuples are randomly generated in each realization.

is moved towards higher SNR compared to the results shown in Fig. 3.1 (a).

3.2 The Finite Snapshot Case

For the finite snapshot case, where data smoothing is applied, an eigenvector-based algorithm using random weighting factors is also developed. This section describes the algorithm. The performance analysis to the proposed algorithm in noisy case is detailed in Section 3.3. This algorithm can be considered as an improved version of the IMDF algorithm introduced in Chapter 2. For simplicity of exposition, the algorithm is developed in the noiseless case.

Given (1.11) in the noiseless case, one can apply the smooth operator $\mathcal{S}$ defined in Lemma 1 to every snapshot $\boldsymbol{x}(t)$, and obtain

$$\boldsymbol{X}_{\mathcal{S}}(t) := \mathcal{S}[\boldsymbol{x}(t)] = \boldsymbol{G}\boldsymbol{D}\big(\boldsymbol{c}(t)\big)\boldsymbol{H}^T, \tag{3.8}$$

where

$$\boldsymbol{G} := \boldsymbol{A}_1^{(K_1)} \odot \boldsymbol{A}_2^{(K_2)} \cdots \odot \boldsymbol{A}_N^{(K_N)},$$
$$\boldsymbol{H} := \boldsymbol{A}_1^{(L_1)} \odot \boldsymbol{A}_2^{(L_2)} \cdots \odot \boldsymbol{A}_N^{(L_N)}.$$

The positive integers K_n and L_n, $n = 1, \cdots, N$, are chosen such that

$$K_n + L_n = M_n + 1, \quad 1 \leq n \leq N. \tag{3.9}$$

To further explore the data structure, one can perform the forward-backward smoothing on the data vector $\boldsymbol{x}(t)$ in (1.11). Define

$$\boldsymbol{y}(t) := \boldsymbol{\Pi}_M \boldsymbol{x}^*(t), \tag{3.10}$$

where $\boldsymbol{\Pi}_M$ is an $M \times M$ backward permutation matrix. It can be verified that

$$\boldsymbol{y}(t) = \boldsymbol{A}\widetilde{\boldsymbol{c}}(t),$$

where $\widetilde{\boldsymbol{c}}(t) = [\widetilde{c}_1(t), \widetilde{c}_2(t), \cdots, \widetilde{c}_F(t)]^T$ with $\widetilde{c}_f(t) = c_f^*(t)e^{-j\phi_f}$, and ϕ_f is defined in (2.39). Applying the same technique to $\boldsymbol{y}(t)$ that is used to construct $\boldsymbol{X}_S(t)$ from $\boldsymbol{x}(t)$, one can obtain

$$\boldsymbol{Y}_S(t) := \mathcal{S}[\boldsymbol{y}(t)] = \boldsymbol{G}\boldsymbol{D}(\widetilde{\boldsymbol{c}}(t))\boldsymbol{H}^T.$$

Then one can collect all the smoothed data matrices to obtain

$$\widetilde{\boldsymbol{X}} := \begin{bmatrix} \boldsymbol{X}_S(1) & \boldsymbol{X}_S(2) & \cdots & \boldsymbol{X}_S(T) & \boldsymbol{Y}_S(T) & \boldsymbol{Y}_S(T-1) & \cdots & \boldsymbol{Y}_S(1) \end{bmatrix}. \tag{3.11}$$

Define $K := \prod_{n=1}^N K_n$ and $L := \prod_{n=1}^N L_n$. The size of $\widetilde{\boldsymbol{X}}$ is $K \times 2TL$. It can be verified that

$$\widetilde{\boldsymbol{X}} = \boldsymbol{G}(\boldsymbol{H} \odot \boldsymbol{B})^T = \boldsymbol{G}\widetilde{\boldsymbol{H}}^T, \tag{3.12}$$

where $\boldsymbol{B}$ and $\widetilde{\boldsymbol{H}}$ are defined as

$$\boldsymbol{B} := \begin{bmatrix} \boldsymbol{C}^T \\ \boldsymbol{\Pi}_T \boldsymbol{C}^H \boldsymbol{D}(\boldsymbol{\beta}) \end{bmatrix}, \tag{3.13}$$

$$\widetilde{\boldsymbol{H}} := \boldsymbol{A}_1^{(L_1)} \odot \boldsymbol{A}_2^{(L_2)} \cdots \odot \boldsymbol{A}_N^{(L_N)} \odot \boldsymbol{B}. \tag{3.14}$$

A key step of the algorithm is the construction of $\widetilde{\boldsymbol{X}}$ to ensure that it is of rank F almost surely. In (3.12), $\boldsymbol{G}$ is the Khatri-Rao product of multiple Vandermonde matrices, invoking Property p3) of Khatri-Rao product, $\boldsymbol{G}$ has full column rank almost surely if $\prod_{n=1}^{N} K_n \geq F$. Similar to Lemma 3, one can prove that $\widetilde{\boldsymbol{H}}$ has full column rank almost surely if $2T\prod_{n=1}^{N} L_n \geq F$. Based on the Sylvester's inequality

$$\operatorname{rank}(\boldsymbol{G}) + \operatorname{rank}\left(\widetilde{\boldsymbol{H}}^T\right) - F \leq \operatorname{rank}\left(\boldsymbol{G}\widetilde{\boldsymbol{H}}^T\right) \leq \min\left\{\operatorname{rank}(\boldsymbol{G}), \operatorname{rank}\left(\widetilde{\boldsymbol{H}}^T\right)\right\},$$

$\widetilde{\boldsymbol{X}}$ is of rank F almost surely. The singular value decomposition (SVD) of $\widetilde{\boldsymbol{X}}$ yields

$$\widetilde{\boldsymbol{X}} = \boldsymbol{U}_s \boldsymbol{\Sigma}_s \boldsymbol{V}_s^H, \tag{3.15}$$

where $\boldsymbol{U}_s$ has F columns that together span the column space of $\widetilde{\boldsymbol{X}}$. Since the same space is spanned by the columns of $\boldsymbol{G}$, there exists an $F \times F$ nonsingular matrix $\boldsymbol{T}^{-1}$ such that

$$\boldsymbol{U}_s = \boldsymbol{G}\boldsymbol{T}^{-1}. \tag{3.16}$$

Similar to (3.3), one can obtain matrix pencil $\boldsymbol{U}_1 = \boldsymbol{J}_1\boldsymbol{U}_s$ and $\boldsymbol{U}_2 = \boldsymbol{J}_2\boldsymbol{U}_s$, where the selection matrix $\boldsymbol{J}_1$ and $\boldsymbol{J}_2$ is define similar to (3.2). Notice that $\boldsymbol{J}_{1,n}$ and $\boldsymbol{J}_{2,n}$ change to

$$\boldsymbol{J}_{1,n} := \left[\boldsymbol{I}_{K_n-1}\ \boldsymbol{0}_{(K_n-1)\times 1}\right], \quad \boldsymbol{J}_{2,n} := \left[\boldsymbol{0}_{(K_n-1)\times 1}\ \boldsymbol{I}_{K_n-1}\right]. \tag{3.17}$$

According to the Property p2) of Khatri-Rao product in Chapter 2: $(\boldsymbol{A}\otimes\boldsymbol{B})(\boldsymbol{C}\odot\boldsymbol{D}) = \boldsymbol{A}\boldsymbol{C} \odot \boldsymbol{B}\boldsymbol{D}$, it can be verified that

$$\boldsymbol{U}_1 = \boldsymbol{P}\boldsymbol{T}^{-1}, \quad \boldsymbol{U}_2 = \boldsymbol{P}\boldsymbol{D}(\boldsymbol{\zeta})\boldsymbol{T}^{-1}. \tag{3.18}$$

where $\boldsymbol{P} = \boldsymbol{A}_1^{(K_1-1)} \odot \boldsymbol{A}_2^{(K_2-1)} \odot \cdots \odot \boldsymbol{A}_N^{(K_N-1)}$. It is clear that $\boldsymbol{P}$ has full column rank almost surely if $F \leq \prod_{n=1}^{N}(K_n - 1)$. In (3.18), $\boldsymbol{\zeta} := [\zeta_1, \zeta_2, \cdots, \zeta_F]^T$, and $\zeta_f = \sum_{n=1}^{N} \alpha_n e^{j\omega_{f,n}}$. One can resolve $\boldsymbol{T}$ from the matrix pencil $\boldsymbol{U}_1$ and $\boldsymbol{U}_2$ via least-squares or total-least-squares approaches [3]. Here, one can use the least-squares approach. $\boldsymbol{T}$ is retrieved from the following EVD up to column permutation and scaling ambiguity.

$$\boldsymbol{U}_1^{\dagger}\boldsymbol{U}_2 = \boldsymbol{T}\boldsymbol{D}(\boldsymbol{\zeta})\boldsymbol{T}^{-1}. \tag{3.19}$$

Clearly one can choose $\{\alpha_n\}_{n=1}^{N}$ to ensure that the elements of $\boldsymbol{\zeta}$ are distinct even if there exist identical frequencies in one or more dimensions, but this is not guaranteed by randomly generated $\{\alpha_n\}_{n=1}^N$. Chapter 4 will discuss how to choose the weighting factors. Suppose that the EVD of $\boldsymbol{U}_1^\dagger \boldsymbol{U}_2$ gives $\boldsymbol{T}_{\text{sp}} = \boldsymbol{T}\boldsymbol{\Lambda}\boldsymbol{\Delta}$, where $\boldsymbol{\Lambda}$ is a nonsingular diagonal column scaling matrix and $\boldsymbol{\Delta}$ is a permutation matrix. Once one obtain $\boldsymbol{T}_{\text{sp}}$, he can retrieve $\boldsymbol{P}$ up to column permutation and scaling ambiguity according to

$$\boldsymbol{P}_{\text{sp}} = \boldsymbol{U}_1 \boldsymbol{T}_{\text{sp}} = \boldsymbol{P}\boldsymbol{\Lambda}\boldsymbol{\Delta}. \tag{3.20}$$

Notice that $\boldsymbol{P}$ is the Khatri-Rao product of N Vandermonde matrices, and there are F columns in $\boldsymbol{P}$. The N frequencies of the same N-D component appear in the same column of $\boldsymbol{P}$. In other words, for fixed f, $\{\omega_{f,n}\}_{n=1}^N$ appear in the same column of $\boldsymbol{P}$. Thanks to this structure, one can obtain F N-D frequency components by dividing suitably chosen elements of the aforementioned columns of $\boldsymbol{P}_{\text{sp}}$. Therefore the column scaling and permutation will not have a material effect on the algorithm. For this reason, subscript "$_{\text{sp}}$" is dropped from now on as long as it is clear from the context. Suppose that $\{e^{j\omega_{f,n}}\}_{n=1}^N$ appears in the f-th column of $\boldsymbol{P}$. Then, $e^{j\omega_{f,n}}$ can be obtained by anyone of the following quotients

$$e^{j\omega_{f,n}} = \frac{p_{k,f}}{p_{k-K_n',f}}, \qquad \text{mod}\ (k-1, K_{n-1}') \geq K_n', \ \text{for } f = 1, \ldots, F, \tag{3.21}$$

where $1 \leq k \leq K_0'$, $p_{k,f}$ is the (k,f)-th element of $\boldsymbol{P}$. Define $K_n' := \prod_{p=n+1}^N (K_p - 1)$ for $0 \leq n \leq N-1$, and $K_N' := 1$. Notice that the frequencies are automatically paired because the frequencies ($\omega_{f,n}$, $n = 1, \ldots, N$) of the same N-D component (the f-th component) are obtained from the same column of $\boldsymbol{P}$.

If the data observations are noisy as given in (1.11), the above algorithm will give an estimate of $\boldsymbol{P}$ as $\widehat{\boldsymbol{P}}$. In order to reduce the MSEs of frequency estimates, one can use the average of all the quotients in (3.21) to obtain an estimate of the N-D exponential. In other words, $e^{j\omega_{f,n}}$ is estimated by the following average

$$\widehat{e^{j\omega_{f,n}}} = \frac{1}{\mu_n} \sum_{\substack{k=1 \\ \text{mod}\left(k-1, K_{n-1}'\right) \geq K_n'}}^{K_0'} \frac{\widehat{p}_{k,f}}{\widehat{p}_{k-K_n',f}}, \qquad n = 1, \ldots, N, \tag{3.22}$$

where $\mu_n = K_0'(K_n-2)/(K_n-1)$. The average is also the so-called "circular mean" in direction statistics [36]. Finally the frequency estimates are obtained by

$$\widehat{\omega}_{f,n} = \mathcal{I}\left(\ln \widehat{e^{j\omega_{f,n}}}\right). \tag{3.23}$$

After the frequency estimates are obtained, the amplitude matrix $\boldsymbol{C}$ can be obtained by solving (1.11) using a least-squares approach.

It can be verified that the ID bound of above algorithm is

$$F \leq \max_{\substack{K_n+L_n=M_n+1\\ 1\leq K_n\leq M_n\\ 1\leq n\leq N}} \min\left(\prod_{n=1}^{N}(K_n-1),\ 2T\prod_{n=1}^{N}L_n\right). \tag{3.24}$$

When $N=2$, the identifiability condition (3.24) becomes

$$F \leq \max_{\substack{K_1+L_1=M_1+1\\ K_2+L_2=M_2+1}} \min\left\{(K_1-1)(K_2-1),\ 2TL_1L_2\right\}. \tag{3.25}$$

The characters of the ID bound is given in [32], which is similar to those of the IMDF algorithm in Chapter 2.

3.3 Perturbation Analysis in the Finite Snapshot Case

This section uses perturbation analysis to derive the theoretical MSEs for the proposed algorithm in Section 3.2 in the noisy case. The following two assumptions are needed for this analysis.

1. In the noiseless case, the nonzero singular values of $\widetilde{\boldsymbol{X}}$ in (3.15) are distinct, and the eigenvalues of $\boldsymbol{U}_1^{\dagger}\boldsymbol{U}_2$ in (3.19) are distinct.

2. The eigenvectors in $\boldsymbol{T}$ of (3.19) are normalized, and the permutation matrix $\boldsymbol{\Delta} = \boldsymbol{I}$ since permutation does not affect the MSEs.

3.3.1 The Perturbation of Singular Vectors in SVD

Singular Value Decomposition (SVD) plays an important role in signal processing. For example, subspace-based algorithms in array signal processing depends on SVD or eigenvalue decomposition to obtain a basis of the desired signal and/or noise subspaces. Principal component analysis (PCA) uses SVD to retrieve principal components from noisy observations. Therefore, perturbation analysis of SVD is important for these algorithms. Consider the following observation matrix

$$\widehat{\boldsymbol{X}} = \boldsymbol{X} + \Delta \boldsymbol{X}, \tag{3.26}$$

where $\widehat{\boldsymbol{X}}$ is a perturbed version of the data matrix $\boldsymbol{X}$, and $\Delta \boldsymbol{X}$ is the perturbation. The subspace decomposition of $\boldsymbol{X}$ is

$$\boldsymbol{X} = \boldsymbol{U}\Sigma\boldsymbol{V}^H = \boldsymbol{U}_s\Sigma_s\boldsymbol{V}_s^H + \boldsymbol{U}_n\Sigma_n\boldsymbol{V}_n^H = \boldsymbol{U}_s\Sigma_s\boldsymbol{V}_s^H, \tag{3.27}$$

where $\boldsymbol{U}_s$, associated with nonzero singular values, spans the column space of $\boldsymbol{X}$, which is also called the signal subspace, and $\boldsymbol{U}_n$, associated with zero singular values ($\boldsymbol{\Sigma}_n = \boldsymbol{0}$), spans the orthogonal space of $\boldsymbol{U}_s$, which is also called the noise subspace. Similarly, the subspace decomposition of $\widehat{\boldsymbol{X}}$ is given by

$$\widehat{\boldsymbol{X}} = \widehat{\boldsymbol{U}}\widehat{\Sigma}\widehat{\boldsymbol{V}}^H = \widehat{\boldsymbol{U}}_s\widehat{\Sigma}_s\widehat{\boldsymbol{V}}_s^H + \widehat{\boldsymbol{U}}_n\widehat{\Sigma}_n\widehat{\boldsymbol{V}}_n^H. \tag{3.28}$$

Due to the perturbation $\Delta \boldsymbol{X}$, all the quantities in the right hand side (RHS) of (3.28) may differ from those in the RHS of (3.27). For example, one can express $\widehat{\boldsymbol{U}}_s$ as $\widehat{\boldsymbol{U}}_s = \boldsymbol{U}_s + \Delta\boldsymbol{U}_s$, where $\Delta\boldsymbol{U}_s$ is the perturbation of the singular vectors that span the signal subspace.

Perturbation analysis results for subspace decomposition have been well documented in the literature. For example, the impact of the phase factor on the statistics of eigenvectors of sample covariance matrices is investigated in [39]. The perturbation of invariance subspace is considered in [40, 41]. Based on the results in [40, 41], the first-order approximation of $\Delta\boldsymbol{U}_s$ is given in [28]. Furthermore, [42] and [43] give the second-order approximation using similar arguments as in [28]. It is shown in [28, 42, 43] that the contribution of the noise subspace spanned by $\boldsymbol{U}_n$ to $\Delta\boldsymbol{U}_s$ is of the first order of $\Delta\boldsymbol{X}$. It is argued that the contribution of the signal subspace spanned by $\boldsymbol{U}_s$ to $\Delta\boldsymbol{U}_s$ is of the second order of $\Delta\boldsymbol{X}$, and thus the contribution of the signal subspace to $\Delta\boldsymbol{U}_s$ is ignored when the first-order perturbation is considered.

This subsection analyze the perturbation of $\boldsymbol{U}_s$ from another perspective. Different from the existing results in [43, 42, 28], it is found that the contribution of the signal subspace to $\Delta\boldsymbol{U}_s$ is also of the first order of $\Delta\boldsymbol{X}$.

Let the SVD of an $M \times N$ (assuming $M \leq N$) matrix $\boldsymbol{X}$ in the noiseless case be given as in (3.27). Suppose that $\boldsymbol{X}$ is of rank F ($F \leq M$). The columns in $\boldsymbol{U}_s$, associated with the F non-zero distinct real singular values (i.e., $\sigma_1 > \sigma_2 > \cdots > \sigma_F$), span the signal subspace, while the columns in $\boldsymbol{U}_n$, associated with the remaining $M-F$ zero singular values (i.e., $\sigma_{F+1} = \cdots = \sigma_M = 0$), span the orthogonal noise subspace. The perturbed data matrix is $\widehat{\boldsymbol{X}} = \boldsymbol{X} + \Delta\boldsymbol{X}$, whose SVD is given by (3.28). The f-th left singular vector of $\widehat{\boldsymbol{X}}$ is $\widehat{\boldsymbol{u}}_f = \boldsymbol{u}_f + \Delta\boldsymbol{u}_f$, for $f = 1, \ldots, F$.

The result on the first-order perturbation of singular vectors is summarized in the following theorem.

Theorem 4: A first-order approximation of the perturbation $\Delta\boldsymbol{u}_f$ is given by

$$\Delta\boldsymbol{u}_f = \boldsymbol{U}_s\boldsymbol{D}_f\boldsymbol{U}_s^H\Delta\boldsymbol{X}\boldsymbol{v}_f\sigma_f + \boldsymbol{U}_s\boldsymbol{D}_f\Sigma_s\boldsymbol{V}_s^H\Delta\boldsymbol{X}^H\boldsymbol{u}_f + \boldsymbol{U}_n\boldsymbol{U}_n^H\Delta\boldsymbol{X}\boldsymbol{v}_f\sigma_f^{-1}, \tag{3.29}$$

where $\boldsymbol{v}_f$ is the right singular vector corresponding to σ_f, and $\boldsymbol{D}_f$ is a diagonal matrix of size $F \times F$ with the (f,f)-th element being zero and (g,g)-th element being $\frac{1}{\sigma_f^2-\sigma_g^2}$, for $g = 1, \ldots, F$, $g \neq f$. Collectively, one can express $\Delta\boldsymbol{U}_s$ as

$$\Delta\boldsymbol{U}_s = [\Delta\boldsymbol{u}_1 \;\; \Delta\boldsymbol{u}_2 \;\; \cdots \;\; \Delta\boldsymbol{u}_F] = \boldsymbol{U}_s\boldsymbol{R} + \boldsymbol{U}_n\boldsymbol{U}_n^H\Delta\boldsymbol{X}\boldsymbol{V}_s\Sigma_s^{-1}, \tag{3.30}$$

where $\boldsymbol{R} = \boldsymbol{D} \diamond \left(\boldsymbol{U}_s^H\Delta\boldsymbol{X}\boldsymbol{V}_s\Sigma_s + \Sigma_s\boldsymbol{V}_s^H\Delta\boldsymbol{X}^H\boldsymbol{U}_s\right)$. Here $\diamond$ denotes the Hadamard (element-wise) product, the (g,f)th element of $\boldsymbol{D}$ is $\boldsymbol{D}(g,f) = \frac{1}{\sigma_f^2-\sigma_g^2}$, for $g \neq f$, and $\boldsymbol{D}(f,f) = 0$.

Proof: The following assumptions are needed to prove the theorem:

a1) The phase factors of singular vectors are chosen such that the perturbation $\Delta\boldsymbol{u}_f$ is orthogonal to $\boldsymbol{u}_f$ for $f = 1, \ldots, F$(see [39] for more detail).

a2) The perturbation $\Delta \boldsymbol{X}$ is so small that the order of the singular values in $\widehat{\boldsymbol{\Sigma}}_s$ are the same as the order of the corresponding singular values in $\boldsymbol{\Sigma}_s$.

Using $\boldsymbol{X}^H \boldsymbol{u}_f = \sigma_f \boldsymbol{v}_f$ and $\boldsymbol{X} \boldsymbol{v}_f = \sigma_f \boldsymbol{u}_f$, one can obtain

$$\boldsymbol{X}\boldsymbol{X}^H \boldsymbol{u}_f = \sigma_f^2 \boldsymbol{u}_f. \tag{3.31}$$

Since only the first-order perturbation is considered, one can use differentiation to obtain the first-order quantities. Differentiating (3.31), one can obtain

$$\Delta \boldsymbol{X}\boldsymbol{X}^H \boldsymbol{u}_f + \boldsymbol{X}\Delta \boldsymbol{X}^H \boldsymbol{u}_f + \boldsymbol{X}\boldsymbol{X}^H \Delta \boldsymbol{u}_f = 2\sigma_f \Delta\sigma_f \boldsymbol{u}_f + \sigma_f^2 \Delta \boldsymbol{u}_f. \tag{3.32}$$

Since $\boldsymbol{u}_f$ is a normalized eigenvector of $\boldsymbol{X}\boldsymbol{X}^H$, according to [44, Chapter 2] and Assumption a1), one can express $\Delta \boldsymbol{u}_f$ as a linear combination of all other eigenvectors such that

$$\Delta \boldsymbol{u}_f = \sum_{g=1,\ g \neq f}^{M} k_{f,g} \boldsymbol{u}_g, \tag{3.33}$$

where $k_{f,g}$ is the coefficient to be determined. Substituting (3.33) into (3.32) and left multiplying both sides by $\boldsymbol{u}_g^H$, one can obtain

$$k_{f,g} = \frac{\boldsymbol{u}_g^H \left(\Delta \boldsymbol{X}\boldsymbol{X}^H + \boldsymbol{X}\Delta \boldsymbol{X}^H\right) \boldsymbol{u}_f}{\sigma_f^2 - \sigma_g^2}, \tag{3.34}$$

where one should use the facts that $\boldsymbol{u}_g^H \boldsymbol{X}\boldsymbol{X}^H \boldsymbol{u}_f = \sigma_g^2 \delta_{g,f}$ and $\boldsymbol{u}_g^H \boldsymbol{u}_f = \delta_{g,f}$, with $\delta_{g,f}$ being the Kronecker delta symbol ($\delta_{g,f} = 1$ when $g = f$, and $\delta_{g,f} = 0$ when $g \neq f$). Note that (3.33) and (3.34) are consistent with the results on the perturbation of eigenvectors of a covariance matrix given in [39]. Substituting (3.34) to (3.33), and noticing that $\sigma_g = 0$ for $g \geq F + 1$, one can get

$$\begin{aligned} \Delta \boldsymbol{u}_f &= \sum_{\substack{g=1 \\ g \neq f}}^{F} \boldsymbol{u}_g \frac{\boldsymbol{u}_g^H \left(\Delta \boldsymbol{X}\boldsymbol{X}^H + \boldsymbol{X}\Delta \boldsymbol{X}^H\right) \boldsymbol{u}_f}{\sigma_f^2 - \sigma_g^2} + \sum_{g=F+1}^{M} \boldsymbol{u}_g \frac{\boldsymbol{u}_g^H \left(\Delta \boldsymbol{X}\boldsymbol{X}^H + \boldsymbol{X}\Delta \boldsymbol{X}^H\right) \boldsymbol{u}_f}{\sigma_f^2} \\ &= \boldsymbol{U}_s \boldsymbol{D}_f \boldsymbol{U}_s^H \left(\Delta \boldsymbol{X}\boldsymbol{X}^H + \boldsymbol{X}\Delta \boldsymbol{X}^H\right) \boldsymbol{u}_f + \boldsymbol{U}_n \boldsymbol{U}_n^H \left(\Delta \boldsymbol{X}\boldsymbol{X}^H + \boldsymbol{X}\Delta \boldsymbol{X}^H\right) \boldsymbol{u}_f \sigma_f^{-2}. \end{aligned} \tag{3.35}$$

Using $\boldsymbol{X}^H\boldsymbol{u}_f = \sigma_f\boldsymbol{v}_f$ and $\boldsymbol{X} = \boldsymbol{U}_s\Sigma_s\boldsymbol{V}_s^H$, one can write the first term in the RHS of (3.35) as

$$\boldsymbol{U}_s\boldsymbol{D}_f\boldsymbol{U}_s^H(\Delta\boldsymbol{X}\boldsymbol{X}^H + \boldsymbol{X}\Delta\boldsymbol{X}^H)\boldsymbol{u}_f = \boldsymbol{U}_s\boldsymbol{D}_f\boldsymbol{U}_s^H\Delta\boldsymbol{X}\boldsymbol{v}_f\sigma_f + \boldsymbol{U}_s\boldsymbol{D}_f\Sigma_s\boldsymbol{V}_s^H\Delta\boldsymbol{X}^H\boldsymbol{u}_f. \tag{3.36}$$

Since $\boldsymbol{U}_n^H\boldsymbol{X} = \boldsymbol{0}$, the second summation term in the RHS of (3.35) can be written as

$$\boldsymbol{U}_n\boldsymbol{U}_n^H(\Delta\boldsymbol{X}\boldsymbol{X}^H + \boldsymbol{X}\Delta\boldsymbol{X}^H)\boldsymbol{u}_f\sigma_f^{-2} = \boldsymbol{U}_n\boldsymbol{U}_n^H\Delta\boldsymbol{X}\boldsymbol{v}_f\sigma_f^{-1}. \tag{3.37}$$

Summarizing (3.35)–(3.37), one can obtain (3.29) and (3.30). ■

In (3.30), the term $\boldsymbol{U}_s\boldsymbol{R}$ is the contribution of the signal subspace to $\Delta\boldsymbol{U}_s$, while the term $\boldsymbol{U}_n\boldsymbol{U}_n^H\Delta\boldsymbol{X}\boldsymbol{V}_s\Sigma_s^{-1}$ is the contribution of the noise subspace. It is clear that the contributions of both the signal subspace and the noise subspace are of the first order of $\Delta\boldsymbol{X}$.

It can be observed from (3.30) that if any singular value becomes close to σ_f, then the perturbation in $\boldsymbol{u}_f$ grows toward infinity. This is because if $\sigma_g - \sigma_f$ is small enough, σ_g and σ_f become nearly identical singular values, the corresponding singular vectors are not unique any more. In this case, the first-order approximation fails to approach $\widehat{\boldsymbol{u}}_f - \boldsymbol{u}_f$. Higher-order terms may have to be taken into account.

Remark 2: In [43], it is argued that the contribution of the signal subspace is of the second order of $\Delta\boldsymbol{X}$, and the first-order perturbation of $\boldsymbol{U}_s$ is thus given as

$$\Delta\boldsymbol{U}_s = \boldsymbol{U}_n\boldsymbol{U}_n^H\Delta\boldsymbol{X}\boldsymbol{V}_s\Sigma_s^{-1}. \tag{3.38}$$

The argument in [43] depends on the existence of a matrix $\boldsymbol{P}$ in Theorem 4.11 of [40] (or equivalently Theorem 2.7 of [41]). For a square matrix $\boldsymbol{X}$, it is easy to verify that $\boldsymbol{U}_s$ is an invariant subspace of $\boldsymbol{X}$. Define

$$\beta = \operatorname{sep}(\Sigma_s\boldsymbol{V}_s^H\boldsymbol{U}_s, 0) - \|\boldsymbol{U}_s^H\Delta\boldsymbol{X}\boldsymbol{U}_s\| - \|\boldsymbol{U}_n^H\Delta\boldsymbol{X}\boldsymbol{U}_n\|, \tag{3.39}$$

where $\operatorname{sep}(\boldsymbol{A}, \boldsymbol{B}) = \inf_{\|\boldsymbol{P}\|=1}\|\boldsymbol{P}\boldsymbol{A} - \boldsymbol{B}\boldsymbol{P}\|$. Theorem 4.11 of [40] states that if $\beta > 0$ and

$$\frac{\|\boldsymbol{U}_n^H\Delta\boldsymbol{X}\boldsymbol{U}_s\|\left(\|\Sigma_s\boldsymbol{V}_s^H\boldsymbol{U}_n\| + \|\boldsymbol{U}_s^H\Delta\boldsymbol{X}\boldsymbol{U}_n\|\right)}{\beta^2} < \frac{1}{4}, \tag{3.40}$$

then there exists a unique $(M-F)\times F$ matrix $\boldsymbol{P}$ satisfying

$$\|\boldsymbol{P}\| < 2\frac{\|\boldsymbol{U}_n^H \Delta \boldsymbol{X} \boldsymbol{U}_s\|}{\beta}, \tag{3.41}$$

such that $\boldsymbol{U}_s + \boldsymbol{U}_n\boldsymbol{P}$ is an invariant subspace of $\widehat{\boldsymbol{X}}$. Since $\widehat{\boldsymbol{U}}_s$ is also an invariant subspace of $\widehat{\boldsymbol{X}}$, there exists a unique $F \times F$ non-singular transformation matrix $\boldsymbol{Q}$ such that $\widehat{\boldsymbol{U}}_s = (\boldsymbol{U}_s + \boldsymbol{U}_n\boldsymbol{P})\,\boldsymbol{Q}$. Because $\widehat{\boldsymbol{U}}_s^H \widehat{\boldsymbol{U}}_s = \boldsymbol{I}$, it can be further verified that

$$\boldsymbol{Q}^H\left(\boldsymbol{I} + \boldsymbol{P}^H\boldsymbol{P}\right)\boldsymbol{Q} = \boldsymbol{I}. \tag{3.42}$$

In [43], $\boldsymbol{Q}$ is constrained to be a Hermitian matrix $(\boldsymbol{I} + \boldsymbol{P}^H\boldsymbol{P})^{-\frac{1}{2}}$, from which it is concluded that $\boldsymbol{Q} - \boldsymbol{I}$ has no first-order terms of $\Delta\boldsymbol{X}$, and thus the contribution of $\boldsymbol{U}_s$ to the first-order perturbation of $\boldsymbol{U}_s$ is ignored. However, any matrix $\boldsymbol{Q} = (\boldsymbol{I} + \boldsymbol{P}^H\boldsymbol{P})^{-\frac{1}{2}}\boldsymbol{\Phi}$, where $\boldsymbol{\Phi}^H\boldsymbol{\Phi} = \boldsymbol{I}$ (i.e., $\boldsymbol{\Phi}$ is a unitary matrix), also satisfies (3.42). Since $\boldsymbol{Q}$ is a unique transformation matrix associating $\widehat{\boldsymbol{U}}_s$ and $\boldsymbol{U}_s + \boldsymbol{U}_n\boldsymbol{P}$, it is not necessary that $\boldsymbol{Q} = (\boldsymbol{I} + \boldsymbol{P}^H\boldsymbol{P})^{-\frac{1}{2}}$. In fact, as will be demonstrated in the follow example, in some cases $\boldsymbol{Q}$ may not be a Hermitian matrix, and $\boldsymbol{Q} - \boldsymbol{I}$ may contain first-order terms of $\Delta\boldsymbol{X}$. In this case, the contribution of $\boldsymbol{U}_s$ to the first-order approximation of $\Delta\boldsymbol{U}_s$ cannot be ignored.

Note that the classical approaches such as the one used in [28] that start from $\widehat{\boldsymbol{U}}_s = \boldsymbol{U}_s + \boldsymbol{U}_n\boldsymbol{P}$, where $\boldsymbol{U}_s = \begin{bmatrix}\boldsymbol{u}_1 & \boldsymbol{u}_2 & \cdots & \boldsymbol{u}_F\end{bmatrix}$, does not yield (3.30). In fact, the result is (3.38). Furthermore, this approach requires the assumption that $\Delta\boldsymbol{U}_s$ is orthogonal to $\boldsymbol{U}_s$, which means $\Delta\boldsymbol{u}_g^H\boldsymbol{u}_f = 0$ for $1 \le g, f \le F$. While in (3.33), it is only assumed that $\Delta\boldsymbol{u}_f^H\boldsymbol{u}_f = 0$ for $f = 1, \ldots, F$. Because the latter assumption is weaker than that of [28], the corresponding result in (3.30) is more general.

A simple deterministic example is demonstrated here. Consider the perturbed data matrix $\widehat{\boldsymbol{X}} = \boldsymbol{X} + \Delta\boldsymbol{X}$, where

$$\boldsymbol{X} = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \Delta\boldsymbol{X} = \begin{bmatrix} 0 & \epsilon & 0 \\ \epsilon & 0 & \epsilon \\ 0 & \epsilon & 0 \end{bmatrix}, \tag{3.43}$$

and ϵ is a small real number. It is clear that the singular values of $\boldsymbol{X}$ are $\sigma_1 = 2$, $\sigma_2 = 1$, $\sigma_3 = 0$, and the signal subspace and noise subspace are spanned by

$$\boldsymbol{U}_s = [\boldsymbol{e}_1 \;\; \boldsymbol{e}_2], \quad \boldsymbol{U}_n = [\boldsymbol{e}_3], \tag{3.44}$$

where $\boldsymbol{e}_i$ denotes the i-th column of $\boldsymbol{I}_3$. The singular values of $\widehat{\boldsymbol{X}}$ are $1+p$, 1, and $p-1$, where $p = \sqrt{1+2\epsilon^2}$. The singular vectors corresponding to the singular values $1+p$ and 1 are

$$\widehat{\boldsymbol{u}}_1 = \frac{1}{p}\begin{bmatrix} \frac{1+p}{2} & \epsilon & \frac{-1+p}{2} \end{bmatrix}^T, \quad \widehat{\boldsymbol{u}}_2 = \frac{1}{p}\begin{bmatrix} -\epsilon & 1 & \epsilon \end{bmatrix}^T. \tag{3.45}$$

Therefore, the true perturbations of $\boldsymbol{u}_1$ and $\boldsymbol{u}_2$ are

$$\Delta\boldsymbol{u}_1 = \widehat{\boldsymbol{u}}_1 - \boldsymbol{e}_1 = \frac{1}{p}\begin{bmatrix} \frac{1-p}{2} & \epsilon & \frac{-1+p}{2} \end{bmatrix}^T, \quad \Delta\boldsymbol{u}_2 = \widehat{\boldsymbol{u}}_2 - \boldsymbol{e}_2 = \frac{1}{p}\begin{bmatrix} -\epsilon & 1-p & \epsilon \end{bmatrix}^T. \tag{3.46}$$

According to the Taylor expansion, $\frac{1}{p} = (1+2\epsilon^2)^{-\frac{1}{2}} = 1 - \frac{1}{2}2\epsilon^2 + \frac{3}{8}4\epsilon^4 + \cdots = 1 + \mathcal{O}\{\epsilon^2\}$, where $\mathcal{O}\{\epsilon^2\}$ stands for the higher-order terms of ϵ. Therefore, (3.46) can be rewritten as

$$\Delta\boldsymbol{U}_s = \begin{bmatrix} \Delta\boldsymbol{u}_1 & \Delta\boldsymbol{u}_2 \end{bmatrix} = \begin{bmatrix} \mathcal{O}\{\epsilon^2\} & -\epsilon + \mathcal{O}\{\epsilon^3\} \\ \epsilon + \mathcal{O}\{\epsilon^3\} & \mathcal{O}\{\epsilon^2\} \\ \mathcal{O}\{\epsilon^2\} & \epsilon + \mathcal{O}\{\epsilon^3\} \end{bmatrix}. \tag{3.47}$$

Next one can check if the first-order perturbation results, (3.30) or (3.38), can be used as a first-order approximation of $\Delta\boldsymbol{U}_s$ in (3.47). The first-order approximations of $\Delta\boldsymbol{U}_s$ are

$$\Delta\boldsymbol{U}_s = \begin{bmatrix} 0 & -\epsilon \\ \epsilon & 0 \\ 0 & \epsilon \end{bmatrix} \quad \text{by (3.30), and} \quad \Delta\boldsymbol{U}_s = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & \epsilon \end{bmatrix} \quad \text{by (3.38).} \tag{3.48}$$

It is clear that (3.30) gives a correct first-order approximation while (3.38) can not.

Finally, one can solve the matrices $\boldsymbol{P}$ and $\boldsymbol{Q}$ in Remark 1. Let

$$\widehat{\boldsymbol{U}}_s = \frac{1}{p}\begin{bmatrix} \frac{1+p}{2} & -\epsilon \\ \epsilon & 1 \\ \frac{-1+p}{2} & \epsilon \end{bmatrix} = (\boldsymbol{U}_s + \boldsymbol{U}_n\boldsymbol{P})\boldsymbol{Q}. \tag{3.49}$$

Substituting (3.44) into (3.49), one has

$$\boldsymbol{P} = \left[\frac{1-p}{1+p}, \quad \frac{2\epsilon}{1+p}\right], \quad \boldsymbol{Q} = \frac{1}{p}\begin{bmatrix} \frac{1+p}{2} & -\epsilon \\ \epsilon & 1 \end{bmatrix}. \tag{3.50}$$

It can be verified that $\boldsymbol{P}$ and $\boldsymbol{Q}$ satisfy (3.42). However, it can also be verified that $\boldsymbol{Q} \neq (\boldsymbol{I}+\boldsymbol{P}^H\boldsymbol{P})^{-\frac{1}{2}}$, and $\boldsymbol{Q}-\boldsymbol{I}$ is not a Hermitian matrix and contains some first-order terms of ϵ. This shows that (3.38) can not be used as the first-order approximation of $\Delta\boldsymbol{U}_s$ in this example.

3.3.2 The Perturbation of $\boldsymbol{U}_s$

Suppose that the perturbed data matrix is $\widehat{\widetilde{\boldsymbol{X}}} = \widetilde{\boldsymbol{X}} + \Delta\widetilde{\boldsymbol{X}}$, where $\Delta\widetilde{\boldsymbol{X}}$ is the perturbation term, which is related to the noise term in (1.11), $\boldsymbol{W}$, such that (c.f., (3.11))

$$\Delta\widetilde{\boldsymbol{X}} = \Big[\mathcal{S}[\boldsymbol{w}(1)] \quad \cdots \quad \mathcal{S}[\boldsymbol{w}(T)] \quad \mathcal{S}[\Pi_M \boldsymbol{w}^*(T)] \quad \cdots \quad \mathcal{S}[\Pi_M \boldsymbol{w}^*(1)]\Big], \tag{3.51}$$

Let the SVD of $\widetilde{\boldsymbol{X}}$ in the noiseless case be

$$\widetilde{\boldsymbol{X}} = \boldsymbol{U}_s\Sigma_s\boldsymbol{V}_s^H + \boldsymbol{U}_n\Sigma_n\boldsymbol{V}_n^H, \tag{3.52}$$

where the vectors in $\boldsymbol{U}_s$, associated with the F non-zero singular values, span the signal subspace, while the vectors in $\boldsymbol{U}_n$, associated with the zero singular values, span the orthogonal subspace of $\boldsymbol{U}_s$. It is clear that $\Sigma_n = \boldsymbol{0}$, and the diagonal elements of $\boldsymbol{\Sigma}_s$ in decreasing order are $\lambda_1 \geq \lambda_2 \geq \ldots \lambda_F$. The perturbed data matrix is $\widehat{\widetilde{\boldsymbol{X}}} = \widetilde{\boldsymbol{X}} + \Delta\widetilde{\boldsymbol{X}}$, whose SVD is given by

$$\widehat{\widetilde{\boldsymbol{X}}} = \widehat{\boldsymbol{U}}_s\widehat{\Sigma}_s\widehat{\boldsymbol{V}}_s^H + \widehat{\boldsymbol{U}}_n\widehat{\Sigma}_n\widehat{\boldsymbol{V}}_n^H. \tag{3.53}$$

The perturbed signal subspace is $\widehat{\boldsymbol{U}}_s = \boldsymbol{U}_s + \Delta\boldsymbol{U}_s$. Subsection 3.3.1 gives a first-order approximation of $\Delta\boldsymbol{U}_s$. The result is general and available for other applications whenever SVD appears. Suppose that the frequencies estimated from $\widehat{\widetilde{\boldsymbol{X}}}$ are $\widehat{\omega}_{f,n}$, which is written as $\widehat{\omega}_{f,n} = \omega_{f,n} + \Delta\omega_{f,n}$, for $f = 1, \ldots, F$ and $n = 1, \ldots, N$. In the following, the theoretical MSEs of $\widehat{\omega}_{f,n}$ are derived based on the Theorem 4.

Next one can express $\Delta \boldsymbol{u}_f$ as a function of $\boldsymbol{W}$ by substituting (3.51) to (3.29). First, the vector $\boldsymbol{v}_f$ is separated into $2T$ equal-sized sub-vectors such that

$$\boldsymbol{v}_f^T := \left[\boldsymbol{v}_{f,1}^T \ \boldsymbol{v}_{f,2}^T \ \cdots \ \boldsymbol{v}_{f,T}^T \ \widetilde{\boldsymbol{v}}_{f,T}^T \ \cdots \ \widetilde{\boldsymbol{v}}_{f,1}^T\right].$$

Let $\boldsymbol{v}_{f,t}^T := [v_{f,t}(1) \ \cdots \ v_{f,t}(L)]$ and

$$\widetilde{\boldsymbol{v}}_{f,t}^T := [\widetilde{v}_{f,t}(1) \ \cdots \ \widetilde{v}_{f,t}(L)],$$

for $t = 1, \cdots, T$. Define

$$\boldsymbol{Q}_{f,t} : = \boldsymbol{J}_{1,\cdots,1} v_{f,t}(1) + \boldsymbol{J}_{1,\cdots,2} v_{f,t}(2) + \cdots + \boldsymbol{J}_{L_1,L_2,\cdots,L_N} v_{f,t}(L), \tag{3.54}$$

$$\widetilde{\boldsymbol{Q}}_{f,t} : = \boldsymbol{J}_{1,\cdots,1} \widetilde{v}_{f,t}(1) + \boldsymbol{J}_{1,\cdots,2} \widetilde{v}_{f,t}(2) + \cdots + \boldsymbol{J}_{L_1,L_2,\cdots,L_N} \widetilde{v}_{f,t}(L), \tag{3.55}$$

It can be verified that

$$\mathcal{S}\left[\boldsymbol{w}(t)\right] \boldsymbol{v}_{f,t} = \boldsymbol{Q}_{f,t} \boldsymbol{w}(t),$$
$$\mathcal{S}\left[\boldsymbol{\Pi}_M \boldsymbol{w}^*(t)\right] \widetilde{\boldsymbol{v}}_{f,t} = \widetilde{\boldsymbol{Q}}_{f,t} \boldsymbol{\Pi}_M \boldsymbol{w}^*(t),$$

and thus the first two terms on the RHS of (3.29) can be written as

$$\boldsymbol{U}_n \boldsymbol{U}_n^H \Delta \widetilde{\boldsymbol{X}} \boldsymbol{v}_f \lambda_f^{-1} + \boldsymbol{U}_s \boldsymbol{S}_f \boldsymbol{U}_s^H \Delta \widetilde{\boldsymbol{X}} \boldsymbol{v}_f \lambda_f = \sum_{t=1}^{T} \boldsymbol{E}_f \boldsymbol{Q}_{f,t} \boldsymbol{w}(t) + \sum_{t=1}^{T} \boldsymbol{E}_f \widetilde{\boldsymbol{Q}}_{f,t} \boldsymbol{\Pi}_M \boldsymbol{w}^*(t), \tag{3.56}$$

where $\boldsymbol{E}_f := \boldsymbol{U}_n \boldsymbol{U}_n^H \lambda_f^{-1} + \boldsymbol{U}_s \boldsymbol{S}_f \boldsymbol{U}_s^H \lambda_f$. For the third term on the RHS of (3.29), one can divide the matrix $\boldsymbol{U}_s \boldsymbol{S}_f \boldsymbol{\Sigma}_s \boldsymbol{V}_s^H$ into $2T$ blocks such that

$$\boldsymbol{U}_s \boldsymbol{S}_f \boldsymbol{\Sigma}_s \boldsymbol{V}_s^H = \left[\boldsymbol{K}_{f,1} \ \boldsymbol{K}_{f,2} \ \cdots \ \boldsymbol{K}_{f,T} \ \widetilde{\boldsymbol{K}}_{f,T} \ \cdots \ \widetilde{\boldsymbol{K}}_{f,1}\right].$$

Then, it can be verified that

$$\boldsymbol{U}_s \boldsymbol{S}_f \boldsymbol{\Sigma}_s \boldsymbol{V}_s^H \Delta \widetilde{\boldsymbol{X}}^H \boldsymbol{u}_f = \sum_{t=1}^{T} \boldsymbol{K}_{f,t} \mathcal{U}_f \boldsymbol{w}^*(t) + \sum_{t=1}^{T} \widetilde{\boldsymbol{K}}_{f,t} \mathcal{U}_f \boldsymbol{\Pi}_M \boldsymbol{w}(t), \tag{3.57}$$

where

$$\mathcal{U}_f := \begin{bmatrix} \boldsymbol{u}_f^T \boldsymbol{J}_{1,1,\cdots,1} \\ \boldsymbol{u}_f^T \boldsymbol{J}_{1,1,\cdots,2} \\ \vdots \\ \boldsymbol{u}_f^T \boldsymbol{J}_{1,1,\cdots,L_N} \\ \boldsymbol{u}_f^T \boldsymbol{J}_{1,1,\cdots,2,1} \\ \vdots \\ \boldsymbol{J}_{L_1,L_2,\cdots,L_N} \end{bmatrix}.$$

Summing (3.56) and (3.57), one can write (3.29) as

$$\Delta \boldsymbol{u}_f = \sum_{t=1}^{T} \boldsymbol{\Phi}_{f,t} \boldsymbol{w}(t) + \sum_{t=1}^{T} \widetilde{\boldsymbol{\Phi}}_{f,t} \boldsymbol{w}^*(t), \tag{3.58}$$

where

$$\boldsymbol{\Phi}_{f,t} = \boldsymbol{E}_f \boldsymbol{Q}_{f,t} + \widetilde{\boldsymbol{K}}_{f,t} \mathcal{U}_f \Pi_M, \tag{3.59}$$

$$\widetilde{\boldsymbol{\Phi}}_{f,t} = \boldsymbol{E}_f \widetilde{\boldsymbol{Q}}_{f,t} \Pi_M + \boldsymbol{K}_{f,t} \mathcal{U}_f. \tag{3.60}$$

Notice that one can write $\boldsymbol{K}_{f,t}$ and $\widetilde{\boldsymbol{K}}_{f,t}$ as

$$\boldsymbol{K}_{f,t} = \boldsymbol{U}_s \boldsymbol{S}_f \Sigma_s \boldsymbol{V}_s^H \left[0_{(t-1)L\times L}^T \;\; \boldsymbol{I}_{L\times L} \;\; 0_{(2T-t)L\times L}^T \right]^T \tag{3.61}$$

$$\widetilde{\boldsymbol{K}}_{f,t} = \boldsymbol{U}_s \boldsymbol{S}_f \Sigma_s \boldsymbol{V}_s^H \left[0_{(2T-t)L\times L}^T \;\; \boldsymbol{I}_{L\times L} \;\; 0_{(t-1)L\times L}^T \right]^T \tag{3.62}$$

3.3.3 The Perturbation of $\boldsymbol{T}$

Given $\Delta \boldsymbol{U}_s$, the perturbations of $\boldsymbol{U}_1$ and $\boldsymbol{U}_2$ in (3.3) are

$$\Delta \boldsymbol{U}_1 = \boldsymbol{J}_1 \Delta \boldsymbol{U}_s, \quad \Delta \boldsymbol{U}_2 = \boldsymbol{J}_2 \Delta \boldsymbol{U}_s. \tag{3.63}$$

According to (3.18), one can obtain $\boldsymbol{U}_1 \boldsymbol{T} \boldsymbol{D}(\boldsymbol{\zeta}) = \boldsymbol{U}_2 \boldsymbol{T}$. Let $\boldsymbol{T} = [\boldsymbol{t}_1 \; \boldsymbol{t}_2 \; \cdots \; \boldsymbol{t}_F]$, then

$$\boldsymbol{U}_1 \boldsymbol{t}_f \zeta_f = \boldsymbol{U}_2 \boldsymbol{t}_f. \tag{3.64}$$

Differentiating (3.64), one can obtain

$$(\Delta \boldsymbol{U}_1 \zeta_f - \Delta \boldsymbol{U}_2)\boldsymbol{t}_f + \boldsymbol{U}_1 \boldsymbol{t}_f \Delta \zeta_f = -(\boldsymbol{U}_1 \zeta_f - \boldsymbol{U}_2)\Delta \boldsymbol{t}_f. \tag{3.65}$$

Since $\boldsymbol{t}_f$ is the f-th normalized eigenvector of $\boldsymbol{U}_1^\dagger \boldsymbol{U}_2$, according to [44, Chapeter 2], one can write $\Delta \boldsymbol{t}_f$ as a linear combination of the eigenvectors $\boldsymbol{t}_g$, $1 \le g \le F$, $g \ne f$, such that $\Delta \boldsymbol{t}_f = \sum_{\substack{g=1 \\ g\ne f}}^F k_{gf} \boldsymbol{t}_g$. One can substitute it into (3.65) and obtain

$$(\Delta \boldsymbol{U}_1 \zeta_f - \Delta \boldsymbol{U}_2)\boldsymbol{t}_f + \boldsymbol{U}_1 \boldsymbol{t}_f \Delta \zeta_f = -(\boldsymbol{U}_1 \zeta_f - \boldsymbol{U}_2) \sum_{g=1, g\ne f}^{F} k_{gf} \boldsymbol{t}_g. \tag{3.66}$$

From (3.18), one can get $\boldsymbol{P}_{\text{sp}}^\dagger \boldsymbol{U}_2 = \boldsymbol{D}(\boldsymbol{\zeta}) \boldsymbol{P}_{\text{sp}}^\dagger \boldsymbol{U}_1$. Suppose that $\boldsymbol{P}_{\text{sp}}^\dagger = [\boldsymbol{s}_1 \; \boldsymbol{s}_2 \; \cdots \; \boldsymbol{s}_F]^T$, then $\boldsymbol{s}_f^T(\boldsymbol{U}_1 \zeta_f - \boldsymbol{U}_2) = \boldsymbol{0}^T$. Also from (3.18), it can be verified that $\boldsymbol{s}_g^T \boldsymbol{U}_1 \boldsymbol{t}_{g'} = \delta_{g,g'}$ and $\boldsymbol{s}_g^T \boldsymbol{U}_2 \boldsymbol{t}_{g'} = \zeta_g \delta_{g,g'}$, where $\delta_{g,g'}$ is the Kronecker delta ($\delta_{g,g'} = 1$ when $g = g'$; $\delta_{g,g'} = 0$ when $g \ne g'$). If one multiply both sides of (3.66) by $\boldsymbol{s}_f^T$, he can obtain

$$\Delta \zeta_f = -\boldsymbol{s}_f^T (\Delta \boldsymbol{U}_1 \zeta_f - \Delta \boldsymbol{U}_2) \boldsymbol{t}_f. \tag{3.67}$$

Note that this analysis is similar to the perturbation analysis of eigenvalues in ESPRIT-type algorithms [45, 46, 28, 47]. The expression of $\Delta \zeta_f$ as in (3.67) was also obtained in [47, 28, 45, 46]. However, here the main interest lies in the perturbation of eigenvectors. One can multiply both sides of (3.66) by $\boldsymbol{s}_g^T$ ($g \ne f$) and obtain

$$k_{gf} = \frac{-\boldsymbol{s}_g^T (\Delta \boldsymbol{U}_1 \zeta_f - \Delta \boldsymbol{U}_2) \boldsymbol{t}_f}{\zeta_f - \zeta_g}, \quad 1 \le g \le F, \;\; g \ne f.$$

Therefore one can obtain $\Delta \boldsymbol{t}_f = \boldsymbol{R}_f \Delta \boldsymbol{U}_s \boldsymbol{t}_f$, where

$$\boldsymbol{R}_f := \sum_{\substack{g=1 \\ g\ne f}}^{F} \frac{-\boldsymbol{t}_g \boldsymbol{s}_g^T (\boldsymbol{J}_1 \zeta_f - \boldsymbol{J}_2)}{\zeta_f - \zeta_g} = -\boldsymbol{T} \boldsymbol{D}_f \boldsymbol{P}_{\text{sp}}^\dagger (\boldsymbol{J}_1 \zeta_f - \boldsymbol{J}_2). \tag{3.68}$$

Here $\boldsymbol{D}_f$ is a diagonal matrix with $[\boldsymbol{D}_f]_{g,g} = \frac{1}{\zeta_f - \zeta_g}$, for $g \ne f$ and $[\boldsymbol{D}_f]_{f,f} = 0$. Since the f-th column of $\boldsymbol{P}$ is obtained using $\boldsymbol{p}_f = \boldsymbol{U}_1 \boldsymbol{t}_f$, one has

$$\Delta \boldsymbol{p}_f = \Delta \boldsymbol{U}_1 \boldsymbol{t}_f + \boldsymbol{U}_1 \Delta \boldsymbol{t}_f = (\boldsymbol{J}_1 + \boldsymbol{U}_1 \boldsymbol{R}_f) \Delta \boldsymbol{U}_s \boldsymbol{t}_f. \tag{3.69}$$

3.3.4 The Perturbation of $e^{\omega_{f,n}}$

Now it is ready to obtain the perturbation of $\omega_{f,n}$. Differentiating (3.23), one has

$$\Delta\omega_{f,n} = \mathcal{I}\left(\frac{\Delta e^{j\omega_{f,n}}}{e^{j\omega_{f,n}}}\right), \tag{3.70}$$

where $e^{j\omega_{f,n}}$ is obtained using (3.22). Differentiating (3.22) would result in $\Delta e^{j\omega_{f,n}}$.

Eqn. (3.22) can be expanded into

$$\widehat{e^{j\omega_{f,n}}} = \frac{1}{\mu_n}\sum_{q=0}^{Q-1}\sum_{r=1}^{K'_{n-1}-K'_n}\frac{\widehat{p}_{qK'_{n-1}+K'_n+r,f}}{\widehat{p}_{qK'_{n-1}+r,f}}, \tag{3.71}$$

where $Q := \Pi_{m=1}^{n-1}(K_m - 1)$. Then, differentiating (3.71), one can obtain $\Delta e^{j\omega_{f,n}} =$

$$\frac{1}{\mu_n}\sum_{q=0}^{Q-1}\sum_{r=1}^{K'_{n-1}-K'_n}\frac{\Delta p_{qK'_{n-1}+K'_n+r,f}\,p_{qK'_{n-1}+r,f} - p_{qK'_{n-1}+K'_n+r,f}\,\Delta p_{qK'_{n-1}+r,f}}{p^2_{qK'_{n-1}+r,f}}.$$

Therefore, using $\frac{p_{qK'_{n-1}+K'_n+r,f}}{p_{qK'_{n-1}+r,f}} = e^{j\omega_{f,n}}$, one has

$$\begin{aligned}\frac{\Delta e^{j\omega_{f,n}}}{e^{j\omega_{f,n}}} &= \frac{1}{\mu_n}\sum_{q=0}^{Q-1}\sum_{r=1}^{K'_{n-1}-K'_n}\left(\frac{\Delta p_{qK'_{n-1}+K'_n+r,f}}{p_{qK'_{n-1}+K'_n+r,f}} - \frac{\Delta p_{qK'_{n-1}+r,f}}{p_{qK'_{n-1}+r,f}}\right)\\ &= \frac{1}{\mu_n}\sum_{q=0}^{Q-1}\left\{-\frac{\Delta p_{qK'_{n-1}+1,f}}{p_{qK'_{n-1}+1,f}} - \frac{\Delta p_{qK'_{n-1}+2,f}}{p_{qK'_{n-1}+2,f}}\cdots - \frac{\Delta p_{qK'_{n-1}+K'_n,f}}{p_{qK'_{n-1}+K'_n,f}}\right.\\ &\quad\left. + \frac{\Delta p_{qK'_{n-1}+K'_{n-1}-K'_n+1,f}}{p_{qK'_{n-1}+K'_{n-1}-K'_n+1,f}} + \cdots + \frac{\Delta p_{qK'_{n-1}+K'_{n-1},f}}{p_{qK'_{n-1}+K'_{n-1},f}}\right\}\end{aligned} \tag{3.72}$$

Notice that $\boldsymbol{p}_f^T = p_{1,f}[1\ e^{j\omega_{f,1}}\ \cdots\ e^{j(K_1-2)\omega_{f,1}}] \otimes \cdots \otimes [1\ e^{j\omega_{f,N}}\ \cdots\ e^{j(K_N-2)\omega_{f,N}}]$. Therefore $p(qK'_{n-1}+r, f)$ $(q = 0, \cdots, Q-1$ and $r = 1, \cdots, K'_{n-1})$ is the product of the $(q+1)$-th element of $p_{1,f}[1\ e^{j\omega_{f,1}}\ \cdots\ e^{j(K_1-2)\omega_{f,1}}] \otimes \cdots \otimes [1\ e^{j\omega_{f,n-1}}\ \cdots\ e^{j(K_{n-1}-2)\omega_{f,n-1}}]$ and the r-th element of $[1\ e^{j\omega_{f,n}}\ \cdots, e^{j(K_n-2)\omega_{f,n}}] \otimes \cdots \otimes [1\ e^{j\omega_{f,N}}\ \cdots\ e^{j(K_N-2)\omega_{f,N}}]$. Similarly, one can express (3.72) as

$$\frac{\Delta e^{j\omega_{f,n}}}{e^{j\omega_{f,n}}} = \frac{1}{\mu_n}\boldsymbol{\tau}^T_{f,n}\Delta\boldsymbol{p}_f. \tag{3.73}$$

Here $\boldsymbol{\tau}_{f,n}$ is a vector that contain all the corresponding denominators of terms in (3.72). It can be verified that the $(qK'_{n-1}+r)$-th element of $\boldsymbol{\tau}^T_{f,n}$ is the product of the

$(q+1)$-th element of $p_{1,f}^{-1}[1 \; e^{-j\omega_{f,1}} \; \cdots \; e^{-j(K_1-2)\omega_{f,1}}] \otimes \cdots \otimes [1 \; e^{-j\omega_{f,n-1}} \; \cdots \; e^{-j(K_{n-1}-2)\omega_{f,n-1}}]$ and the r-th element of $[-1 \; \underbrace{0 \; \cdots \; 0}_{K_n-3 \text{ zeros}} \; e^{-j(K_n-2)\omega_{f,n}}] \otimes [1 \; e^{-j\omega_{f,n+1}} \; \cdots \; e^{-j(K_{n+1}-2)\omega_{f,n+1}}] \otimes \cdots \otimes [1 \; e^{-j\omega_{f,N}} \; \cdots \; e^{-j(K_N-2)\omega_{f,N}}]$. Therefore $\boldsymbol{\tau}_{f,n}^T$ can be written as

$$\begin{aligned}
\boldsymbol{\tau}_{f,n}^T &:= p_{1,f}^{-1} \left(\boldsymbol{\phi}_{f,1} \otimes \cdots \otimes \boldsymbol{\phi}_{f,n-1}\right) \otimes \boldsymbol{\varphi}_{f,n} \otimes \left(\boldsymbol{\phi}_{f,n+1} \otimes \cdots \otimes \boldsymbol{\phi}_{f,N}\right) \quad : K_0' \times 1, \quad (3.74) \\
\boldsymbol{\phi}_{f,n} &= \left[1 \; e^{-j\omega_{f,n}} \; \cdots \; e^{-j(K_n-2)\omega_{f,n}}\right]^T : (K_n - 1) \times 1, \\
\boldsymbol{\varphi}_{f,n} &= \left[-1 \; 0 \; \cdots \; 0 \; e^{-j(K_n-2)\omega_{f,n}}\right]^T : (K_n - 1) \times 1. \qquad (3.75)
\end{aligned}$$

Here $p_{1,f}$ is $(1, f)$-th element of $\boldsymbol{P}_{\text{sp}}$, which is also the f-th diagonal element of $\boldsymbol{\Lambda}$. Notice that the nonzero elements of $\boldsymbol{\tau}_{f,n}^T$ are the reciprocals of corresponding $\{p_{k,f}\}_{k=1}^{K_0'}$ with scaling ambiguity, and the zero elements are distributed regularly. Substituting (3.58) to (3.69) and (3.69) to (3.73), one has

$$\frac{\Delta e^{j\omega_{f,n}}}{e^{j\omega_{f,n}}} = \frac{1}{\mu_n} \sum_{t=1}^{T} \left[\boldsymbol{\xi}_{f,n,t}^T \boldsymbol{w}(t) + \widetilde{\boldsymbol{\xi}}_{f,n,t}^T \boldsymbol{w}^*(t)\right], \tag{3.76}$$

where $\boldsymbol{t}_f^T := [t_{f,1} \; t_{f,2} \; \cdots \; t_{f,F}]$, and

$$\boldsymbol{\xi}_{f,n}^T(t) = \boldsymbol{\tau}_{f,n}^T (\boldsymbol{J}_1 + \boldsymbol{U}_1 \boldsymbol{R}_f) \sum_{g=1}^{F} t_{f,g} \boldsymbol{\Phi}_{g,t}, \tag{3.77}$$

$$\widetilde{\boldsymbol{\xi}}_{f,n}^T(t) = \boldsymbol{\tau}_{f,n}^T (\boldsymbol{J}_1 + \boldsymbol{U}_1 \boldsymbol{R}_f) \sum_{g=1}^{F} t_{f,g} \widetilde{\boldsymbol{\Phi}}_{g,t}, \tag{3.78}$$

and $\boldsymbol{\Phi}_{g,t}$ and $\widetilde{\boldsymbol{\Phi}}_{g,t}$ are defined in (3.58).

3.3.5 Simplification of (3.77) and (3.78)

The equation (3.77) and (3.78) can be simplified. In order to simplify those equations, one need the following lemma.

Lemma 10: $\boldsymbol{\tau}_{f,n}^T (\boldsymbol{J}_1 + \boldsymbol{U}_1 \boldsymbol{R}_f) \boldsymbol{U}_s = \boldsymbol{0}$, $f = 1, \ldots, F$, and $n = 1, \ldots, N$.

Proof: Since $\boldsymbol{P}_{\text{sp}}^{\dagger} = \boldsymbol{T}^{-1} \boldsymbol{U}_1^{\dagger}$ and $\boldsymbol{U}_1^{\dagger} \boldsymbol{U}_2 = \boldsymbol{T} \boldsymbol{D}(\boldsymbol{\zeta}) \boldsymbol{T}^{-1}$, one has

$$\boldsymbol{R}_f \boldsymbol{U}_s = -\boldsymbol{T} \boldsymbol{D}_f \boldsymbol{T}^{-1} \boldsymbol{T} (\zeta_f \boldsymbol{I}_F - \boldsymbol{D}(\boldsymbol{\zeta})) \boldsymbol{T}^{-1} = -\boldsymbol{I}_F + \boldsymbol{t}_f \boldsymbol{\rho}_f^T, \tag{3.79}$$

where $\boldsymbol{\rho}_f^T$ is the f-th row of $\boldsymbol{T}^{-1}$. With (3.79), and noticing that $\boldsymbol{U}_1\boldsymbol{t}_f = \boldsymbol{p}_f$ and $\boldsymbol{\tau}_{f,n}^T\boldsymbol{p}_f = 0$, one can obtain

$$\boldsymbol{\tau}_{f,n}^T(\boldsymbol{J}_1 + \boldsymbol{U}_1\boldsymbol{R}_f)\boldsymbol{U}_s = \boldsymbol{\tau}_{f,n}^T\boldsymbol{U}_1 + \boldsymbol{\tau}_{f,n}^T\boldsymbol{U}_1(-\boldsymbol{I}_F + \boldsymbol{t}_f\boldsymbol{y}_f^T) = \boldsymbol{0}^T.$$

Therefore, Lemma 10 is proved. ■

Now Substituting (3.59) and (3.60) to (3.77) and (3.78), and invoking Lemma 10, one can cancel out all the terms that contain $\boldsymbol{U}_s$ and obtain

$$\boldsymbol{\xi}_{f,n}^T(t) = \boldsymbol{\tau}_{f,n}^T(\boldsymbol{J}_1 + \boldsymbol{U}_1\boldsymbol{R}_f)\sum_{g=1}^{F} t_{f,g}\boldsymbol{U}_n\boldsymbol{U}_n^H\lambda_g^{-1}\boldsymbol{Q}_{g,t}, \tag{3.80}$$

$$\widetilde{\boldsymbol{\xi}}_{f,n}^T(t) = \boldsymbol{\tau}_{f,n}^T(\boldsymbol{J}_1 + \boldsymbol{U}_1\boldsymbol{R}_f)\sum_{g=1}^{F} t_{f,g}\boldsymbol{U}_n\boldsymbol{U}_n^H\lambda_g^{-1}\widetilde{\boldsymbol{Q}}_{g,t}\Pi_M, \tag{3.81}$$

3.3.6 The Theoretic Mean and MSE of the Estimates

Since $E\left[\boldsymbol{w}(t)\right] = 0$, it is ready to see

$$E\left[\Delta\omega_{f,n}\right] = 0. \tag{3.82}$$

This means that the estimation is unbiased if only the first-order perturbation is considered. Using the following properties of complex Gaussian noise,

$$\begin{aligned} &E\left[\mathcal{R}(\boldsymbol{w}(t))(\mathcal{R}(\boldsymbol{w}^T(s)))\right] = \frac{\sigma^2}{2}\boldsymbol{I}_{M_1M_2}\delta_{t,s}, \quad E\left[\mathcal{I}(\boldsymbol{w}(t))(\mathcal{I}(\boldsymbol{w}^T(s)))\right] = \frac{\sigma^2}{2}\boldsymbol{I}_{M_1M_2}\delta_{t,s}, \\ &E\left[\mathcal{R}(\boldsymbol{w}(t))(\mathcal{I}(\boldsymbol{w}^T(s)))\right] = E\left[\mathcal{I}(\boldsymbol{w}(t))(\mathcal{R}(\boldsymbol{w}^T(s)))\right] = \boldsymbol{0}, \end{aligned} \tag{3.83}$$

one can obtain the theoretic MSEs for the frequency estimates as

$$E\left[(\Delta\omega_{f,n})^2\right] = \frac{\sigma^2}{2\mu_n^2}\sum_{t=1}^{T}\left[\left\|\mathcal{R}\left(\boldsymbol{\xi}_{f,n}^T(t) - \widetilde{\boldsymbol{\xi}}_{f,n}^T(t)\right)\right\|^2 + \left\|\mathcal{I}\left(\boldsymbol{\xi}_{f,n}^T(t) + \widetilde{\boldsymbol{\xi}}_{f,n}^T(t)\right)\right\|^2\right], \tag{3.84}$$

for $n = 1, \dots, N$, and $f = 1, \dots, F$.

Remark 3: In (3.30), the contribution of the signal subspace $\boldsymbol{U}_s\boldsymbol{R}$ is in the signal subspace, thus one may call it "in-space" perturbation. Since the contribution of the noise subspace $\boldsymbol{U}_n\boldsymbol{U}_n^H\Delta\boldsymbol{X}\boldsymbol{V}_s\boldsymbol{\Sigma}_s^{-1}$ is orthogonal to the signal subspace, one may call it "out-of-space" perturbation. In the above performance analysis, it is found that the "in-space" perturbation does not impact the performance of the proposed eigenvector-based frequency estimation algorithm. This fact is actually true for any subspace-based algorithm that only need the signal subspace or an arbitrary basis of the signal subspace. One can explain this phenomenon using the concept of subspace angles. Let the space spanned by $\boldsymbol{U}_s$ be $\text{span}(\boldsymbol{U}_s)$, and similarly for $\text{span}(\widehat{\boldsymbol{U}}_s)$. As a metric of the distance between $\text{span}(\boldsymbol{U}_s)$ and $\text{span}(\widehat{\boldsymbol{U}}_s)$, let us examine the principal angle $\angle\left(\text{span}(\boldsymbol{U}_s), \text{span}(\widehat{\boldsymbol{U}}_s)\right)$ [48]. Since columns in $\boldsymbol{U}_s$ and $\widehat{\boldsymbol{U}}_s$ are both orthonormal basis, and the principal angle is relatively small, according to [48], one has

$$\sin\left(\angle(\text{span}(\boldsymbol{U}_s), \text{span}(\widehat{\boldsymbol{U}}_s))\right) = \|\widehat{\boldsymbol{U}}_s^H - \widehat{\boldsymbol{U}}_s^H\boldsymbol{U}_s\boldsymbol{U}_s^H\|_2 = \|\boldsymbol{U}_n\boldsymbol{U}_n^H\Delta\boldsymbol{X}\boldsymbol{V}_s\boldsymbol{\Sigma}_s^{-1}\|_2. \tag{3.85}$$

Eqn. (3.85) states that the angle between the subspaces spanned by $\boldsymbol{U}_s$ and $\widehat{\boldsymbol{U}}_s$ is only determined by the "out-of-space" perturbation.

CHAPTER 4

OPTIMIZATION OF WEIGHTING FACTORS

This chapter analyzes the impact of the weighting factors adopted in the previous chapter to the performance of frequency estimation. It is found that the random selected weighting factors can not guarantee the performance and in order to improve the performance in the worst case the weighting factors should be optimized. Based on the theoretical MSE derived in the previous chapter, the optimization criterion to choose weighting factors is derived. Interestingly, the optimization criterion has a very simple interpretation in term of the eigenvalue distribution of $\boldsymbol{U}_1\boldsymbol{U}_2^\dagger$. That is to say, good weighting factors should increase the eigenvalue dispersiveness of $\boldsymbol{U}_1\boldsymbol{U}_2^\dagger$. Simulation results are presented to demonstrate the derived optimization criterion is effective to improve the performance of eigenvector-based frequency estimation than randomly selected weighting factors and other existing frequency estimation algorithms.

4.1 Optimizing Weighting Factors to Minimize Variances

Based on the MSEs of estimation obtained in the previous section, one can optimize the weighting factors $\{\alpha_n\}_{n=1}^N$ to minimize the MSEs and improve the performance of the proposed algorithm. Notice that minimizing (3.84) with respect to $\{\alpha_n\}_{n=1}^N$ is too complex to be solved. However, the MSEs can by regarded as asymptotical error variances when the noise power σ^2 approximates zero, and thus one can take its ratio with the Cramér-Rao Bound (CRB) as an equivalent performance measure, and then use inequalities to bound this ratio so that only entries related to

$\{\alpha_n\}_{n=1}^N$ need to be minimized.

Then the asymptotical efficiency of the estimation of $\omega_{f,n}$ is defined as

$$\eta_{f,n} := \lim_{\sigma^2 \to 0} \frac{E[\Delta\omega_{f,n}^2]}{\mathrm{var}_{\mathrm{CRB}}(\widehat{\omega}_{f,n})}, \tag{4.1}$$

where $\mathrm{var}_{\mathrm{CRB}}(\widehat{\omega}_{f,n})$ is CRB of N-D frequency estimation problem given in (7.2). One can bound $\eta_{f,n}$ using properties of matrix norms such that

$$\eta_{f,n} \le \frac{1}{\mu_n^2 b_{(n-1)F+f}} \left\|\boldsymbol{\tau}_{f,n}\right\|^2 \left(\|\boldsymbol{J}_1\| + \|\boldsymbol{U}_1\| \, \|\boldsymbol{R}_f\|\right)^2$$
$$\sum_{t=1}^{T}\left(\left\|\sum_{g=1}^{F} t_{f,g}\left(\boldsymbol{\Phi}_{g,t} + \widetilde{\boldsymbol{\Phi}}_{g,t}\right)\right\|^2 + \left\|\sum_{g=1}^{F} t_{f,g}\left(\boldsymbol{\Phi}_{g,t} - \widetilde{\boldsymbol{\Phi}}_{g,t}\right)\right\|^2\right).$$

In the RHS of the above inequality, only $\boldsymbol{R}_f$ depends on the weighting factors $\{\alpha_n\}_{n=1}^N$. Furthermore, $\|\boldsymbol{R}_f\| \le \|\boldsymbol{T}\| \, \|\boldsymbol{D}_f\| \, \|\boldsymbol{P}^{\dagger}\| \, \|(\boldsymbol{J}_1\zeta_f - \boldsymbol{J}_2)\|$. Define $\boldsymbol{\alpha} := [\alpha_1 \ \cdots \ \alpha_N]^T$ and $\boldsymbol{\omega}_f := [e^{j\omega_{f,1}} \ \cdots \ e^{j\omega_{f,N}}]^T$. Notice that $\zeta_f = \boldsymbol{\omega}_f^T\boldsymbol{\alpha}$ (see (3.18)). Ignoring all the entries that do not depend on $\boldsymbol{\alpha}$, one can find the upper bound of the efficiency $\eta_{f,n}$ is decided by

$$\|\boldsymbol{D}_f\| \, \|(\boldsymbol{J}_1\zeta_f - \boldsymbol{J}_2)\| = \sum_{\substack{g=1 \\ g \ne f}}^{F} \frac{\left\|\sum_{n=1}^{N} \alpha_n(\boldsymbol{J}_1 e^{j\omega_{f,n}} - \widetilde{\boldsymbol{J}}_{2,n})\right\|}{|(\boldsymbol{\omega}_f^T - \boldsymbol{\omega}_g^T)\boldsymbol{\alpha}|}. \tag{4.2}$$

Without loss of generality, it is assumed that $\|\boldsymbol{\alpha}\| \le 1$ since $\boldsymbol{\alpha}$ appears in both the numerator and the denominator of the RHS of (4.2). If $\|\boldsymbol{\alpha}\| \le 1$, then $|\alpha_n| \le 1$ for $n = 1, \cdots, N$. In this case,

$$\|\boldsymbol{D}_f\| \, \|(\boldsymbol{J}_1\zeta_f - \boldsymbol{J}_2)\| \le N\left(\|\boldsymbol{J}_1\| + \left\|\widetilde{\boldsymbol{J}}_{2,n}\right\|\right)\gamma_f(\boldsymbol{\alpha}), \tag{4.3}$$

where $\gamma_f(\boldsymbol{\alpha}) := \sum_{\substack{g=1 \\ g \ne f}}^{F} \frac{1}{|(\boldsymbol{\omega}_f^T - \boldsymbol{\omega}_g^T)\boldsymbol{\alpha}|}$. In order to minimize the upper bound of the average efficiency

$$\eta = \frac{1}{NF}\sum_{f=1}^{F}\sum_{n=1}^{N}\eta_{f,n}, \tag{4.4}$$

it is desired to minimize the following cost function

$$\gamma(\boldsymbol{\alpha}) = \frac{1}{2}\sum_{f=1}^{F}\gamma_f(\boldsymbol{\alpha}) = \sum_{f=1}^{F-1}\sum_{g=f+1}^{F}\frac{1}{|(\boldsymbol{\omega}_f^T - \boldsymbol{\omega}_g^T)\boldsymbol{\alpha}|}. \tag{4.5}$$

An optimal $\boldsymbol{\alpha}$ can be obtained by

$$\boldsymbol{\alpha}_{\text{opt}} = \arg\min_{\boldsymbol{\alpha}} \gamma(\boldsymbol{\alpha}), \quad \text{subject to} \quad \|\boldsymbol{\alpha}\| \leq 1. \tag{4.6}$$

In fact the optimization criterion (4.6) has a simple interpretation in term of the eigenvalue distribution of $\boldsymbol{U}_1^{\dagger}\boldsymbol{U}_2$. The difference of eigenvalues is $d_{f_1,f_2} := \zeta_{f_1} - \zeta_{f_2} = (\boldsymbol{\omega}_{f_1}^T - \boldsymbol{\omega}_{f_2}^T)\boldsymbol{\alpha}$, for $1 \leq f_1, f_2 \leq F$. Notice that these differences appear as the denominators in the entries of $\gamma(\boldsymbol{\alpha})$ (see (4.5)) and the diagonal elements of $\boldsymbol{D}_f$ (see (3.68)). If $\boldsymbol{\alpha}$ is approximately orthogonal to some $\boldsymbol{\omega}_{f_1}^T - \boldsymbol{\omega}_{f_2}^T$, d_{f_1,f_2} is close to zero, then $\gamma(\boldsymbol{\alpha})$ will be very large, and the performance of the algorithm will degrade dramatically due to high MSE.

Furthermore, an example is used to show the effect of $\boldsymbol{\alpha}$ on η. Suppose that three 3-D frequency components are to be estimated from 3 snapshots of 6×6×6 noisy data samples as given in (1.3). The 3-D frequency components are

$$f = 1: \quad (\omega_{1,1}, \omega_{1,2}, \omega_{1,3}) = (0.70\pi, 0.50\pi, 0.20\pi), \tag{4.7}$$

$$f = 2: \quad (\omega_{2,1}, \omega_{2,2}, \omega_{2,3}) = (0.80\pi, 0.60\pi, 0.20\pi), \tag{4.8}$$

$$f = 3: \quad (\omega_{3,1}, \omega_{3,2}, \omega_{3,3}) = (0.80\pi, 0.50\pi, 0.40\pi). \tag{4.9}$$

Fig. 4.1 depicts the logarithmic plot of η as a function of α_1 and α_2, while α_3 is fixed at $\sqrt{\frac{1}{3}}$. For simplicity of the plot, it is assumed that $|\alpha_1| = \sqrt{\frac{1}{3}}$ and $|\alpha_2| = \sqrt{\frac{1}{3}}$. Hence the x- and y-axes in Fig. 4.1 are the angles of α_1 and α_2 respectively.

As illustrated in Fig. 4.1, in most cases, $\boldsymbol{\alpha}$ keeps the average efficiency η as small as 1, but when $\boldsymbol{\alpha}$ falls into some "bad regions", the average efficiency becomes very large. A random selected $\boldsymbol{\alpha}$ does not guarantee that η does not fall into the "bad regions". Therefore one should use (4.6) to optimize the choice of $\boldsymbol{\alpha}$.

The optimization problem (4.6) is a so called sum-of-ratios fractional programming problem, which is a difficult global optimization problem [49]. There is no efficient algorithm available to solve it to date. From Fig. 4.1 it is found that the "bad regions" are often regular and small, and most choices of $\boldsymbol{\alpha}$ are fairly good to

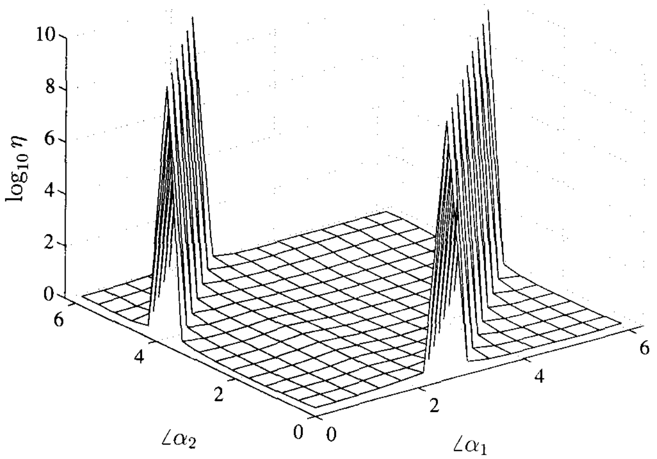


Figure 4.1. The average efficiency as a function of α_1 and α_2 ($|\alpha_1| = \sqrt{\frac{1}{3}}$, $|\alpha_2| = \sqrt{\frac{1}{3}}$, and $\alpha_3 = \sqrt{\frac{1}{3}}$).

obtain a small average efficiency. One can use grid search in the super-sphere $\|\boldsymbol{\alpha}\| \leq 1$ to find a moderate initial value of $\boldsymbol{\alpha}$, then use a Newton type algorithm to find an optimal $\{\alpha_n\}_{n=1}^{N}$. In order to reduce the complexity, one may set $|\alpha_n| = \sqrt{\frac{1}{N}}$, for $n = 1, \cdots, N$, and the search grid does not need to be fine (for example, the step size of angle in one dimension can be set to π/F). Alternatively one can use the following method to obtain a moderate initial value of $\{\alpha_n\}_{n=1}^{N}$. Define

$$\epsilon(\boldsymbol{\alpha}) := \min_{1\leq f<g\leq F} \left|(\boldsymbol{\omega}_f^T - \boldsymbol{\omega}_g^T)\boldsymbol{\alpha}\right|. \tag{4.10}$$

Then it is clear that $\gamma(\boldsymbol{\alpha}) \leq \frac{F(F-1)}{\epsilon(\boldsymbol{\alpha})}$. If one can solve the following optimization problem

$$\boldsymbol{\alpha}_0 = \arg\max_{\|\boldsymbol{\alpha}\|\leq 1} \epsilon(\boldsymbol{\alpha}), \tag{4.11}$$

the upper bound of $\gamma(\boldsymbol{\alpha})$ is minimized. The optimization problem (4.11) can be solved using a sequential quadratic programming (SQP) method, which is a common quasi-Newton type algorithm available in many optimization packages such as the optimization toolbox in Matlab.

TABLE 4.1

An improved eigenvector-based algorithm using optimal weighting factors

---

1. Given (1.11), follow (3.8)–(3.15) to obtain $\boldsymbol{U}_s$.
2. Randomly select $\boldsymbol{\alpha}$ subject to $|\alpha_n| = \sqrt{\frac{1}{N}}$, $n = 1, \ldots, N$, compute $\{\widehat{\omega}_{f,n}\}_{n=1}^{N}$, $f = 1, \ldots, F$, using (3.2)–(3.23).
3. Based on $\{\widehat{\omega}_{f,n}\}_{n=1}^{N}$, for $f = 1, \ldots, F$, obtain an updated $\boldsymbol{\alpha}_{\text{opt}}$ by first solving (4.11) using SQP to get initials, then solving (4.6) using a Newton method.
4. Compute updated $\{\widehat{\omega}_{f,n}\}_{n=1}^{N}$, $1 \le f \le F$, with $\boldsymbol{\alpha}_{\text{opt}}$ using (3.2)–(3.23).
5. Iterate Steps 3 and 4 until frequency estimates converge (typically one execution of Steps 3-4 is sufficient).

Remark 4: The optimization problem (4.11) can be formulated as a fixed tolerance problem [50], therefore it can be solved using SQP. Typically, the SQP procedure includes a sequence of iterations. In each iteration, there are three steps:

1. Update the Hessian matrix of the Lagrangian function, for example, using the BFGS method [51, Chapter 8].
2. Solve a quadratic programming subproblem to obtain a suitable step size. For example, an active-set strategy [51, Chapter 16] can be applied.
3. Test to determine if the iteration can be stopped by evaluating a merit function, which can be chosen as the target function itself.

The algorithm using an optimal $\boldsymbol{\alpha}$ for N-D frequency estimation is described in Table 4.1. Notice that the first step, which involves the SVD of $\widetilde{\boldsymbol{X}}$, is most computationally complex. However this step is executed only once. To appreciate the proposed algorithm in Table 4.1, three methods to obtain an optimal $\boldsymbol{\alpha}$ are compared. The difference is only in Step 3 of Table 4.1, where one may: (a) use the solution of (4.11) as $\boldsymbol{\alpha}_{\text{opt}}$, referred to as "Minmax" here; (b) use the solution of (4.11) as initials, then solve (4.6) to obtain $\boldsymbol{\alpha}_{\text{opt}}$, referred to as "Minmax+Newton", which is the proposed algorithm; and (c) solve (4.6) by grid search first then refine it using a Newton method, referred to as "Grid+Newton". These approaches are applied to estimate three 3-D frequency components from 3 snapshots of $6 \times 6 \times 6$ noisy data samples. The frequency components are given in (4.7)-(4.9). Fig. 4.2 (a) depicts the root mean-square error (RMSE) versus signal-to-noise ratio (SNR). The RMSE is obtained by averaging over all frequencies after only one iteration of Steps 3-4, except for the case indicated with "Random α" where only Steps 1-2 are executed. The SNR is defined as (c.f., (1.12)):

$$\mathrm{SNR} = 10\log_{10}\frac{\|\boldsymbol{AC}\|^2}{M_1 M_2 M_3 T \sigma^2}. \tag{4.12}$$

The corresponding CRB on standard deviation (STD) is also plotted.

It can be seen from Fig. 4.2 (a) that the three optimization methods provide similar performance, and all outperform the case with randomly chosen $\boldsymbol{\alpha}$. It turns out one iteration of Steps 3-4 is sufficient, as demonstrated by Fig. 4.2 (b), where the RMSEs of frequency estimates are plotted versus the number of iterations of Steps 3-4. Zero iteration corresponds to the case with only randomly chosen $\boldsymbol{\alpha}$. It can be observed that one execution of Steps 3 and 4 is sufficient as further iterations only provide negligible performance improvement if any. This is largely due to that "bad regions" only account for a very small percentage of the total area as shown in Fig. 4.1. Also note that here the main aim is to avoid $\boldsymbol{\alpha}$ falling into those "bad regions", rather than finding an absolute optimal solution, therefore there is no convergence issue as there are many good choices of $\boldsymbol{\alpha}$. Furthermore, it can be seen that "Minmax+Newton" and "Grid+Newton" are comparable, and both are slightly better than "Minmax". Because "Grid+Newton" has a higher complexity than "Minmax+Newton", it is better to choose "Minmax+Newton" with one iteration as the proposed algorithm in Table 4.1 for optimizing the weighting factors, which is the algorithm used in the simulations of Section 4.2.

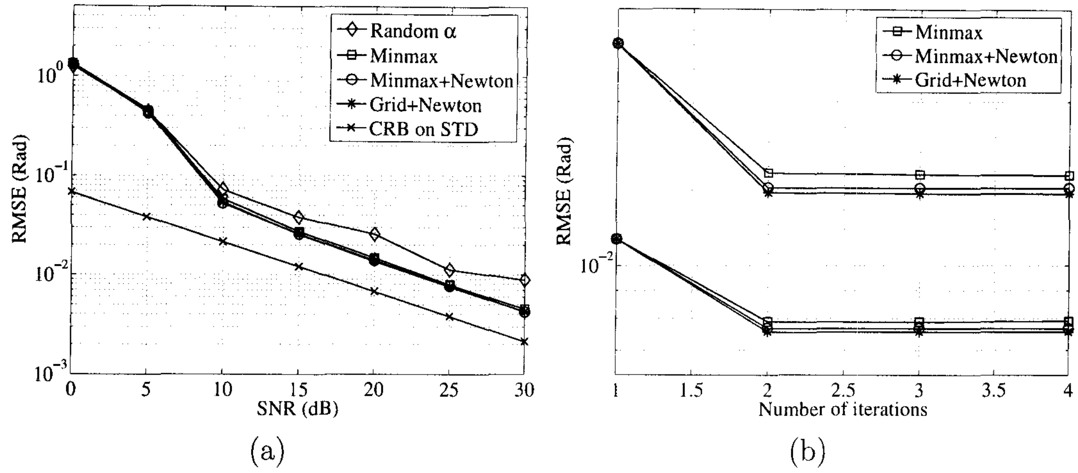


Figure 4.2. (a) RMSE of different optimization methods versus SNR; (b) RMSE of different optimization methods versus the number of iterations of Steps 3-4.

4.2 Simulation Results

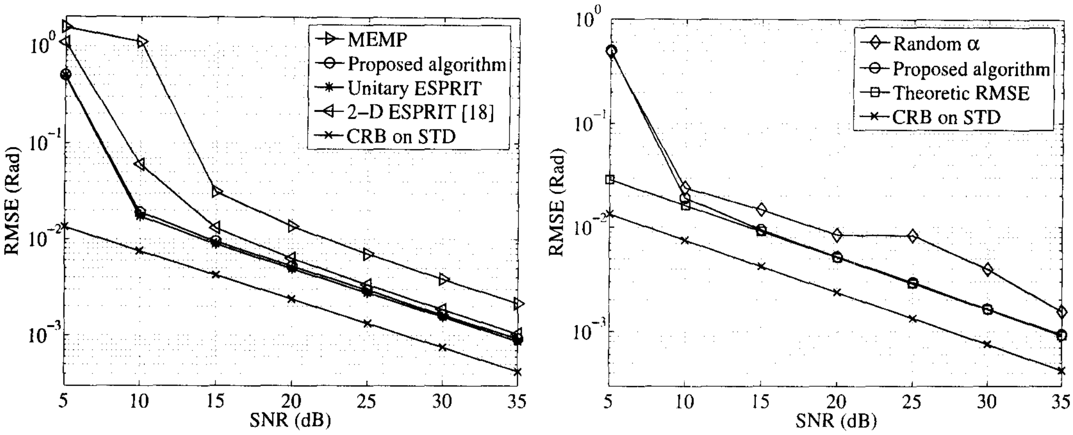
This section presents the Monte Carlo simulation results to demonstrate the performance of the proposed algorithm in Table 4.1, which is also compared to other N-D frequency estimation algorithms as well as the associated CRB.

4.2.1 2-D Identical Frequency Estimation from Single Snapshot

In the first experiment, the proposed algorithm, MEMP [21], Unitary ESPRIT [13], and 2-D ESPRIT [23] are applied to estimate three 2-D frequencies from a 20×20 noisy data set. The amplitudes, $c_f(1)$ for $f = 1, \ldots, F$, are set to be one for this case. The three frequency pairs are

$$(\omega_{1,1}, \omega_{1,2}) = (0.55\pi, 0.20\pi),$$
$$(\omega_{2,1}, \omega_{2,2}) = (0.60\pi, 0.20\pi),$$
$$(\omega_{3,1}, \omega_{3,2}) = (0.60\pi, 0.25\pi).$$

Notice that there are identical frequencies in both dimensions. Fig. 4.3 depicts the performance comparisons. Fig. 4.3 (a) plots the simulated RMSE of various algorithms and the average CRB on STD in the two dimensions. The RMSE results



(a) (b)

Figure 4.3. (a) Comparison of different algorithms for 2-D frequency estimation from single snapshot; (b) Comparison of optimized $\boldsymbol{\alpha}$ and randomly chosen $\boldsymbol{\alpha}$.

are averaged over all frequencies and obtained through 1000 realizations. For the proposed algorithm, the smoothing parameters $(\{K_n\}_{n=1}^2, \{L_n\}_{n=1}^2)$ are chosen such that the identifiability bound in (3.25) can be achieved. The smoothing parameters of other algorithms are chosen according to their corresponding references. As shown in Fig. 4.3 (a), the proposed algorithm offers comparable performance as that of the Unitary ESPRIT algorithm, and outperforms 2-D ESPRIT and MEMP.

Fig. 4.3 (b) compares the optimized weighting factors to randomly chosen ones, where "Random α" means zero iteration of Steps 3-4 in Table 4.1. The theoretical RMSE is obtained from the square root of the average of (3.84), for $n = 1, 2$ and $F = 1, \ldots, 3$, with an optimized $\boldsymbol{\alpha}$ by solving (4.6) using the true frequencies, which serves as a benchmark since in the algorithm $\boldsymbol{\alpha}$ is optimized when the true frequencies are unknown. It is clear that the proposed algorithm significantly outperforms the one with random weighting factors, and the simulated RMSE of the proposed algorithm matches well to the theoretical RMSE for moderate to high SNR range.

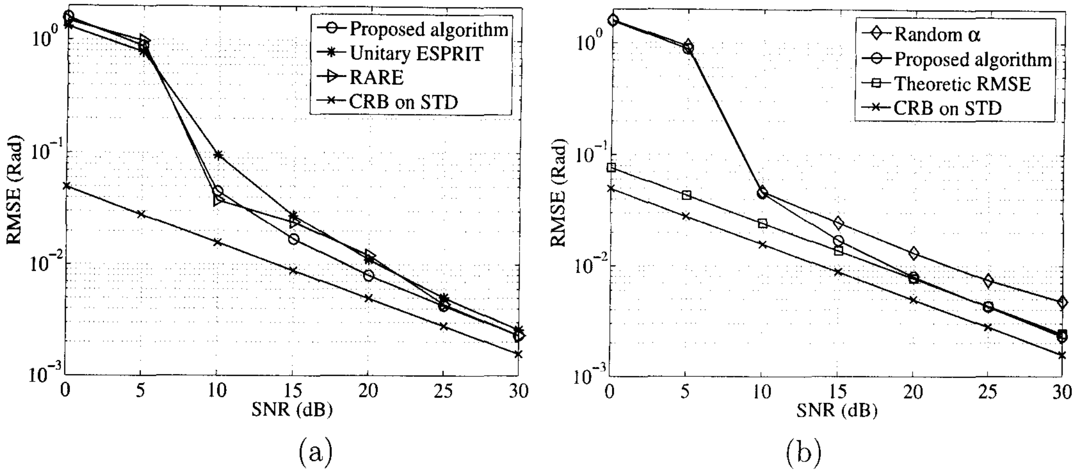


Figure 4.4. (a) Comparison of different algorithms for 2-D frequency estimation from multiple snapshots; (b) Comparison of optimized $\boldsymbol{\alpha}$ and randomly chosen $\boldsymbol{\alpha}$.

4.2.2 2-D Close Frequency Estimation from Multiple Snapshots

In the second experiment, the proposed algorithm, Unitary ESPRIT and RARE are applied to estimate three 2-D frequencies from 10 snapshots of noisy data, each of size 12×12, as given in (1.11). The amplitudes, $c_f(t)$, for $f = 1, \ldots, F$ and $t = 1, \ldots, T$, are drawn from a complex Gaussian distribution. The three frequency pairs are

$$(\omega_{1,1}, \omega_{1,2}) = (0.72\pi, 0.62\pi),$$
$$(\omega_{2,1}, \omega_{2,2}) = (0.74\pi, 0.58\pi),$$
$$(\omega_{3,1}, \omega_{3,2}) = (0.76\pi, 0.60\pi).$$

Notice that frequencies are close to each other in both dimensions. Fig. 4.4 plots the simulated RMSE of various algorithms, along with the corresponding CRB and the theoretical RMSE of the proposed algorithm. Multidimensional data smoothing is also performed for the Unitary ESPRIT algorithm and the RARE algorithm. It can be seen from Fig. 4.4 (a) that the proposed algorithm offers competitive performance when compared with the Unitary ESPRIT and RARE algorithms. Fig. 4.4 (b) confirms again that optimized weighting factors outperform randomly chosen weighting

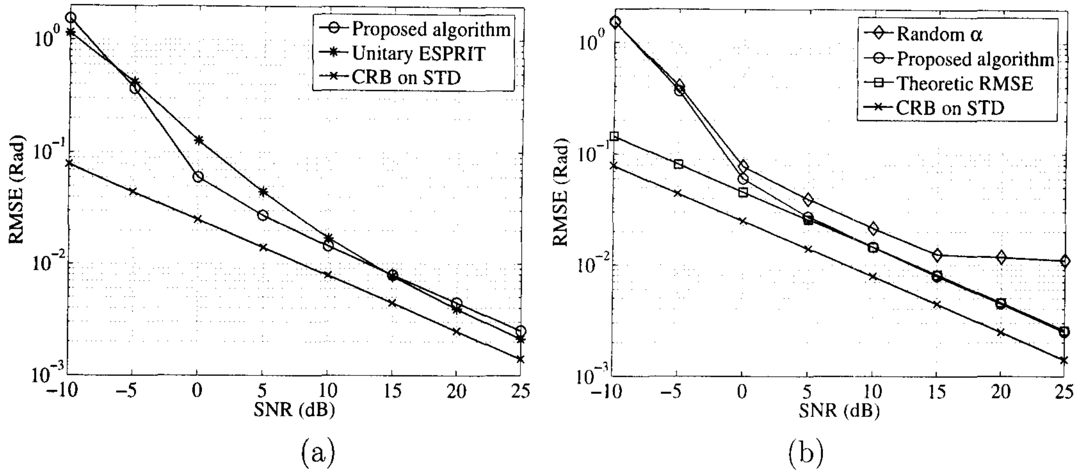


Figure 4.5. (a) Comparison of different algorithms for 3-D frequency estimation from single snapshot; (b) Comparison of optimized $\boldsymbol{\alpha}$ and randomly chosen $\boldsymbol{\alpha}$.

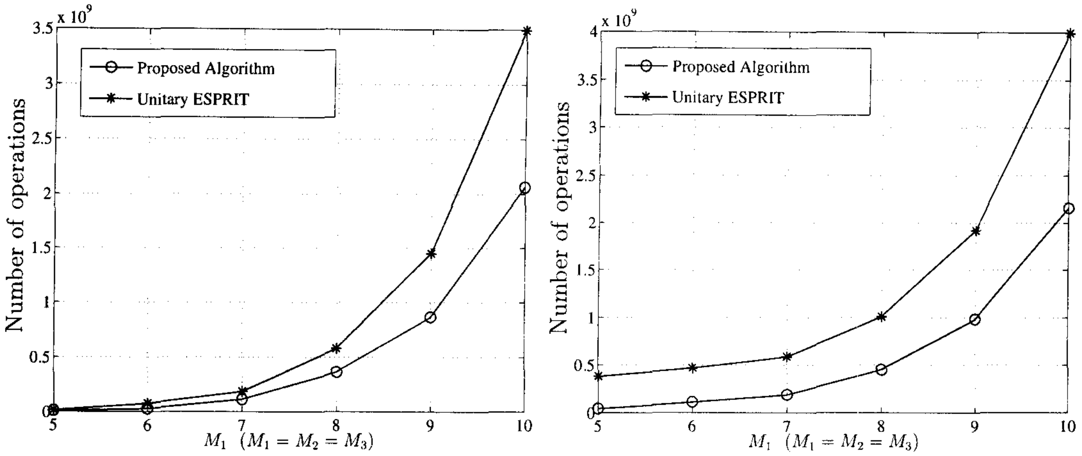
factors, and the simulated RMSE matches the theoretical RMSE at high SNR.

4.2.3 3-D Identical Frequency Estimation from Multiple Snapshots

In the third experiment, the proposed algorithm and the Unitary ESPRIT algorithm are applied to estimate three 3-D frequencies from 10 snapshots of 6×6×6 noisy data samples. The frequencies components are given in (4.7)-(4.9). Notice that there are identical frequencies in all dimensions. The amplitudes are drawn from a complex Gaussian distribution. Fig. 4.5 shows the performance comparisons. From Fig. 4.5, it can be found that the proposed algorithm also offer competitive performance in 3-D frequencies estimation compared to the Unitary ESPRIT algorithm. Notice that the simulated RMSE of the proposed algorithm matches its theoretical RMSE for moderate to high SNR range.

4.2.4 Complexity Comparison

Since it is difficult to analytically calculate the complexity order of the optimization step of the proposed algorithm, here the Performance API (PAPI) tool [52] of Matlab is used to count the number of floating point operations. The complexity



(a) (b)

Figure 4.6. The number of floating point operations versus M_1 when: (a) $F = 3$; (b) $F = 30$.

of proposed algorithm is compared to that of N-D Unitary ESPRIT [13], whose analytical form of complexity order is also unavailable due to the iteration nature of the simultaneous Schur decomposition. The two algorithms are applied to estimate 3-D frequencies from a data set of size $M_1 \times M_2 \times M_3$. Note that both algorithms obtain the signal subspace using SVD with a similar data smoothing step in the single snapshot case, therefore the cost of the frequency estimation and pairing steps determines the difference in complexity. In order to have a fair comparison, the same smoothing parameters are used in both algorithms. It is set that $M_1 = M_2 = M_3$. Figs. 4.6 (a) and 4.6 (b) plot the number of floating point operations of the two algorithms as a function of M_1 for $F = 3$ and $F = 30$, respectively. It can be observed that the proposed algorithm has lower complexity than Unitary ESPRIT and the difference increases as the number of frequency components increases.

CHAPTER 5

APPLICATION IN TIME-VARYING CHANNEL ESTIMATION

Multidimensional frequency estimation has plenty of applications in communications and signal process. This chapter applies the proposed 2-D frequency estimation algorithm to time-varying channel estimation in Orthogonal Frequency Division Multiplexing (OFDM) communication systems.

Recently adaptive OFDM wireless communication systems were proposed to overcome the limitations of conventional OFDM systems [53]. In such systems, the estimation of the time-varying wireless channel is an important problem for system design. Not only does the receiver require channel state information (CSI) for reliable symbol estimation, but the transmitter also uses CSI feedback from receiver to adaptively vary power, modulation and code rate on each subcarrier. Therefore robust and fast channel estimation approaches at the receiver side, which can track time-varying channel instantaneously, and long range channel prediction algorithms for the transmitter, which can predict future CSI based on current channel estimates, are both desired.

Channel identification and prediction for OFDM systems have gained considerable attention. Both non-parametric channel estimation algorithms and parametric approaches were proposed. In [54], a robust non-parametric MMSE channel estimator was developed for OFDM. However, it was shown that the parametric channel estimator outperforms its non-parametric counterpart in [55], where a deterministic parametric model was used to represent the multipath channel. In [55], the delay parameters of each path were initially estimated through 1-D ESPRIT algorithm and then tracked by a delay locked loop. The same parametric channel model was adopted in [56], where a subspace tracking algorithm was proposed to track the delay parameters. Parametric channel models were also adopted to predict channel variation in [57, 58]. In summary, parametric channel models were widely applied to estimate, track and predict time-varying channel in OFDM systems.

It is noticed that only multipath (frequency-selective) channel model was considered in most previous work. For frequency- and time-selective (doubly-selective) fading channels, where the Doppler spreading can not be ignored, the coefficients in the multipath channel model vary rapidly that effective tracking is difficult. In [57, 58], deterministic models for doubly selective channel were considered to predict channel variations. In [58], 1-D ESPRIT algorithm was applied in two stages to estimate the delay and Doppler parameters sequentially.

Although 1-D ESPRIT algorithms were used in channel estimation in OFDM system, its high dimension extension, multidimensional frequency estimation algorithms, were not considered in this problem. Actually, these algorithms were extensively used in wireless channel sounding [15]. This chapter proposes to use proposed 2-D frequency estimation algorithm to estimate the parameters in the deterministic doubly-selective channel model. The algorithm possesses several advantages in identifiability, performance and complexity. The estimated parameters are then used to predict channel variations.

5.1 System Model

5.1.1 Adaptive OFDM System

The adaptive OFDM system model considered in this chapter is presented in Fig. 5.1.1. The input of the system is information bitstream at the transmitter, and the output is restored information bitstream at the receiver. The main difference of the adaptive OFDM system and the conventional OFDM system is that the channel information obtained at the receiver is fed back to the transmitter by the channel predicator in the block diagram. The adaptive coder in the transmitter performs adaptive coding and modulation according to the near future CSI [53].

At the transmitter, the m-th OFDM symbol $\boldsymbol{s}^{(m)}$ contain N frequency signals

$$\boldsymbol{s}^{(m)} := [s_0^{(m)} \ s_1^{(m)} \ \cdots \ s_{N-1}^{(m)}]^T, \ m = 0, \ldots, M-1.$$

QAM modulation is assumed. The OFDM symbols are converted into time-domain samples $\{t_n^{(m)}\}$ according to the normalized N-point inverse DFT operation

$$t_n^{(m)} = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} s_k^{(m)} e^{j\frac{2\pi}{N}kn}, \quad -N_c \leq n < N. \tag{5.1}$$

For each time-domain block with length N, a cyclic prefix (i.e., samples in the head of the block) of length N_c ($N_c \leq N$) is appended at the end of the block. Then these samples are transmitted sequentially over a noisy multipath channel. The multipath channel is modeled by the time-variant discrete impulse response $h^{(m)}[n, \ell]$, defined as the time-n response to an impulse applied at time $n - \ell$. Assuming a causal channel with maximum delay spread $N_h \leq N_c$, the received sample collected during the m-th OFDM symbol are

$$r_n^{(m)} = \sum_{l=0}^{N_h-1} h^{(m)}[n, \ell] t_{n-\ell}^{(m)} + w_n^{(m)}, \tag{5.2}$$

where $w_n^{(m)}$ is complex additive white Gaussian noise (AWGN) with variance σ^2. Due to the introduction of the cyclic prefix, the linear convolution is equivalent to circular convolution. Since the cyclic prefix is corrupted by inter-block interference, it is discarded at receiver side. The receiver then computes an N-point DFT of $\{r_n^{(m)}\}$ and obtain signals in frequency domain

$$x_d^{(m)} = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} r_n^{(m)} e^{-j\frac{2\pi}{N}dn}, \quad 0 \leq d < N. \tag{5.3}$$

One can substitute (5.1) and (5.2) to (5.3) and obtain

$$x_d^{(m)} = \sum_{k=0}^{N-1} H^{(m)}[d, k] s_k^{(m)}, \quad 0 \leq d < N, \tag{5.4}$$

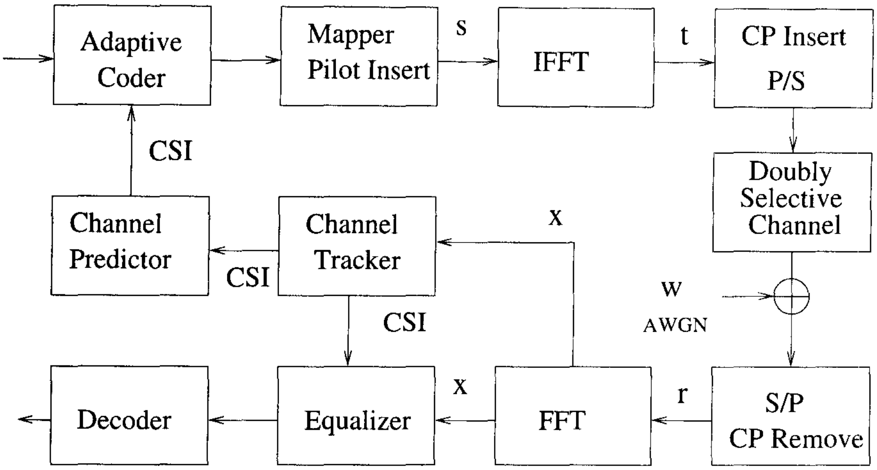


Figure 5.1. Adaptive OFDM system.

where $H^{(m)}[d,k]$ is the channel spectrum in the Doppler domain in the m-th symbol duration

$$H^{(m)}[d,k] := \frac{1}{N}\sum_{n=0}^{N-1}\sum_{l=0}^{N-1} h^{(m)}[n,\ell]e^{j\frac{2\pi}{N}(kn-dn-\ell k)}, \tag{5.5}$$

for $0 \leq d < N$ and $0 \leq k < N$.

5.1.2 Time-Varying Channel Models

The frequency- and time-selective or doubly-selective channel is described by a linear time-varying system [59]

$$h(t,\tau) = \sum_{\ell=1}^{L} a_\ell(t)\delta(\tau - \tau_\ell(t)). \tag{5.6}$$

Assuming a far-field discrete scatterer model, $a_\ell(t)$ can be further decomposed as [59]

$$a_\ell(t) = \sum_{r=1}^{R_\ell} a_{\ell,r}(t)e^{j2\pi v_{\ell,r}(t)t}. \tag{5.7}$$

Then, the channel model (5.6) can be written as

$$h(t,\tau) = \sum_{p=1}^{P} a_p(t)e^{j2\pi v_p(t)t}\delta(\tau - \tau_p(t)), \tag{5.8}$$

where $P = \sum_{\ell=1}^{L} R_\ell$. Notice some $\tau_p(t)$ may be identical. It is assumed that the complex amplitude $a_p(t)$, the time delay $\tau_p(t)$, and Doppler frequency shift $v_p(t)$ vary slowly when compared with duration of one OFDM symbol, and can be regarded as invariant parameters. Therefore a_p and τ_p' and v_p are used to represent the amplitude, delay and Doppler for the p-th path in the sequel. Suppose the duration of one OFDM symbol is T_s, then

$$h^{(m)}[n,\ell] = h\left(mT_s + \frac{nT_s}{N+N_c}, \frac{\ell T_s}{N+N_c}\right). \tag{5.9}$$

Substituting (5.8) and (5.9) to (5.5), and defining $\tau_p := \tau_p' \frac{N+N_c}{N}$, which can be regarded as a delay parameter of p-th path, one has

$$H^{(m)}[d,k] = \frac{1}{N}\sum_{p=1}^{P} a_p e^{j2\pi m T_s v_p} e^{-j2\pi k \frac{\tau_p}{T_s}} S^{(m)}[p,k-d],$$

where $S^{(m)}[p,k-d]$ is the p-th path Doppler spectrum in one OFDM symbol duration T_s

$$S^{(m)}[p,k-d] = \frac{1 - e^{j2\pi v_p T_s \frac{N}{N+N_c}}}{1 - e^{j2\pi\left(\frac{T_s v_p}{N+N_c} + \frac{k}{N} - \frac{d}{N}\right)}}. \tag{5.10}$$

If it is assumed that $|v_p T_s \frac{N}{N+N_c}| \ll 1$, $p = 1, \ldots, P$, then

$$S^{(m)}[p,k-d] = \begin{cases} N, & d = k, \\ 0, & d \neq k. \end{cases} \tag{5.11}$$

Therefore (5.4) becomes

$$x_n^{(m)} = H^{(m)}[n] s_n^{(m)}, \tag{5.12}$$

for $0 \leq m < M$ and $0 \leq n < N$. Here $H^m[n]$ is the long range channel spectrum

$$H^{(m)}[n] = \sum_{p=1}^{P} a_p e^{j2\pi m T_s v_p} e^{-j2\pi n \frac{\tau_p}{T_s}}. \tag{5.13}$$

5.1.3 Pilot Pattern

In a pilot-assisted OFDM system, the pilot tones $s_{n_\ell}^{(m_k)}$ are evenly inserted in the time-frequency plane:

$$m_k = m_1 + (k-1)\Delta M, \quad k = 1, \dots, K,$$
$$n_\ell = n_1 + (\ell - 1)\Delta N, \quad \ell = 1, \dots, L.$$

A least-squares estimate of $H_{\mathrm{LS}}^{(m_k)}[n_\ell]$ is obtained by the division of the received signal over the known pilot symbols

$$H_{\mathrm{LS}}^{(m_k)}[n_\ell] = \frac{x_{n_\ell}^{(m_k)}}{s_{n_\ell}^{(m_k)}}. \tag{5.14}$$

Once $H_{\mathrm{LS}}^{(m_k)}[n_\ell]$ is obtained, the model parameters can be estimated from $H_{\mathrm{LS}}^{(m_k)}[n_\ell]$ through (5.13). This problem is 2-D frequency estimation. In the following, a 2-D frequency estimation algorithm is proposed to estimate the model parameters $\{a_p, v_p, \tau_p\}_{p=1}^{P}$ from $\{H_{\mathrm{LS}}^{(m_k)}[n_\ell]\}_{\ell=1}^{L}$, $k = 1, \dots, K$.

5.2 Channel Identification and Prediction

Given $K \times L$ least-squares estimates of channel spectrum $H_{\mathrm{LS}}^{(m_k)}[n_\ell]$ with

$$H_{k,\ell} := H_{\mathrm{LS}}^{(m_k)}[n_\ell] = \sum_{p=1}^{P} c_p e^{j2\pi(k-1)\Delta M T_s v_p} e^{-j2\pi(\ell-1)\Delta N \frac{\tau_p}{T_s}} \tag{5.15}$$

for $k = 1, \dots, K$, and $\ell = 1, \dots, L$, where

$$c_p = a_p e^{j2\pi m_1 T_s v_p} e^{-j2\pi n_1 \frac{\tau_p}{T_s}},$$

It is desired to estimate the channel parameter sets $\{a_p, v_p, \tau_p\}_{p=1}^{P}$.

Given (5.15), one can define two Vandermonde matrices: $\Theta \in \mathbb{C}^{K \times P}$ and

$\boldsymbol{\Phi} \in \mathbb{C}^{L \times P}$ such that

$$\boldsymbol{\Theta} := [\boldsymbol{\theta}_1 \; \boldsymbol{\theta}_2 \; \cdots \; \boldsymbol{\theta}_P], \tag{5.16}$$

$$\boldsymbol{\theta}_p := \left[1 \; e^{j2\pi\Delta M T_s \upsilon_p} \; \ldots \; e^{j2\pi(K-1)\Delta M T_s \upsilon_p}\right]^T,$$

$$\boldsymbol{\Phi} := [\boldsymbol{\phi}_1 \; \boldsymbol{\phi}_2 \; \cdots \; \boldsymbol{\phi}_P], \tag{5.17}$$

$$\boldsymbol{\phi}_p := \left[1 \; e^{-j2\pi\Delta N \frac{\tau_p}{T_s}} \; \ldots \; e^{-j2\pi(L-1)\Delta N \frac{\tau_p}{T_s}}\right]^T.$$

The data model in (5.15) can be written in matrix form as

$$\boldsymbol{H} = \boldsymbol{\Theta}\boldsymbol{D}(\boldsymbol{c})\boldsymbol{\Phi}^T, \tag{5.18}$$

where $\boldsymbol{c} = [c_1, c_2, \ldots, c_P]^T$ and $[\boldsymbol{H}]_{k,\ell} = H_{k,\ell}$. Eqn. (5.18) can also be written in vector form. Let

$$\boldsymbol{h} = [H_{1,1} \;\; H_{1,2} \; \cdots \; H_{1,L} \;\; H_{2,1} \; \cdots \; H_{K,L}]^T, \tag{5.19}$$

then it can be verified that

$$\boldsymbol{h} = (\boldsymbol{\Theta} \odot \boldsymbol{\Phi})\boldsymbol{c}. \tag{5.20}$$

Here $\odot$ stands for the Khatri-Rao (column-wise Kronecker) product, i.e.,

$$\boldsymbol{\Theta} \odot \boldsymbol{\Phi} := [\boldsymbol{\theta}_1 \otimes \boldsymbol{\phi}_1 \; \boldsymbol{\theta}_2 \otimes \boldsymbol{\phi}_2 \; \cdots \; \boldsymbol{\theta}_P \otimes \boldsymbol{\phi}_P].$$

The Khatri-Rao product will be used in the sequel.

If the smoothing operator defined in (2.1) is applied, then it can be verified that [38]

$$\mathcal{S}(\boldsymbol{h}) = \boldsymbol{G}\boldsymbol{D}(\boldsymbol{c})\boldsymbol{E}^T, \tag{5.21}$$

where

$$\boldsymbol{G} := \boldsymbol{\Theta}^{(K_1)} \odot \boldsymbol{\Phi}^{(L_1)}, \quad \boldsymbol{E} := \boldsymbol{\Theta}^{(K_2)} \odot \boldsymbol{\Phi}^{(L_2)}.$$

To further explore the data structure, one can perform the backward smoothing on the data vector $\boldsymbol{h}$ in (1.5). Define

$$\widetilde{\boldsymbol{h}} = \Pi_M \boldsymbol{h}^*, \tag{5.22}$$

where $\mathbf{\Pi}_M$ is an $M \times M$ permutation matrix with ones on its anti-diagonal. It can be verified that

$$\widetilde{\boldsymbol{h}} := (\boldsymbol{\Theta} \odot \boldsymbol{\Phi})\widetilde{\boldsymbol{c}},$$

where $\widetilde{\boldsymbol{c}} = [\widetilde{c}_1, \widetilde{c}_2, \cdots, \widetilde{c}_P]^T$ with $\widetilde{c}_p = c_p^* e^{-j\beta_p}$, and β_f is defined as

$$\beta_p = 2\pi(K-1)\Delta M T_s v_p - 2\pi(L-1)\Delta N \frac{\tau_p}{T_s}. \tag{5.23}$$

Applying the same 2-D smoothing technique defined in (2.3) to $\widetilde{\boldsymbol{h}}$, one can obtain

$$\mathcal{S}(\widetilde{\boldsymbol{h}}) = \boldsymbol{G}\boldsymbol{D}(\widetilde{\boldsymbol{c}})\boldsymbol{E}^T. \tag{5.24}$$

Then, one can define a big matrix using the two matrices $\mathcal{S}(\boldsymbol{h})$ and $\mathcal{S}(\widetilde{\boldsymbol{h}})$

$$\mathcal{H} := \left[\mathcal{S}(\boldsymbol{h}) \;\; \mathcal{S}(\widetilde{\boldsymbol{h}})\right]. \tag{5.25}$$

Notice that $\mathcal{H}$ is matrix of size $K_1L_1 \times 2K_2L_2$.

The maximal possible number of pathes the algorithm can resolve is called the identifiability bound, which is denoted as $P_{\max}$. It can be proved that [38]

$$P_{\max} = \begin{cases} \Gamma(K-1, L), & K < L, \\ \Gamma(K, L-1), & K > L, \end{cases} \tag{5.26}$$

where $\Gamma(K, L)$ is the function defined as

$$\Gamma(K, L) = \max_{\substack{K_1+K_2=K+1 \\ L_2+L_2=L+1}} \min(K_1L_1,\ 2K_2L_2). \tag{5.27}$$

Generally, $P_{\max}$ can be as large as about $0.34KL$ in noiseless case. In the algorithm, the 2-D smoothing parameters K and L are carefully chosen such that $P_{\max}$ achieve its maximal value [38].

5.2.1 Estimation of the Model Order and Signal Subspace

The signal subspace can be obtained by the singular value decomposition (SVD) to the following matrix

$$\boldsymbol{R} := \frac{1}{2K_2L_2}\mathcal{H}\mathcal{H}^H = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^H, \tag{5.28}$$

The choice of model order P is obtained by the Minimized Description Length (MDL) algorithm [60]. Suppose the eigenvalues of $\boldsymbol{R}$ in (5.28) are $\lambda_1 \geq \lambda_2 \cdots \lambda_P \geq \lambda_{P+1} \cdots \geq \lambda_M$, and the power of the AWGN is σ^2, then the MDL cost function is

$$\begin{aligned}\mathrm{MDL}(P) = {} & 2K_2L_2(K_1L_1 - P)\ln\left(\frac{1}{K_1L_1 - P}\prod_{p=P+1}^{K_1L_1}\lambda_p\right) \\ & - 2K_2L_2\ln\prod_{p=P+1}^{K_1L_1}\lambda_p + \frac{1}{2}P(2K_1L_1 - P + 1)\ln(2K_2L_2)\,.\end{aligned} \tag{5.29}$$

One should choose P such that $\mathrm{MDL}(P)$ is minimized. Once P is determined, the signal subspace $\boldsymbol{U}_s$ is composed by the P principle eigenvectors of $\boldsymbol{R}$. From (5.28), it is noticed that the column spaces spanned by $\boldsymbol{U}_s$ and $\boldsymbol{G}$ are equivalent, therefore there exists a nonsingular transformation matrix $\boldsymbol{T}^{-1}$ such that

$$\boldsymbol{U}_s = \boldsymbol{G}\boldsymbol{T}^{-1}. \tag{5.30}$$

5.2.2 Estimation of the Transformation Matrix $\boldsymbol{T}$

In the following, weighting factors are used to construct a matrix, whose eigenvector matrix is $\boldsymbol{T}$. The construction approach is similar to (3.2) in Chapter 3.

One can construct a matrix pencil along the first dimension,

$$\boldsymbol{U}_{1,1} := \left(\left[\boldsymbol{I}_{K_1-1}\ \boldsymbol{0}_{(K_1-1)\times 1}\right] \otimes \boldsymbol{I}_{L_1}\right)\boldsymbol{U}_s, \tag{5.31}$$

$$\boldsymbol{U}_{1,2} := \left(\left[\boldsymbol{0}_{(K_1-1)\times 1}\ \boldsymbol{I}_{K_1-1}\right] \otimes \boldsymbol{I}_{L_1}\right)\boldsymbol{U}_s. \tag{5.32}$$

Then it can be verified that

$$\boldsymbol{U}_{1,1}^{\dagger}\boldsymbol{U}_{1,2} := \boldsymbol{T}^{-1}\boldsymbol{D}(\boldsymbol{v})\boldsymbol{T},$$

where

$$\boldsymbol{v} := \left[e^{j2\pi\Delta MT_s v_1}\ e^{j2\pi\Delta MT_s v_2}\ \cdots\ e^{j2\pi\Delta MT_s v_P}\right]^T.$$

Similarly one can construct a matrix pencil along the second dimension

$$\boldsymbol{U}_{2,1} := \left(\boldsymbol{I}_{K_1} \otimes \left[\boldsymbol{I}_{L_1-1}\ \boldsymbol{0}_{(L_1-1)\times 1}\right]\right)\boldsymbol{U}_s, \tag{5.33}$$

$$\boldsymbol{U}_{2,2} := \left(\boldsymbol{I}_{K_1} \otimes \left[\boldsymbol{0}_{(L_1-1)\times 1}\ \boldsymbol{I}_{L_1-1}\right]\right)\boldsymbol{U}_s. \tag{5.34}$$

Then, it can be verified that

$$\boldsymbol{U}_{2,1}^{\dagger}\boldsymbol{U}_{2,2} := \boldsymbol{T}^{-1}\boldsymbol{D}(\boldsymbol{\tau})\boldsymbol{T},$$

where

$$\boldsymbol{\tau} := \left[e^{-j2\pi\Delta N\frac{\tau_1}{T_s}}\ e^{-j2\pi\Delta N\frac{\tau_2}{T_s}}\ \cdots\ e^{-j2\pi\Delta M\frac{\tau_P}{T_s}}\right]^T.$$

Next, using two complex weighting factors α and β satisfying $|\alpha| \leq 1$ and $|\beta| \leq 1$, one can perform eigenvalue decomposition (EVD) to the following matrix

$$\boldsymbol{Q} := \alpha\boldsymbol{U}_{1,1}^{\dagger}\boldsymbol{U}_{1,2} + \beta\boldsymbol{U}_{2,1}^{\dagger}\boldsymbol{U}_{2,2} = \boldsymbol{T}^{-1}\boldsymbol{D}(\boldsymbol{\zeta})\boldsymbol{T}. \tag{5.35}$$

where $\boldsymbol{\zeta} := [\zeta_1\ \zeta_2\ \cdots\ \zeta_P]^T$ and

$$\zeta_p := \alpha e^{j2\pi\Delta MT_s v_p} + \beta e^{-j2\pi\Delta N\frac{\tau_p}{T_s}}, \tag{5.36}$$

for $p = 1, \ldots, P$. The weighting factors α and β are randomly chosen at the first time, once one has the estimates of $e^{2\pi\Delta MT_s v_p}$ and $e^{-2\pi\Delta N\frac{\tau_p}{T_s}}$, he can obtain better weighting factors by solving the following optimization problem. As is shown in Chapter 4, these optimization criteria can be obtained from the perturbation analysis.

The weighting factors $\{\alpha, \beta\}$ are chosen so as to minimize the following cost function

$$\gamma(\alpha, \beta) = \sum_{p=1}^{P}\sum_{q=p+1}^{P}\frac{1}{|\zeta_p - \zeta_q|}, \tag{5.37}$$

$$(\alpha, \beta)_{\text{opt}} = \arg\min_{|\alpha|\leq 1, |\beta|\leq 1}\gamma(\alpha, \beta). \tag{5.38}$$

In order to solve the optimization problem (5.38) using a quadratic programming (SQP) method, one can obtain an initial value of (α, β) by solving the following optimization problem

$$\epsilon(\alpha, \beta) := \min_{1\leq p,q\leq P}|\zeta_p - \zeta_q|.$$

$$(\alpha, \beta)_0 = \arg\max_{|\alpha|\leq 1, |\beta|\leq 1}\epsilon(\alpha, \beta). \tag{5.39}$$

The eigenvector matrix of EVD in (5.35) is

$$\boldsymbol{T}_{sp} := \boldsymbol{T}\boldsymbol{\Lambda}\boldsymbol{\Delta}, \tag{5.40}$$

where $\boldsymbol{\Lambda}$ is a nonsingular diagonal column scaling matrix and $\boldsymbol{\Delta}$ is a permutation matrix.

5.2.3 Estimation of Model Parameters

Once the transformation matrix is obtained, one can obtain the estimate of matrix $\boldsymbol{G}$ according to (c.f. (5.30))

$$\widehat{\boldsymbol{G}}_{sp} = \boldsymbol{U}_s\boldsymbol{T}_{sp}. \tag{5.41}$$

Notice that $\boldsymbol{G}$ is the Khatri-Rao product of two Vandermonde matrices $\boldsymbol{\Theta}^{(K_1)}$ and $\boldsymbol{\Phi}^{(L_1)}$, and there are P columns in $\boldsymbol{G}$. The Doppler and delay parameters of the same path appear in the same column of $\boldsymbol{G}$. In other words, for fixed p, v_p and τ_p appear in the same column of $\boldsymbol{G}$. Thanks to this structure, the exponentials containing v_p and τ_p can be obtained by dividing suitably chosen elements of the aforementioned columns of $\boldsymbol{G}_{sp}$. Therefore the column scaling and permutation will not have material effect on the algorithm. Since there are many quotients can be regarded as the estimate of the exponentials, one can take average over them to reduce estimation error variance. Suppose the Doppler and delay parameters of the p-th path appear in the p-th column of $\widehat{\boldsymbol{G}}_{sp}$, then $e^{j2\pi\Delta MT_s v_p}$ can be estimated through

$$e^{j2\pi\widehat{\Delta MT_s}v_p} = \frac{1}{(K_1-1)L_1}\sum_{n=L_1+1}^{K_1L_1}\frac{\widehat{g}_{n,p}}{\widehat{g}_{n-L_1,p}}. \tag{5.42}$$

Similarly, $e^{-j2\pi\Delta N\frac{\tau_p}{T_s}}$ can be estimated by

$$\widehat{e^{-j2\pi\Delta N\frac{\tau_p}{T_s}}} = \frac{1}{K_1(L_1-1)}\sum_{\substack{n=2\\ \text{mod }(n,L_1)\neq 1}}^{K_1L_1}\frac{\widehat{g}_{n,p}}{\widehat{g}_{n-1,p}}. \tag{5.43}$$

The Doppler frequency and delay parameters of the p-th path can be obtained by

$$\widehat{v_p} = \mathcal{I}\left(\log \widehat{e^{j2\pi\Delta MT_s v_p}}\right) / \left(2\pi\Delta MT_s\right), \tag{5.44}$$

$$\widehat{\tau_p} = \mathcal{I}\left(\log \widehat{e^{-j2\pi\Delta N\frac{\tau_p}{T_s}}}\right) T_s / \left(-2\pi\Delta N\right), \tag{5.45}$$

where $\mathcal{I}(\cdot)$ stands for the imaginary part. After the frequency estimates are obtained, the amplitude $\boldsymbol{c}$ can be obtained by applying a least-squares approach to (5.20)

$$\widehat{\boldsymbol{c}} = \left(\widehat{\boldsymbol{\Theta}} \odot \widehat{\boldsymbol{\Phi}}\right)^{\dagger} \boldsymbol{h}, \tag{5.46}$$

where $\widehat{\boldsymbol{\Theta}}$ and $\widehat{\boldsymbol{\Phi}}$ are obtained by substituting the estimates $\widehat{v_p}$ and $\widehat{\tau_p}$ into (5.16) and (5.17). The complex amplitude of p-th path a_p is obtained by

$$\widehat{a_p} = \widehat{c_p} e^{-j2\pi m_1 T_s \widehat{v_p} + j2\pi n_1 \frac{\widehat{\tau_p}}{T_s}}. \tag{5.47}$$

The proposed algorithm for channel identification in OFDM system is summarized in Table 5.1.

5.2.4 Channel Prediction

After all the parameters $\{v_p,\ \tau_p,\ a_p\}_{p=1}^{P}$ in the channel model (5.13) are obtained, the channel variation at near future (for example, at the time when (m_K+i)-th symbol is transmitted) can be predicted by assuming that these parameters do not change or only change slightly. The predicted channel is

$$H^{(m_K+i)}[n] = \sum_{p=1}^{P} a_p e^{j2\pi(m_K+i)T_s v_p} e^{-j2\pi n\frac{\tau_p}{T_s}}. \tag{5.48}$$

5.3 Simulation Results

This section presents the Monte Carlo simulation results. The simulated channel wireless fading has $N_h = 6$ taps. A modified Jakes simulator introduced in [61] with 8 propagation paths is applied to simulate each taps since the simulator has desired statistics for Rayleigh fading channels even when the number of sinusoids is

TABLE 5.1

OFDM channel identification using joint 2-D frequency estimation

(1) Obtain the least-squares estimates of the channel by (5.14).

(2) Determine the smoothing parameter K_1 and L_1 by solving (5.26), perform 2-D smoothing to form $\mathcal{H}$ by (5.25).

(3) Compute the sample covariance matrix $\boldsymbol{R}$ by (5.28), and perform SVD to $\boldsymbol{R}$ and determine the model order P by the MDL principle, and obtain signal subspace $\boldsymbol{U}_s$.

(4) Compute $\boldsymbol{U}_{1,1}^{\dagger}\boldsymbol{U}_{1,2}$ and $\boldsymbol{U}_{2,1}^{\dagger}\boldsymbol{U}_{2,2}$.

(5) Randomly choose α and β, perform EVD to $\boldsymbol{Q}$ by (5.35), obtain $\boldsymbol{G}$ by (5.41), and obtain estimates of exponentials $e^{j2\pi\Delta M T_s v_p}$ and $e^{-j2\pi\Delta N \frac{\tau_p}{T_s}}$ by (5.42) and (5.43), respectively.

(6) Based on the estimates of $e^{j2\pi\Delta M T_s v_p}$ and $e^{-j2\pi\Delta N \frac{\tau_p}{T_s}}$, compute $\boldsymbol{\zeta}$ by (5.36), obtain better factors α and β by solving (4.11) and then (5.38), repeat step (5) using the updated factors.

(7) Obtain the Doppler shift and delay of the path by (5.44), estimate $\boldsymbol{c}$ by (5.46) and a_p by (5.47).

small. Consider an OFDM system with bandwidth BW $=$ 5 MHz and carrier frequency 2.6 GHz as in IEEE 802.16 [62]. The sampling factor is 144/125, and the sampling rate $f_s = \mathrm{BW} \cdot 144/125 = 5.76$ MHz. There are $N = 256$ subcarriers in the system and the length of the cyclic prefix for each OFDM symbol is $N_c = 64$, therefore the duration of one OFDM symbols is $T_s = (N + N_c)/f_s = 55.56$ μs. Suppose the maximum Doppler frequency $f_d = 180$ Hz, which is corresponding to a mobility of 75 km/h. Notice the $\tau_{\max} = (N_h - 1)/f_s = 0.87$ μs. The pilots are distributed in the frequency-time plane evenly and satisfy the reconstruction conditions mentioned in [63]:

$$f_d T_s \Delta M \leq \frac{1}{2}, \quad \tau_{\max} \frac{f_s}{N} \Delta N \leq \frac{1}{2}. \tag{5.49}$$

In the simulation, $M = 800$, $m_1 = 1$, $n_1 = 1$, $\Delta M = 25$, $\Delta N = 8$, $K = 24$, and $L = 32$. Therefore, the channel is interpolated in the duration from the first to the 576-th symbol, where pilot symbols are inserted, while the channel is predicted in the duration from the 577-th to the 800-th symbol, where no pilot is applied. Notice there are only a few pilot symbols in the system. One advantage of the proposed method is that even with few pilot symbols, the maximum number of identifiable path $P_{\max}$ can be as large as about $\lfloor 0.34KL \rfloor$.

Since it was shown in [55] that parametric channel estimation outperforms its non-parametric counterpart, one can only consider parametric channel estimation in the simulation. The proposed algorithm is compared to the $2\times$1-D ESPRIT algorithm proposed in [58]. The signal-to-noise ratio (SNR) is defined as (cf. (5.2)):

$$\mathrm{SNR} = -10 \log \sigma^2. \tag{5.50}$$

Fig. 5.2 plots the channel trace of the amplitude of the first tap at SNR $= 15$ dB, where the horizonal axis represent the time in the unit of T_s. It can be seen that the channel amplitudes can be tracked for both algorithms in the duration when the pilot symbols are available. After the 576-th symbol, the estimation error increase with the growth of prediction length. A snapshot of the predicted channel in frequency

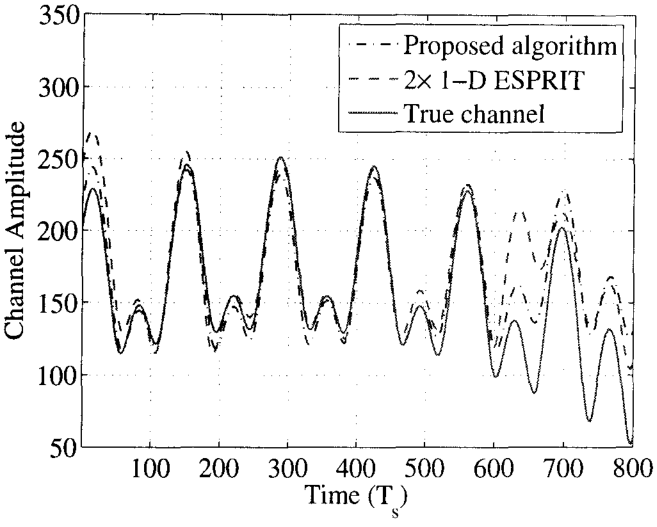


Figure 5.2. Channel trace of the first tap at SNR = 15 dB.

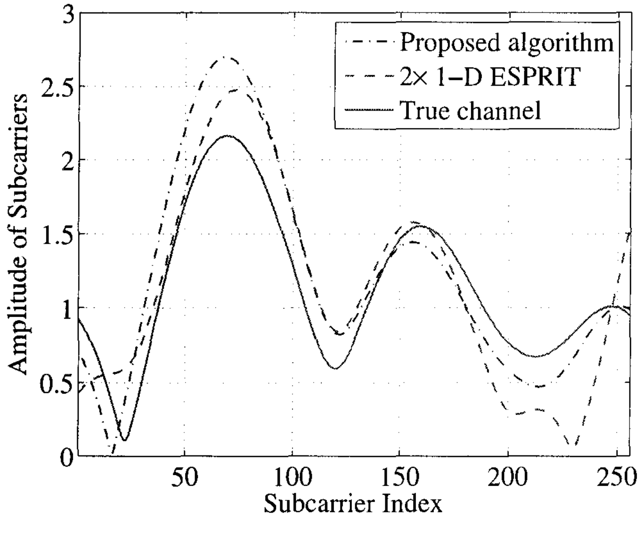
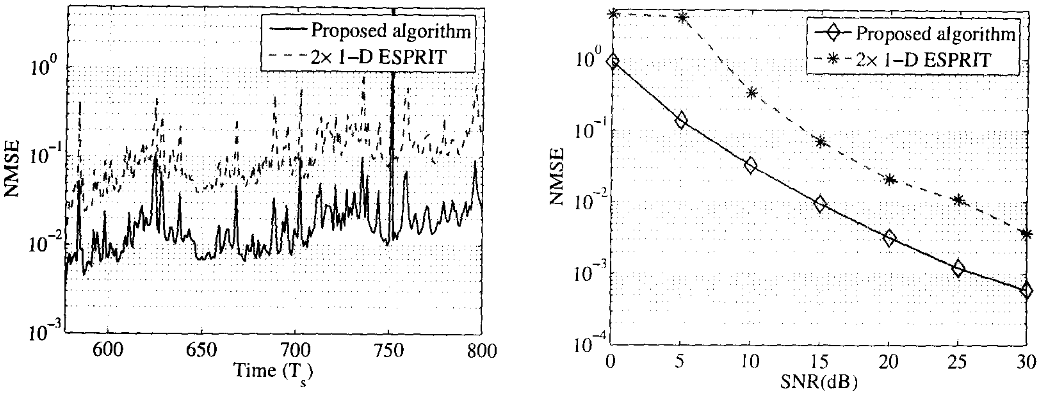


Figure 5.3. Snapshot of predicted channel at 610-th symbol at SNR = 15 dB.



(a) (b)

Figure 5.4. Channel NMSE versus (a) time at SNR=15 dB and (b) SNR when the 610-th symbol is received.

domain is plotted in Fig. 5.3, where the amplitudes of the 256 subcarriers are depicted at SNR = 15 dB when the 610-th symbol is received.

Fig. 5.4(a) plots the normalized mean square error (NMSE) of the channel estimation in each OFDM symbol for the two algorithms at SNR = 15 dB. The horizonal axis represent the time in the unit of T_s. The NMSE are averaged over all NMSE's that are collected from 400 realizations. It can be seen from Fig. 5.4(a) that the proposed algorithm outperforms the 2 × 1-D ESPRIT algorithm.

Fig. 5.4(b) plots the NMSE of the predicted channel for the two algorithms versus SNR when the 610-th symbol is received. It can be seen from Fig. 5.4(b) that the prediction performance of proposed algorithm is better that of the 2 × 1-D ESPRIT algorithm in all SNR range.

CHAPTER 6

APPLICATION IN TRACKING OF MULTIPLE MOVING TARGETS

It is clear that many DOA estimation problems in radar signal process can be reduced to multidimensional frequency estimation. This chapter develops a fast eigenvector-based frequency estimation algorithm using adaptive weighting factors to tracking the 2-D DOA of multiple move targets. The update of those weighting factors is based on the optimization criterion derived in Chapter 4.

The problem of tracking the direction-of-arrivals (DOAs) of multiple moving targets in radar signal processing has attracted much interests in recent years. Various methods based on adaptive filter and Bayesian statistics were proposed [64]. However, most previous work assumed that the DOAs were one dimension vectors. A uniform rectangular array (URA) can be applied to track the DOAs in two dimensions: elevation and azimuth. The problem of jointly estimating the two dimensional (2-D) angles is actually a 2-D frequency estimation problem. Effectively associating the 2-D angles of the same target is critical. The Unitary ESPRIT algorithm was applied to jointly estimate 2-D DOAs in [65]. Based on simultaneous Schur decomposition, the Unitary ESPRIT algorithm was generalized to multidimensional case in [13]. The problem of jointly tracking 2-D DOA was considered in [66], where a subspace tracking algorithm (i.e., Bi-SVD) was applied to track the subspaces of the structural matrices in an adaptive MI-ESPRIT algorithm. The MI-ESPRIT algorithm employed adaptive simultaneous Schur decomposition to estimate the 2-D DOAs.

As pointed out in Chapter 2, the eigenvector-based frequency estimation algorithms do not need a frequency pairing step or joint diagonalization process, and have relatively lower computational complexity. It is natural to consider developing low complexity frequency tracking algorithms based on these eigenvector-based algorithms.

This chapter proposes a new 2-D DOA tracking algorithm based on the optimization criterion derived in Chapter 4. Similar to [66], subspace tracking technique is applied to estimate the instantaneous signal subspace, but another subspace tracking algorithm - LOAFR1 [67] is adopted, since the LOAFR1 algorithm demonstrates better performance over other subspace tracking algorithms and has the same complexity order as the remaining steps of the proposed algorithm. The signal subspace shares the same column space as that of a structural matrix, which is the Khatri-Rao product of several Vandermonde matrices. The transformation matrix connecting the structural matrix and the signal subspace is estimated through a so-called "weighted diagonalization" method, which performs eigenvalue decomposition (EVD) of an adaptively weighted matrix. The weighting factors are updated adaptively according to current angle estimates and an optimization criterion similar to those in Chapter 4. The sequential quadratic programming (SQP) process in Chapter 4 is avoided here, thus computational complexity is reduced. The computational order of the proposed tracking algorithm is as low as $\mathcal{O}(MNF^2)$, where M and N are the dimension sizes of the URA array in elevation and azimuth directions, F is the number of targets. It can be used to estimate the 2-D DOA of multiple moving targets at every snapshot and track the DOA trajectories in real time. The performance of the proposed algorithm is evaluated by numerical simulations.

6.1 Data Model for 2-D DOA Tracking

This chapter assumes that the number of targets does not change in the tracking process. Suppose there are F moving targets that are to be tracked by a URA of size $M \times N$. Suppose the elevation and azimuth angles of the f-th target at tth snapshot are $\theta_f(t)$ and $\phi_f(t)$, which are the angles with respect with x-axis and y-axis;

the inter-sensor spacings in x-axis and y-axis are Δ_x and Δ_y, respectively. Define

$$\omega_f(t) = \frac{2\pi}{\lambda}\Delta_x \cos\theta_f(t), \quad \nu_f(t) = \frac{2\pi}{\lambda}\Delta_y \cos\phi_f(t), \tag{6.1}$$

where λ is the wavelength. The output signal of the URA at the (m, n)-th sensor is modeled as [65]

$$x_{m,n}(t) = \sum_{f=1}^{F} c_f(t) e^{j(m-1)\omega_f(t)} e^{j(n-1)\nu_f(t)}, \ t = 1, \ldots, T,$$

where $m = 1, \ldots, M$ and $n = 1, \ldots, N$. If one chooses $\Delta_x = \Delta_y = \frac{\lambda}{2}$, and assumes $0 < \theta_f(t), \phi_f(t) \leq \pi$, there is one-to-one mapping between $\omega_f(t)$ and $\theta_f(t)$, and between $\nu_f(t)$ and $\phi_f(t)$, where $-\pi \leq \omega_f(t) < \pi$ and $-\pi \leq \nu_f(t) < \pi$. For this reason, $\omega_f(t)$ and $\phi_f(t)$ are referred as the 2-D DOA of the f-th target in the sequel.

If the f-th target moves, $(\omega_f(t), \nu_f(t))$ travels a trajectory in the $\Pi \times \Pi$ plane, where $\Pi = [-\pi, \pi)$. The problem of 2-D DOA tracking is to estimate F trajectories $\{(\omega_f(t), \nu_f(t))\}_{f=1}^{F}$ from the observations $\{x_{m,n}(t)\}$ for $t = 1, \ldots, T$. Define the snapshot vector as

$$\boldsymbol{x}(t) = [x_{1,1}(t) \ \ x_{1,2}(t) \ \cdots \ x_{1,N}(t) \ \ x_{2,1}(t) \ \cdots \ x_{M,N}(t)]^T,$$

and the amplitude vector $\boldsymbol{c}(t) := [c_1(t) \ c_2(t) \ \cdots \ c_F(t)]^T$, then it can be verified that

$$\boldsymbol{x}(t) = \big(\boldsymbol{A}(t) \odot \boldsymbol{B}(t)\big)\boldsymbol{c}(t) = \boldsymbol{G}(t)\boldsymbol{c}(t), \quad t = 1, \ldots, T, \tag{6.2}$$

where $\boldsymbol{A}(t)$ and $\boldsymbol{B}(t)$ are Vandermonde matrices with generators $\{e^{j\omega_f(t)}\}_{f=1}^{F}$ and $\{e^{j\nu_f(t)}\}_{f=1}^{F}$ respectively, and the structural matrix $\boldsymbol{G}(t) := \boldsymbol{A}(t) \odot \boldsymbol{B}(t)$. Suppose that the signal subspace of the time-varying correlation matrix of $\boldsymbol{x}(t)$ is $\boldsymbol{Q}(t)$, then $\boldsymbol{Q}(t)$ and $\boldsymbol{G}(t)$ share the same column space, therefore there exists a nonsingular transformation matrix $\boldsymbol{T}(t)$ of size $F \times F$ such that

$$\boldsymbol{G}(t) = \boldsymbol{Q}(t)\boldsymbol{T}(t). \tag{6.3}$$

The proposed fast recursive 2-D frequency tracking has four steps: firstly, apply the LOAFR1 algorithm to estimate the signal subspace $\boldsymbol{Q}(t)$; secondly, estimate the transformation matrix $\boldsymbol{T}(t)$ by performing EVD to an adaptive weighted matrix; then, obtain the estimates of $\boldsymbol{G}(t)$ through (6.3) and estimate the 2-D DOAs by dividing the elements in $\boldsymbol{G}(t)$; finally, update the weighting factors according to the current estimates of DOA. The updated weighting factors are applied to estimate $\boldsymbol{T}(t+1)$ in the next snapshot. The proposed algorithm is detailed in the following.

6.2 Subspace Tracking Based on the LOAFR1 Algorithm

The LOAFR1 subspace tracking algorithm [67] is adopted to estimate the signal subspace $\boldsymbol{Q}(t)$ from $\boldsymbol{x}(t)$ adaptively. The initialization process of the LOAFR1 algorithm is

$$\begin{aligned} &\boldsymbol{Q}(0) = \begin{bmatrix} \boldsymbol{I}_F \\ 0_{(MN-F)\times F} \end{bmatrix}, \quad \boldsymbol{P}(0) = \begin{bmatrix} \boldsymbol{I}_F \\ 0_{(MN-F)\times F} \end{bmatrix}, \\ &\Theta(0) = \boldsymbol{I}_F, \qquad \rho = 0.85. \end{aligned}$$

At time t, the updating process of the LOAFR1 algorithm is

$$\boldsymbol{h}(t) = \boldsymbol{Q}^H(t-1)\boldsymbol{x}(t), \tag{6.4}$$

$$\boldsymbol{P}(t) = \rho\boldsymbol{P}(t-1)\Theta(t-1) + (1-\rho)\boldsymbol{x}(t)\boldsymbol{h}^H(t), \tag{6.5}$$

$$\boldsymbol{Q}(t)\boldsymbol{L}(t) = \boldsymbol{P}(t), \qquad \text{by truncated QR factorization}, \tag{6.6}$$

$$\Theta(t) = \boldsymbol{Q}^H(t-1)\boldsymbol{Q}(t). \tag{6.7}$$

6.3 Estimation of the Transformation Matrix $\boldsymbol{T}(t)$

This section presents a method to solve $\boldsymbol{T}(t)$ using weighting factors. One can construct a matrix pencil along the elevation dimension of $\boldsymbol{Q}(t)$ such that

$$\boldsymbol{U}_{1,1}(t) := \left(\left[\boldsymbol{I}_{M-1}\ 0_{(M-1)\times 1}\right] \otimes \boldsymbol{I}_N\right)\boldsymbol{Q}(t), \tag{6.8}$$

$$\boldsymbol{U}_{1,2}(t) := \left(\left[0_{(M-1)\times 1}\ \boldsymbol{I}_{M-1}\right] \otimes \boldsymbol{I}_N\right)\boldsymbol{Q}(t). \tag{6.9}$$

Then, it can be verified that

$$\boldsymbol{M}_1(t) := \boldsymbol{U}_{1,1}^{\dagger}(t)\boldsymbol{U}_{1,2}(t) = \boldsymbol{T}^{-1}(t)\boldsymbol{D}\big(\boldsymbol{\omega}(t)\big)\boldsymbol{T}(t),$$

where $\boldsymbol{\omega}(t) := \big[e^{j\omega_1(t)}\ e^{j\omega_2(t)}\ \cdots\ e^{j\omega_F(t)}\big]^T$. Similarly one can construct a matrix pencil along the azimuth dimension of $\boldsymbol{Q}(t)$ such that

$$\boldsymbol{U}_{2,1}(t) := \big(\boldsymbol{I}_M \otimes \big[\boldsymbol{I}_{N-1}\ \boldsymbol{0}_{(N-1)\times 1}\big]\big)\,\boldsymbol{Q}(t), \tag{6.10}$$

$$\boldsymbol{U}_{2,2}(t) := \big(\boldsymbol{I}_M \otimes \big[\boldsymbol{0}_{(N-1)\times 1}\ \boldsymbol{I}_{N-1}\big]\big)\,\boldsymbol{Q}(t), \tag{6.11}$$

and it can be verified that

$$\boldsymbol{M}_2(t) := \boldsymbol{U}_{2,1}^{\dagger}(t)\boldsymbol{U}_{2,2}(t) = \boldsymbol{T}^{-1}(t)\boldsymbol{D}(\boldsymbol{\nu}(t))\boldsymbol{T}(t),$$

where $\boldsymbol{\nu}(t) := \big[e^{j\nu_1(t)}\ e^{j\nu_2(t)}\ \cdots\ e^{j\nu_F(t)}\big]^T$. The two variable complex weighting factors $\alpha_1(t-1)$ and $\alpha_2(t-1)$ are initialized randomly and updated adaptively according to Section 6.5. The EVD of the following weighted matrix is

$$\boldsymbol{M}(t) := \alpha_1(t-1)\boldsymbol{M}_1(t) + \alpha_2(t-1)\boldsymbol{M}_2(t) \tag{6.12}$$

$$= \boldsymbol{T}^{-1}(t)\boldsymbol{D}(\boldsymbol{\zeta}(t))\boldsymbol{T}(t), \tag{6.13}$$

where $\boldsymbol{\zeta}(t) := [\zeta_1(t)\ \zeta_2(t)\ \cdots\ \zeta_F(t)]^T$, and

$$\zeta_f(t) := \alpha_1(t-1)e^{j\omega_f(t)} + \alpha_2(t-1)e^{j\nu_f(t)}, \tag{6.14}$$

for $f = 1, \ldots, F$. Consider the column scaling and permutation ambiguity, the EVD of $\boldsymbol{M}(t)$ of (6.13) gives

$$\boldsymbol{T}_{\text{sp}}(t) := \boldsymbol{T}\Lambda(t)\boldsymbol{\Delta}(t), \tag{6.15}$$

where $\Lambda(t)$ is a nonsingular diagonal column scaling matrix and $\boldsymbol{\Delta}(t)$ is a permutation matrix.

6.4 Estimation of DOAs

Once the transformation matrix $\boldsymbol{T}_{\text{sp}}(t)$ is obtained, one can estimate $\boldsymbol{G}(t)$ according to (c.f. (6.3)) $\boldsymbol{G}_{\text{sp}}(t) = \boldsymbol{Q}(t)\boldsymbol{T}_{\text{sp}}(t)$. Since $\boldsymbol{G}(t) = \boldsymbol{A}(t)\odot\boldsymbol{B}(t)$, the elevation and azimuth angles of the same target appear in the same column of $\boldsymbol{G}(t)$. Thanks to this structure, one can obtain the exponentials containing $\{\omega_f(t)\}$ and $\{\nu_f(t)\}$ by dividing suitably chosen elements of the columns of $\boldsymbol{G}_{\text{sp}}(t)$. Therefore the column scaling will not have material effect on the algorithm. Since there are many quotients that can be regarded as the estimate of the exponentials, one can take average over them to reduce estimation error variance in the case when the observation is noisy. This method is the so called "circular mean" in direction statistics. However, because of the existence of the permutation ambiguity $\boldsymbol{\Delta}(t)$, the order of estimated angles is different from the true order in (6.2). Suppose the elevation and azimuth angles of the g-th target appear in the f-th column of $\boldsymbol{G}_{\text{sp}}(t)$, then $e^{j\omega_g(t)}$ can be estimated by

$$e^{j\omega_g(t)} = \frac{1}{(M-1)N} \sum_{n=N+1}^{MN} \frac{g_{n,f}(t)}{g_{n-N,f}(t)}, \quad f, g \in \{1, \dots, F\}. \tag{6.16}$$

where $g_{n,f}(t)$ is the (n, f)-th element of $\boldsymbol{G}_{\text{sp}}(t)$. Similarly, $e^{j\nu_g(t)}$ can be estimated by

$$e^{j\nu_g(t)} = \frac{1}{M(N-1)} \sum_{\substack{n=2 \\ \text{mod }(n,N)\neq 1}}^{MN} \frac{g_{n,f}(t)}{g_{n-1,f}(t)}, \quad f, g \in \{1, \dots, F\}. \tag{6.17}$$

Finally $\omega_g(t)$ and $\nu_g(t)$ of the g-th target can be obtained by

$$\omega_g(t) = \mathcal{I}\left(\log e^{j\omega_g(t)}\right), \quad \nu_g(t) = \mathcal{I}\left(\log e^{j\nu_g(t)}\right), \tag{6.18}$$

where $\mathcal{I}(\cdot)$ stands for the imaginary part.

6.5 Update of the Weighting Factors

Similar to Section 4.1, the weighting factors $\alpha_1(t-1)$ and $\alpha_2(t-1)$ in (6.12) can be optimized so that the performance of eigenvector-based estimation can be improved. However the optimization algorithm (i.e., SQP) used in Section 4.1 is computational demanding and not suitable for a tracking algorithm. Here a low-complexity method is proposed to dynamically adjust the weighting factors. If it is assumed that $|\alpha_i(t)| \leq 0.5$, for $i = 1, 2$, then it is obvious that $|\zeta_f(t)| \leq 1$, for

$f = 1, \ldots, F$. For the EVD in (6.13), it is shown in [24] that the perturbation of the f-th eigenvector $\boldsymbol{t}_f(t)$ is

$$\Delta \boldsymbol{t}_f(t) = \boldsymbol{T}(t)\boldsymbol{D}_f(t)\boldsymbol{T}^{-1}(t)\Delta \boldsymbol{M}(t)\boldsymbol{t}_f(t), \tag{6.19}$$

where $\boldsymbol{D}_f(t)$ is diagonal matrix with $[\boldsymbol{D}_f(t)]_{f,f} = 0$ and $[\boldsymbol{D}_f(t)]_{g,g} = \frac{1}{\zeta_f(t)-\zeta_g(t)}$ for $g \neq f$. In order to minimize the estimation errors, one should minimize the perturbation $\Delta \boldsymbol{T}(t)$ and thus the norm of all $\boldsymbol{D}_f(t)$, $f = 1, \ldots, F$. Therefore it is desired to solve the following optimization problem

$$\boldsymbol{\zeta}_{\text{opt}}(t) = \arg\min_{\boldsymbol{\zeta}(t)} \Gamma(\boldsymbol{\zeta}(t)), \qquad |\zeta_f(t)| \leq 1, f = 1, \ldots, F. \tag{6.20}$$

$$\Gamma(\boldsymbol{\zeta}(t)) := \sum_{f=1}^{F} \sum_{\substack{g=1 \\ g\neq f}}^{F} \frac{1}{|\zeta_f(t) - \zeta_g(t)|}. \tag{6.21}$$

It can be verified that $\Gamma(\boldsymbol{\zeta}(t)) \leq F(F-1)\frac{1}{\beta(\boldsymbol{\zeta}(t))}$, where

$$\beta(\boldsymbol{\zeta}(t)) = \min_{\substack{g=1 \\ g\neq f}} |\zeta_f(t) - \zeta_g(t)|. \tag{6.22}$$

In order to solve (6.20), one can maximize $\beta(\boldsymbol{\zeta}(t))$. If $F \leq 6$, it is easy to prove that the optimal $\{\zeta_f(t)\}_{f=1}^{F}$ should distribute regularly in the unit circle such that

$$\boldsymbol{\zeta}_{\text{opt}} = \begin{bmatrix} 1 & e^{j2\pi/F} & \cdots & e^{j2\pi(F-1)/F} \end{bmatrix}^T. \tag{6.23}$$

For $F > 6$, the problem is related to the circle packing problem and solved in [68]. Define $\boldsymbol{\alpha}(t) := [\alpha_1(t)\ \alpha_2(t)]^T$. Since $\boldsymbol{\omega}(t)$ and $\boldsymbol{\nu}(t)$ change continuously, at time $t+1$, Eqn. (6.14) become

$$\boldsymbol{\zeta}(t+1) = \begin{bmatrix} \boldsymbol{\omega}(t+1) & \boldsymbol{\nu}(t+1) \end{bmatrix} \boldsymbol{\alpha}(t) \approx \begin{bmatrix} \boldsymbol{\omega}(t) & \boldsymbol{\nu}(t) \end{bmatrix} \boldsymbol{\alpha}(t) \tag{6.24}$$

In order to make $\boldsymbol{\zeta}(t+1)$ close to the optimal distribution as in (6.23), one can update $\boldsymbol{\alpha}(t)$ by solving the following least-squares (LS) problem

$$\boldsymbol{\alpha}(t) = \arg\min_{\boldsymbol{\alpha}} \left\| \begin{bmatrix} \boldsymbol{\omega}(t) & \boldsymbol{\nu}(t) \end{bmatrix} \boldsymbol{\alpha} - \boldsymbol{\zeta}_{\text{opt}} \right\|. \tag{6.25}$$

TABLE 6.1
The complexity order of the 2-D DOA tracking algorithm

Algorithm Operations	Complexity Order
$\boldsymbol{h}(t) = \boldsymbol{Q}^H(t-1)\boldsymbol{x}(t)$ in (6.4)	MNF
Updating $\boldsymbol{P}(t)$ as (6.5)	$MNF^2 + MNF + F^2$
QR factorization to $\boldsymbol{P}(t)$	$2MNF^2$
$\Theta(t) = \boldsymbol{Q}^H(t-1)\boldsymbol{Q}(t)$ in (6.7)	MNF^2
$\boldsymbol{M}_1(t) = \boldsymbol{U}^{\dagger}_{1,1}(t)\boldsymbol{U}_{1,2}(t)$	$2MNF^2 + 8F^3/3$
$\boldsymbol{M}_2(t) = \boldsymbol{U}^{\dagger}_{2,1}(t)\boldsymbol{U}_{2,2}(t)$	$2MNF^2 + 8F^3/3$
EVD of $\boldsymbol{M}(t)$	$25F^3$
$\boldsymbol{G}(t) = \boldsymbol{Q}(t)\boldsymbol{T}(t)$	MNF^2
Estimation of $e^{\omega_g(t)}$ by (6.16)	$(M-1)NF$
Estimation of $e^{\nu_g(t)}$ by (6.17)	$M(N-1)F$
$\min \left\Vert [\boldsymbol{\omega}(t) \;\; \boldsymbol{\nu}(t)]\, \boldsymbol{\alpha}(t) - \boldsymbol{\zeta}_{\text{opt}} \right\Vert$	$4F$
Total	$9MNF^2 + 30F^3$

The updated $\boldsymbol{\alpha}(t)$ is employed in (6.12) at time $t+1$.

Since the DOA of one target vary continuous, one can eliminate the permutation ambiguity $\boldsymbol{\Delta}(t)$ and associate the DOA estimate at time t with those at time $t-1$ by minimizing the difference of the exponentials at time t and those at time $t-1$

$$\boldsymbol{p}_o = \arg\min_{\boldsymbol{p}} \left\{ \sum_{f=1}^{F} \left[\left| e^{j\omega_{p(f)}(t)} - e^{j\omega_f(t-1)} \right|^2 + \left| e^{j\nu_{p(f)}(t)} - e^{j\nu_f(t-1)} \right|^2 \right] \right\},$$

Here $\boldsymbol{p} = \{p(1) \; p(2) \; \cdots \; p(F)\}$ is a permutation of $\{1 \; 2 \; \cdots \; F\}$. $e^{j\omega_{p(f)}(t)}$ and $e^{j\nu_{p(f)}(t)}$ are the estimates at time t obtained in (6.16) and (6.17) respectively. The current DOA estimates are appended to the DOA trajectory as $\omega_f(t) = \omega_{p_o(f)}(t)$ and $\nu_f(t) = \nu_{p_o(f)}(t)$.

The consumed flops of the proposed algorithm is counted in Table 6.5. In practise, Gauss elimination is used to solve the equation $\boldsymbol{U}^H_{1,1}(t)\boldsymbol{U}_{1,1}(t)\boldsymbol{X} = \boldsymbol{U}^H_{1,1}(t)\boldsymbol{U}_{2,1}(t)$ to obtain $\boldsymbol{M}_1(t)$. The method of normal equations [37] is employed to solve the LS problem (6.25).

6.6 Simulation Results

This section presents the simulation results. Consider three moving targets, whose DOAs vary as linear functions of time. Therefore $\omega_f(t)$ and $\nu_f(t)$ are sinusoidal functions of time

$$\omega_1(t) = 0.45\pi \sin(60t) + 0.45\pi,\ \nu_1(t) = 0.5\pi \cos(100t) + 0.33\pi,$$

$$\omega_2(t) = 0.4\pi \sin(120t) + 0.33\pi,\ \nu_2(t) = 0.3\pi \cos(50t) + 0.5\pi,$$

$$\omega_3(t) = 0.2\pi \sin(80t) + 0.5\pi,\ \nu_3(t) = 0.16\pi \sin(90t + 0.25\pi) + 0.2\pi.$$

In this specific case, since the variation frequency of $\omega_f(t)$ and $\nu_f(t)$ are relative low (8Hz-20Hz), the sampling period $T_s = 0.0001$s is chosen here. The tracking duration is from 0 s to 0.1 s, therefore there are $T = 1000$ snapshots. If the sampling period increases, the tracking performance become worse because the data sample is not enough to track the relatively fast variation of DOAs. If the sampling period decreases, the tracking duration should be increased accordingly. Since the tracking algorithm require some time to converge and DOAs estimates before convergence is not accurate. The size of the URA is 10×10. The observation signal from the array is polluted by complex AWGN, i.e., Eqn. (6.2) becomes

$$\boldsymbol{x}(t) = \boldsymbol{G}(t)\boldsymbol{c}(t) + \boldsymbol{n}(t),$$

where $\boldsymbol{n}(t)$ is noise with variance σ^2. The signal-to-noise ratio (SNR) is defined as $\mathrm{SNR} = -10\log_{10}\sigma^2$. The amplitudes $\{c_f(t)\}_{t=1}^{T}$ are drawn from independent normal distributions. Fig. 6.1 illustrates the true and estimated DOA trajectories of the moving targets in the $\Pi \times \Pi$ plane at SNR = 2dB for 5 noise realizations. The point in the plane with coordinates $(\omega_f(t), \nu_f(t))$ represents the 2-D DOA of the f-th target at time t. As time elapses, these points form three trajectory curves that describe the variation of the 2-D DOAs of the three moving targets. As is seen from Fig. 6.1, the estimated trajectories match well to the true trajectories. Fig. 6.2 and Fig. 6.3 plot the variation of the elevation and azimuth angles, respectively, as functions of time

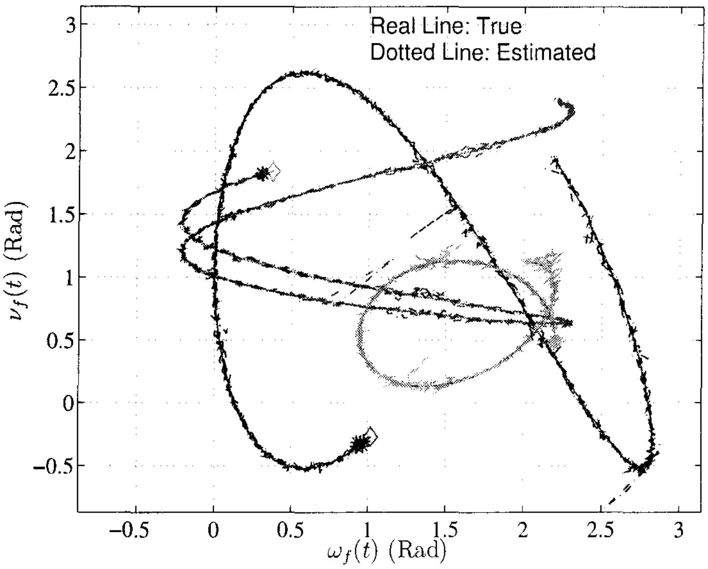


Figure 6.1. The trajectory of the true and estimated 2-D DOAs for three moving targets (SNR = 2dB)

in the unit of T_s at SNR = 2dB for 5 noise realizations. These results demonstrate that the proposed algorithm can track the 2-D DOA variations closely.

The performance of the proposed algorithm is evaluated in various SNRs through Monte Carlo simulation. The scenario is the same as the previous experiment except that the number of snapshot is $T = 200$. For comparison, another 2-D frequency tracking algorithm is implemented by combining the LOAFR1 subspace tracking [67] and the 2-D Unitary ESPRIT algorithm [65], which is called "LOAFR1 + 2D Unitary ESPRIT". Different from the proposed algorithm, this algorithm uses the 2-D Unitary ESPRIT algorithm to estimate $(\omega_f(t), \nu_f(t))$ from $\boldsymbol{Q}(t)$. The normalized mean square error (NMSE) is defined as

$$\mathrm{NMSE} = \frac{1}{2FT}\sum_{t=1}^{T}\sum_{f=1}^{F}\left[\left|\frac{\widehat{\omega}_f(t) - \omega_f(t)}{\omega_f(t)}\right|^2 + \left|\frac{\widehat{\nu}_f(t) - \nu_f(t)}{\nu_f(t)}\right|^2\right],$$

where $\widehat{\omega}_f(t)$ and $\widehat{\nu}_f(t)$ are estimated DOAs. The NMSE of the two algorithms that is averaged over 1000 realizations is plotted in Fig. 6.4. It is evident that the proposed algorithm outperforms the "LOAFR1 + 2D Unitary ESPRIT" algorithm in moderate

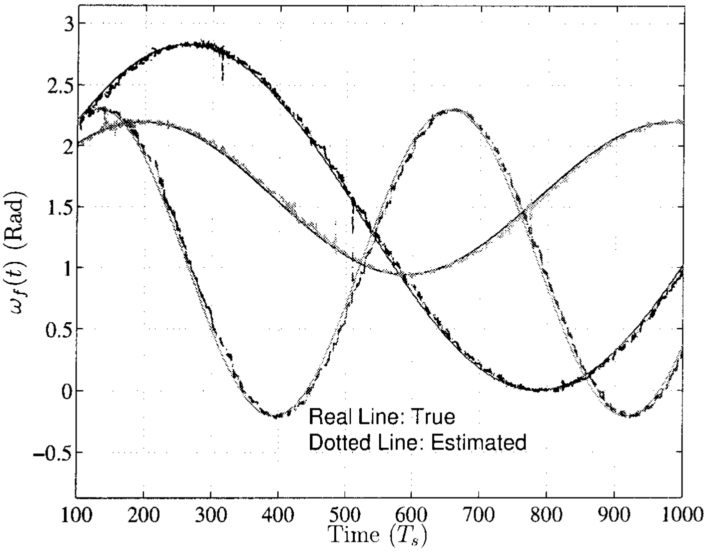


Figure 6.2. The true and estimated elevation angles for three moving targets (SNR = 2dB)

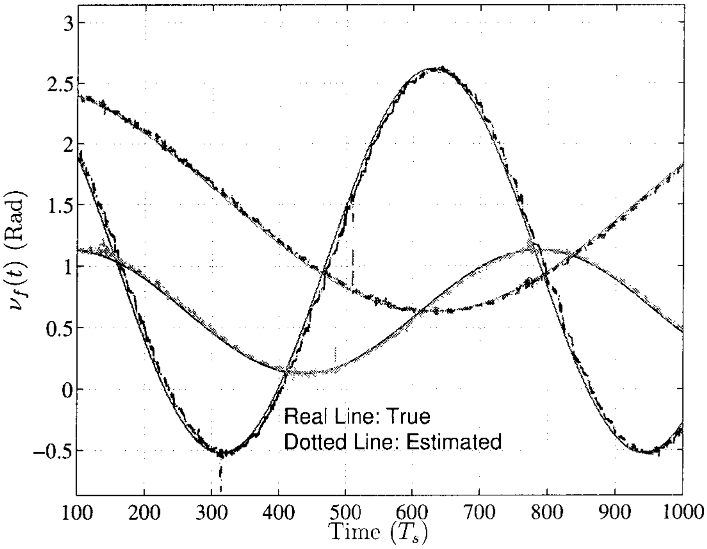


Figure 6.3. The true and estimated azimuth angles for three moving targets (SNR = 2dB)

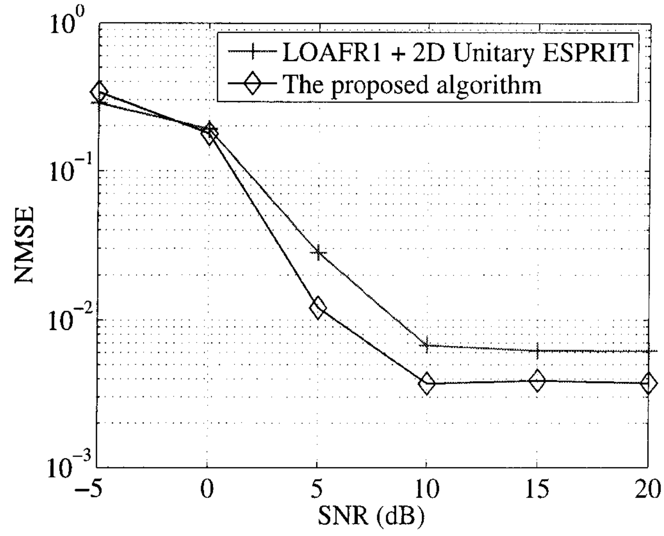


Figure 6.4. The NMSE versus SNR

and high SNR range in Fig. 6.4. This demonstrates the advantage of the proposed adaptive weighted diagonalization scheme since the subspace tracking steps of the two algorithms are identical. Notice that in the high SNR range the floor of NMSE implies these two trackers are both biased estimators due to the estimation delay.

An interesting problem under investigation is the tracking of multiple moving targets when the signal powers fluctuate as the targets move, and the number of signals may also vary. For example, if the number of targets varies, an adaptive order selection approach similar to that of [67] may be considered, and the least squares problem (6.25) also changes.

Another interest problem is the tracking performance of several very fast targets. If the targets are not too fast, the tracking algorithm gives some intermediate points in the trajectories of these targets. However, if the targets are so fast that even the LOAFR1 subspace tracking algorithm can not converge timely, the whole tracking algorithm will fail.

CHAPTER 7

CONCLUSION

Summarily, this dissertation develops a framework on eigenvector-based multidimensional frequency estimation and systematically studies the identifiability, performance, and complexity of several proposed algorithms. Application examples are also presented. A summary comparison is given in Table 7.1 for the algorithms derived in this dissertation and some of those found in the literature. All the algorithms follow the following framework for eigenvector-based N-D frequency estimation, which contains four main steps:

1. Perform N-D smoothing defined in Lemma 2 and FB smoothing to the observation data so as to obtain a large smoothed data matrix. This step is critical to improve the statistical identifiability and alleviate the small sample effects when the number of snapshot is limited.

2. Determine the model order (i.e. the number of components) and perform the SVD to the smoothed data matrix to obtain the signal subspace. This step is often the most computational expensive and dominate the complexity order of the whole algorithm.

3. Estimate the transformation matrix (i.e. $\boldsymbol{T}$) using the matrix pencil method or joint diagonalization method. To some degree, this step is critical for the performance of the algorithm.

4. After the transformation matrix is obtained, the N-D frequencies can be resolved by first estimating a structural matrix (i.e. $\boldsymbol{G}$ or $\boldsymbol{P}$) and then computing

TABLE 7.1

Algebraic algorithms for N-D frequency estimation

Algorithms	Data Smoothing	Obtain $\boldsymbol{U}_s$	Estimate the transformation matrix $\boldsymbol{T}$	Estimate $\{e^{j\omega_{f,n}}\}_{f=1}^{F}$
2-D MEMP [21]	2-D smoothing, backward smoothing	SVD	matrix pencil of two large matrices	eigenvalue-based
MDF [24]	N-D smoothing, backward smoothing	SVD	matrix pencil of two large matrices	eigenvector-based
IMDF [38]	N-D smoothing, backward smoothing, and rotation invariance	SVD	matrix pencil along the first dimension	eigenvector-based
2-D ESPRIT [23]	N-D smoothing, backward smoothing	SVD	joint diagonalization using random weighting factors	eigenvalue-based
Unitary ESPRIT [13]	N-D smoothing, backward smoothing, and rotation invariance	SVD	Simultaneously Schur decomposition	eigenvalue-based
Optimized IMDF [32]	N-D smoothing, backward smoothing, and rotation invariance	SVD	matrix pencils using optimal weighting factors	eigenvector-based
2-D Channel Estimation [33]	2-D smoothing, backward smoothing, and rotation invariance	SVD	joint diagonalization using optimal weighting factors	eigenvector-based
Adaptive 2-D DOA Tracking [69]	N/A	subspace tracking	joint diagonalization using adaptive weighting factors	eigenvector-based

the N-D frequencies by dividing the elements in the structural matrix (i.e. (2.24) and (3.22)). The quotients are often averaged to improve the estimation based on the "circular mean" method. In order to estimate the structural matrix, the eigenvector matrix is required, therefore the approach are named eigenvector-based frequency estimation.

For the eigenvalue-based algorithms, the N-D frequencies are either estimated from the eigenvalues of several matrices using joint diagonalization (e.g. simultaneous Schur decomposition), or obtained separately in individual dimensions and then paired by an extra step. Since the joint diagonalization and eigenvalue pairing step are often computational demanding, eigenvector-based algorithms, which consume less computation resources, are attractive.

The key contributions of this dissertation include

1. The IMDF algorithm offer the most relaxed statistical identifiability bound to date. However, it may not be operational when there are identical or very close

frequencies in some dimensions.

2. The eigenvector-based algorithm using random weighting factors are then proposed to estimate very close or identical frequencies in certain dimensions. The performance of the proposed algorithm is analyzed in detail, based on which an optimization strategy is also proposed to obtain optimal weighting factors.
3. The proposed algorithms are also applied to time-varying channel estimation in OFDM communication systems and 2-D DOA tracking in radar signal processing.
4. The identifiability and performance of the proposed algorithms are studied analytically. All the theoretic analysis is confirmed by computer simulations. Furthermore, the simulation results show that the proposed algorithms can offer superior performance over existing algorithms for the N-D frequency estimation problem.

REFERENCES

[1] B. Ottersten and T. Kailath, "Direction-of-Arrival estimation for wideband signals using the ESPRIT algorithm," *IEEE Trans. Acoustics, Speech, and Signal Processing*, vol. 38, no. 2, pp. 317–327, Feb. 1990.

[2] M. Haardt and J. A. Nossek, "Unitary ESPRIT: How to obtain increased estimation accuracy with a reduced computational burden," *IEEE Trans. Signal Processing*, vol. 43, no. 5, pp. 1232–1242, May 1995.

[3] R. Roy and T. Kailath, "ESPRIT – estimation of signal parameters via rotational invariance techniques," *IEEE Trans. Acoustics, Speech, and Signal Processing*, vol. 37, no. 7, pp. 984–995, July 1989.

[4] A. J. van der Veen, P. B. Ober, and E. D. Deprettere, "Azimuth and elevation computation in high resolution DOA estimation," *IIEEE Trans. Signal Processing*, vol. 40, no. 3, pp. 1828–1832, July 1992.

[5] M. D. Zoltowski, M. Haardt, and C. P. Mathews, "Closed-form 2-D angle estimation with rectangular arrays in element space or beamspace via unitary ESPRIT," *IEEE Trans. Signal Processing*, vol. 44, no. 2, pp. 316–328, Feb. 1996.

[6] A. L. Swindlehurst and T. Kailath, "Azimuth/elevation direction finding using regular array geometries," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 29, no. 3, pp. 145–156, Jan. 1993.

[7] A.N. Lemma, A.-J. van der Veen, and E.F. Deprettere, "Joint angle-frequency estimation using multi-resolution ESPRIT," in *Proc. the 1998 IEEE International Conference on Acoustics, Speech, and Signal Processing*, May 1998.

[8] X. Liu, N. D. Sidiropoulos, and A. Swami, "Joint hop timing and frequency estimation for collision resolution in frequency-hopped networks," *IEEE Trans. Wireless Communications*, vol. 4, no. 6, pp. 3063–3074, Nov. 2005.

[9] A.-J. V. D. Veen, M. C. Vanderveen, and A. Paulraj, "Joint angle and delay estimation using shift-invariance techniques," *IEEE Trans. Signal Processing*, vol. 46, no. 2, pp. 405–418, Feb. 1998.

[10] B. H. Fleury, M. Tschudin, R. Heddergott, D. Dahlhaus, and K. I. Pedersen, "Channel parameter estimation in mobile radio environment using the SAGE algorithm," *IEEE Journal on Selected Areas in Comm.*, vol. 17, no. 3, pp. 434–450, Mar. 1999.

[11] K. Kalliola, H. Laitinen, P. Vainikainen, M. Toeltsch, J. Laurila, and E. Bonek, "3-D double directional radio chnnel characterization for urban macrocellular applications," *IEEE Trans. Antennas Propag.*, vol. 51, no. 11, pp. 3122–3133, Nov. 2003.

[12] M. Steinbauer, A. F. Molisch, and E. Bonek, "The double-directional radio channel," *IEEE Antennas Propag. Mag.*, vol. 43, no. 4, pp. 51–63, Aug. 2003.

[13] M. Haardt and J. A. Nossek, "Simultaneous Schur decomposition of several non-symmetric matrices to achieve automatic pairing in multidimensional harmonic retrieval problems," *IEEE Trans. Signal Processing*, vol. 46, no. 1, pp. 161–169, Jan. 1998.

[14] M. Pesavento, C. F. Mecklenbräuker, and J. F. Böhme, "Multidimensional rank reduction estimator for parametric MIMO channel models," *EURASIP Journal on Applied Signal Processing*, vol. 2004, pp. 1354–1363, Sept. 2004.

[15] A. B. Gershman and N. D. Sidiropoulos, Eds., *Space-Time Processing for MIMO Communications*, chapter Multidimensional harmonic retrieval with applications in MIMO wireless channel sounding, pp. 41–75, Wiley Press, 2005.

[16] Wei-Ge Chen, G. B. Giannakis, and N. Nandhakumar, "A harmonic retrieval framework for discontinuous motion estimation," *IEEE Trans. Image Processing*, vol. 7, no. 9, pp. 1241–1257, Sept. 1998.

[17] A. Box and L. Lerner, "Two-dimensional NMR spectroscopy," *Amer. Assoc. for the Advancement of Science*, vol. 232, no. 1, pp. 960–967, May 1986.

[18] G. S. Armstrong, V. A. Mandelshtam, A. J. Shaka, and B. Bendiak, "Rapid high-resolution four-dimension NMR spectroscopy using the filter diagonalization method and its advantages for detailed structural elusidation of oligosaccharides," *Jounal of Magnetic Resonance*, vol. 173, pp. 160–168, 2005.

[19] P. Stoica and A. Nehorai, "Music, maximum likelihood, and cramer-rao bound," *IEEE Trans. Acoustics, Speech, and Signal Processing*, vol. 37, no. 5, pp. 720–741, May 1989.

[20] Y. Li, J. Razavilar, and K. J. R. Liu, "A high-resolution technique for multi-dimensional NMR spectroscopy," *IEEE Trans. Biomedical Engineering*, vol. 45, no. 1, pp. 78–86, Jan. 1998.

[21] Y. Hua, "Estimating two-dimensional frequencies by matrix enhancement and matrix pencil," *IEEE Trans. Signal Processing*, vol. 40, no. 9, pp. 2267–2280, Sept. 1992.

[22] Y. Hua and K. A. Meraim, "Techniques of eigenvalues estimation and association," *Digital Signal Processing*, vol. 7, no. 4, pp. 253–259, Oct. 1997.

[23] S. Rouquette and M. Najim, "Estimation of frequencies and damping factors by two-dimensional ESPRIT type methods," *IEEE Trans. Signal Processing*, vol. 49, pp. 237–245, Jan. 2001.

[24] X. Liu and N. D. Sidiropoulos, "Almost sure identifiability of multidimensional constant modulus harmonic retrieval," *IEEE Trans. Signal Processing*, vol. 50, no. 9, pp. 2366–2368, Sept. 2002.

[25] J. Li, P. Stoica, and D. Zheng, "An efficient algorithm for two-dimensional frequency estimation," *Multidimensional Systems and Signal Processing*, vol. 7, no. 2, pp. 151–178, Apr. 1996.

[26] H. Yang and Y. Hua, "On rank of block Hankel matrix for 2-D frequency detection and estimation," *IEEE Trans. Signal Processing*, vol. 44, no. 4, pp. 1046–1048, Apr. 1996.

[27] N. D. Sidiropoulos, "Generalizing carathéodory's uniqueness of harmonic parameterization to n dimensions," *IEEE Trans. Information Theory*, vol. 47, no. 5, pp. 1687–1690, May 2001.

[28] F. Li, H. Liu, and R. J. Vaccaro, "Performance analysis for DOA estimation algorithms: unification, simplification, and observations," *IEEE Trans. Aerospace and Electronic Systems*, vol. 29, no. 4, pp. 1170–1184, Oct. 1993.

[29] Y. Hua, F. A. Baqai, Y. Zhu, and D. J. Heilbronn, "Imaging of point scatterers from step-frequency ISAR data," *IEEE Trans. Aerospace And Electroic Systems*, vol. 29, no. 1, pp. 195–205, Jan. 1993.

[30] J. Liu and X. Liu, "Eigenvector-based N-D frequency estimation from sample covariance matrix," *IEEE Signal Processing Letter*, vol. 14, no. 3, 2007.

[31] J. Liu and X. Liu, "Optimizing eigenvector based frequency estimation in the presence of identical frequencies in multiple dimensions," in *Proc. the 7th IEEE Workshop on Signal Processing Advances in Wireless Communications (SPAWC 2006), Cannes, France*, July 2006.

[32] J. Liu and X. Liu, "Multidimensional frequency estimation with finite snapshots in the presence of identical frequencies," *IEEE Trans. Signal Processing*, Apr. 2007, accepted.

[33] J. Liu and X. Liu, "Time-varying channel identification and prediction in OFDM systems using 2-D frequency estimation," in *Proc. Military Communications Conference, Monterey, CA*, Oct. 2006, pp. 1635–1639.

[34] J. Brewer, "Kronecker products and matrix calculus in system theory," *IEEE Trans. Circuits and Systems*, vol. 25, no. 9, pp. 772–781, Sept. 1978.

[35] T. Jiang and N. D. Sidiropoulos, "Almost-sure identifiability of multidimensional harmonic retrieval," *IEEE Trans. Signal Processing*, vol. 49, no. 9, pp. 49–59, Sept. 2001.

[36] K. V. Mardia and P. Jupp, Eds., *Directional Statistics, 2nd edition*, Wiley, 2000.

[37] G. H. Golub and C. F. Van Loan, *Marix Computation (3rd Ed.)*, John Hopkins University Press, 1996.

[38] J. Liu and X. Liu, "An eigenvector-based approach for multidimensional frequency estimation with improved identificability," *IEEE Trans. Signal Processing*, vol. 54, no. 12, Dec. 2006.

[39] B. Friedlander and A.Weiss, "On the second-order statistics of the eigenvectors of sample covariance matrices," *IEEE Trans. Signal Processing*, vol. 46, no. 11, pp. 3136–3139, Nov. 1998.

[40] G. W. Stewart, "Error and perturbation bounds for subspaces associated with certain eigenvalue problems," *SIAM Review*, vol. 15, no. 4, pp. 727–764, Oct. 1973.

[41] G. W. Stewart, "Stochastic perturbation theory," *SIAM Review*, vol. 32, no. 4, pp. 579–610, Dec. 1990.

[42] R. J. Vaccaro, "A second-order perturbation expansion for the SVD," *SIAM Journal of Matrix Analysis and Applications*, vol. 15, no. 2, pp. 661–671, Apr. 1994.

[43] Z. Xu, "Perturbation analysis for subspace decomposition with applications in subspace-based algorithms," *IEEE Trans. Signal Processing*, vol. 50, no. 11, pp. 2820–2930, Nov. 2002.

[44] J. H. Wilkinson, *The Algebraic Eigenvalue Problem*, Oxford Univ. Press: New York, 1965.

[45] B. D. Rao and K. V. S. Hari, "Performance analysis of ESPRIT and TAM in determining the direction of arrival of plane waves in noise," *IEEE Trans. Signal Processing*, vol. 37, no. 12, pp. 1990–1995, Dec. 1989.

[46] Y. Hua and T. K. Sarkar, "Matrix pencil method for estimting parameters of exponentially damped/undamped sinusoids in noise," *IEEE Trans. Acoustics, Speech, and Signal Processing*, vol. 36, no. 5, pp. 814–824, May 1990.

[47] A. N. Lemma, A.-J. van der Veen, and E. F. Deprettere, "Analysis of joint angle-frequency estimation using ESPRIT," *IEEE Trans. Signal Processing*, vol. 51, no. 5, pp. 1264–1283, May 2003.

[48] A. V. Knyazev and M. E. Argentati, "Principal angles between subspaces in an a-based scalar product: algorithms and perturbation estimates," *SIAM Journal on Scientific Computing*, vol. 23, no. 6, pp. 2008–2040, June 2001.

[49] R. W. Freund and F. Jarre, "Solving the sum-of-ratios problem by an interior point method," *Journal of Global Optimization*, vol. 19, no. 1, pp. 83–102, Jan. 2001.

[50] R. K. Brayton, S. W. Director, G. D. Hachtel, and L. Vidigal, "A new algorithm for statistical circuit design based on quasi-newton methods and function splitting," *IEEE Trans. Circuits and Systems*, vol. 26, no. 9, pp. 784–794, Sept. 1979.

[51] J. Nocedal and S. J. Wright, *Numerical Optimization*, New York: Springer-Verlag Press, 1999.

[52] S. Browne, J. Dongarra, N. Garner, and P. Mucci, "A portable programming interface for performance evaluation on modern processors," *The International Journal of High Performance Computing Applications*, vol. 14, no. 3, pp. 189–204, 2000.

[53] M. R. Souryal and R. L. Pickholtz, "Adaptive modulation with imperfect channel information in ofdm," in *Proc. Int. Conf. Comm.*, June 2001, pp. 1861–1865.

[54] Y. Li, L. J. Cimini, and N. R. Sollenberger, "Robust channel estimation for OFDM systems with rapid dispersive fading channels," *IEEE Trans. Commun.*, vol. 46, no. 7, pp. 902–915, July 1998.

[55] B. Yang, K. B. Letaief, R. S. Cheng, and Z. Cao, "Channel estimation for OFDM transmission in multipath fading channels based on parametric channel modeling," *IEEE Trans. Commun.*, vol. 49, no. 3, pp. 467–479, Mar. 2001.

[56] O. Simeone, Y. Bar-Ness, and U. Spagnolini, "Pilot-based channel estimation for ofdm systems by tracking the delay-subspace," *IEEE Trans wireless communication*, vol. 3, no. 1, pp. 315–325, Jan. 2004.

[57] L. Dong, G. Xu, and H. Ling, "Prediction of fast fading mobile radio channels in wideband communication systems," in *Proc. Globecom*, Nov. 2001, pp. 3287–3297.

[58] I. C. Wong and B. L. Evans, "Joint channel estimation and prediction for OFDM systems," in *Proc. IEEE Globecom*, Nov. 2005, pp. 2255–2259.

[59] W. C. Jakes, *Microwave Mobile Communications*, Wiley Press, 1974.

[60] G. S. Armstrong, V. A. Mandelshtam, A. J. Shaka, and B. Bendiak, "Detection of the number of coherent signals by the MDL principle," *IEEE Trans. Acoustics, Speech and Signal processing*, vol. 37, no. 8, pp. 1190 1196, Aug. 1989.

[61] Y. R. Zheng and C. Xiao, "Simulation models with correct statistical properties for rayleigh fading channels," *IEEE Trans. Commun.*, vol. 51, no. 6, pp. 920–928, June 2003.

[62] IEEE Standard 802.16e, "Air interface for fixed and mobile broadband wireless access systems," *IEEE Trans. Signal Processing*, 2004.

[63] Y. Li, "Pilot-symbol-aided channel estimtion for ofdm in wireless systems," *IEEE Trans. Veh. Tech.*, vol. 49, no. 4, pp. 1207 1215, July 2000.

[64] J. Xin and A. Sano, "Efficient subspace-based algorithm for adaptive bearing estimation and tracking," *IEEE Trans. Signal Processing*, vol. 53, no. 12, pp. 4485–4505, Dec. 2005.

[65] M. D. Zoltowski, M. Haardt, and C. P. Mathews, "Close-form 2-D angle estimation with rectangular arrays in element space or beamspace via Unitary ESPRIT," *IEEE Trans. Signal Processing*, vol. 44, no. 2, pp. 316–328, Feb. 1996.

[66] P. Strobach, "Bi-iteration multiple invariance subspace tracking and adaptive esprit," *IEEE Trans. Signal Processing*, vol. 48, no. 2, pp. 442 456, Feb. 2000.

[67] P. Strobach, "Low-rank adaptive filter," *IEEE Trans. Signal Processing*, vol. 44, no. 12, pp. 2932–2947, Dec. 1996.

[68] S. Kravitz, "Packing cylinders into cylindrical containers," *Mathematics Magazine*, vol. 40, no. 2, pp. 65 71, Mar. 1967.

[69] J. Liu and X. Liu, "Joint 2-D DOA tracking for multiple moving targets using adaptive frequency estimaiton," in *International Conf. on Acoustics, Speech, and Signal Processing*, Apr. 2007, vol. 2, pp. 1113 1116.

Appendix

Appendix A. Proof of Lemma 2 in Chapter 2

Lemma 1 is a special case of Lemma 2 when $N = 2$. One can only prove Lemma 2. Given (1.11), one can prove (2.30) by mathematical induction. Define

$$\begin{aligned} \boldsymbol{G} &:= \boldsymbol{A}_1^{(K_1)} \odot \boldsymbol{A}_2^{(K_2)} \odot \cdots \odot \boldsymbol{A}_N^{(K_N)} \\ \boldsymbol{H}_n &:= \boldsymbol{A}_n^{(L_n)} \odot \cdots \odot \boldsymbol{A}_{N-1}^{(L_{N-1})} \odot \boldsymbol{A}_N^{(L_N)} \end{aligned}$$

then (2.30) can be written as $\boldsymbol{X}_{\mathcal{S}} = \boldsymbol{G}\boldsymbol{D}(\boldsymbol{c})\boldsymbol{H}_1^T$. one can first prove that the first L_N columns of $\boldsymbol{X}_{\mathcal{S}}$ are $\boldsymbol{G}\boldsymbol{D}(\boldsymbol{c})\boldsymbol{H}_N^T$. Using Property p2), a typical column of $\boldsymbol{X}_{\mathcal{S}}$ can be written as

$$\begin{aligned} \boldsymbol{J}_{l_1,l_2,\cdots,l_N}\boldsymbol{x} &= \left(\boldsymbol{J}_{l_1}^{K_1} \otimes \boldsymbol{J}_{l_2}^{K_2} \otimes \cdots \otimes \boldsymbol{J}_{l_N}^{K_N}\right)\left(\boldsymbol{A}_1 \odot \boldsymbol{A}_2 \odot \cdots \odot \boldsymbol{A}_N\right)\boldsymbol{c} \\ &= \left(\boldsymbol{J}_{l_1}^{K_1}\boldsymbol{A}_1 \odot \boldsymbol{J}_{l_2}^{K_2}\boldsymbol{A}_2 \odot \cdots \odot \boldsymbol{J}_{l_N}^{K_N}\boldsymbol{A}_N\right)\boldsymbol{c} \\ &= \boldsymbol{G}\boldsymbol{D}\left(\boldsymbol{\omega}_1^{l_1-1}\right)\boldsymbol{D}\left(\boldsymbol{\omega}_2^{l_2-1}\right)\cdots\boldsymbol{D}\left(\boldsymbol{\omega}_N^{l_N-1}\right)\boldsymbol{c} \end{aligned}$$

where $\boldsymbol{\omega}_n^l := \left[e^{jl\omega_{1,n}}, \cdots, e^{jl\omega_{F,n}}\right]^T$, for $n = 1, \cdots, N$. Therefore the first L_N columns of $\mathcal{S}(\boldsymbol{x})$ are

$$\begin{aligned} \left[\boldsymbol{J}_{1,1,\cdots,1}\boldsymbol{x} \quad \boldsymbol{J}_{1,1,\cdots,2}\boldsymbol{x} \quad \cdots \quad \boldsymbol{J}_{1,1,\cdots,L_N}\boldsymbol{x}\right] &= \boldsymbol{G}\left[\boldsymbol{c} \quad \boldsymbol{D}(\boldsymbol{\omega}_N^1)\boldsymbol{c} \quad \cdots \quad \boldsymbol{D}(\boldsymbol{\omega}_N^{L_N-1})\boldsymbol{c}\right] \\ &= \boldsymbol{G}\boldsymbol{D}(\boldsymbol{c})\left[\boldsymbol{\omega}_N^0 \quad \boldsymbol{\omega}_N^1 \quad \cdots \quad \boldsymbol{\omega}_N^{L_N-1}\right] \\ &= \boldsymbol{G}\boldsymbol{D}(\boldsymbol{c})\boldsymbol{H}_N^T. \end{aligned}$$

Assume the first $L_n \cdots L_N$ columns of $\boldsymbol{X}_{\mathcal{S}}$ are $\boldsymbol{G}\boldsymbol{D}(\boldsymbol{c})\boldsymbol{H}_n$, then the first $L_{n-1} \cdots L_N$ columns of $\boldsymbol{X}_{\mathcal{S}}$ are

$$\boldsymbol{G}\boldsymbol{D}(\boldsymbol{c})\begin{bmatrix} \boldsymbol{H}_n\boldsymbol{I}_F \\ \boldsymbol{H}_n\boldsymbol{D}\left(\boldsymbol{\omega}_{n-1}^1\right) \\ \vdots \\ \boldsymbol{H}_n\boldsymbol{D}\left(\boldsymbol{\omega}_{n-1}^{L_{n-1}-1}\right) \end{bmatrix}^T = \boldsymbol{G}\boldsymbol{D}(\boldsymbol{c})\left(\boldsymbol{A}_{n-1}^{(L_{n-1})} \odot \boldsymbol{H}_n\right)^T = \boldsymbol{G}\boldsymbol{D}(\boldsymbol{c})\boldsymbol{H}_{n-1}^T.$$

According to the principle of mathematical induction, Lemma 2 follows.

Appendix B. Proof of Lemma 3 in Chapter 2

Proof: Define $K := \prod_{n=1}^{N} K_n$. One can prove that the determinant of $\widetilde{\boldsymbol{G}}$ is nonzero when $2TK = F$. Since when $2TK < F$ or $2TK > F$, the proof still holds for corresponding column-reduced or row-reduced square sub-matrix.

Suppose $2TK = F$, $|\widetilde{\boldsymbol{G}}|$ is a polynomial of $(T+N)F$ variables $\{e^{j\omega_{f,n}}, c_{f,t}\}, n = 1, \cdots, N,\ f = 1, \cdots, F,\ t = 1 \cdots, T$, hence is analytic in $\mathbb{C}^{(T+N)F}$. According to Lemma 2 in [35], if one can find one point in $\mathbb{C}^{(T+N)F}$ such that $|\widetilde{\boldsymbol{G}}| \neq 0$, $\widetilde{\boldsymbol{G}}$ is full column rank almost surely. If one chooses the $(T+N)F$ variables such that

$$\begin{aligned} e^{j\omega_{f,n}} &= e^{j\alpha_f K_{b,n}}, \ \ 1 \leq n \leq N, \ \ \alpha_f = 2\pi \frac{f-1}{F}, \\ c_{f,t} &= c e^{j\alpha_f (tK + \beta)}, \\ \beta &= \frac{-1}{2}\left((2T+1)K + \sum_{n=1}^{N} (M_n - 1) K_{b,n} \right). \end{aligned}$$

where $c > 0$ and

$$K_{b,n} := \begin{cases} \prod_{p=n+1}^{N} K_p, & 1 \leq n \leq N-1 \\ 1 & n = N \end{cases} \tag{7.1}$$

The f-th column of $\boldsymbol{Q}$ is

$$c\left[e^{j\alpha_f (K+\beta)}, e^{j\alpha_f (2K+\beta)}, \cdots, e^{j\alpha_f (2TK+\beta)}\right]^T,$$

thus $\boldsymbol{Q}$ is a Vandermonde matrix. Now $\widetilde{\boldsymbol{G}}$ can be written as a Vandermonde matrix with generators equally-spaced on the unit circle, so $|\widetilde{\boldsymbol{G}}| \neq 0$. Therefore $\widetilde{\boldsymbol{G}}$ is almost sure full rank.

Appendix C. CRB of Multidimensional Frequency Estimation

One can derive the CRB of N-D frequency estimation from (1.3) using a method similar to that of [19]. The result is given as follows:

$$\mathrm{var}_{\mathrm{CRB}}(\widehat{\omega}_{f,n}) = \frac{\sigma^2}{2} b_{(n-1)F+f}, \tag{7.2}$$

where $b_{(n-1)F+f}$ is the $(nF - F + f)$-th element of following vector

$$\boldsymbol{b} = \mathrm{Diag}\left\{ \left[\sum_{t=1}^{T} \mathcal{R}\Big(\boldsymbol{C}_e^H(t) \boldsymbol{D}^H (\boldsymbol{I}_M - \boldsymbol{A}(\boldsymbol{A}^H \boldsymbol{A})^{-1} \boldsymbol{A}^H) \boldsymbol{D} \boldsymbol{C}_e(t) \Big) \right]^{-1} \right\}$$

where $\boldsymbol{C}_e(t) := \boldsymbol{I}_N \otimes \boldsymbol{D}(\boldsymbol{c}(t))$, and

$$\begin{aligned}
\boldsymbol{D} &= \left[\boldsymbol{d}_{1,1}\ \boldsymbol{d}_{2,1}\ \cdots\ \boldsymbol{d}_{F,1}\ \boldsymbol{d}_{1,2}\ \cdots\ \boldsymbol{d}_{F,N}\right] : M \times NF, \\
\boldsymbol{d}_{f,n} &= \left(\boldsymbol{\theta}_{f,1} \otimes \cdots \otimes \boldsymbol{\theta}_{f,n-1}\right) \otimes \boldsymbol{\vartheta}_{f,n} \otimes \left(\boldsymbol{\theta}_{f,n+1} \otimes \cdots \otimes \boldsymbol{\theta}_{f,N}\right) \quad : M \times 1, \\
\boldsymbol{\theta}_{f,n} &= \left[1\ e^{j\omega_{f,n}}\ \cdots\ e^{j(M_n-1)\omega_{f,n}}\right]^T : M_n \times 1, \\
\boldsymbol{\vartheta}_{f,n} &= \left[0\ je^{j\omega_{f,n}}\ \cdots\ j(M_n-1)e^{j(M_n-1)\omega_{f,n}}\right]^T : M_n \times 1.
\end{aligned}$$

In all simulations, the RMSEs of various algorithms are compared to the square root of the corresponding CRB on variance.

Appendix D. Notations and Abbreviations

Notations

$\boldsymbol{A}^*$	conjugate of $\boldsymbol{A}$
$\boldsymbol{A}^T$	transpose of $\boldsymbol{A}$
$\boldsymbol{A}^H$	Hermitian transpose of $\boldsymbol{A}$
$\boldsymbol{A}^\dagger$	pseudo-inverse of $\boldsymbol{A}$
$\otimes$	Kronecker product
$\odot$	Khatri-Rao (column-wise Kronecker) product
$\boldsymbol{I}_p$	a $p \times p$ identity matrix
$\boldsymbol{0}_{M \times N}$	an $M \times N$ zero matrix
$\boldsymbol{D}(\boldsymbol{a})$	a diagonal matrix with $\boldsymbol{a}$ as its diagonal
$\boldsymbol{A}^{(m)}$	a sub-matrix of $\boldsymbol{A}$ formed by its first m rows
$\lVert\boldsymbol{A}\rVert$	the Frobenius norm of $\boldsymbol{A}$
$\lVert\boldsymbol{A}\rVert_2$	the l_2 norm of $\boldsymbol{A}$ (the largest eigenvalue)
$[\boldsymbol{A}]_{f,n}$ or $a_{f,n}$	the (f, n)-th element of $\boldsymbol{A}$
$E(\cdot)$	expectation
$\mathcal{R}(\cdot)$	real part of complex variables
$\mathcal{I}(\cdot)$	imaginary part of complex variables
Upper bold face letters	matrices
Lower bold face letters	column vectors

Abbreviations

AWGN	Additive White Gaussian noise
CSI	Channel State Information
CRB	Cramér Rao lower Bound
DOA	Direction-of-Arrival
DFT	Discrete Fourier Transform
ESPRIT	Estimation of Signal Parameters via Rotational Invariance Techniques
EVD	EigenValue Decomposition
FB	Forward Backward
FFT	Fast Fourier Transform
ID	IDentifiability
IMDF	Improved Multi-Dimensional Folding
JADE	Joint Approximate Diagonalization of Eigenmatrices
LHS	Left Hand Side
LS	Least Square
MEMP	Matrix Enhanced Matric Pencil
MMSE	Minimal Mean Square Error
MSE	Mean Square Error
MUSIC	MUltiple SIgnal Classification or Characterization
NMSE	Normalized Mean Square Error
OFDM	Orthogonal Frequency Division Multiplexing
PAPI	Performance APplication Interface
PCA	Principal Component Analysis
QAM	Quadrature amplitude modulation
RARE	RAnk Reduction Estimator
RMSE	Root Mean Square Error
RHS	Right Hand Side
SVD	Singular Value Decomposition
SNR	Signal-Noise-Ratio
STD	STandard Deviation
ULA	Uniform Linear Array
URA	Uniform Rectangular Array

CURRICULUM VITAE

NAME: Jun Liu

ADDRESS: Department of Electrical and Computer Engineering
University of Louisville
Louisville, KY 40292

EDUCATION: M.S., Electrical Engineering,
Huazhong University of Science and Technologies
1999-2001

B.S., Electrical Engineering,
Huazhong University of Science and Technologies
1995-1999

PREVIOUS: Software Engineer,
Lucent Technologies, Bell Labs, Beijing, China
2001-2004

RESEARCH: Communications and Signal Processing

TEACHING: ECE 320 Network Analysis GTA
ECE 653 Digital Communication GTA
ECE 530 Communications and Modulation GTA

SOCIETIES: Student member of IEEE

AWARDS: The 25th MILCOM Student Travel Grants, Oct. 2000

PUBLICATIONS:

JOURNAL PUBLICATIONS:

1. J. Liu and X. Liu, "An eigenvector-based approach for multidimensional frequency estimation with improved identifiability," *IEEE Trans. Signal Processing*, vol. 54, no. 12, pp. 4543–4556, Dec. 2006.
2. J. Liu and X. Liu, "Eigenvector-based N-D frequencies from sample covariance matrix," *IEEE Trans. Signal Processing Letters*, vol. 14, no. 3, pp. 209–212, Mar. 2007.
3. J. Liu, X. Liu and X. Ma, "Multidimensional frequency estimation from finite snapshots in the presence of identical frequencies," *IEEE Trans. Signal Processing*, Nov. 2007.
4. J. Liu, X. Liu and X. Ma, "A first-order perturbation analysis of the singular vectors in singular value decomposition," *IEEE Trans. Signal Processing*, revised, June 2007.
5. J. Liu, X. Liu and X. Ma, "Joint diagonalization via weighted generalized eigenvalue decomposition," *IEEE Trans. Signal Processing*, revised, July 2007.

CONFERENCE PUBLICATIONS

6. J. Liu and X. Liu, "An algebraic approach for two-dimensional frequency estimation with improved identifiability," in *Proc. IEEE Workshop on Signal Processing Advances in Wireless Communications*, pp. 655–659, New York City, NY, Jun. 2005.
7. M.-T. Sun, X. Ma, J. Liu, and X. Liu, "A greedy smart path pruning strategy for geographical routing in wireless networks," in *Proc. Military Communications Conference*, pp. 1326–1332, Atlantic City, NJ, Oct. 2005.
8. J. Liu and X. Liu, "Statistical identifiability of multidimensional frequency estimation with finite snapshots," in *Proc. the 4th IEEE Workshop on Sensor Array and Multi-channel Processing*, pp. 491–495, Waltham, MA, Jul. 12-14, 2006.
9. J. Liu and X. Liu, "Optimizing eigenvector based frequency estimation in the presence of identical frequencies in multiple dimensions," in *Proc. the 7th IEEE Workshop on Signal Processing Advances in Wireless Communications*, Cannes, France, Jul. 2-5, 2006.

10. J. Liu and X. Liu, "Time-varying channel identification and predication in OFDM systems using 2-D frequency estimation," in *Proc. Military Communications Conference*, pp. 1635 1639, Oct. 23-25, 2006.

11. J. Liu and X. Liu, "Joint 2-D DOA tracking for multiple moving targets using adaptive frequency estimation," *Proc. the 32nd International Conference on Acoustics, Speech, and Signal Processing*, vol. 2, pp. 1113 1116, Honolulu, HI, Apr. 15-20, 2007.

12. J. Liu, X. Liu, and X. Ma, "First-order perturbation analysis of the singular vectors in singular value decomposition," *IEEE Statistical Signal Processing Workshop*, Madison, WI, Aug. 26-29, 2007.