



M.Sc. Thesis

Rank Detection Based on Generalized Eigenvalue Threshold in Arbitrary Noise

Bingxiang Zhong B.Sc.

Abstract

Rank detection is crucial in array processing applications, as many algorithms rely on accurately estimating the rank of the data matrix to ensure optimal performance. Under Gaussian white noise, rank can be detected through eigenvalue analysis. However, in arbitrary noise, prewhitening the data matrix with the noise covariance matrix is necessary, and rank detection is achieved by examining the generalized eigenvalues. Existing methods often assume the noise covariance structure or require a large number of noise samples. This thesis focuses on addressing the rank detection problem in scenarios with limited noise samples and arbitrary noise environments.

Firstly, we investigate the largest generalized eigenvalue threshold for the prewhitened data sample covariance matrix according to the random matrix theory. We develop a rank detection algorithm based on the threshold via a sequential test, and provide the performance analysis. A series of simulations demonstrate its superiority over conventional methods such as Minimum Description Length (MDL) and Akaike's Information Criterion (AIC).

Secondly, since the Short-time Fourier Transform (STFT) is commonly used for non-stationary signal analysis, we extend our rank detection method to the STFT domain. The correlations introduced by the STFT have a significant impact on the distribution of the noise. Therefore, we develop a technique to remove correlations among time-frequency bins based on exact expressions of these correlations. After successfully eliminating these correlations, our proposed rank detection method achieves enhanced reliability and performance in the STFT domain.

Lastly, we evaluate the effectiveness of our rank detection method in speech enhancement applications. Simulations confirm that utilizing the estimated rank improves speech quality compared to using the known number of sources.

Rank Detection Based on Generalized Eigenvalue Threshold in Arbitrary Noise

THESIS

submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE

in

ELECTRICAL ENGINEERING

by

Bingxiang Zhong B.Sc.
born in Jiangxi, China

This work was performed in:

Signal Processing Systems Group
Department of Microelectronics
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology



Delft University of Technology

Copyright © 2023 Signal Processing Systems Group
All rights reserved.

DELFT UNIVERSITY OF TECHNOLOGY
DEPARTMENT OF
MICROELECTRONICS

The undersigned hereby certify that they have read and recommend to the Faculty of Electrical Engineering, Mathematics and Computer Science for acceptance a thesis entitled “**Rank Detection Based on Generalized Eigenvalue Threshold in Arbitrary Noise**” by **Bingxiang Zhong B.Sc.** in partial fulfillment of the requirements for the degree of **Master of Science**.

Dated: 29 June 2023

Chairman:

prof.dr.ir. A.J. van der Veen

Advisor:

prof.dr.ir. A.J. van der Veen

Committee Members:

dr.ir. R.C. Hendriks

dr.ir. R. Heusdens

Abstract

Rank detection is crucial in array processing applications, as many algorithms rely on accurately estimating the rank of the data matrix to ensure optimal performance. Under Gaussian white noise, rank can be detected through eigenvalue analysis. However, in arbitrary noise, prewhitening the data matrix with the noise covariance matrix is necessary, and rank detection is achieved by examining the generalized eigenvalues. Existing methods often assume the noise covariance structure or require a large number of noise samples. This thesis focuses on addressing the rank detection problem in scenarios with limited noise samples and arbitrary noise environments.

Firstly, we investigate the largest generalized eigenvalue threshold for the prewhitened data sample covariance matrix according to the random matrix theory. We develop a rank detection algorithm based on the threshold via a sequential test, and provide the performance analysis. A series of simulations demonstrate its superiority over conventional methods such as Minimum Description Length (MDL) and Akaike's Information Criterion (AIC).

Secondly, since the Short-time Fourier Transform (STFT) is commonly used for non-stationary signal analysis, we extend our rank detection method to the STFT domain. The correlations introduced by the STFT have a significant impact on the distribution of the noise. Therefore, we develop a technique to remove correlations among time-frequency bins based on exact expressions of these correlations. After successfully eliminating these correlations, our proposed rank detection method achieves enhanced reliability and performance in the STFT domain.

Lastly, we evaluate the effectiveness of our rank detection method in speech enhancement applications. Simulations confirm that utilizing the estimated rank improves speech quality compared to using the known number of sources.

Acknowledgments

It feels like just yesterday when I first arrived in the Netherlands and stepped foot on the TU Delft campus, even though it was about 2 years ago. Time really flies! Over these past 2 years, I have experienced a lot of things, both moments of joy and times when I felt down. Now, I am happy to reach the end of my master's journey. I want to express my gratitude to those who have helped, supported and accompanied me on this journey.

First, I would like to express my heartfelt gratitude to my advisor prof.dr.ir. A.J. van der Veen for his assistance throughout my thesis research. Not only did he guide me to complete the final part of my master's study, but he also taught me valuable life lessons about facing difficulties, managing stress, and thinking critically. His patience and support have kept me going and made me feel confident when dealing with research challenges. His instructions and ideas have always inspired me and made the process of exploring the unknown enjoyable.

Next, I want to extend my thanks to dr.ir. R.C. Hendriks and dr.ir. R. Heusdens for being my thesis committee members. Their feedback and suggestions have been incredibly valuable to my thesis. Their support and encouragement have boosted my confidence throughout this process. I am truly thankful for their expertise and guidance.

Lastly, I want to express my heartfelt appreciation to my friends and family for their unwavering support and unconditional love throughout this journey. I will never be who I am today without them. Thank you all for being by my side!

Bingxiang Zhong B.Sc.
Delft, The Netherlands

Contents

Abstract	v
Acknowledgments	vii
1 Introduction	1
1.1 Research Question	2
1.2 Thesis Outline	2
2 Problem Formulation and Related Work	3
2.1 Signal Model	3
2.2 Tools from Linear Algebra for Rank Detection	4
2.2.1 Eigenvalue Decomposition (EVD)	4
2.2.2 Prewhitening Technique	5
2.2.3 Generalized Eigenvalue Decomposition (GEVD)	5
2.3 Related Work on Rank Detection	6
2.3.1 Rank Detection for Gaussian White Noise	7
2.3.2 Rank Detection for Colored and Correlated Noise	8
2.4 Problem Formulation	9
3 Threshold for Prewhitened Sample Covariance Matrix	11
3.1 Threshold for Gaussian White Noise	11
3.2 Threshold for Prewhitened Noise	12
3.2.1 The distribution of Prewhitened Noise	12
3.2.2 Adjustment of the Threshold β for Prewhitened Noise	12
3.2.3 Threshold for F-type Matrices	13
3.3 Threshold in the Signal-plus-Noise Case	18
3.4 Chapter Conclusion	20
4 Threshold-based Rank Detection Algorithm	21
4.1 Sequential Rank Detection Algorithm	21
4.2 Performance Analysis	22
4.3 Rank Detection Simulations	26
4.3.1 Rank Detection for $N_b = \infty$	26
4.3.2 Rank Detection for $N_b \neq \infty$	28
4.4 Chapter Conclusion	28
5 Threshold-based Rank Detection Method in the STFT Domain	31
5.1 Short-Time Fourier Transform (STFT)	31
5.2 Problems of Threshold Methods in the STFT domain	32
5.3 Correlation in STFT	33
5.3.1 Time correlation	34
5.3.2 Frequency Correlation	35

5.3.3	Time-Frequency Correlation	36
5.4	Removing the Correlations	37
5.4.1	Temporal Prewhitening (TP)	38
5.4.2	Frequential Prewhitening (FP)	38
5.4.3	Time-Frequency Prewhitening (TFP)	39
5.5	Simulations	40
5.5.1	Eigenvalue Distribution Recheck	40
5.5.2	Rank Detection for Stationary Source	42
5.5.3	Rank Tracking for Time-varying Source	44
5.6	Chapter Conclusion	45
6	Threshold-based Rank Detection Method for Speech Enhancement	47
6.1	Multi-microphone Signal Model	47
6.2	Beamforming	49
6.2.1	GEVD-based Beamformers	50
6.2.2	Beamforming After Temporal Prewhitening	51
6.3	Implementation Detail	52
6.4	Simulations	54
6.4.1	Simulation Setup	54
6.4.2	Evaluation Method	56
6.4.3	Rank Detection Accuracy of Threshold Method	57
6.4.4	Speech Enhancement Performance	60
6.5	Chapter Conclusion	61
7	Conclusion and Future Work	63
7.1	Conclusion	63
7.2	Future Work	64
7.2.1	Theoretical Advancements	64
7.2.2	Practical Applications	64
A	Tracy-Widom Distribution in the Real Case	65
B	Temporal Prewhitening Algorithm	67
B.1	General Temporal Prewhitening	67
B.2	Cholesky-Forward-Substitution Algorithm	68
B.2.1	Algorithm for $D = 1$	69
B.2.2	Algorithm for $D > 1$	70

List of Figures

3.1	Empirical CDF of the largest eigenvalue of the SCM of prewhitened noise, compared to the theoretical CDF and the modified CDF. $M = 4, N_a = 200, N_b = 200, \tilde{N}_a = 80$	13
3.2	Empirical CDF of the largest eigenvalue of the SCM of prewhitened noise, compared to the theoretical CDF when $N_b = \infty$	14
3.3	Theoretical and empirical CDF of the largest eigenvalue of the SCM of prewhitened noise.	16
3.4	CDF of the largest eigenvalue of the SCM of prewhitened noise for the correlated noise.	18
3.5	CDF of the $d + 1$ -st eigenvalue of the SCM of prewhitened data matrix with d number of sources, and the Tracy-Widom model with parameters $(M - d, N_a - d, N_b)$	20
4.1	Empirical and theoretical CDF of the d -th eigenvalue with d sources.	24
4.2	Empirical and theoretical probability detection as a function of M , with $N_a = N_b = 8M$	24
4.3	Empirical and theoretical probability detection for three sources when changing the signal strength of the third source p_3 , and fixing the first two signals strengths $p_1 = 12, p_2 = 10$	25
4.4	Probability detection as a function of signal strength for $M = 8, N_a = N_b = 200$	25
4.5	The rank detection performance of different methods for $N_b = \infty$, while varying the rank d, N_a	27
4.6	The rank detection performance of different methods, while varying the rank d, N_a and N_b	29
5.1	Diagram of STFT on a data vector.	32
5.2	Empirical CDF of the largest eigenvalue of the SCM of prewhitened noise after STFT, compared to the CDF of the Tracy-Widom model for different window functions	33
5.3	Time correlation for the same frequency bin ($k = 2$) as a function of time lag τ , considering different window functions, for $L = 256, K = 256, H = 128$	35
5.4	Frequency correlation for the same time frame ($m = 3$) as a function of frequency difference δ , considering different window functions, for $L = 256, K = 256, H = 128$	36
5.5	Sliding window processing diagram.	39
5.6	Empirical CDF of the largest eigenvalue of the SCM of prewhitened noise after TFP and CDF of the Tracy-Widom model for different window functions.	41

5.7	Empirical CDF of the largest eigenvalue of the SCM of prewhitened noise after TFP and CDF of the Tracy-Widom model when including the first frequency bin ($k = 0$).	41
5.8	Empirical CDF of the largest eigenvalue of the SCM of prewhitened noise after TP and CDF of the Tracy-Widom model for different window functions at the same frequency bin ($k = 2$).	42
5.9	The probability of detection as a function of SNR for $d = 1$ in STFT domain at one frequency bin ($k = 2$).	43
5.10	True rank of the generated signals in the time domain.	44
5.11	The probability of detection when the rank is changing over time. . . .	45
5.12	Rank tracking results of SF and MF settings at 30 dB SNR.	46
6.1	Implementation diagram of threshold method in speech enhancement. .	52
6.2	Probability of detection as a function of SNR of microphone self-noise for different window functions.	53
6.3	The room geometry for simulations with $M = 6$ microphones and $d = 2$ sources.	55
6.4	Plots of speech signals and their VAD labels used in the simulations. .	55
6.5	Spectrum plots of target speech signals at the reference microphone, when the noise is not presented.	56
6.6	Rank detection probability using threshold method for (a) one target source and (b) two target sources.	58
6.7	Rank estimation results for one or two target sources at frequency bin 20, input SNR 25 dB.	58
6.8	Detection probability for two target sources at frequency bin 20, input SNR 25 dB, $N_a = 20$, $N_b = 100$	59
6.9	Rank estimation results for two target sources at frequency bin 20, input SNR 25 dB, $N_a = 20$, $N_b = 100$	59
6.10	Speech enhancement performance using the maximum rank and the estimated rank of the threshold method, for one target source.	60
6.11	Speech enhancement performance using the maximum rank and the estimated rank of the threshold method, for two target sources.	61

List of Tables

6.1	Simulation parameters	55
-----	---------------------------------	----

Introduction

Among the recent decades, array processing has become increasingly an indispensable technology with wide-ranging applications in various domains, revolutionizing our daily lives and industrial practices. From enabling seamless wireless communication to enhancing medical imaging, radar systems, industrial quality control, and speech processing, array processing techniques have become integral to modern society.

Within the field of array processing, rank detection, also known as the detection of the number of independent sources, is of utmost importance as it serves as the necessary first step for subsequent array processing applications. For instance, signal processing algorithms based on subspace methods, like the ESPRIT or MUSIC algorithms that are used in radar systems for direction-of-arrival (DoA) estimation, rely on accurate rank detection [1, 2]. Speech enhancement and separation tasks require precise knowledge of the number of sources to ensure optimal speech quality. Similarly, in wireless communications, the identification and mitigation of the interference also need the accurate estimation of the number of independent sources. Therefore, achieving an accurate estimation of the rank plays a vital role in optimizing system performance and overall operational efficiency.

Ideally, the rank can be determined by observing the eigenvalues of the sample covariance matrix of the collected data samples. However, in scenarios with limited observations, low signal-to-noise ratio (SNR), and the presence of noise, observing the eigenvalues becomes challenging, necessitating the use of more advanced methods. Numerous techniques have been developed to accurately detect the rank, with a specific focus on scenarios involving Gaussian white noise [3, 4, 5, 6]. Prominent approaches include the Minimum Description Length (MDL) and Akaike's Information Criterion (AIC), which leverage information theoretic criteria [3]. Additionally, methods such as [7] have established precise thresholds for eigenvalues obtained from noise, enabling the development of robust rank detection algorithms.

However, rank detection becomes more challenging when dealing with colored and correlated noise environments. In such cases, a prewhitening technique is often employed to decorrelate the noise by using the noise covariance matrix, and the estimation of the noise covariance matrix is required in practice. The rank detection then is achieved by analyzing the generalized eigenvalues of the matrix pencil formed by the data and noise covariance matrices. Although methods originally designed for Gaussian white noise can be used, their effectiveness may be compromised when working with a limited number of noise records used for prewhitening. Many existing specialized methods rely on the assumptions of the structure of noise covariance matrix or the availability of a large number of noise samples to ensure reliable performance [8, 9, 10]. Therefore, the development of specialized techniques capable of overcoming these limitations is necessary.

Motivated by these research gaps, this thesis aims to address the challenges of rank

detection in the presence of arbitrary noise. Specifically, we will focus on utilizing the threshold method to determine the rank. This method involves counting the number of generalized eigenvalues that exceed a certain threshold. Additionally, we will extend this method to the domain of Short-time Fourier Transformation (STFT), which is commonly used for analyzing and processing non-stationary signals.

1.1 Research Question

The primary research question that guides this study is as follows:

- *How can we achieve precise rank detection based on the threshold method in the presence of colored or correlated noise, considering the finite number of data samples and noise samples?*

1.2 Thesis Outline

To answer the research question, the thesis consists of seven chapters. In this current chapter, we provide a brief introduction to the rank detection problem and outline the main research questions explored in this thesis. The subsequent chapters are organized as follows:

Chapter 2: This chapter begins by presenting essential mathematical concepts for a comprehensive understanding of the research, including the basic signal model and relevant linear algebra tools. Additionally, a brief literature review will be provided. Finally, we present the problem formulation and outline specific research objectives based on the findings from the literature review.

Chapter 3: In this chapter, we review the eigenvalue threshold for Gaussian white noise and investigate its applicability to prewhitened noise. Furthermore, we present the precise mathematical expression of the threshold for the largest generalized eigenvalue of prewhitened noise. Lastly, we introduce a more accurate threshold for the signal-plus-noise case.

Chapter 4: We propose a sequential test approach for the rank detection algorithm based on the threshold method, followed by an analysis of its performance. We also present the simulation results of the proposed rank detection algorithm and compare them with those obtained using MDL and AIC.

Chapter 5: This chapter extends the rank detection method to the STFT domain. Firstly, we discuss the time and frequency correlations introduced by the STFT, which can impact the performance of the proposed rank detection method. Subsequently, we present techniques to mitigate these correlations, and show simulations of rank detection after removing the correlations, comparing the results with those obtained without removing them.

Chapter 6: In this chapter, we introduce the proposed rank detection method into the context of speech enhancement. Simulations are employed to evaluate the effectiveness of the rank detection method in this application.

Chapter 7: The final chapter concludes our findings and provides recommendations for future research.

Problem Formulation and Related Work

2

In this chapter, we will begin by providing fundamental mathematical concepts, including the signal model and important linear algebra tools, which are crucial for better understanding the subsequent chapters of this thesis. Next, we will present a review of existing literature on rank detection. By analyzing the related work, we gain valuable insights and motivation for our own research. Furthermore, we will provide a more specific problem formulation for the main research question.

2.1 Signal Model

Consider a common signal model in array processing, where a sensor array of M sensors is used to collect N independent observations from d sources. Each observation is denoted as the vector $\mathbf{x}[n]$ of dimension M , for $n = 1, \dots, N$. The observation vector is modelled as

$$\mathbf{x}[n] = \mathbf{H}\mathbf{s}[n] + \mathbf{n}[n], n = 1, \dots, N \quad (2.1)$$

where $\mathbf{n}[n]$ denotes a M -dimensional complex noise vector, $\mathbf{H} = [\mathbf{h}_1, \dots, \mathbf{h}_d]$ is the complex steering matrix of size $M \times d$, $\mathbf{s}[n] = [s_1[n], \dots, s_d[n]]^T$ is a d -dimensional complex source vector. $(\cdot)^T$ denotes the transpose operator.

Few assumptions are made for this signal model. The number of sources d is smaller than the number of sensors M . The column vectors in matrix $\mathbf{H}$ contain the transfer function of the source from source position to sensors. These vectors are assumed to be linearly independent, resulting in $\mathbf{H}$ being a full-column rank matrix. The source is assumed to follow a complex zero-mean Gaussian distribution with covariance matrix $\mathbb{E}[\mathbf{s}\mathbf{s}^H] = \mathbf{R}_S$, where $\mathbb{E}[\cdot]$ represents the expectation operator. The covariance matrix $\mathbf{R}_S$ has a size of $d \times d$ and is a full-rank matrix. We assume that the noise is drawn from a complex Gaussian distribution $\mathcal{CN}(0, \mathbf{R}_N)$, and the covariance matrix $\mathbf{R}_N$ is a full-rank matrix of size $M \times M$. Additionally, the noise is assumed to be independent of the source, i.e., $\mathbb{E}[\mathbf{s}\mathbf{n}^H] = \mathbf{0}$. The covariance matrix of $\mathbf{x}[n]$ is then given by

$$\begin{aligned} \mathbf{R}_X &= \mathbf{H}\mathbf{R}_S\mathbf{H}^H + \mathbf{R}_N \\ &= \mathbf{\Phi} + \mathbf{R}_N, \end{aligned} \quad (2.2)$$

with $(\cdot)^H$ the complex conjugate transpose. Matrix $\mathbf{\Phi}$ is of rank d , given the assumptions defined above.

We can stack all the observations column-wise, resulting in the signal model given by

$$\begin{aligned} \mathbf{X} &= \mathbf{H}\mathbf{S} + \mathbf{N} \\ &= \mathbf{X}_S + \mathbf{N}, \end{aligned} \quad (2.3)$$

where $\mathbf{X} = [\mathbf{x}[1], \dots, \mathbf{x}[N]] \in \mathbb{C}^{M \times N}$ denotes the data matrix, $\mathbf{S} = [\mathbf{s}[1], \dots, \mathbf{s}[N]] \in \mathbb{C}^{d \times N}$ is the source matrix, and $\mathbf{N} = [\mathbf{n}[1], \dots, \mathbf{n}[N]] \in \mathbb{C}^{M \times N}$ is the noise matrix.

In practice, the covariance matrix is often unknown and needs to be estimated. One common approach is to use the sample covariance matrix (SCM), which is obtained by averaging the outer products of observations. The SCM of $\mathbf{X}$ is given by

$$\hat{\mathbf{R}}_X = \frac{1}{N} \mathbf{X} \mathbf{X}^H. \quad (2.4)$$

As N approaches infinity, the sample covariance matrix becomes an increasingly accurate estimate of the true covariance matrix. Estimation of the noise covariance matrix follows a similar procedure, where noise samples are collected during signal-free periods.

2.2 Tools from Linear Algebra for Rank Detection

The rank detection problem is to find the number of sources, denoted by d , based on the noisy observations. The number of observations, the SNR, and the noise distribution can all affect the rank detection performance. Before we delve into more specific methods, the necessary tools from linear algebra are introduced first for the rank detection in this section.

2.2.1 Eigenvalue Decomposition (EVD)

As matrix $\mathbf{R}_X$ is Hermitian, the EVD of the data covariance matrix is given by [11]

$$\mathbf{R}_X = \mathbf{U} \mathbf{\Lambda} \mathbf{U}^H, \quad (2.5)$$

where the matrix $\mathbf{U}$, with dimensions $M \times M$, contains the eigenvectors, while $\mathbf{\Lambda} = \text{diag}\{\lambda_1, \dots, \lambda_M\}$ is a diagonal matrix of size $M \times M$ consisting of the eigenvalues. Since the covariance matrix is Hermitian and positive definite, the eigenvector matrix $\mathbf{U}$ is a unitary matrix, i.e., $\mathbf{U} \mathbf{U}^H = \mathbf{I}_M$, and the eigenvalues are real and positive.

The rank of the data matrix can be determined by examining its eigenvalues. In an ideal scenario without noise, the collected data matrix $\mathbf{X}$ is equal to the matrix $\mathbf{X}_S$. If $d < M$, we can observe d non-zero eigenvalues and $M - d$ zero eigenvalues. Therefore, the rank of the data matrix can be straightforwardly determined by analyzing the eigenvalues.

In the case of Gaussian white noise, the noise covariance matrix is given by $\mathbf{R}_N = \sigma_n^2 \mathbf{I}_M$, where σ_n^2 represents the noise power and $\mathbf{I}_M$ denotes the identity matrix of size $M \times M$. The presence of noise influences the eigenvalues by increasing each eigenvalue by the noise power σ_n^2 . Consequently, if the eigenvalues are sorted in descending order, we will have

$$\lambda_1 \geq \dots \geq \lambda_d \geq \lambda_{d+1} = \dots = \lambda_M = \sigma_n^2. \quad (2.6)$$

Hence, by setting the noise power as the threshold, it becomes convenient to determine the rank of the data matrix. This can be achieved by simply counting the number of eigenvalues that surpass the threshold value.

2.2.2 Prewhitening Technique

In practical scenarios, it is common to encounter non-white noise, where the noise covariance matrix does not exhibit uniform diagonal entries. For example, in the case of uncalibrated sensors, the noise power in every sensor is different, making it challenging to determine an eigenvalue threshold to accurately detect the number of sources. Moreover, when the noise is spatially correlated, the eigenvectors of the signal subspace are affected, rendering the threshold method ineffective. To overcome these challenges, the prewhitening technique is often employed. This technique transforms the noise distribution into a 'white noise' distribution, where the covariance matrix becomes diagonal and equal to the identity matrix. By applying prewhitening, the correlated noise no longer influences the eigenvectors, allowing for a more reliable determination of the number of sources.

The prewhitening technique involves a linear transformation of the data matrix using a whitening matrix. The whitening matrix is typically defined as the inverse of the square root of the noise covariance matrix, denoted as $\mathbf{R}_N^{-1/2}$. The square root of the noise covariance matrix can be computed through the EVD of the noise covariance matrix, Cholesky decomposition, or QR decomposition. For example, with EVD, we can write $\mathbf{R}_N$ as

$$\begin{aligned}\mathbf{R}_N &= \mathbf{U}_N \mathbf{\Lambda}_N \mathbf{U}_N^H \\ &= \mathbf{U}_N \mathbf{\Lambda}_N^{1/2} \mathbf{U}_N^H \mathbf{U}_N \mathbf{\Lambda}_N^{1/2} \mathbf{U}_N^H \\ &= \mathbf{R}_N^{1/2} \mathbf{R}_N^{1/2},\end{aligned}$$

where $\mathbf{U}_N$ are the eigenvectors and $\mathbf{\Lambda}_N$ are the eigenvalues. $\mathbf{R}_N^{1/2}$ is the unique Hermitian square root of $\mathbf{R}_N$.

Let $\tilde{\mathbf{X}}$ denote the prewhitened data matrix, and $\tilde{\mathbf{X}} = \mathbf{R}_N^{-1/2} \mathbf{X}$. The prewhitened data covariance matrix then is given by

$$\begin{aligned}\mathbf{R}_{\tilde{\mathbf{X}}} &= \mathbf{R}_N^{-1/2} \mathbf{\Phi} \mathbf{R}_N^{-1/2} + \mathbf{R}_N^{-1/2} \mathbf{R}_N \mathbf{R}_N^{-1/2} \\ &= \mathbf{R}_N^{-1/2} \mathbf{\Phi} \mathbf{R}_N^{-1/2} + \mathbf{I}_M.\end{aligned}\tag{2.7}$$

This suggests that the EVD of the covariance matrix of the prewhitened data $\tilde{\mathbf{X}}$ can be expressed as follows:

$$\mathbf{R}_{\tilde{\mathbf{X}}} = \tilde{\mathbf{U}}(\tilde{\mathbf{\Lambda}}_S + \mathbf{I}_M)\tilde{\mathbf{U}}^H,\tag{2.8}$$

where if the number of sources is assumed to be $d < M$, the eigenvalues of $\mathbf{R}_{\tilde{\mathbf{X}}}$ will exhibit the relationship:

$$\lambda_1 \geq \dots \geq \lambda_d \geq \lambda_{d+1} = \dots = \lambda_M = 1.\tag{2.9}$$

Thus, the number of sources can be easily determined by counting the eigenvalues of $\mathbf{R}_{\tilde{\mathbf{X}}}$ that are greater than one.

2.2.3 Generalized Eigenvalue Decomposition (GEVD)

However, calculating the square root of $\mathbf{R}_N$ to obtain $\mathbf{R}_N^{1/2}$ may introduce inaccuracies in the data. To avoid this issue, an alternative approach is to employ the generalized

eigenvalue decomposition (GEVD) on the matrix pencil $(\Phi, \mathbf{R}_N)$ [11]:

$$\begin{aligned}\Phi &= \mathbf{F}\Lambda_S\mathbf{F}^H \\ \mathbf{R}_N &= \mathbf{F}\mathbf{I}_M\mathbf{F}^H,\end{aligned}\tag{2.10}$$

where the matrix $\mathbf{F}$ corresponds to the generalized eigenvectors and has dimensions $M \times M$, Λ_S represents the diagonal matrix containing the generalized eigenvalues. It is important to note that obtaining the source data matrix directly is often impossible. To address this limitation, we can employ the GEVD on the matrix pencil $(\mathbf{R}_X, \mathbf{R}_N)$ with slight modifications made to the generalized eigenvalues. According to Eq. (2.2), we have

$$\mathbf{R}_X = \mathbf{F}(\Lambda_S + \mathbf{I}_M)\mathbf{F}^H.\tag{2.11}$$

The prewhitened data covariance matrix then can be expressed as

$$\mathbf{R}_{\tilde{X}} = \mathbf{R}_N^{-1/2}\mathbf{F}(\Lambda_S + \mathbf{I}_M)\mathbf{F}^H\mathbf{R}_N^{-1/2}.\tag{2.12}$$

By comparing Eq. (2.12) with Eq. (2.8), we can observe that the prewhitening method and GEVD are equivalent. Specifically, we have $\mathbf{R}_N^{-1/2}\mathbf{F} = \tilde{\mathbf{U}}$ and $\Lambda_S = \tilde{\Lambda}_S$. Therefore, the generalized eigenvalues of the matrix pencil $(\mathbf{R}_X, \mathbf{R}_N)$ are identical to the eigenvalues of the prewhitened data covariance matrix.

The GEVD of the matrix pencil $(\mathbf{R}_X, \mathbf{R}_N)$ can also be viewed as the EVD of $\mathbf{R}_N^{-1}\mathbf{R}_X$, since we have

$$\begin{aligned}\mathbf{R}_N^{-1}\mathbf{R}_X &= \mathbf{F}^{-H}\mathbf{I}_M\mathbf{F}^{-1}\mathbf{F}(\Lambda_S + \mathbf{I}_M)\mathbf{F}^H \\ &= \mathbf{F}^{-H}(\Lambda_S + \mathbf{I}_M)\mathbf{F}^H.\end{aligned}\tag{2.13}$$

Denote $\mathbf{F}^{-H}$ as $\mathbf{K}$. Then, we have

$$\mathbf{R}_N^{-1}\mathbf{R}_X = \mathbf{K}(\Lambda_S + \mathbf{I}_M)\mathbf{K}^{-1}.\tag{2.14}$$

which is the EVD of $\mathbf{R}_N^{-1}\mathbf{R}_X$. It is evident from Eq. (2.11) and Eq. (2.14) that the generalized eigenvalues of the matrix pencil $(\mathbf{R}_X, \mathbf{R}_N)$ are identical to the eigenvalues of $\mathbf{R}_N^{-1}\mathbf{R}_X$. However, as $\mathbf{R}_N^{-1}\mathbf{R}_X$ is not Hermitian, $\mathbf{K}$ is not a unitary matrix ($\mathbf{K}^H \neq \mathbf{K}^{-1}$). In conclusion, the eigenvalues of the covariance matrix of $\tilde{\mathbf{X}}$ are equivalent to those of $\mathbf{R}_N^{-1}\mathbf{R}_X$, and to the generalized eigenvalues of the matrix pencil $(\mathbf{R}_X, \mathbf{R}_N)$.

2.3 Related Work on Rank Detection

In practical applications, the true covariance matrices are typically unknown and need to be estimated. Additionally, the number of observations is finite, resulting in distinct eigenvalues that prevent a straightforward separation between the source and noise components using a simple threshold, as described in Eq. (2.6) or Eq. (2.9). To address these challenges, more sophisticated methods and algorithms are required to detect the rank or determine a suitable threshold.

This section presents several commonly used methods for rank detection, considering different noise distributions, namely Gaussian white noise and non-white noise. The methods are presented separately for each noise type, taking into account the specific characteristics and challenges associated with each.

2.3.1 Rank Detection for Gaussian White Noise

2.3.1.1 Information Theoretic Criteria

For Gaussian white noise, AIC and MDL based on the information theoretic criteria are two well-known and popular rank detecting methods [3]. The rank is set to the argument $q = 0, \dots, M-1$ that minimizes the defined criteria, which balances the trade-off between high likelihood and a low number of free parameters [3]. Let $\hat{\lambda}_i, i = 1, \dots, M$ denote the eigenvalues of $\hat{\mathbf{R}}_X$. The MDL estimator is given by

$$\hat{d}_{MDL} = \arg \min_q - (M - q)N \ln \left[\frac{\prod_{i=q+1}^M \hat{\lambda}_i^{1/(M-q)}}{\frac{1}{M-q} \sum_{i=q+1}^M \hat{\lambda}_i} \right] + \frac{1}{2}q(2M - q) \ln N, \quad (2.15)$$

and the AIC estimator is given by

$$\hat{d}_{AIC} = \arg \min_q - 2(M - q)N \ln \left[\frac{\prod_{i=q+1}^M \hat{\lambda}_i^{1/(M-q)}}{\frac{1}{M-q} \sum_{i=q+1}^M \hat{\lambda}_i} \right] + 2q(2M - q). \quad (2.16)$$

Comparing MDL and AIC, we observe that the first term of MDL is the same as AIC when we ignore the constant factor 2 in AIC. However, MDL introduces an extra factor of $\frac{1}{2} \ln N$ in the second term. This difference greatly impacts the performance of MDL, making it a consistent estimator. In other words, as the number of observations approaches infinity, the estimated rank converges to the true rank with a probability of one [3]. MDL generally provides reliable estimation results, but it may not be effective in scenarios with very low SNR. On the other hand, AIC exhibits good performance at low SNR but tends to overestimate the rank and is not a consistent detector [4]. It is important to note that both MDL and AIC are based on large sample asymptotics and may encounter challenges when dealing with small sample sizes.

Improvements have been made to address the problems of MDL and AIC. Many methods take into account the statistical characteristics of the eigenvalues of the sample covariance matrix. For example, in [5], authors considered the exact distribution of the eigenvalues of Wishart matrices, while in [6], authors employed the order statistics to give a more accurate estimation of the eigenvalues in small data samples.

2.3.1.2 Threshold Method

Another approach is to employ a threshold-based method where a threshold is set for the eigenvalues, and the number of signals is determined by counting the number of eigenvalues that exceed the threshold. If we assume the eigenvalues from the signals are much larger than from the noise, a reasonable threshold then can be the largest eigenvalue from noise. Therefore, many threshold-based rank detection methods focus on the behavior of the largest eigenvalue from the noise.

Bai and Yin [12] provided an upper bound of the largest eigenvalue of the SCM of Gaussian white noise. This upper bound is determined by the noise power and the size

of the matrix. As the number of observations increases, the largest eigenvalue tends to converge to this upper bound. Using this upper bound as a threshold can be an effective approach to detect the rank when the eigenvalues of the signals are significantly larger than those of the noise, or when a large number of data samples are available.

A more accurate threshold considering the finite data samples is given in [13, 14], based on the random matrix theory. This theory shows that the largest eigenvalue of the noise SCM follows the Tracy-Widom distribution to the order of $O(n^{-2/3})$. Based on this distribution, it is possible to determine a threshold that achieves a constant false alarm rate (CFAR), ensuring a desired level of accuracy in detecting the rank. In the work of Zhou et al. [15], a computationally efficient method called SURV was proposed as a counterpart to generalized singular value decomposition (GSVD). SURV utilizes the largest eigenvalue threshold derived from the Tracy-Widom distribution to perform noise removal by eliminating the noise components from the data.

In [7], Kritchman and Nadler developed a rank estimator based on this random matrix theory, in which the rank is found via a sequence of hypothesis tests. The authors also proved that the estimator is weakly consistent, but if the false alarm rate decreases with the sample size, then it will be strongly consistent. The performance of this rank estimator based on random matrix theory was shown to outperform traditional methods such as MDL and AIC in various scenarios.

Authors in [16] made further advancements in refining the threshold when signals are present. Moreover, they discovered that the largest eigenvalue of the Hankel structure of the noise SCM follows a generalized extreme value (GEV) distribution. The authors also provided an empirical expression for the eigenvalue distribution based on extensive experimental analysis.

2.3.2 Rank Detection for Colored and Correlated Noise

Things would become differently for the non-white noise, characterized by a covariance matrix that is not diagonal with identical elements. In such cases, a prewhitening technique is commonly employed to transform the noise into a 'white' noise model. This can also be achieved through the GEVD on the matrix pencil $(\hat{\mathbf{R}}_X, \hat{\mathbf{R}}_N)$. If the noise covariance matrix is known or the number of noise samples used for prewhitening is sufficiently large, the prewhitened noise approximates Gaussian white noise, enabling the utilization of methods designed for Gaussian white noise.

However, challenges arise when the noise covariance matrix is unknown and estimated from a limited number of noise data samples. In such cases, the prewhitening process may not fully eliminate the non-white characteristics of the noise. Consequently, the prewhitened noise may exhibit different properties from Gaussian white noise, rendering the application of methods designed for Gaussian white noise invalid for rank detection in this context.

Addressing rank detection in the presence of non-white noise is an area that has received relatively limited attention. Some approaches extend the information theoretic criteria to address colored and correlated noise. For instance, in [8], the authors considered colored noise and assume a specific model for the noise covariance matrix, leading to the development of modified versions of MDL and AIC based on this model. Another approach, described in [10], proposes a quasi-maximum likelihood (quasi-ML)

estimator of the data covariance matrix to account for colored noise, which is then used to construct criteria for rank detection. Furthermore, Zhao et al. [9] extended the criteria to arbitrary noise settings and provided a consistent estimator, although the performance with finite data samples is not explicitly discussed. Additionally, [17] shed light on rank detection based on random matrix theory, building upon the work of Johnstone [18]. However, the rank detection algorithm proposed in [17] lacks a consolidated derivation, performance comparison with other methods, and comprehensive analysis.

Furthermore, it is important to note that the aforementioned methods, whether designed for Gaussian white noise or non-white noise, all assume stationarity of the signal. For non-stationary signals such as speech and audio, the STFT is commonly employed, and the characteristics of the noise and signal may change. This can affect the effectiveness of methods based on these characteristics for rank detection.

2.4 Problem Formulation

Motivated by the related work, this thesis is aimed to solve the rank detection problem using the threshold-based method when the noise is colored or correlated. We will mainly base ourselves on the theory from Johnstone [18]. To answer the main research question, given the finite number of noisy observations $\mathbf{X}$ and pure noise samples $\mathbf{N}$, we are going to solve the following problems:

- Find the threshold for the largest generalized eigenvalue from noise subspace of the GEVD on the matrix pencil $(\hat{\mathbf{R}}_X, \hat{\mathbf{R}}_N)$.
- Develop a reliable rank detection algorithm based on the threshold.
- Extend the rank detection method to the STFT domain.

Finally, we will test the effectiveness of the rank detection method in an array processing application.

Threshold for Prewhitened Sample Covariance Matrix

3

In this chapter, we will investigate the threshold for the largest eigenvalue of the prewhitened SCM. We will begin by reviewing the eigenvalue threshold for the Gaussian white noise, and check if this threshold can be applied to the prewhitened covariance matrix. Next, we will present the Tracy-Widom approximation of the distribution of the largest eigenvalue for F-type matrices, which also represents the theoretical distribution of the largest eigenvalue of the prewhitened covariance matrix. To demonstrate the validity of the theory, we will provide simulations.

3.1 Threshold for Gaussian White Noise

We first review the results summarized in [13, 14] of the largest eigenvalue distribution of the white Gaussian covariance matrix. Consider the complex Gaussian white noise matrix $\mathbf{A} \in \mathbb{C}^{M \times N_a}$ from the distribution $\mathcal{CN}(\mathbf{0}, \mathbf{I}_M)$. The SCM is given by $\hat{\mathbf{R}}_A = \frac{1}{N_a} \mathbf{A} \mathbf{A}^H$ and follows the Wishart distribution. For $M, N_a \rightarrow \infty$ and $M/N_a \rightarrow \zeta \in (0, \infty)$, the distribution of the largest eigenvalue $\hat{\lambda}_1$ of $\hat{\mathbf{R}}_A$ then can be approximated by the Tracy-Widom distribution of order 2:

$$\frac{N_a \hat{\lambda}_1 - \rho_{M, N_a}}{v_{M, N_a}} \xrightarrow{\mathcal{D}} \mathcal{TW}_2, \quad (3.1)$$

with centering and scaling constants as

$$\begin{aligned} \rho_{M, N_a} &= (\sqrt{N_a} + \sqrt{M})^2, \\ v_{M, N_a} &= (\sqrt{N_a} + \sqrt{M}) \left(\frac{1}{\sqrt{N_a}} + \frac{1}{\sqrt{M}} \right)^{\frac{1}{3}}, \end{aligned}$$

where slight modifications are required when the data matrix is real. See Appendix A.

For the largest eigenvalue, we are interested in a threshold β that achieves a certain false alarm rate $1 - p$, such that the probability $P(\hat{\lambda}_1 > \beta) = 1 - p$. Let F_2^{-1} denote the inverse Cumulative Distribution Function (CDF) of Tracy-Widom distribution of order 2. The probability can be written as

$$\begin{aligned} P(\hat{\lambda}_1 > \beta) = 1 - p &\Leftrightarrow P\left(\frac{N_a \hat{\lambda}_1 - \rho_{M, N_a}}{v_{M, N_a}} < \frac{N_a \beta - \rho_{M, N_a}}{v_{M, N_a}}\right) = p \\ &\Leftrightarrow \frac{N_a \beta - \rho_{M, N_a}}{v_{M, N_a}} = F_2^{-1}(p). \end{aligned}$$

The threshold β is given by

$$\beta = \frac{1}{N_a} (\rho_{M, N_a} + v_{M, N_a} F_2^{-1}(p)). \quad (3.2)$$

where F_2^{-1} can be approximated by an inverse gamma distribution [14]. If the noise power is σ_n^2 , the threshold should be adjusted to be $\sigma_n^2\beta$.

3.2 Threshold for Prewhitened Noise

The purpose of using prewhitening or GEVD is to whiten the noise data and ensure that it exhibits the property of white noise. However, it is important to determine whether the prewhitened noise is truly white noise or not. If it is, then the threshold for Gaussian white noise can be used. In this section, we will investigate this issue and test the effectiveness of that threshold. The correct threshold then will be given.

3.2.1 The distribution of Prewhitened Noise

Without loss of generality, assume we have two complex Gaussian random matrices $\mathbf{A} \in \mathbb{C}^{M \times N_a}$ and $\mathbf{B} \in \mathbb{C}^{M \times N_b}$, where both matrices are observations from $\mathcal{CN}(\mathbf{0}, \mathbf{I}_M)$. Their corresponding SCMs are given by $\hat{\mathbf{R}}_A = \frac{1}{N_a} \mathbf{A} \mathbf{A}^H$ and $\hat{\mathbf{R}}_B = \frac{1}{N_b} \mathbf{B} \mathbf{B}^H$, respectively. To ensure the invertibility of the SCM, the number of snapshots N_b of $\mathbf{B}$ should be greater than or equal to the number of sensors.

We use the matrix $\mathbf{B}$ to prewhiten matrix $\mathbf{A}$, and the prewhitened noise matrix is given by $\tilde{\mathbf{A}} = \hat{\mathbf{R}}_B^{-1/2} \mathbf{A} \in \mathbb{C}^{M \times N_a}$. The prewhitened matrix $\tilde{\mathbf{A}}$ can be regarded as the linear transformation of the Gaussian white noise, and the observations in $\tilde{\mathbf{A}}$ follow the distribution $\mathcal{CN}(\mathbf{0}, \hat{\mathbf{R}}_B^{-1})$. The SCM of the prewhitened noise is given by $\hat{\mathbf{R}}_{\tilde{A}} = \frac{1}{N_a} \tilde{\mathbf{A}} \tilde{\mathbf{A}}^H$.

If N_b is finite, the matrix $\hat{\mathbf{R}}_B^{-1}$ will not be equal to the identity matrix. As a result, the threshold defined in Equation Eq. (3.2) is invalid and inappropriate for the largest eigenvalue of the prewhitened covariance matrix. Therefore, modifications to the parameters of the threshold are required.

3.2.2 Adjustment of the Threshold β for Prewhitened Noise

Since the prewhitened noise is not exactly equal to white noise, the largest eigenvalue of the SCM of the prewhitened noise does not follow the Tracy-Widom distribution described in Eq. (3.2). As shown in Fig. 3.1, the empirical CDF of the largest eigenvalue of the SCM of prewhitened noise deviates greatly from the Tracy-Widom model with parameter $(M = 4, N_a = 200)$, which is represented by the green line. However, the empirical CDF exhibits a similar shape to the Tracy-Widom distribution, and its behavior is constant with fixed M , N_a , and N_b .

We have observed that the threshold can still be useful by modifying the parameter N_a to a smaller value denoted by $\tilde{N}_a$. The modified Tracy-Widom model with $\tilde{N}_a = 80$ fits better to the empirical CDF, as shown by the red curve in Fig. 3.1. However, a mathematical method for finding the parameter $\tilde{N}_a$ is currently unknown.

One possible solution is to use empirical results to determine the suitable $\tilde{N}_a$ for all triples of (M, N_a, N_b) and fit them into a certain type of function. However, this approach is time-consuming and prone to overfitting if not all possible parameter triples are included. Additionally, when the number of observations is small, $\tilde{N}_a$ might be smaller than the number of sensors or might not exist.

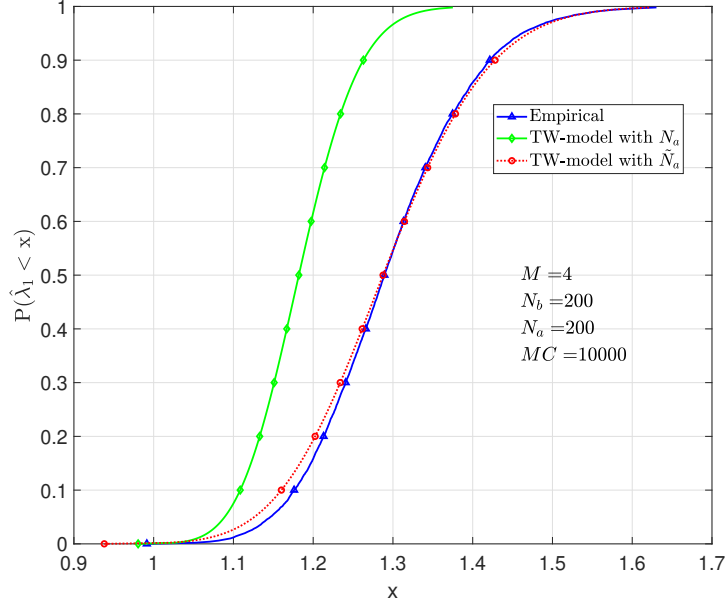


Figure 3.1: Empirical CDF of the largest eigenvalue of the SCM of prewhitened noise, compared to the theoretical CDF and the modified CDF. $M = 4$, $N_a = 200$, $N_b = 200$, $\tilde{N}_a = 80$.

The threshold is valid without any modifications when the length of matrix $\mathbf{B}$ is infinite. In that case, the sample covariance matrix of $\mathbf{B}$ equals the identity matrix. Hence, we have

$$\lim_{N_b \rightarrow \infty} \tilde{\mathbf{A}} = \lim_{N_b \rightarrow \infty} \hat{\mathbf{R}}_B^{-1/2} \mathbf{A} = \mathbf{I}_M^{-1/2} \mathbf{A} = \mathbf{A},$$

which indicates that the prewhitened data matrix will have the Gaussian distribution $\mathcal{CN}(\mathbf{0}, \mathbf{I}_M)$. In this case, the threshold defined in Eq. (3.2) is effective in providing an upper bound for the largest eigenvalue, and the Tracy-Widom model fits perfectly with the empirical curve, as shown in Fig. 3.2. Therefore, if the noise covariance matrix is known apriori, the threshold β will be a precise upper bound of the largest eigenvalue and can achieve a certain false alarm rate.

3.2.3 Threshold for F-type Matrices

In this small section, we will begin by reviewing the results of the distribution of the largest eigenvalue of F-type matrices [17, 18, 19]. The F-type matrix is given by $\hat{\mathbf{R}}_B^{-1} \hat{\mathbf{R}}_A$, where $\mathbf{A}$ and $\mathbf{B}$ are matrices with the same notation as before. In a more general case, Johnstone [18] considered the distribution of the largest eigenvalue of $(\hat{\mathbf{R}}_B + \hat{\mathbf{R}}_A)^{-1} \hat{\mathbf{R}}_A$. The results for both matrices are almost the same, with small differences in some constant scaling numbers. We focus on the largest eigenvalue of $\hat{\mathbf{R}}_B^{-1} \hat{\mathbf{R}}_A$ since its eigenvalues are identical to those of the prewhitened covariance matrix.

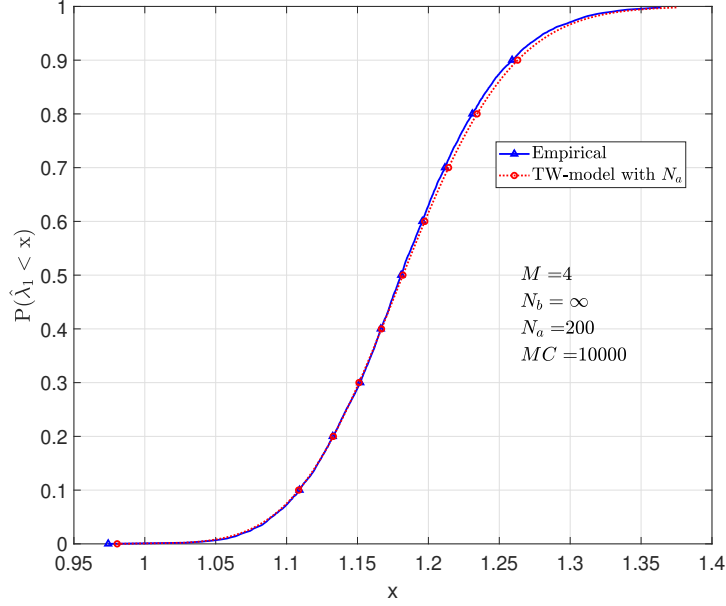


Figure 3.2: Empirical CDF of the largest eigenvalue of the SCM of prewhitened noise, compared to the theoretical CDF when $N_b = \infty$.

Assume M, N_a, N_b satisfy the following equations when $M \rightarrow \infty$:

$$\lim_{M \rightarrow \infty} \frac{\min(M, N_a)}{N_a + N_b} > 0, \quad (3.3)$$

$$\lim_{M \rightarrow \infty} \frac{M}{N_b} < 1. \quad (3.4)$$

The condition stated in Eq. (3.3) indicates that if M goes to infinity, N_a and N_b should also go to infinity. In other words, for a finite number of sensors M , the number of observations N_a and N_b must also be finite. Additionally, as noted in Eq. (3.4), the number of samples N_b must exceed the number of sensors M to ensure that the SCM of $\mathbf{B}$ is invertible.

Then, the logarithm of the largest eigenvalue $\hat{\lambda}_1$ of the F type matrix $\hat{\mathbf{R}}_B^{-1} \hat{\mathbf{R}}_A$ can be approximated to order $O(n^{-2/3})$ by the Tracy-Widom law with proper centering and scaling [17, 18]:

$$\frac{\ln\left(\frac{N_a}{N_b} \hat{\lambda}_1\right) - \rho_{M, N_a, N_b}}{v_{M, N_a, N_b}} \xrightarrow{\mathcal{D}} \mathcal{TW}_2. \quad (3.5)$$

To describe the centering and scaling constants, a parameter K is defined as $K = \min(M, N_a)$. The centering and scaling constants can then be obtained as follows:

$$\rho_{M, N_a, N_b} = \frac{\frac{u_K}{\tau_K} + \frac{u_{K-1}}{\tau_{K-1}}}{\frac{1}{\tau_K} + \frac{1}{\tau_{K-1}}}, \quad (3.6)$$

$$v_{M,N_a,N_b} = \frac{2}{\frac{1}{\tau_K} + \frac{1}{\tau_{K-1}}}, \quad (3.7)$$

where

$$u_K = 2 \ln \tan \frac{\gamma_K + \phi_K}{2},$$

$$\tau_K^3 = \frac{16}{(2K + N_b - M + |N_a - M| + 1)^2} \frac{1}{\sin^2(\gamma_K + \phi_K) \sin(\gamma_K) \sin(\phi_K)},$$

and

$$\begin{aligned} \sin^2\left(\frac{\gamma_K}{2}\right) &= \frac{K + 1/2}{2K + N_b - M + |N_a - M| + 1}, \\ \sin^2\left(\frac{\phi_K}{2}\right) &= \frac{K + |N_a - M| + 1/2}{2K + N_b - M + |N_a - M| + 1}. \end{aligned}$$

For the case of real data, the centering and scaling constants need to be modified. More details can be found in Appendix A. To determine the threshold for the largest eigenvalue, we can set the probability $P(\hat{\lambda}_1 > \beta)$ to a desired false alarm rate of $1 - p$, as we have done previously. We can then express the probability in the same way as before:

$$\begin{aligned} P(\hat{\lambda}_1 > \beta) = 1 - p &\Leftrightarrow P\left(\frac{\ln\left(\frac{N_a}{N_b}\hat{\lambda}_1\right) - \rho_{M,N_a,N_b}}{v_{M,N_a,N_b}} < \frac{\ln\left(\frac{N_a}{N_b}\beta\right) - \rho_{M,N_a,N_b}}{v_{M,N_a,N_b}}\right) = p \\ &\Leftrightarrow \frac{\ln\left(\frac{N_a}{N_b}\beta\right) - \rho_{M,N_a,N_b}}{v_{M,N_a,N_b}} = F_2^{-1}(p). \end{aligned}$$

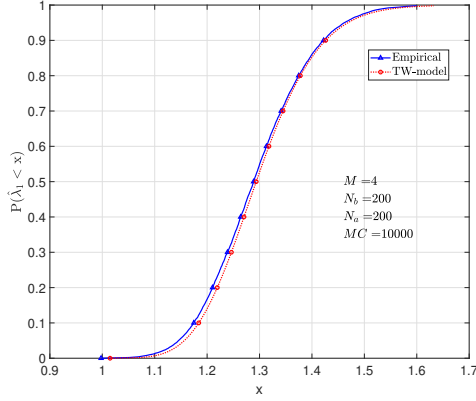
Then, the threshold is given by

$$\beta = \frac{N_b}{N_a} \exp(\rho_{M,N_a,N_b} + v_{M,N_a,N_b} F_2^{-1}(p)). \quad (3.8)$$

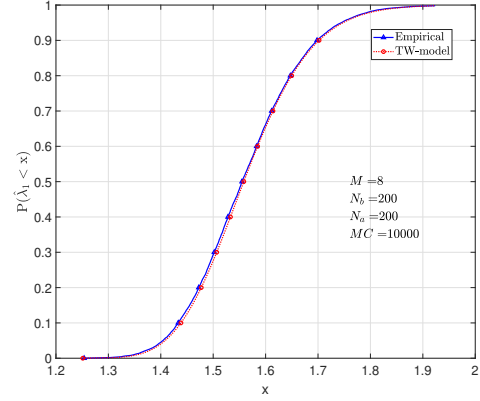
The threshold defined in this case is more complex than that of the Gaussian white noise scenario. It involves more parameters and functions, as well as the logarithm transformation of the largest eigenvalue. The introduction of the logarithm transformation is crucial for ensuring the accuracy of the approximation, especially when the number of observations is small [18].

For simulations, we select various values of (M, N_a, N_b) and verify the reliability of the threshold β by plotting the CDF of the Tracy-Widom model, as defined in Eq. (3.5), and the empirical CDF of the largest eigenvalue of the SCM of prewhitened noise. To ensure statistical significance, we conduct 10,000 Monte Carlo experiments and present the results in Fig. 3.3.

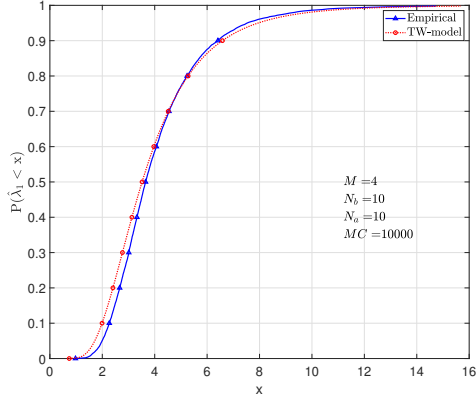
As demonstrated in Fig. 3.3, the empirical curve of the largest eigenvalue of the SCM of the prewhitened noise fits perfectly with the Tracy-Widom model in every simulation scenario, even when the number of observations is reduced or the number of sensors is increased. However, since the Tracy-Widom distribution is an approximation, deviations may occur when the number of observations is small, as shown in



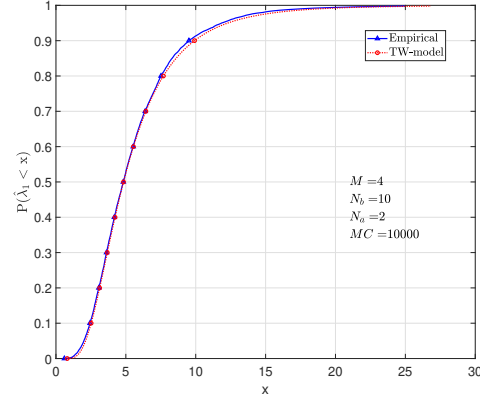
(a) CDF of the largest eigenvalue for $M = 4, N_a = N_b = 200$.



(b) CDF of the largest eigenvalue for $M = 8, N_a = N_b = 200$.



(c) CDF of the largest eigenvalue for $M = 4, N_a = N_b = 10$.



(d) CDF of the largest eigenvalue for $M = 4, N_a = 2, N_b = 10$.

Figure 3.3: Theoretical and empirical CDF of the largest eigenvalue of the SCM of prewhitened noise.

Fig. 5.2c. Additionally, the model can provide an accurate approximation of the empirical eigenvalue distribution even when N_a is smaller than M . This feature could be valuable in applications where parameters need to be updated adaptively based on a single data vector per time, but the number of data observations collected is smaller than the number of sensors.

Consider a more general case where the noise is not white noise. Let $\mathbf{N}_A$ and $\mathbf{N}_B$ be two independent Gaussian noise matrices, each of size $M \times N_a$ and $M \times N_b$, respectively. Both matrices are drawn from the distribution $\mathcal{CN}(\mathbf{0}, \mathbf{\Sigma})$, where $\mathbf{\Sigma}$ is a positive-definite Hermitian matrix. We can express $\mathbf{N}_A$ and $\mathbf{N}_B$ as linear transformations of the Gaussian white noise matrices $\mathbf{A}$ and $\mathbf{B}$ respectively as follows:

$$\begin{aligned}\mathbf{N}_A &= \mathbf{T}\mathbf{A}, \\ \mathbf{N}_B &= \mathbf{T}\mathbf{B},\end{aligned}$$

where $\mathbf{T}$ is the lower-triangular Cholesky decomposition matrix of Σ , i.e., $\Sigma = \mathbf{T}\mathbf{T}^H$.

The threshold defined in Eq. (3.8) is even effective when the noise is not white, as the eigenvalues of $\hat{\mathbf{R}}_{N_B}^{-1}\hat{\mathbf{R}}_{N_A}$ and $\hat{\mathbf{R}}_B^{-1}\hat{\mathbf{R}}_A$ are identical.

Proof:

The sample covariance matrices of $\mathbf{N}_A$ and $\mathbf{N}_B$ are given by:

$$\hat{\mathbf{R}}_{N_A} = \frac{1}{N_a}\mathbf{N}_A\mathbf{N}_A^H = \frac{1}{N_a}\mathbf{T}\mathbf{A}\mathbf{A}^H\mathbf{T}^H,$$

and

$$\hat{\mathbf{R}}_{N_B} = \frac{1}{N_b}\mathbf{N}_B\mathbf{N}_B^H = \frac{1}{N_b}\mathbf{T}\mathbf{B}\mathbf{B}^H\mathbf{T}^H.$$

Hence, we have

$$\begin{aligned}\hat{\mathbf{R}}_{N_B}^{-1}\hat{\mathbf{R}}_{N_A} &= \{\mathbf{T}(\frac{1}{N_b}\mathbf{B}\mathbf{B}^H)\mathbf{T}^H\}^{-1}\mathbf{T}(\frac{1}{N_a}\mathbf{A}\mathbf{A}^H)\mathbf{T}^H \\ &= \mathbf{T}^{-H}\hat{\mathbf{R}}_B^{-1}\mathbf{T}^{-1}\mathbf{T}\hat{\mathbf{R}}_A\mathbf{T}^H \\ &= \mathbf{T}^{-H}\hat{\mathbf{R}}_B^{-1}\hat{\mathbf{R}}_A\mathbf{T}^H.\end{aligned}\tag{3.9}$$

The invertibility of the matrix $\mathbf{T}$ implies that $\hat{\mathbf{R}}_{N_B}^{-1}\hat{\mathbf{R}}_{N_A}$ is a similarity transformation of $\hat{\mathbf{R}}_B^{-1}\hat{\mathbf{R}}_A$. As a result, the eigenvalues of $\hat{\mathbf{R}}_{N_B}^{-1}\hat{\mathbf{R}}_{N_A}$ are identical to those of $\hat{\mathbf{R}}_B^{-1}\hat{\mathbf{R}}_A$. Thus, the threshold defined in Eq. (3.8) remains effective even when the noise is not white.

We perform the same simulations as before but generate the correlated noise matrices $\mathbf{N}_A$ and $\mathbf{N}_B$ from the Gaussian distribution with the following covariance matrix:

$$\Sigma = \begin{bmatrix} 2 & 1 & 0.5 & 0.2 \\ 1 & 1 & 0.4 & 0.5 \\ 0.5 & 0.4 & 1.6 & 0.3 \\ 0.2 & 0.5 & 0.3 & 0.5 \end{bmatrix}.$$

As shown in Fig. 3.4, the Tracy-Widom model again provides an accurate approximation to the empirical eigenvalue distribution for the case of correlated noise. This result confirms that the threshold defined in Eq. (3.8) is effective even when the noise is not white. It also shows that the centering and scaling constants derived in [18, 19] are applicable in the case of correlated noise.

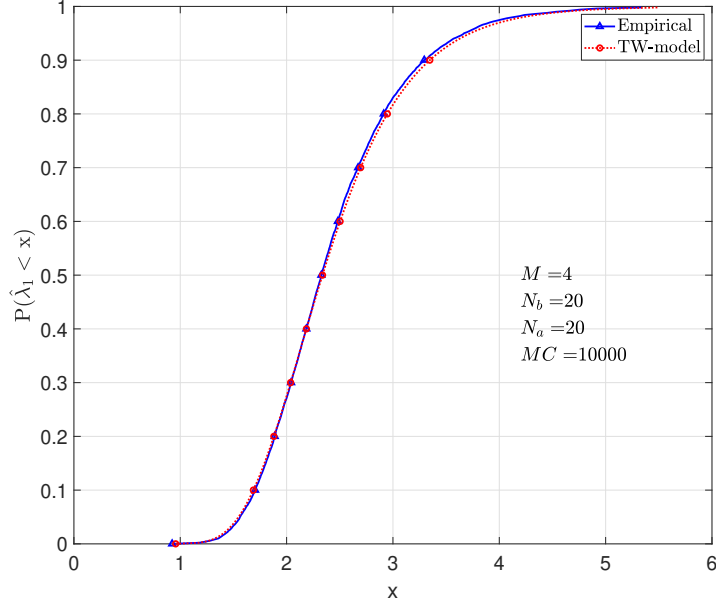


Figure 3.4: CDF of the largest eigenvalue of the SCM of prewhitened noise for the correlated noise.

It is important to note that the threshold derived above assumes that both N_a and N_b are finite. In practice, we may encounter situations where the noise covariance matrix is known or, equivalently, N_b is infinite. In this case, the threshold for the prewhitened noise given by Eq. (3.8) is no longer valid, and we need to use the threshold for the Gaussian white noise case defined in Eq. (3.2). In conclusion, the threshold for the prewhitened noise is given by:

$$\beta(M, N_a, N_b) = \begin{cases} \frac{N_b}{N_a} \exp(\rho_{M, N_a, N_b} + v_{M, N_a, N_b} F_2^{-1}(p)) & N_b \neq \infty \\ \frac{1}{N_a} (\rho_{M, N_a} + v_{M, N_a} F_2^{-1}(p)) & N_b = \infty. \end{cases} \quad (3.10)$$

3.3 Threshold in the Signal-plus-Noise Case

We now provide a more accurate threshold when the signal is corrupted by noise. Specifically, we consider a signal model described in Eq. (2.3). Here we show again the signal model, but for ease of the description, we denote the noise matrix incorporated into the signal matrix as $\mathbf{N}_A$. The signal model is given by

$$\mathbf{X} = \mathbf{X}_S + \mathbf{N}_A, \quad (3.11)$$

where $\mathbf{X}_S \in \mathbb{C}^{M \times N_a}$ is the zero mean random Gaussian source matrix with rank d , and $\mathbf{N}_A$ is the random Gaussian noise matrix with zero mean and any covariance matrix. We prewhiten the noise in $\mathbf{X}$ using the noise data matrix $\mathbf{N}_B$ of size $M \times N_b$, collected during the signal-free period. The prewhitened data matrix is denoted as $\tilde{\mathbf{X}}$.

To detect the number of sources in $\mathbf{X}$, we compare the eigenvalues of the SCM of $\tilde{\mathbf{X}}$ to a threshold. While the threshold defined in Eq. (3.10) can be used directly

for rank detection, it may result in low detection probability for small SNR. In [17], a more accurate threshold for rank detection is presented. The authors claimed that the distribution of the largest eigenvalue from the noise in a prewhitened covariance matrix of a rank- d matrix $\mathbf{X}$ is equivalent to the largest eigenvalue distribution of the noise-only case with parameters $(M - d, N_a - d, N_b)$. That is

$$\frac{\ln\left(\frac{N_a}{N_b}\hat{\lambda}_{d+1}\right) - \rho_{M-d, N_a-d, N_b}}{v_{M-d, N_a-d, N_b}} \xrightarrow{\mathcal{D}} \mathcal{TW}_2. \quad (3.12)$$

However, they do not provide a theoretical proof for this statement. In the following, we will offer our explanation for this claim.

The prewhitened data matrix is of size $M \times N_a$ and can be written as:

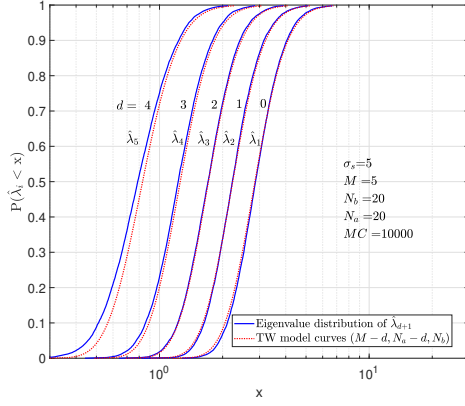
$$\begin{aligned} \tilde{\mathbf{X}} &= \hat{\mathbf{R}}_{N_B}^{-1/2} \mathbf{X}_S + \hat{\mathbf{R}}_{N_B}^{-1/2} \mathbf{N}_A \\ &= \tilde{\mathbf{X}}_S + \tilde{\mathbf{N}}_A, \end{aligned} \quad (3.13)$$

where $\tilde{\mathbf{N}}_A$ can be approximately regarded as the noise drawing from the Gaussian white noise. The accuracy of this approximation depends on the number of the noise samples N_b used for prewhitening. In general, the prewhitened data matrix is the addition of a low-rank source matrix and the Gaussian white noise matrix. Then, the largest eigenvalue corresponding to the noise subspace of $\tilde{\mathbf{R}}_{\tilde{\mathbf{X}}}$ converges to the largest eigenvalue distribution of the noise-only covariance matrix of size $(M - d) \times (N_a - d)$. A proof of this convergence can be found in [16]. As a result, the $d + 1$ -st eigenvalue of $\tilde{\mathbf{R}}_{\tilde{\mathbf{X}}}$ has the same distribution of the largest eigenvalue as the largest eigenvalue of the noise-only situation, but with the parameters $(M - d, N_a - d, N_b)$ used to calculate the centering and scaling constants in Eq. (3.10). Then, the threshold is modified as

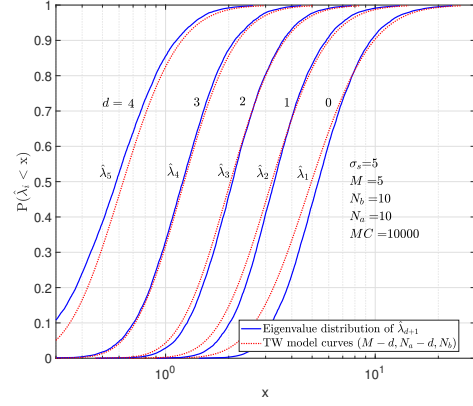
$$\beta(M, N_a, N_b, d) = \begin{cases} \frac{N_b}{N_a} \exp(\rho_{M-d, N_a-d, N_b} + v_{M-d, N_a-d, N_b} F_2^{-1}(p)) & N_b \neq \infty \\ \frac{1}{N_a} (\rho_{M-d, N_a-d} + v_{M-d, N_a-d} F_2^{-1}(p)) & N_b = \infty. \end{cases} \quad (3.14)$$

For simulations, we generate the source data matrix from the Gaussian distribution $\mathcal{CN}(\mathbf{0}, \sigma_s^2 \mathbf{I}_d)$. Without loss of generality, the noise observations are collected from the standard Gaussian distribution $\mathcal{CN}(\mathbf{0}, \mathbf{I}_M)$. We then prewhiten the data matrix using the noise matrix $\mathbf{N}_B$ of size $M \times N_b$.

To evaluate the accuracy of the eigenvalue approximation, we plot the empirical CDF of the $d + 1$ -st eigenvalue of the SCM of prewhitened data matrix in Fig. 3.5. We compare this distribution to the Tracy-Widom model with parameters $(M - d, N_a - d, N_b)$. As shown in the figure, the fit is relatively good for large values of N_a and N_b . However, for small values of N_a and N_b , deviations occur due to the prewhitened noise violating the assumption of Gaussian white noise. Nevertheless, the fit is still highly accurate in the high probability range, demonstrating the validity of the threshold for a low false alarm rate. Additionally, it is important to note that the value of the threshold β increases when d decreases, as we can see in the figure where the CDF of smaller d is on the right side of larger d .



(a) CDF of the $d + 1$ -st eigenvalue for $M = 5, N_a = N_b = 20$.



(b) CDF of the $d + 1$ -st eigenvalue for $M = 5, N_a = N_b = 10$.

Figure 3.5: CDF of the $d + 1$ -st eigenvalue of the SCM of prewhitened data matrix with d number of sources, and the Tracy-Widom model with parameters $(M - d, N_a - d, N_b)$.

3.4 Chapter Conclusion

In this chapter, we have shown that the largest eigenvalue of the SCM of prewhitened noise follows the Tracy-Widom distribution, similar to the largest eigenvalue of the SCM of Gaussian white noise. However, the Tracy-Widom distribution for prewhitened noise has distinct centering and scaling constants. By utilizing this distribution, the threshold for achieving a desired false alarm rate is derived, which is a function of matrix dimensions (M, N_a, N_b) and the false alarm rate. Furthermore, in the case of the signal-plus-noise scenario with a rank- d source matrix, the threshold is modified to be calculated with the parameters $(M - d, N_a - d, N_b)$. Simulations conducted shown that the empirical CDF fits well with the CDF of the Tracy-Widom distribution, validating the findings.

Threshold-based Rank Detection Algorithm

4

In this chapter, we will develop a rank detection algorithm that utilizes the threshold defined in the previous chapter. The main objective of this algorithm is to detect the rank of the covariance matrix in a more efficient and accurate way. We will compare the performance of our algorithm with two commonly used rank detection algorithms, namely MDL and AIC.

4.1 Sequential Rank Detection Algorithm

Knowing the $d + 1$ -st eigenvalue distribution allows us to establish a more accurate threshold for distinguishing between the signal and noise subspaces. Based on this, we can now present a rank detection algorithm for the prewhitened covariance matrix, which takes as input the eigenvalues, the dimensions of the data matrices, and a desired false alarm rate. The algorithm uses a sequential test approach to determine the rank of the data matrix. Specifically, for $k = 1, \dots, \min(M, N_a) - 1$, we consider the hypotheses:

$$\begin{aligned}\mathcal{H}_k &: \text{rank}(\hat{\mathbf{R}}_{\tilde{X}}) = k, \\ \mathcal{H}'_k &: \text{rank}(\hat{\mathbf{R}}_{\tilde{X}}) > k.\end{aligned}$$

The null hypothesis $\mathcal{H}_k$ is tested starting from $k = 0$. If $\mathcal{H}'_0$ is accepted, k will be increased until $\mathcal{H}_k$ is accepted. The complete algorithm is summarized in Algorithm 1. It is worth noting that the eigenvalues of the SCM of prewhitened noise are identical to the generalized eigenvalues of the matrix pencil $(\hat{\mathbf{R}}_X, \hat{\mathbf{R}}_{N_B})$, which can also be used as input for the algorithm.

Algorithm 1 Sequential rank detection algorithm

Input: M, N_a, N_b , false alarm rate P_{FA} , eigenvalues $\hat{\lambda}_i$ for $i = 1, 2, \dots, M$
Initialization: $\hat{d} = 0, k = -1$ $\triangleright k$ could be any number except 0.
while $k \neq \hat{d}$ **do**
 $k = \hat{d}$
 Compute the threshold β with Eq. (3.14)
 $\hat{d} = \#\{\hat{\lambda}_i > \beta\}$
end while
Output: the estimated rank $\hat{d}$

The detecting procedure is different from that of MDL and AIC. In MDL and AIC, we need to calculate the criteria for every possible rank and choose the rank that minimizes the criteria [3]. In other words, the rank is increased by a step size of 1 from 0 to the maximum value, and the criteria value is evaluated for each rank. However, in

our approach, the rank can be increased in larger steps, and the step size is determined by the number of eigenvalues surpassing the threshold. We stop the procedure once the estimated rank remains unchanged. In general, we do not evaluate every possible rank, making our algorithm more efficient than the MDL and AIC.

4.2 Performance Analysis

We will analyze the performance of the proposed detection algorithm and present the theoretical probability of detection. A similar analysis for a signal corrupted by Gaussian white noise (i.e., $N_b = \infty$) can be found in [7]. We will adopt the same approach to analyze the detection algorithm for the prewhitened data. In our analysis, we assume that both $\mathbf{N}_A$ and $\mathbf{N}_B$ are collected from Gaussian white noise with zero mean and unit variance, and the signal is zero mean Gaussian distributed, with known covariance matrix $\mathbf{P} = \text{diag}\{p_1, \dots, p_d\}$. The eigenvalues corresponding to the signal components are then asymptotically equal to $p_i + 1$, for $1 \leq i \leq d$.

Let us review the detection procedure in our algorithm. For ease of notation, we use β_k to represent $\beta(M, N_a, N_b, k)$. Initially, we set the estimated rank to 0 and compare all the eigenvalues with the threshold β_0 . Subsequently, we increase the estimated rank to k (i.e., the number of eigenvalues that are greater than β_0) and compare the eigenvalues with the new threshold β_k , repeating this process until the estimated rank remains the same. To ensure the feasibility of our analysis, we assume $\hat{\lambda}_d$ has multiplicity one and $\hat{\lambda}_d \ll \hat{\lambda}_{d-1}$, such that the primary error is the misidentification of the d -th eigenvalue. In order to detect the true rank d , the eigenvalues should at least satisfy the following condition:

$$\hat{\lambda}_d > \beta_{d-1} > \beta_d > \hat{\lambda}_{d+1}, \quad (4.1)$$

where $\hat{\lambda}_{d+1}$ corresponds to the largest eigenvalue of the noise components, and is bounded by β_d with a probability of $1 - P_{FA}$. The value of β_{d-1} is larger than the value of β_d with a probability of one. Thus, we can approximate the probability of correct detection as:

$$P(\hat{d} = d) = P(\hat{\lambda}_d > \beta_{d-1}, \beta_d > \hat{\lambda}_{d+1}) \approx (1 - P_{FA})P(\hat{\lambda}_d > \beta_{d-1}), \quad (4.2)$$

where we assume the independence, which might not hold strictly.

To compute this probability, we can rely on the theory presented in [20], which provides the distribution for the eigenvalues from the signal subspace of $\hat{\mathbf{R}}_{\hat{X}}$. Specifically, for $\hat{\lambda}_i > \frac{c_2 + r}{1 - c_2}$, where $1 \leq i \leq d$, the asymptotic distribution of the eigenvalue $\hat{\lambda}_i$ can be approximated by a normal distribution:

$$t(\hat{\lambda}_i - \tau(p_i)) \xrightarrow{\mathcal{D}} \mathcal{N}(0, r^2 \delta^2(p_i)), \quad (4.3)$$

where $c_1 = \frac{M}{N_a - d}$, $c_2 = \frac{M}{N_b}$, $t = \sqrt{M}$ and

$$\tau(p_i) = \frac{(p_i + c_1)(p_i + 1)}{p_i - c_2(p_i + 1)},$$

$$\delta^2(p_i) = \frac{p_i^2(1+p_i)^2(p_i^2 - c_2(1+p_i)^2 - c_1)}{(c_2 - p_i + c_2 p_i)^4},$$

$$r^2 = c_1 + c_2 - c_1 c_2.$$

$\tau(p_i)$ denotes the asymptotic limit of the eigenvalue and converges to $p_i + 1$ as $c_1 \rightarrow 0$ and $c_2 \rightarrow 0$. The behavior of the signal eigenvalues of the prewhitened covariance matrix depends on the signal strength and the number of sensors and observations, as explained in Eq. (4.3). However, this approximation can be more accurate for a finite number of data samples with two modifications we consider here.

The first modification is similar to that in [7], in which the interactions between signals are also taken into account. For finite observations, the eigenvalues of the sample covariance matrix of the signal are not exactly equal to the signal strengths we set. The eigenvalues might be slightly smaller than the presetting signal strengths. In [21], the authors provided an expectation value of the i -th ($i > 1$) eigenvalue given the signal strengths up to $O(1/n^2)$,

$$\tilde{p}_i = \mathbb{E}[\hat{\lambda}_i] = p_i - \frac{p_i}{N_a} \sum_{j=1}^{i-1} \frac{p_j}{p_j - p_i}. \quad (4.4)$$

We will use this expectation value $\tilde{p}_i$ instead of p_i in our analysis.

In the second modification, we replace M with $M - d$ in Eq. (4.3). This modification is based on the fact that our main interest is the distribution of $\hat{\lambda}_d$. However, we have found that the approximation of the distribution in Eq. (4.3) is less accurate for the d -th eigenvalue, when there are a finite number of data samples, as shown in Fig. 4.1a.

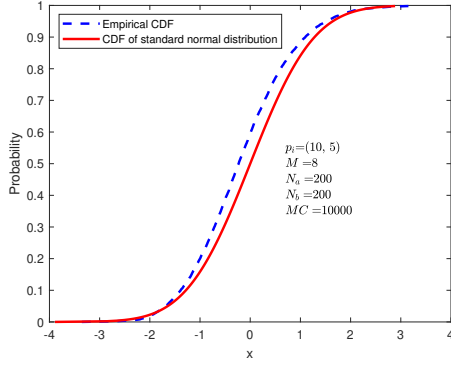
To obtain a more accurate distribution of the d -th eigenvalue for finite M , N_a , and N_b , we replace M with $M - d$ for above parameters, and we denote the modified parameters as $\tilde{t} = \sqrt{M - d}$, $\tilde{c}_1 = \frac{M-d}{N_a-d}$, $\tilde{c}_2 = \frac{M-d}{N_b}$. The results for the modification are shown in the Fig. 4.1b, where we observe a better fit of the CDF when using the modified parameters. Although this modification's influence reduces for large data samples, it is necessary for real-world applications that typically have a small and finite number of data samples. However, the theoretical proof of this modification is unknown, and empirical simulations are used to validate it.

Finally, with both the modifications, we can approximate the probability of detecting the correct number of signals as

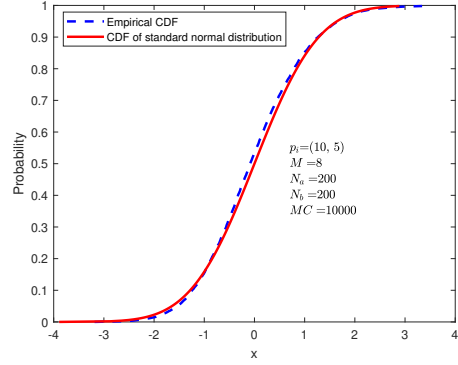
$$P(\hat{d} = d) \approx (1 - P_{FA}) P\{\eta > \tilde{t} \frac{\beta_{d-1} - \tau(\tilde{p}_d)}{r\delta(\tilde{p}_d)}\}, \quad (4.5)$$

where $\eta \sim \mathcal{N}(0, 1)$. $\tau(\tilde{p}_d)$, r , and $\delta(\tilde{p}_d)$ are computed with $\tilde{c}_1$ and $\tilde{c}_2$.

In Fig. 4.2 and Fig. 4.3, we present the results of simulations to support the theoretical analysis. To obtain the empirical results, we use a false alarm rate of 1% and implement the proposed sequential detecting algorithm. As shown in the Fig. 4.2, the empirical curve closely fits the theoretical curve as M increases, and the probability of detection will gradually approach the desired false alarm rate. However, when there are multiple sources and the signal strength of the d -th source is decreased, deviations

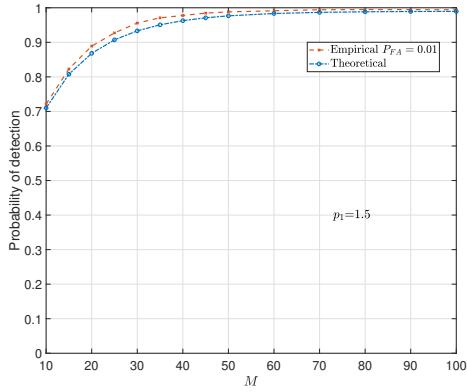


(a) CDF of the d -th eigenvalue without modifications of M for $M = 8, d = 2, N_a = N_b = 200$.

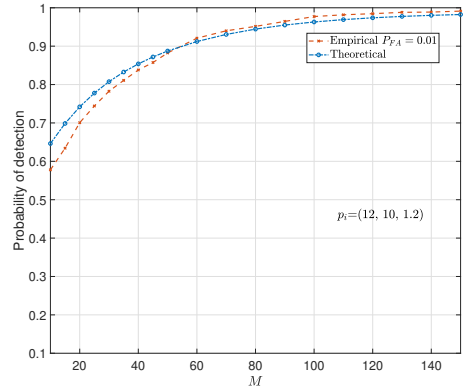


(b) CDF of the d -th eigenvalue with modifications of M for $M = 8, d = 2, N_a = N_b = 200$.

Figure 4.1: Empirical and theoretical CDF of the d -th eigenvalue with d sources.



(a) Empirical and theoretical probability detection for a single source, with signal strength $p_i = 1.5$.



(b) Empirical and theoretical probability detection for three sources, with signal strength $p_i = (12, 10, 1.2)$.

Figure 4.2: Empirical and theoretical probability detection as a function of M , with $N_a = N_b = 8M$.

occur at lower values of M . It is similar when changing the signal strength as shown in Fig. 4.3.

Eq. (4.5) allows us to answer the question of how strong the signal strength p_d should be to achieve at least a $\frac{1}{2}(1 - P_{FA})$ detection probability. For $P(\hat{d} = d) \approx \frac{1}{2}(1 - P_{FA})$, p_d should satisfy the condition:

$$\tau(\tilde{p}_d) = \beta_{d-1}. \quad (4.6)$$

For simplicity, we will ignore the effects of signal interactions. Plugging Eq. (3.10) and the equation for $\tau(p_d)$ above, and p_d can be obtained by solving the following

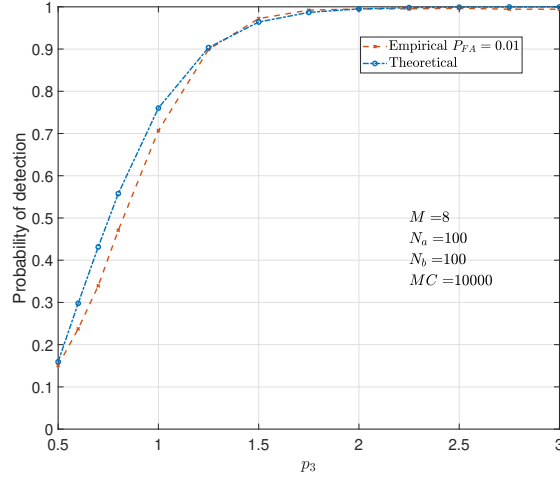
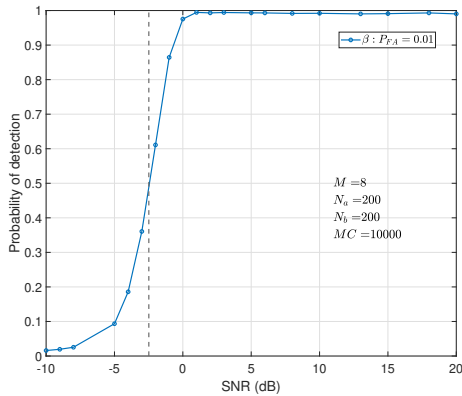


Figure 4.3: Empirical and theoretical probability detection for three sources when changing the signal strength of the third source p_3 , and fixing the first two signals strengths $p_1 = 12, p_2 = 10$.

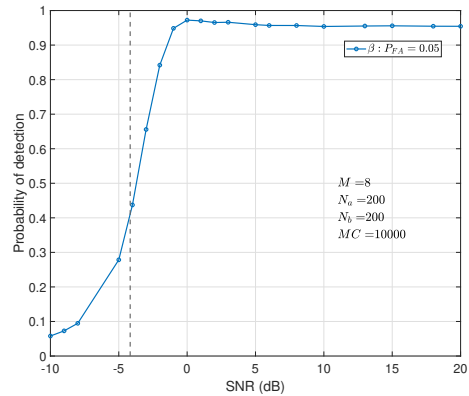
quadratic equation:

$$p_d^2 + (c_1 + 1 - \beta_{d-1} + c_2\beta_{d-1})p_d + c_1 + c_2\beta_{d-1} = 0, \quad (4.7)$$

where for large false alarm rates, the roots of the quadratic equation might be complex numbers. The same problem also occurs in [7]. In such case, we can take the absolute value of the root, but the results might not fit the theoretical analysis perfectly.



(a) Probability detection for a single source.



(b) Probability detection for two sources, with $p_1 = 4p_2$.

Figure 4.4: Probability detection as a function of signal strength for $M = 8, N_a = N_b = 200$.

In Fig. 4.4, we show the probability of detection for our algorithm at false alarm rates of 1% and 5%. The dashed black line represents the signal strength threshold required to achieve a $\frac{1}{2}(1 - P_{FA})$ probability of detection. We observe that the threshold

closely corresponds to a $\frac{1}{2}(1 - P_{FA}) = 0.495$ probability of detection for the 1% false alarm rate. However, for the larger false alarm rate, the detection rate of the signal strength threshold is lower than $\frac{1}{2}(1 - P_{FA}) = 0.475$, possibly due to the occurrence of complex roots in the quadratic equation and the ignorance of the signal interactions.

The analysis and simulations presented here demonstrate that the empirical detection rate of the rank detection algorithm converges to the theoretical rate. However, slight discrepancies arise due to the approximations made for simplifying the analysis. With this analysis, we can anticipate the detection rate when certain information, such as SNR and signal strengths is known.

4.3 Rank Detection Simulations

Two simulations are conducted to validate the effectiveness of the proposed rank detection algorithm. To measure the performance of the detection algorithm, we use the probability of detection, which is defined as:

$$P_D = \frac{\# \text{ of experiments that give the correct rank estimation}}{\# \text{ of total experiments}}$$

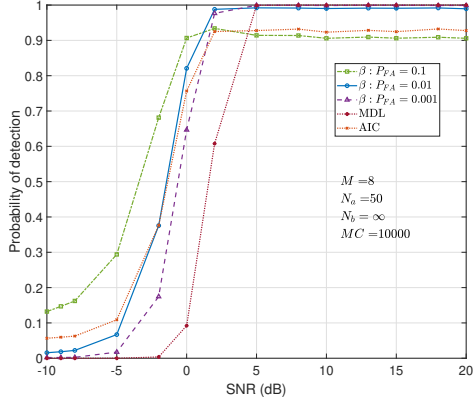
The source and noise matrices are generated as before with various ranks. For both simulations, 10,000 experiments in total are conducted.

4.3.1 Rank Detection for $N_b = \infty$

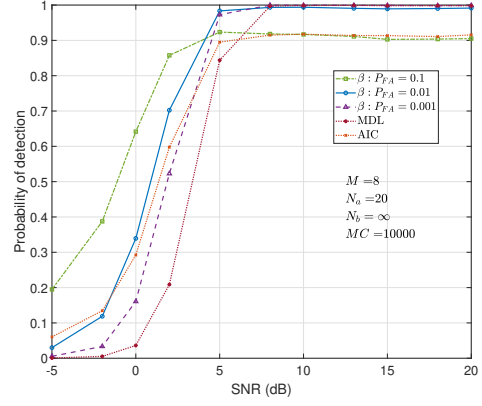
In the first simulation, we assume that the covariance matrix is known, i.e., we have an infinite number of snapshots ($N_b = \infty$). We compare the performance of the proposed method with other rank detectors such as MDL and AIC. The simulation results are shown in Fig. 4.5. We observe that the proposed algorithm can accurately detect the true rank of the signal at a relatively small P_{FA} when the SNR is sufficiently high. Although a larger false alarm rate (0.1) can provide a more accurate estimation at low SNR, its probability of detection at high SNR is 10% lower than that of a higher P_{FA} . The performance of the detection algorithm is affected by the increase in signal rank and decrease in the number of snapshots, leading to a significant degradation in estimation accuracy at low SNR.

MDL shows similar performance to the proposed algorithm for a small P_{FA} when the number of observations N_a is large, as seen in subfigures (a), (c), and (e) of Fig. 4.5. At high SNR, the rank estimation accuracy using MDL is almost 100%, but the performance is worse than the proposed algorithm at low SNR. However, when the signal has a high rank but the number of snapshots is small, using MDL could lead to a higher incorrect rank estimation rate, as observed in subfigure (f) of Fig. 4.5.

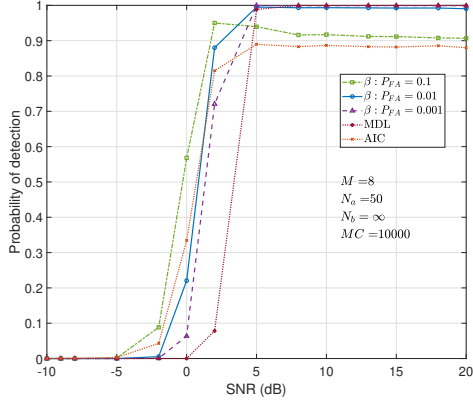
In contrast, AIC exhibits a similar pattern to the proposed algorithm when $P_{FA} = 0.1$, achieving good estimation at low SNR but worse performance at high SNR. However, unlike the proposed algorithm, the performance of AIC is significantly degraded by the decrease in the number of observations or the increase in the rank of the signal.



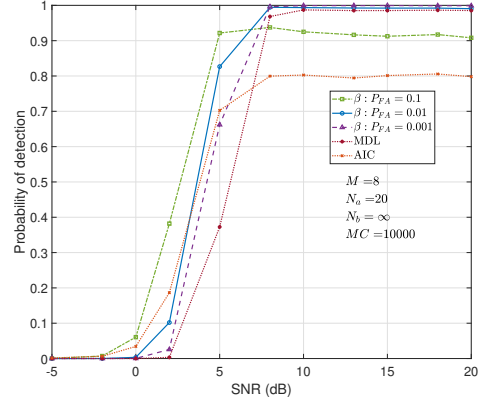
(a) Rank detection probability for $M = 8, d = 1, N_a = 50, N_b = \infty$.



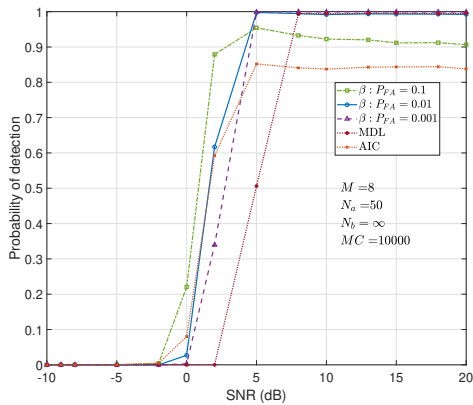
(b) Rank detection probability for $M = 8, d = 1, N_a = 20, N_b = \infty$.



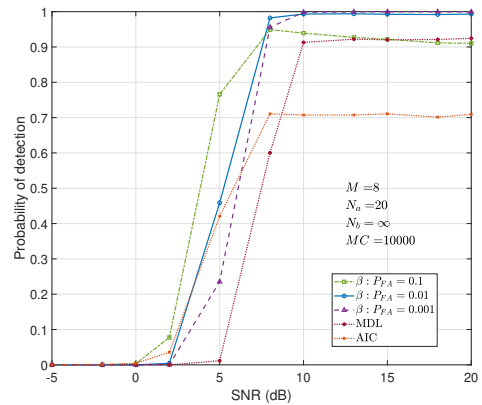
(c) Rank detection probability for $M = 8, d = 3, N_a = 50, N_b = \infty$.



(d) Rank detection probability for $M = 8, d = 3, N_a = 20, N_b = \infty$.



(e) Rank detection probability for $M = 8, d = 5, N_a = 50, N_b = \infty$.



(f) Rank detection probability for $M = 8, d = 5, N_a = 20, N_b = \infty$.

Figure 4.5: The rank detection performance of different methods for $N_b = \infty$, while varying the rank d , N_a .

4.3.2 Rank Detection for $N_b \neq \infty$

In the second simulation, we examine the effect of a finite number of N_b , as shown in Fig. 4.6. The results indicate that the proposed detection algorithm performs similarly to the case when $N_b = \infty$, and it outperforms MDL in all cases, achieving the desired estimation accuracy at high SNR compared to MDL's approximately 93% accuracy. However, the proposed algorithm is less accurate in the very low SNR range, where MDL sometimes performs better than the proposed algorithm. On the other hand, AIC is not able to estimate the rank of the data matrix due to using a limited number of noise samples for prewhitening, resulting in incomplete prewhitening of the noise. We can conclude that the not fully prewhitened noise has a greater impact on AIC than MDL. Therefore, we will discard AIC in the following experiments.

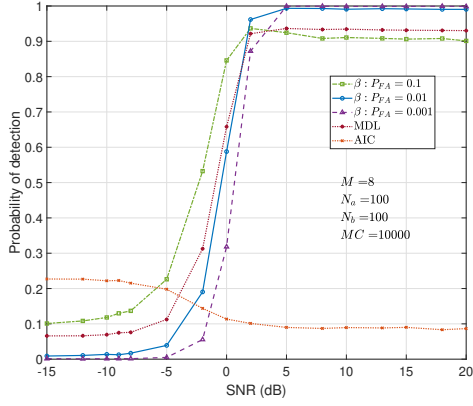
As the number of observations (N_a and N_b) decreases, the proposed method requires a higher SNR to achieve comparable performance to that of larger snapshot numbers. However, for MDL, the degradation in performance is severe as N_a and N_b decrease. This maybe because MDL is designed for rank detection of a signal in a Gaussian white noise environment [3]. In our case, the whiteness of the noise depends on the number of observations. When the number of observations is small, the prewhitened noise has a covariance matrix given by the inverse of $\hat{\mathbf{R}}_{N_B}$. This covariance matrix may have non-zero off-diagonal elements, resulting in correlated prewhitened noise.

When the rank of the data matrix increases, the proposed algorithm requires a higher SNR to achieve a high probability of detection, similar to the effect of decreasing the number of observations. On the other hand, the performance of MDL does not seem to be affected by the increasing rank, and may even slightly improve the estimation accuracy.

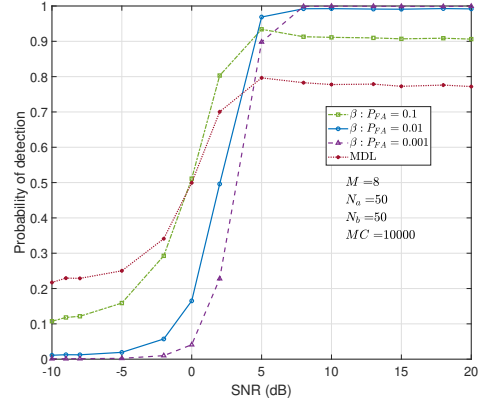
The false alarm rate has a similar influence on the detection algorithm's performance as in the previous case. At false alarm rates of 0.01 and 0.001, the algorithm achieves almost 100% probability of detection at high SNR, while a false alarm rate of 0.1 results in a 10% probability of incorrect rank estimation. However, it is worth noting that the algorithm with a false alarm rate of 0.1 is more robust to noise and has a higher probability of detection than the algorithm with smaller false alarm rates. While setting an extremely low false alarm rate is often desirable, it may not always result in the best performance. In the presence of high noise levels, a slightly higher false alarm rate may be more effective than an extremely small one.

4.4 Chapter Conclusion

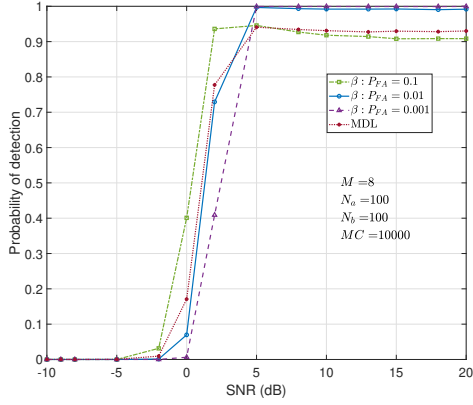
Based on the distribution of the largest eigenvalue of the prewhitened SCM, we have developed a rank detection algorithm via a sequential test approach. The performance analysis and simulations of the rank detection algorithm are presented to demonstrate the effectiveness of the proposed method and its superiority over the MDL and AIC. Moreover, we can control the probability of detection by adjusting the false alarm rate, which provides greater flexibility compared to conventional methods.



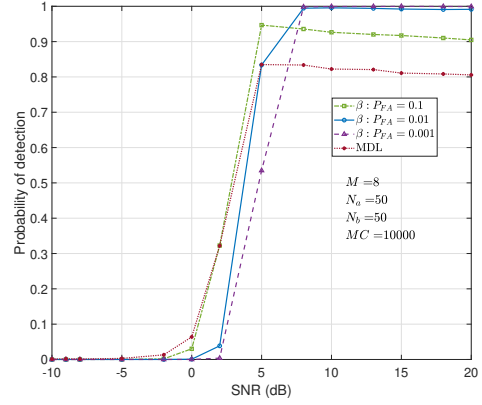
(a) Rank detection probability for $M=8, d=1, N_a=N_b=100$.



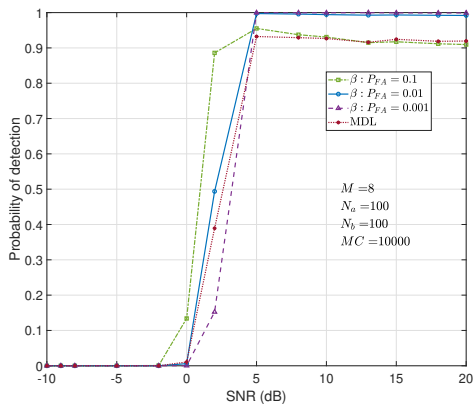
(b) Rank detection probability for $M=8, d=1, N_a=N_b=50$.



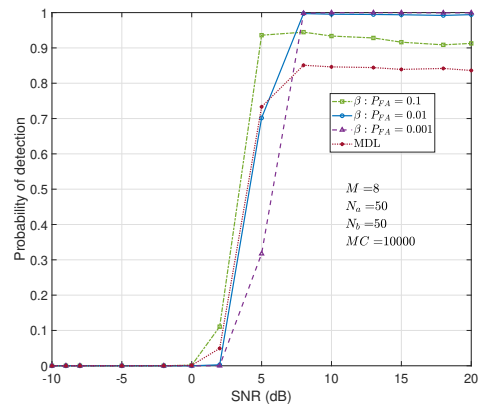
(c) Rank detection probability for $M=8, d=3, N_a=N_b=100$.



(d) Rank detection probability for $M=8, d=3, N_a=N_b=50$.



(e) Rank detection probability for $M=8, d=5, N_a=N_b=100$.



(f) Rank detection probability for $M=8, d=5, N_a=N_b=50$.

Figure 4.6: The rank detection performance of different methods, while varying the rank d , N_a and N_b

Threshold-based Rank Detection Method in the STFT Domain

5

The previous chapter presented the threshold method for detecting the rank of the data matrix in the time domain. However, since many signal processing applications are implemented in the frequency domain, and the Short-Time Fourier Transform (STFT) is often used for non-stationary signals, it is important to consider how our method performs under this condition. In this chapter, we will analyze the potential issues with using STFT and how they may affect the performance of the threshold method. We will address these issues and present simulations to illustrate our findings.

5.1 Short-Time Fourier Transform (STFT)

The STFT is a useful tool in signal analysis and processing, because it can not only capture some time-varying information, but also convert wide-band problems into narrow-band problems. It is suitable to analyze the time-frequency characteristics of non-stationary signals, such as speech and audio signals. Unlike the Fourier transform, the STFT divides a long signal into a series of short time segments, and applies the Fourier transform to each segment independently. This method not only provides frequency resolution but also preserves time information. The STFT is widely used in many speech and audio applications, where the short time segments of the signal are assumed to be stationary. This assumption greatly reduces the complexity and simplifies the processing of speech signals.

The mathematical expression of the STFT for a finite sequence $x[n], n = 0, \dots, N-1$ with a real window function $w[n]$ of length L is given by

$$X_m[k] = \sum_{n=0}^{L-1} w[n]x[n+mH]e^{-j\frac{2\pi}{K}kn}, \quad (5.1)$$

where m indicates the time frame index and $1 \leq m \leq T_f$, $T_f = \lfloor \frac{N-L}{H} \rfloor$ is the total time frames, H is the hop size implying the percentage of overlap, k is the frequency bin index with $0 \leq k \leq K-1$, K is the number of total frequency bins and $K \geq L$, $X_m[k]$ is the STFT coefficient at time frame m and frequency bin k .

To better understand the computation of the STFT, Fig. 5.1 illustrates the process of STFT on the data vector $x[n]$. For multiple channel inputs, the STFT is performed independently on each channel. First, the data vector is divided into T_f segments, each with a length of L . If the hop size $H < L$, adjacent segments will overlap, as shown by the shaded areas in the figure. A window function is then applied to each segment, and FFT is performed on the windowed segments to produce the STFT coefficients. Therefore, the STFT coefficients at the same time frame are correlated across the frequency bins.

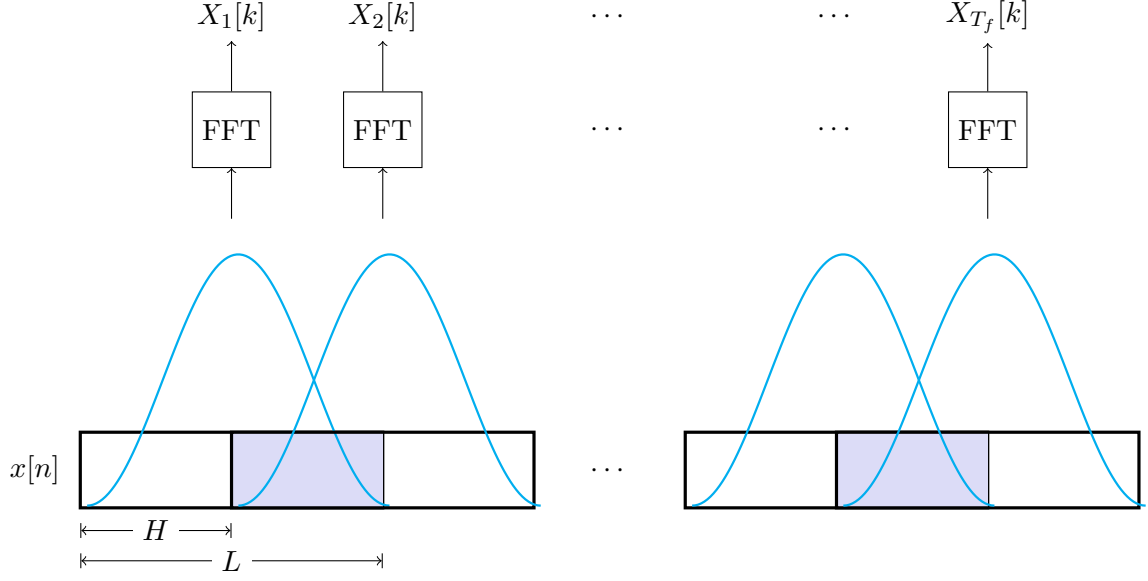


Figure 5.1: Diagram of STFT on a data vector.

To reduce the effects of spectral leakage, triangular shaped windows, such as Hamming and Hanning window functions, are commonly used. However, these window functions have small values on the boundaries, which can result in data loss if no overlap is used [22]. To make full use of the data, the hop size H is typically set to $L/2$ or $L/4$, which corresponds to 50% or 75% overlap, respectively. As a result, adjacent STFT coefficients in the same frequency bin are correlated with each other, meaning that the STFT coefficients are correlated across time.

5.2 Problems of Threshold Methods in the STFT domain

In our proposed threshold method, we assume that the noise matrix is composed of columns of i.i.d. variables from the zero-mean Gaussian distribution, while rows might be correlated with each other. In other words, the noise is correlated spatially across sensors but independent across time. The same assumption is also applied to the source.

However, as we transform the data into the frequency domain via STFT with a hop size $H < L$, we will not only have correlated samples along the frequency bin, but also across all the time frames. Even if we apply the STFT to Gaussian white noise, the resulting STFT coefficients are complex Gaussian noise and not white noise [23]. As a consequence, the largest eigenvalue of the covariance matrix of Gaussian white noise in the STFT domain will not follow the Tracy-Widom distribution, and the largest eigenvalue distribution of the prewhitened SCM after STFT will also be affected. Therefore, the proposed threshold method will not be effective in achieving the desired false alarm rate.

To demonstrate the correlation problem in STFT, we present an example that focuses only on the time correlation aspect. Fig. 5.2 shows the comparison between the empirical CDF of the largest eigenvalue of the prewhitened noise covariance matrix

after STFT and the CDF of the Tracy-Widom distribution. The simulation is conducted by generating two Gaussian white noise matrices, and performing STFT on them with $K = L = 256$. The resulting STFT coefficients of the two noise matrices at one frequency bin have sizes of $M \times N_a$ and $M \times N_b$, respectively. We conduct 10,000 experiments and compare different window functions.

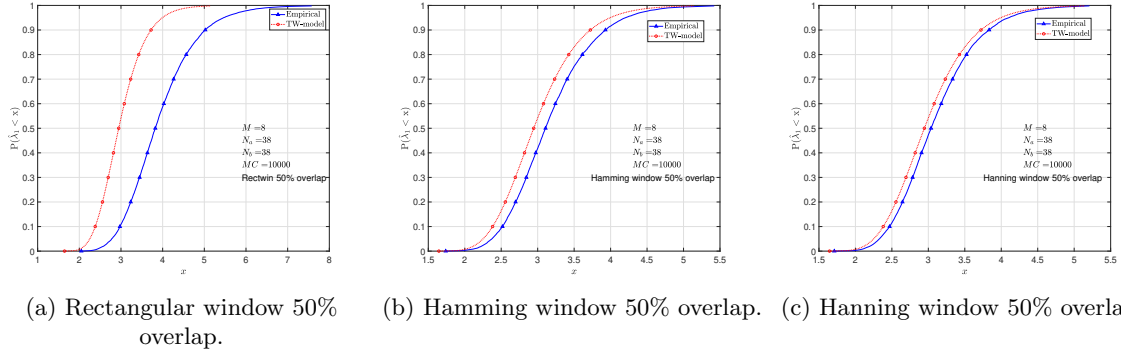


Figure 5.2: Empirical CDF of the largest eigenvalue of the SCM of prewhitened noise after STFT, compared to the CDF of the Tracy-Widom model for different window functions

As shown in Fig. 5.2, the empirical CDF of the largest eigenvalue of the SCM of prewhitened noise after STFT does not fit the Tracy-Widom model. The gap between the empirical and theoretical curves depends on the window function and overlap size. For example, using the Hanning window with 50% overlap gives a closer fit to the TW model, while the rectangular window fails to fit the model.

In many applications, people assume that the STFT of Gaussian white noise remains the same properties and design the signal model based on that assumption. However, after STFT, the noise is no longer Gaussian white noise, and the STFT coefficients have correlated columns at every frequency bin. If the algorithms in the applications are not very sensitive to the noise distribution, the assumption of Gaussian white noise after STFT can still be utilized in some cases, such as using the Hanning window with 50% overlap.

Furthermore, we observe that the CDF of the Tracy-Widom distribution is consistently on the left side of the empirical curve. This indicates that applying the threshold-based rank detection method directly on the STFT coefficients would result in a higher false alarm rate than intended. As our method relies on the accurate knowledge of the noise distribution, we cannot overlook the impact of correlations between the STFT coefficients. Therefore, in the following section, we propose a method to remove these correlations.

5.3 Correlation in STFT

As we mentioned earlier, the STFT introduces both frequency correlation and time correlation in the resulting coefficients. The frequency correlation arises because the STFT coefficients are computed from the same time frame. On the other hand, the

time correlation arises due to the overlapping nature of the STFT analysis. These correlations have a significant impact on the distribution of the signals and can hinder the effectiveness of threshold-based methods in signal analysis and processing. To address this issue, it is necessary to remove these correlations. We will utilize the exact expressions of these correlations in the context of the STFT to achieve this.

Without loss of generality, we investigate the correlation of the Gaussian white noise in STFT domain. Now, assume we have a complex random vector $v[n], n = 0, \dots, N-1$ of i.i.d. variables from the zero mean and σ^2 variance Gaussian distribution. The STFT of this sequence is given by

$$V_m[k] = \sum_{n=0}^{L-1} w[n]v[n+mH]e^{-j\frac{2\pi}{K}kn}. \quad (5.2)$$

5.3.1 Time correlation

The correlations between the STFT coefficients at different time frames are computed via $\mathbb{E}[V_m[k]V_{m+\tau}^*[k]]$ at the same frequency bin. Taking the distribution of the random sequence into account, we derive the expression of the time correlation as follows:

First, plug in the STFT expression for $V_m[k]V_{m+\tau}^*[k]$,

$$\begin{aligned} V_m[k]V_{m+\tau}^*[k] &= \sum_{n_1=0}^{L-1} w[n_1]v[n_1+mH]e^{-j\frac{2\pi}{K}kn_1} \left(\sum_{n_2=0}^{L-1} w[n_2]v[n_2+(m+\tau)H]e^{-j\frac{2\pi}{K}kn_2} \right)^* \\ &= \sum_{n_1=0}^{L-1} \sum_{n_2=0}^{L-1} w[n_1]v[n_1+mH]e^{-j\frac{2\pi}{K}kn_1} w[n_2]v^*[n_2+(m+\tau)H]e^{j\frac{2\pi}{K}kn_2} \\ &= \sum_{n_1=0}^{L-1} \sum_{n_2=0}^{L-1} w[n_1]w[n_2]v[n_1+mH]v^*[n_2+(m+\tau)H]e^{j\frac{2\pi}{K}k(n_2-n_1)}, \end{aligned} \quad (5.3)$$

with $(\cdot)^*$ representing the complex conjugate. Since $v[n]$ are i.i.d., and $\mathbb{E}[v[n]v^*[n]] = \sigma^2$, dropping those terms where $n_1 + mH \neq n_2 + (m+\tau)H$, we have

$$\begin{aligned} \mathbb{E}[V_m[k]V_{m+\tau}^*[k]] &= \mathbb{E}\left[\sum_{n_1=0}^{L-1} \sum_{n_2=0}^{L-1} w[n_1]w[n_2]v[n_1+mH]v^*[n_2+(m+\tau)H]e^{j\frac{2\pi}{K}k(n_2-n_1)}\right] \\ &= \sigma^2 w[\tau H]w[0]e^{j\frac{2\pi}{K}k(0-\tau H)} + \sigma^2 w[\tau H+1]w[1]e^{j\frac{2\pi}{K}k(1-(\tau H-1))} + \dots + \\ &\quad \sigma^2 w[L-1]w[L-1-\tau H]e^{j\frac{2\pi}{K}k(L-1-\tau H-(L-1))} \\ &= \sigma^2 e^{-j\frac{2\pi}{K}k\tau H} \sum_{n=\tau H}^{L-1} w[n]w[n-\tau H]. \end{aligned} \quad (5.4)$$

Note that the STFT coefficients only correlate within the maximum D time lag,

with $D = \lfloor \frac{L-1}{H} \rfloor$. Hence, we have

$$\mathbb{E}[V_m[k]V_{m+\tau}^*[k]] = \begin{cases} \sigma^2 e^{-j\frac{2\pi}{K}k\tau H} \sum_{n=\tau H}^{L-1} w[n]w[n-\tau H] & \tau \leq D; \\ 0 & \text{otherwise.} \end{cases} \quad (5.5)$$

The equation above reveals that the correlation of STFT coefficients across time is a function of the window function and an exponential of the frequency k [24]. This expression is similar to the overlap correlation presented in [22, 25], but with the additional exponential frequency dependence. In Fig. 5.3, we show the correlation at a single frequency bin as a function of time lag τ , normalized by the variance $\mathbb{E}[V_m[k]V_m^*[k]]$.

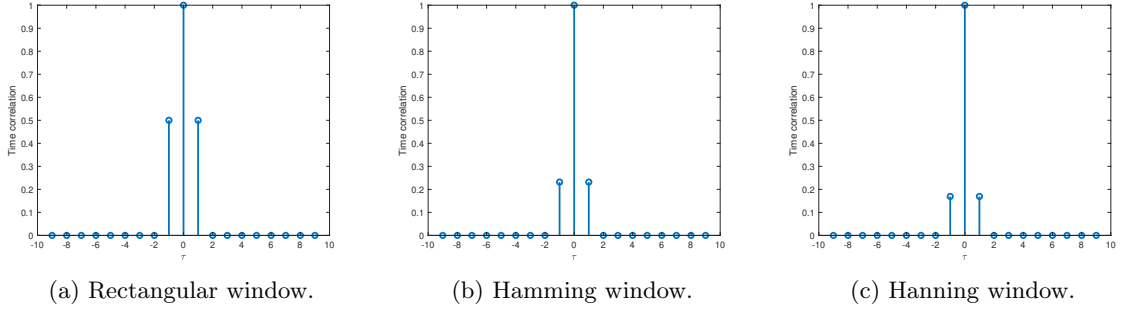


Figure 5.3: Time correlation for the same frequency bin ($k = 2$) as a function of time lag τ , considering different window functions, for $L = 256$, $K = 256$, $H = 128$.

Now we can understand why the Hanning window with 50% overlap fits the TW model closer than other window functions. This is because in this setting, the correlation is relatively small, typically below 0.2. In contrast, for other window functions, such as the rectangular window, the correlation is larger, leading to a larger deviation from the TW distribution. This is consistent with the findings in [22], where the authors suggested that 50% overlap is optimal for the Hanning window, as the overlap correlation is relatively small and the flatness of the spectrum is highest.

5.3.2 Frequency Correlation

For frequency correlation, we calculate $\mathbb{E}[V_m[k]V_m^*[k + \delta]]$ at the same time frame but different frequency bins. δ denotes the frequency difference. Now the expression for $V_m[k]V_m^*[k + \delta]$ is given by

$$\begin{aligned} V_m[k]V_m^*[k + \delta] &= \sum_{n_1=0}^{L-1} w[n_1]v[n_1 + mH]e^{-j\frac{2\pi}{K}kn_1} \left(\sum_{n_2=0}^{L-1} w[n_2]v[n_2 + mH]e^{-j\frac{2\pi}{K}(k+\delta)n_2} \right)^* \\ &= \sum_{n_1=0}^{L-1} \sum_{n_2=0}^{L-1} w[n_1]w[n_2]v[n_1 + mH]v^*[n_2 + mH]e^{j\frac{2\pi}{K}(kn_2 + \delta n_2 - kn_1)}. \end{aligned} \quad (5.6)$$

Similarly, with $v[n]$ i.i.d., dropping those terms where $n_1 + mH \neq n_2 + mH$, we can derive

$$\begin{aligned}
\mathbb{E}[V_m[k]V_m^*[k + \delta]] &= \mathbb{E}\left[\sum_{n_1=0}^{L-1} \sum_{n_2=0}^{L-1} w[n_1]w[n_2]v[n_1 + mH]v^*[n_2 + mH]e^{j\frac{2\pi}{K}(kn_2 + \delta n_2 - kn_1)}\right] \\
&= \sigma^2 w[0]w[0]e^{j\frac{2\pi}{K}0} + \sigma^2 w[1]w[1]e^{j\frac{2\pi}{K}\delta} + \dots + \\
&\quad \sigma^2 w[L-1]w[L-1]e^{j\frac{2\pi}{K}(L-1)\delta} \\
&= \sigma^2 \sum_{n=0}^{L-1} w^2[n]e^{j\frac{2\pi}{K}n\delta}.
\end{aligned} \tag{5.7}$$

It can be noticed that the frequency correlation, unlike the time correlation, is not generally zero for any frequency difference δ . The expression in Equation 5.7 can be viewed as the inverse discrete Fourier transform (DFT) of the window function $w^2[n]$ [24]. For a rectangular window function $w^2[n] = 1$, the frequency correlation $\mathbb{E}[V_m[k]V_m^*[k + \delta]] = \sigma^2$ for $\delta = 0$, and it is zero for any other frequency difference δ . This is because the inverse DFT of the rectangular window is a unit impulse, which indicates that the frequency components are uncorrelated. This aligns with the fact that the DFT of Gaussian white noise remains Gaussian white noise. For other window functions, the frequency correlation approaches zero quickly. In Figure 5.4, we display examples of the frequency correlation for different window functions, and the results are normalized by the correlation for $\delta = 0$.

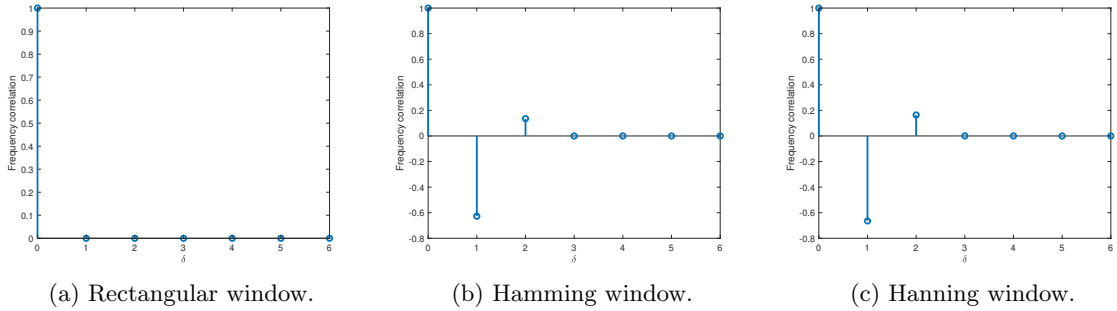


Figure 5.4: Frequency correlation for the same time frame ($m = 3$) as a function of frequency difference δ , considering different window functions, for $L = 256, K = 256, H = 128$.

5.3.3 Time-Frequency Correlation

The time-frequency correlation is then calculate via $\mathbb{E}[V_m[k]V_{m+\tau}^*[k + \delta]]$ between different time frames and frequency bins. Following the same procedure as before, we

have

$$\begin{aligned}
V_m[k]V_{m+\tau}^*[k+\delta] &= \sum_{n_1=0}^{L-1} w[n_1]v[n_1+mH]e^{-j\frac{2\pi}{K}kn_1} \left(\sum_{n_2=0}^{L-1} w[n_2]V[n_2+(m+\tau)H]e^{-j\frac{2\pi}{K}(k+\delta)n_2} \right)^* \\
&= \sum_{n_1=0}^{L-1} \sum_{n_2=0}^{L-1} w[n_1]w[n_2]v[n_1+mH]v^*[n_2+(m+\tau)H]e^{j\frac{2\pi}{K}(kn_2+\delta n_2-kn_1)}.
\end{aligned} \tag{5.8}$$

Then, the correlation can be derived as

$$\begin{aligned}
\mathbb{E}[V_m[k]V_{m+\tau}^*[k+\delta]] &= \mathbb{E}\left[\sum_{n_1=0}^{L-1} \sum_{n_2=0}^{L-1} w[n_1]w[n_2]v[n_1+mH]v^*[n_2+(m+\tau)H]e^{j\frac{2\pi}{K}(kn_2+\delta n_2-kn_1)} \right] \\
&= \sigma^2 w[\tau H]w[0]e^{j\frac{2\pi}{K}(0-k\tau H)} + \sigma^2 w[\tau H+1]w[1]e^{j\frac{2\pi}{K}(\delta-k\tau H)} + \dots + \\
&\quad \sigma^2 w[L-1]w[L-1-\tau H]e^{j\frac{2\pi}{K}((L-1-\tau H)\delta-k\tau H)} \\
&= \sigma^2 e^{-j\frac{2\pi}{K}k\tau H} \sum_{n=\tau H}^{L-1} w[n]w[n-\tau H]e^{j\frac{2\pi}{K}(n-\tau H)\delta} \\
&= \sigma^2 e^{-j\frac{2\pi}{K}(k+\delta)\tau H} \sum_{n=\tau H}^{L-1} w[n]w[n-\tau H]e^{j\frac{2\pi}{K}n\delta}.
\end{aligned} \tag{5.9}$$

Considering the limited value of τ , we have

$$\mathbb{E}[V_m[k]V_{m+\tau}^*[k+\delta]] = \begin{cases} \sigma^2 e^{-j\frac{2\pi}{K}(k+\delta)\tau H} \sum_{n=\tau H}^{L-1} w[n]w[n-\tau H]e^{j\frac{2\pi}{K}n\delta} & \tau \leq D; \\ 0 & \text{otherwise.} \end{cases} \tag{5.10}$$

The time-frequency correlation can be viewed as the combination of the expressions of the time correlation and the frequency correlation. But it cannot be divided as the superposition of the two correlations.

5.4 Removing the Correlations

In this section, we will discuss methods for removing correlations in the STFT coefficients. Let $\mathbf{V}$, with dimensions $K \times T_f$, represent the STFT coefficients of the input signals from one channel. The general approach involves constructing a correlation matrix $\mathbf{C}$, with dimensions $KT_f \times KT_f$, using Equation 5.10. This correlation matrix is then used to prewhiten the STFT coefficients $\mathbf{V}$. To be more specific, we define the whitened STFT coefficients $\mathbf{V}'$ as a matrix of the same size as the original STFT coefficients. The whitened coefficients are obtained using the following expression:

$$vec(\mathbf{V}') = \mathbf{C}^{-1/2}vec(\mathbf{V}), \tag{5.11}$$

where $vec(\cdot)$ represents the vectorization of the matrix by stacking the columns vertically. The matrix $\mathbf{C}$ is positive-definite and Hermitian, which allows us to decompose

it using the Cholesky decomposition, yielding the square root as $\mathbf{C} = \mathbf{L}\mathbf{L}^H$, where $\mathbf{L}$ is a lower triangular matrix. Based on this decomposition, we can express the whitened STFT coefficients $\mathbf{V}'$ as follows:

$$vec(\mathbf{V}') = \mathbf{L}^{-1}vec(\mathbf{V}) \quad (5.12)$$

For multiple channel inputs, we can apply the inverse of $\mathbf{L}$ to each channel independently, effectively whitening the data for each channel.

It is worth noting that constructing the full correlation matrix $\mathbf{C}$ for all frequency bins and time frames, and calculating the inverse of $\mathbf{L}$, can be computationally complex and memory-intensive. The Cholesky decomposition and inverse calculation are both computationally demanding operations with a complexity of $O(n^3)$. Therefore, we need more efficient ways to eliminate correlations. The approach for removing correlations depends on practical considerations and the characteristics of the signals involved. Different scenarios may require different strategies for efficient correlation removal. In this section, we will present three methods to remove time, frequency, and time-frequency correlations, respectively.

5.4.1 Temporal Prewhitening (TP)

For signals such as speech and audio that are not evenly distributed across frequency bins, it is more appropriate to process the frequency bins separately. When we are processing the data at every frequency bin, our main objective is to remove the time correlation, which is referred to as temporal prewhitening (TP).

In this case, the correlation matrix is constructed using Eq. (5.5), and is of size $T_f \times T_f$. Using the standard prewhitening method is still computationally complex if T_f is very large. It is important to note that the correlation is non-zero only within the D time lags, resulting in a banded Toeplitz matrix structure for the correlation matrix. Exploiting this special structure, we develop an algorithm called Cholesky-Forward-Substitution algorithm. This algorithm efficiently computes the inverse of the Cholesky decomposition, enabling us to effectively remove the correlation in an adaptive manner. For $D = 1$ (i.e., using less than or equal to 50% overlap), the computational cost for a single update in this process, considering a single channel, is only $O(1)$. For higher overlaps, the complexity increases to $O(D^2)$ per single update, which is significantly smaller compared to the cubic complexity $O(n^3)$ of standard methods. For a detailed summary of the Cholesky-Forward-Substitution algorithm, please refer to Appendix B.

5.4.2 Frequential Prewhitening (FP)

When processing data at each time frame across the frequency bins, the primary objective is to eliminate frequency correlation, which we will refer to as frequential prewhitening (FP). This process is particularly relevant for signals that are evenly distributed across frequency bins, such as Gaussian signals. By processing the data at each time frame, we can achieve optimal time resolution.

The correlation matrix is then generated based on Eq. (5.7). It is important to note that unlike time correlation, frequency correlation is generally not zero. However, the

correlation diminishes rapidly, allowing us to approximate the correlation matrix $\mathbf{C}$ as a banded Toeplitz matrix. Consequently, the algorithm used for TP can also be applied to remove the frequency correlation.

5.4.3 Time-Frequency Prewhitening (TFP)

Unfortunately, it is not possible to remove the time-frequency correlation by considering TP and FP individually. The reason is that the time and frequency correlations are interlinked within the time-frequency correlation itself. As a result, they cannot be isolated into separate components that exclusively represent time or frequency correlations. Furthermore, the correlation matrix of the time-frequency correlation does not possess the specific structure seen in a banded Toeplitz matrix. Consequently, the algorithm we initially developed for TP is not applicable in this scenario. Investigating methods for efficiently calculating the inverse of the Cholesky decomposition of the time-frequency correlation matrix could be a potential future research. Here, we simply employ the standard method to address the time-frequency correlation.

The process of eliminating the time-frequency correlation is referred to as Time-Frequency Prewhitening (TFP). To achieve this, a sliding window approach is employed instead of prewhitening the entire STFT coefficients. More specifically, this method involves selecting a submatrix of $\mathbf{V}$ that is of interest for processing and subsequently sliding the window across different submatrices, as depicted in Fig. 5.5. The window is slid with a hop size denoted by R , where $1 \leq R \leq N_t$. We define $\mathbf{V}_s$ as the submatrix of $\mathbf{V}$ with dimensions $N_f \times N_t$, where N_f represents the number of frequency bins selected in the submatrix, and N_t denotes the number of time frames. Therefore, the whitening matrix $\mathbf{L}^{-1}$ is of size $N_f N_t \times N_f N_t$, and the condition $N_f N_t \ll K T_f$ holds, leading to a significant reduction in computational complexity and memory requirements.

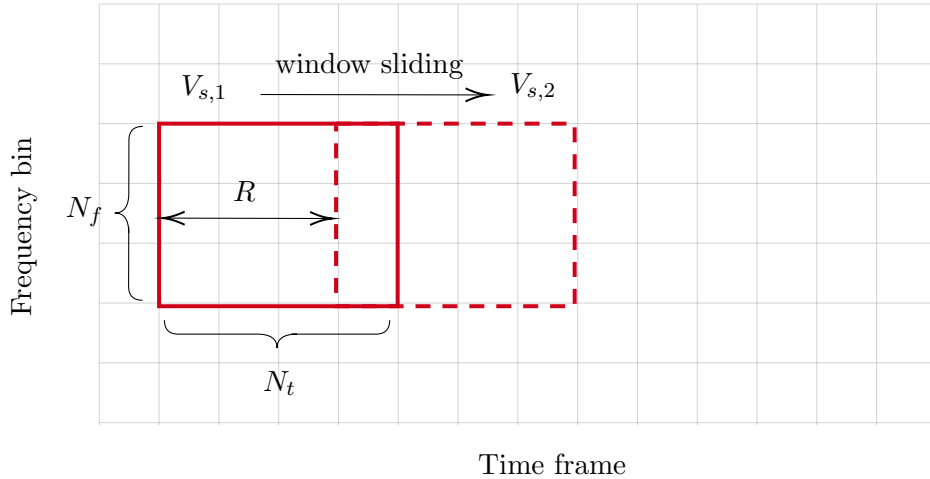


Figure 5.5: Sliding window processing diagram.

Additionally, it is important to note that the time-frequency correlation, as described in Eq. (5.10), is independent of the specific data being processed. Instead, it relies only on factors such as the window function, maximum time lag D , and frequency

difference δ , which are determined prior to removing the correlation. Consequently, it becomes possible to construct the correlation matrix in advance and compute the inverse of its Cholesky decomposition matrix. This calculation only needs to be performed once, after which we can store the resulting matrix to whiten every submatrix of STFT coefficients.

The prewhitening procedure described can be integrated into applications that also utilize a sliding window approach, such as tracking the rank over time. The size of $\mathbf{V}_s$ determines the trade-off between the frequency resolution and temporal resolution, and the choice of N_f and N_t depends on the rank tracking requirements and signal types. For signals that are not evenly distributed across frequency bins, it is more suitable to set $N_f = 1$ and track the rank for each individual frequency bin. Then, only the time correlation needs to be removed. However, setting $N_f = 1$ and having $N_t \gg 1$ results in perfect frequency resolution but relatively lower time resolution. Consequently, this can lead to suboptimal performance in rank tracking. In order to achieve high time resolution, we can reduce the value of N_t , but this also reduces the number of observations available. If we have an insufficient number of observations, accurate rank detection becomes challenging using threshold-based methods.

On the other hand, for certain types of signals are evenly distributed across all frequency bins (e.g., in the case of Gaussian sources) or exhibit energy concentration in specific frequency bins, we can sacrifice frequency resolution by increasing the number of frequency bins N_f to obtain more number of observations. However, it is not feasible to N_f to a very large number, due to the limitations from the signals characteristics and computational complexity.

5.5 Simulations

In the simulation section, we will first examine the largest eigenvalue distribution of the SCM of prewhitened noise after removing the correlation. Subsequently, we will utilize the threshold method to detect the rank in the STFT domain and assess the impact of correlation removal.

5.5.1 Eigenvalue Distribution Recheck

To assess the effectiveness of removing time-frequency correlation, we examine the CDF of the largest eigenvalue of the SCM of prewhitened noise. In this evaluation, we generate two independent random Gaussian white noise matrices, each with a size of 8×5000 , with 8 sensors and 5000 observations. These matrices are then transformed into the frequency domain using the STFT with parameters $K = L = 256$ and a hop size of $H = L/2$.

Next, we apply the time-frequency correlation removal method described in the previous section to the STFT coefficients of the two random matrices. Specifically, we set $N_f = 3$ and $N_t = 10$, and focus on the STFT coefficients corresponding to frequency bins 2, 3, 4, and the first 10 time frames. This results in a reshaped matrix of size 8×30 for each noise STFT coefficients. Let's denote the sizes of the reshaped matrices as $M \times N_a$ and $M \times N_b$, respectively.

We plot the empirical CDF of the largest eigenvalue of the SCM of the prewhitened noise, along with the theoretical Tracy-Widom distribution, in Fig. 5.6. As depicted in the figure, the empirical curve now exhibits a perfect fit with the theoretical curve, indicating that the time-frequency correlation has been effectively removed.

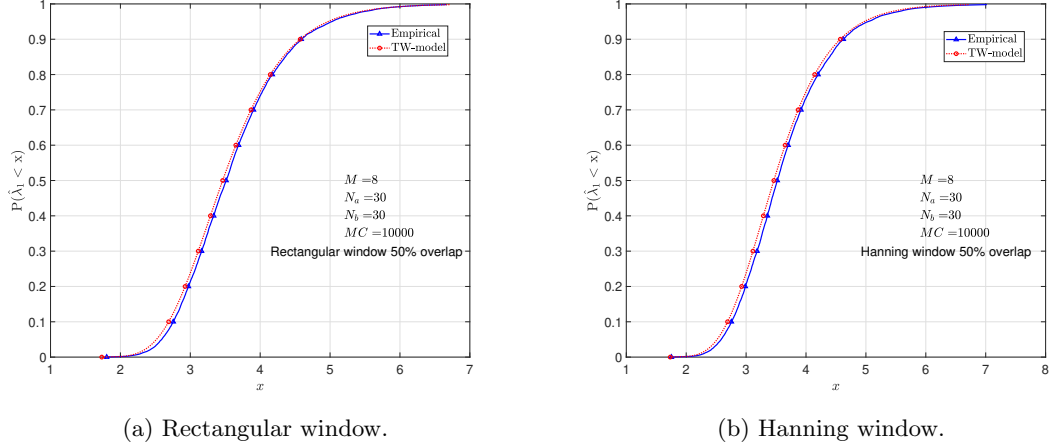


Figure 5.6: Empirical CDF of the largest eigenvalue of the SCM of prewhitened noise after TFP and CDF of the Tracy-Widom model for different window functions.

When selecting frequency bins for evaluating the empirical CDF of the largest eigenvalue, we should avoid including the first or last frequency bin. Including these bins would result in a mismatch between the empirical and theoretical curves. This discrepancy arises because the STFT coefficients at the first and last frequency bins are real numbers, and the Tracy-Widom distribution of complex numbers does not align with the data.

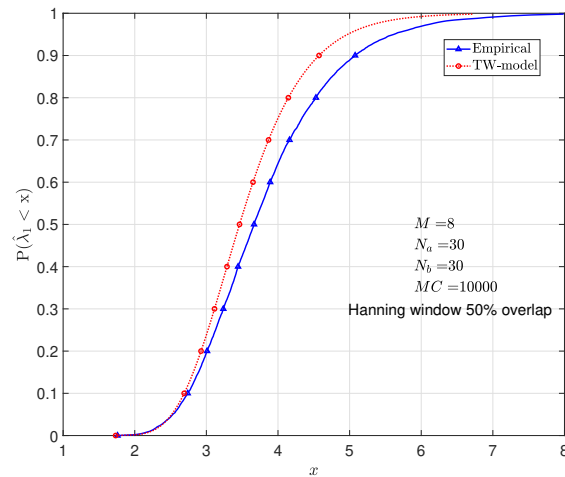


Figure 5.7: Empirical CDF of the largest eigenvalue of the SCM of prewhitened noise after TFP and CDF of the Tracy-Widom model when including the first frequency bin ($k = 0$).

To further emphasize this, let's examine Fig. 5.7, where we select frequency bins 1, 2, and 3. As observed in the figure, the two curves do not match, indicating a lack of fit between the empirical and theoretical distributions. Therefore, when utilizing multiple frequency bins, it is crucial to avoid including the first and last frequency bins to ensure the accuracy of the empirical CDF evaluation.

To evaluate the effectiveness of the efficient prewhitening algorithm for temporal prewhitening, we set $N_f = 1$ and $N_t = 30$. We focus on the frequency bin $k = 2$ and apply TP to the STFT coefficients of the two random matrices, both of size 8×30 .

In Fig. 5.8, we observe the effects of TP on the CDF of the largest eigenvalue. Comparing it with Fig. 5.2, we can see that after applying TP, the empirical curve now exhibits a perfect fit with the Tracy-Widom model for different window functions. This demonstrates the effectiveness of the algorithm in removing time correlations.

Additionally, we compare the computational efficiency of the standard prewhitening method ,i.e., using MATLAB built-in functions 'chol' and 'inv', with the modified prewhitening algorithm. The elapsed time in MATLAB for the standard method is measured to be 32.209 seconds, while the modified prewhitening algorithm takes 28.092 seconds. This indicates that the modified algorithm offers a slight improvement in computational efficiency.

When the number of time frames increases, the computational difference between the two methods also becomes more prominent. The modified algorithm may provide even more significant computational advantages in scenarios where a larger number of time frames are involved. This efficiency improvement can be beneficial, especially when dealing with larger data samples or real-time processing requirements.

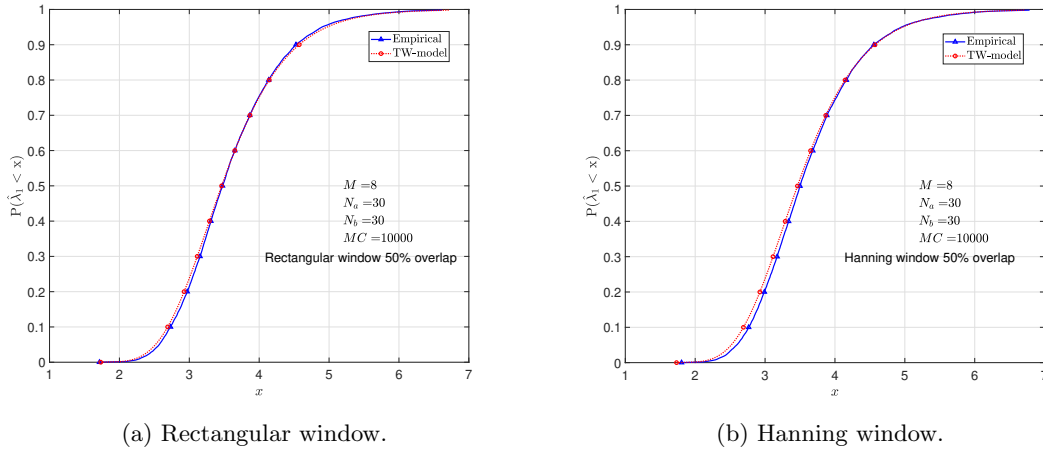


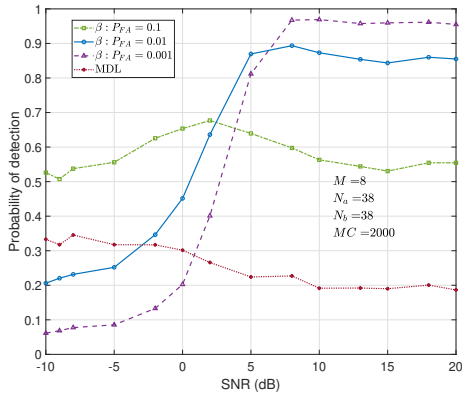
Figure 5.8: Empirical CDF of the largest eigenvalue of the SCM of prewhitened noise after TP and CDF of the Tracy-Widom model for different window functions at the same frequency bin ($k = 2$).

5.5.2 Rank Detection for Stationary Source

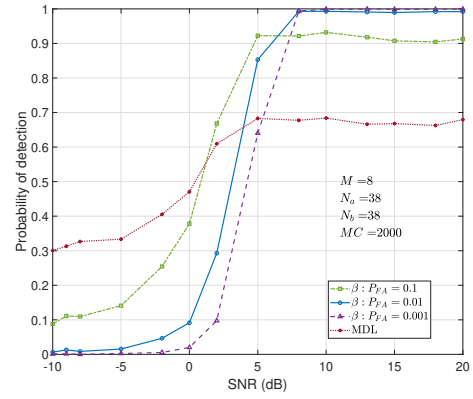
Next, we will analyze the impact of removing the correlations on the rank detection performance. We will begin by examining the case where the source remains stationary

over time. Consequently, it is enough to select STFT coefficients from a single frequency bin to detect the rank, and we only need to remove the time correlation.

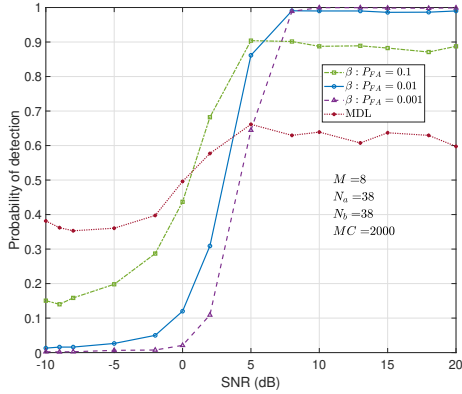
To simulate the scenario, we begin by generating a zero-mean Gaussian source matrix of rank d with 5000 observations, along with a Gaussian white noise matrix of size 8×5000 in the time domain. These matrices are then transformed using the STFT with parameters $k = L = 256$ and $H = L/2$. Next, we construct the noisy data matrix to the signal model defined in Eq. (2.1), where a random steering matrix $\mathbf{H}$ is involved. The sizes of the resulting noisy STFT coefficients and noise STFT coefficients are denoted as $M \times N_a$ and $M \times N_b$, respectively. The noise matrix then is used to prewhiten the noisy data matrix, and subsequently the sequential rank detector is applied to the prewhitened SCM.



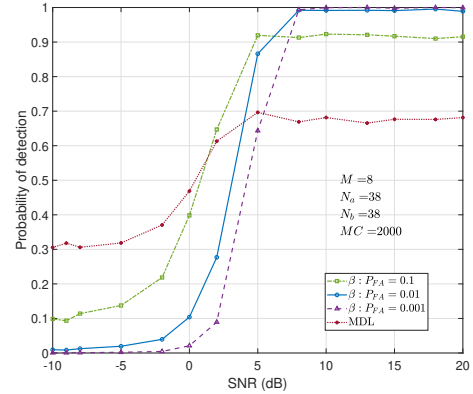
(a) Rectangular window without TP.



(b) Rectangular window with TP.



(c) Hanning window without TP.



(d) Hanning window with TP.

Figure 5.9: The probability of detection as a function of SNR for $d = 1$ in STFT domain at one frequency bin ($k = 2$).

Fig. 5.9 illustrates the probability of detection as a function of SNR when employing different window functions with and without TP. With the implementation of TP, the detection performance can reach the desired false alarm rate for various window func-

tions. Additionally, the MDL method also yields relatively reliable detection results. However, the absence of TP leads to degraded rank detection performance, particularly when using window functions with high time correlation, such as the rectangular window. The high time correlation results in a situation where the probability of detection cannot be controlled by P_{FA} we set for the rank detection algorithm.

5.5.3 Rank Tracking for Time-varying Source

Now, we will demonstrate the usage of the threshold method for rank tracking. In this simulation, the setup is similar to the previous experiments, but with the introduction of a time-varying Gaussian source that changes its rank every 5000 samples. This time-varying behavior is illustrated in Fig. 5.10.

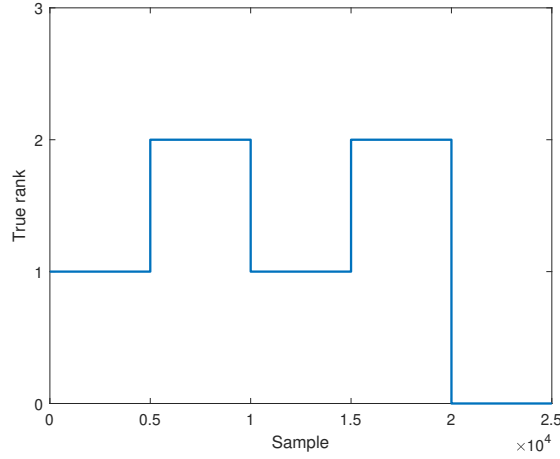


Figure 5.10: True rank of the generated signals in the time domain.

The STFT parameters remain the same as before, but here we use the Hanning window function. To track the rank, we employ a sliding window strategy on the STFT coefficients, as depicted in Figure 5.5. The false alarm rate P_{FA} for the rank detection algorithm is set to 1%. We consider two different settings:

- **Single-Frequency (SF) Setting:** In this setting, we detect the rank for each individual frequency bin. Since the Gaussian source is equally distributed across frequency bins, we simplify the analysis by selecting a single frequency bin, specifically $k = 2$. We set $N_f = 1$ and $N_t = 15$ to capture the rank information in the selected frequency bin. The hop size R is set equal to N_t . The resulting noisy data matrix is of size $M \times N_t$, which is 8×15 . The noise matrix has the same dimensions.
- **Multiple-Frequency (MF) Setting:** In this setting, we aim to improve time resolution by using multiple frequency bins. We choose $N_f = 3$ frequency bins, specifically bins 2, 3, and 4, and set $N_t = 5$ time frames for each rank detection. The hop size R is also set to N_t . The STFT coefficients at the selected time-frequency bins are then vectorized for every channel, resulting in the reshaped

noisy data matrix of size $M \times N_t N_f$, which is 8×15 . The noise matrix has the same dimensions. Additionally, we consider increasing N_f to 6, selecting frequency bins from $k = 2$ to $k = 7$. This expansion of the frequency range leads to a doubling of the number of observations. The resulting matrix has a size of 8×30 .

For the SF setting, TP is applied to remove time correlation, while for the MF setting, TFP is implemented to eliminate time-frequency correlation. The rank detection probability is presented in Fig. 5.11 as a function of the SNR, based on the results of 2000 Monte Carlo experiments. Notably, employing multiple frequency bins allows us to achieve higher detection accuracy compared to using a single frequency bin. When the number of frequency bins is doubled, the number of observations is also doubled, resulting in improved rank detection accuracy at low SNR values.

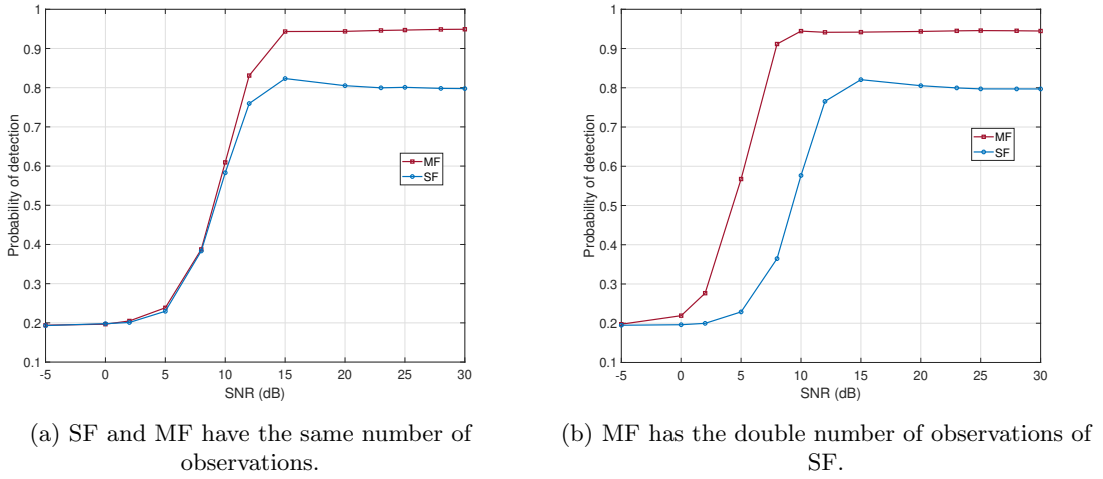


Figure 5.11: The probability of detection when the rank is changing over time.

However, it is observed that the detection performance does not reach the desired false alarm rate of 1%. This can be attributed to the presence of data samples from multiple sources within the detection window, near the points where the rank is shifting. As a result, the rank is overestimated near rank-shifting points, as illustrated in Fig. 5.12.

5.6 Chapter Conclusion

In this chapter, we investigated the correlation problems in STFT. The presence of time and frequency correlations within the STFT has a direct impact on the effectiveness of the threshold method. To address this, we derived expressions for time, frequency, and time-frequency correlations, which are then utilized to construct a correlation matrix for correlation removal. The removal of the correlation depends on the specific application. In scenarios where data is processed for each frequency bin across the entire time frame,

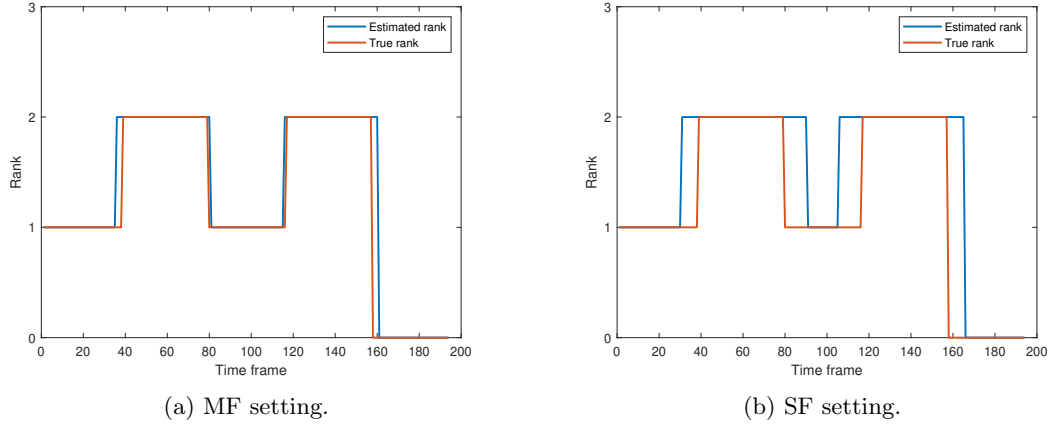


Figure 5.12: Rank tracking results of SF and MF settings at 30 dB SNR.

only time correlation needs to be considered. Further more, we have developed an efficient algorithm to eliminate time correlation.

Simulations conducted show that after removing the correlations, the empirical CDF of the largest eigenvalue of the prewhitened SCM aligns well with the Tracy-Widom model, and the detection rate can also achieve the desired false alarm rate. Additionally, we studied the threshold-based rank detection method for rank tracking problem. Our findings indicate that employing multiple frequency bins compensates for the limited observations when using shorter time frames. This approach provides higher time resolution, resulting in a better rank tracking performance compared to utilizing a single frequency bin alone.

Threshold-based Rank Detection Method for Speech Enhancement

6

In this chapter, we will explore the application of the threshold-based rank detection method in speech enhancement. To begin, we will introduce the multi-microphone signal model and the beamforming algorithm used in our study. Subsequently, we will discuss the implementation details of the threshold method and provide the simulations results of speech enhancement.

6.1 Multi-microphone Signal Model

Suppose we have M microphones placed in a closed room with sound absorbing walls, d speech targets, and some interference. The sound absorbing walls reduce the reflection coefficients, resulting in that the Room Impulse Response (RIR) only contains the direct path and reverberations can be ignored. In the time domain, the received signal at the j -th microphone is given by

$$x_j[t] = \sum_{i=1}^d h_{i,j}[t] * s_i[t] + n_j[t], \quad j = 1, 2, \dots, M \quad (6.1)$$

where $h_{i,j}$ is the acoustic RIR from the i -th source to the j -th microphone, s_i is the i -th target source, n_j is the noise signal received at the j -th microphone.

In order to satisfy the narrowband condition and exploit wide-sense stationary properties, we generally perform STFT to transform the data to frequency domain. By employing the STFT, we can simplify the computation by converting convolutions into multiplications. Let m and k denote as the time frame index and the frequency bin index. The STFT coefficients of the acoustic signal at the j -th microphone are then given by

$$\begin{aligned} X_j[m, k] &= \sum_{i=1}^d H_{i,j}[m, k] S_i[m, k] + N_j[m, k] \\ &= Y_j[m, k] + N_j[m, k]. \end{aligned} \quad (6.2)$$

Since we are going to process the signal for each time frame and frequency bin, the time and frequency indices therefore are dropped for ease of notation. Stacking the received signals from the microphones in a vector, we have the following signal model:

$$\begin{aligned} \mathbf{x} &= \mathbf{H}\mathbf{s} + \mathbf{n} \\ &= \mathbf{y} + \mathbf{n}, \end{aligned} \quad (6.3)$$

where $\mathbf{x} = [X_1, \dots, X_M]^T \in \mathbb{C}^M$ is the vector of the noisy speech STFT coefficients, $\mathbf{y} = [Y_1, \dots, Y_M]^T \in \mathbb{C}^M$ is the vector of received clean speech STFT coefficients, and

$\mathbf{n} = [N_1, \dots, N_M]^T \in \mathbb{C}^M$ is the vector of the noise STFT coefficients. $\mathbf{s} = [S_1, \dots, S_d]^T \in \mathbb{C}^d$ is the vector of target speech STFT coefficients. $\mathbf{H} = [\mathbf{h}_1, \dots, \mathbf{h}_d] \in \mathbb{C}^{M \times d}$ is the steering matrix, with $\mathbf{h}_i = [H_{i,1}, \dots, H_{i,M}]^T \in \mathbb{C}^M$ the steering vector for the i -th source. The elements in the steering vector are also called acoustic transfer functions (ATFs).

The ATF is commonly assumed to be time-invariant and frequency-dependent, and contains crucial information of the directions of the speech sources [26]. In multi-microphone speech enhancement applications, a reference microphone is generally required to output the estimated target speech signal, and the choice of the reference microphone could affect the performance of speech enhancement algorithms [27]. The ATFs can be modified into the relative transfer function (RTF) by normalizing all the ATFs with respect to the ATF at the reference position. Without loss of generality, the first microphone is considered as the reference microphone here. The RTF vector then is given by

$$\begin{aligned} \mathbf{h}_i &= \frac{1}{H_{i,1}} [H_{i,1}, H_{i,2}, \dots, H_{i,M}]^T \\ &= [1, \frac{H_{i,2}}{H_{i,1}}, \dots, \frac{H_{i,M}}{H_{i,1}}]^T, \end{aligned} \quad (6.4)$$

and the RTF matrix is given by stacking the RTF vectors together. We also use $\mathbf{H}$ to denote the RTF matrix.

In general, the target source $\mathbf{s}$ and the noise $\mathbf{n}$ are assumed to be realizations of zero-mean wide-sense stationary processes. Furthermore, they are assumed to be mutually uncorrelated for each time-frequency bin, meaning that their cross-covariance matrix is zero, i.e., $\mathbb{E}[\mathbf{sn}^H] = \mathbf{0}$. Based on this assumption, the covariance matrix of $\mathbf{x}$ is given by

$$\begin{aligned} \mathbf{R}_X &= \mathbb{E}[\mathbf{xx}^H] \\ &= \mathbb{E}[(\mathbf{Hs} + \mathbf{n})(\mathbf{Hs} + \mathbf{n})^H] \\ &= \mathbb{E}[\mathbf{Hss}^H \mathbf{H}^H] + \mathbb{E}[\mathbf{nn}^H] \\ &= \mathbf{H}\mathbf{R}_S \mathbf{H}^H + \mathbf{R}_N \\ &= \mathbf{R}_Y + \mathbf{R}_N, \end{aligned} \quad (6.5)$$

which is also the sum of the covariance matrix of the received target signals and the noise covariance matrix. All covariance matrices are complex matrices of size $M \times M$.

The covariance matrix of the received target sources is usually impossible to be known, while the noise covariance matrix or the noisy speech covariance could be estimated somehow. We can use the SCM to estimate it. More specifically, the covariance matrix is estimated by averaging over a certain number of the most recent time frames. For the noisy speech covariance matrix, the estimation is processed during the speech-plus-noise periods, and given by

$$\hat{\mathbf{R}}_X = \frac{1}{N_a} \sum_{i \in \mathcal{T}_x} \mathbf{x}[i] \mathbf{x}^H[i], \quad (6.6)$$

where $\mathcal{T}_x$ of size N_a is the set containing the time frame indices of noisy speech signals. Let $\mathbf{X}$ denote a size $M \times N_a$ matrix that contains the observed signals $\mathbf{x}[i], i \in \mathcal{T}_x$. The

SCM of the noisy speech can be expressed as

$$\hat{\mathbf{R}}_X = \frac{1}{N_a} \mathbf{X} \mathbf{X}^H \quad (6.7)$$

Similarly, the noise covariance matrix is estimated during the noise-only periods:

$$\hat{\mathbf{R}}_N = \frac{1}{N_b} \sum_{i \in \mathcal{T}_n} \mathbf{n}[i] \mathbf{n}^H[i], \quad (6.8)$$

where $\mathcal{T}_n$ of size N_b is the set containing the time frame indices of the pure noise. With $\mathbf{N}$ of size $M \times N_b$ containing the N_b observations of the noise, the SCM of noise can also be written as

$$\hat{\mathbf{R}}_N = \frac{1}{N_b} \mathbf{N} \mathbf{N}^H. \quad (6.9)$$

To distinguish between speech-plus-noise periods and noise-only periods, the voice activity detector (VAD) plays a crucial role in the system. In our simulations, we assume a perfect VAD, which is obtained using the Audio Labeling tool in MATLAB.

6.2 Beamforming

Speech enhancement can be achieved by applying a beamformer to the STFT coefficients of the noisy speech in each time-frequency bin. This process, commonly known as beamforming, aims to improve the quality of the speech signal. The output of the beamformer is obtained by combining the individual microphone signals using specific beamforming weights, which is given by

$$\hat{Y}_1 = \mathbf{w}^H \mathbf{x}, \quad (6.10)$$

where $\mathbf{w}$ is the beamforming vector of length M , $\hat{Y}_1$ is the estimated clean speech signal received at the reference microphone.

Various types of beamformers are utilized in speech enhancement applications. For example, the Minimum Variance Distortionless Response (MVDR) beamformer [28] reduces noise power while maintaining the spectral characteristics of the desired signal. Another beamforming technique is the Linear Constraints Minimum Variance (LCMV) beamformer [29], which not only minimizes noise variance but also incorporates additional constraints to enhance interference suppression. These beamformers are relatively straightforward to implement, but they require information such as ATFs or DoAs, which need to be determined prior to applying the beamforming algorithms.

Signal subspace-based beamformers offer an alternative approach to speech enhancement without requiring prior information such as DoAs, ATFs, or array structure [30, 31, 32, 33]. These beamforming algorithms exploit the principle that the signal subspace can be decomposed into two subspaces occupied by the clean signal and the noise respectively. This decomposition can be done through EVD and its generalized version. When the noise sources are spatially uncorrelated and limited to microphone self-noise, EVD can be used to denoise the speech signals. However, in the presence

of correlated background noise across the microphones, GEVD is employed, combining the prewhitening technique and EVD to reduce the correlated noise. A general beamformer based on GEVD is called the signal-distortion weighted (SDW) Wiener filter, which balances the trade-off between the signal distortion and noise reduction performance [30]. A low-rank approximation of the SDW Wiener filter was developed and proved to have better performance of noise reduction without increasing the signal distortion [34].

Here, we consider using the GEVD-based beamformers for speech enhancement, since in the threshold method, the GEVD is also used to obtain the generalized eigenvalues for the rank detection. We will show the connections between the GEVD-based beamformers and the threshold method, and see how to combine them together.

6.2.1 GEVD-based Beamformers

First, we will review the GEVD-based beamformers and show the derivation of the optimal beamforming weights $\mathbf{w}$. The first step is to perform GEVD on the matrix pencil $(\mathbf{R}_Y, \mathbf{R}_N)$, with $\mathbf{R}_N \succ 0$:

$$\mathbf{F}^H \mathbf{R}_Y \mathbf{F} = \mathbf{\Lambda}, \quad \mathbf{F}^H \mathbf{R}_N \mathbf{F} = \mathbf{I}_M, \quad (6.11)$$

where $\mathbf{R}_Y, \mathbf{R}_N \in \mathbb{C}^{M \times M}$, $\mathbf{F} \in \mathbb{C}^{M \times M}$ are the right generalized eigenvectors, and $\mathbf{\Lambda} = \text{diag}\{\lambda_1, \dots, \lambda_m\} \in \mathbb{R}^{M \times M}$ are generalized eigenvalues. Let $\mathbf{K} = \mathbf{F}^{-H}$. We have

$$\mathbf{R}_Y = \mathbf{K} \mathbf{\Lambda} \mathbf{K}^H, \quad \mathbf{R}_N = \mathbf{K} \mathbf{I}_M \mathbf{K}^H. \quad (6.12)$$

The pair $(\mathbf{\Lambda}, \mathbf{F})$ are also the eigenvalues and eigenvectors of the matrix $\mathbf{R}_N^{-1} \mathbf{R}_Y$. According to Eq. (6.5), the covariance matrix $\mathbf{R}_X$ can be expressed as

$$\begin{aligned} \mathbf{R}_X &= \mathbf{R}_Y + \mathbf{R}_N \\ &= \mathbf{K} \mathbf{\Lambda} \mathbf{K}^H + \mathbf{K} \mathbf{I}_M \mathbf{K}^H \\ &= \mathbf{K} (\mathbf{\Lambda} + \mathbf{I}_M) \mathbf{K}^H. \end{aligned} \quad (6.13)$$

This implies that if $(\mathbf{\Lambda}, \mathbf{F})$ is the eigenpair of the matrix pencil $(\mathbf{R}_Y, \mathbf{R}_N)$, $(\mathbf{\Lambda} + \mathbf{I}_M, \mathbf{F})$ is then the eigenpair of the matrix pencil $(\mathbf{R}_X, \mathbf{R}_N)$. Since accessing the covariance matrix $\mathbf{R}_Y$ is often challenging, we typically work with the matrix pencil $(\mathbf{R}_X, \mathbf{R}_N)$ instead. By subtracting one from the generalized eigenvalues of the matrix pencil $(\mathbf{R}_X, \mathbf{R}_N)$, we obtain the generalized eigenvalues $\mathbf{\Lambda}$.

The next step is to find the optimal beamforming weights $\mathbf{w}$, which is achieved by minimizing the mean squared error (MSE) between the beamformer output and the received target signal at the reference microphone (the first microphone). With the assumption $\mathbb{E}[\mathbf{y}\mathbf{n}^H] = \mathbf{0}$, we have

$$\begin{aligned} \mathbb{E} |\mathbf{w}^H \mathbf{x} - y_1|^2 &= \mathbb{E} |\mathbf{w}^H \mathbf{y} + \mathbf{w}^H \mathbf{n} - y_1|^2 \\ &= \mathbb{E} |\mathbf{w}^H \mathbf{y} - y_1|^2 + \mathbb{E} |\mathbf{w}^H \mathbf{n}|^2, \end{aligned} \quad (6.14)$$

where the first term represents the signal distortion and the second term is the residual noise variance. We can make compromise between signal distortion and noise reduction

by minimizing the first term and constraining the noise variance. This leads us to the following optimization problem [30, 34]:

$$\begin{aligned} \min \quad & \mathbb{E} |\mathbf{w}^H \mathbf{y} - y_1|^2 \\ \text{s.t.} \quad & \mathbb{E} |\mathbf{w}^H \mathbf{n}|^2 \leq c, \end{aligned} \quad (6.15)$$

where c is the user chosen parameter, with $0 \leq c \leq \sigma_{n_1}^2$. $\sigma_{n_1}^2$ is the noise power at the reference microphone before applying the beamforming technique.

The optimal weights can be found by using the Lagrangian function and taking its derivative to zero, which is given by

$$\mathbf{w} = (\mathbf{R}_Y + \mu \mathbf{R}_N)^{-1} \mathbf{R}_Y \mathbf{e}_1, \quad (6.16)$$

where $\mathbf{e}_1 = [1, 0, \dots, 0]^T \in \mathbb{R}^M$ indicates the position of the reference microphone, and μ is the trade-off parameter for the signal distortion and noise reduction. Hence, these filters are commonly referred to as SDW Wiener filters. By utilizing GEVD expressions of the covariance matrices, we can express the optimal filters as follows:

$$\mathbf{w} = \mathbf{F}(\mathbf{\Lambda} + \mu \mathbf{I}_M)^{-1} \mathbf{\Lambda} \mathbf{K}^H. \quad (6.17)$$

It is often assumed that the covariance matrix $\mathbf{R}_Y$ exhibits low rank, meaning that the number of sources d is smaller than the number of microphones M . By considering this property, we can choose the first d eigenvectors and eigenvalues. Consequently, the beamformer can be expressed as follows:

$$\mathbf{w} = \mathbf{F}_d(\mathbf{\Lambda}_d + \mu \mathbf{I}_d)^{-1} \mathbf{\Lambda}_d \mathbf{K}_d^H, \quad (6.18)$$

where $\mathbf{F}_d$ contains the first d right generalized eigenvectors, $\mathbf{K}_d$ contains the first d left generalized eigenvectors. $\mathbf{\Lambda}_d$ is the diagonal matrix containing the first eigenvalues. The filters are called the low-rank multi-channel Wiener filter (LR-MWF) [35]. Note that many beamforming can be expressed as Eq. (6.18). With $\mu = 0, d = 1$, the beamformer is considered as the MVDR beamformer. When $\mu = 1, d = M$, it is the classical multichannel Wiener filter.

6.2.2 Beamforming After Temporal Prewhitening

To implement the GEVD-based beamformer for speech enhancement, it is crucial to determine the number of sources or the rank of each time-frequency bin. While the maximum rank d_{max} may be known or assumed, the rank of individual time-frequency bins is not always equal to d_{max} since sources are not always simultaneously active at all time-frequency bins [36]. To estimate the rank of each bin, the threshold method can be employed. However, before applying the threshold method, it is necessary to preprocess the collected data matrix $\mathbf{X}$ by removing time correlation introduced by the STFT. This can be achieved through temporal prewhitening, as discussed in the previous chapter.

In the threshold method, the GEVD is employed to obtain generalized eigenvalues for rank detection. It is also utilized in constructing the beamformer. Typically, two

separate GEVDs need to be performed. However, we have discovered that performing only one GEVD on the preprocessed data matrix is sufficient. Let $\mathbf{X}'$ denote the preprocessed noisy speech matrix, and $\mathbf{N}'$ the preprocessed noisy matrix. The temporal prewhitening is done by

$$\mathbf{X}' = \mathbf{X}\mathbf{L}^{-T}, \mathbf{N}' = \mathbf{N}\mathbf{L}^{-T}, \quad (6.19)$$

where $\mathbf{L}$ is the Cholesky decomposition of the time correlation matrix.

By performing the GEVD on the preprocessed matrix pencil $(\hat{\mathbf{R}}_{X'}, \hat{\mathbf{R}}_{N'})$, we can determine the estimated rank as well as obtain the optimal beamformer $\mathbf{w}'$. The output of the beamformer will be the corresponding estimated target signals with removed time correlation at the reference microphone. To recover the original STFT coefficients, the estimated signal matrix should be multiplied by $\mathbf{L}^T$. This operation is equivalent to applying the beamformer to the original data matrix (i.e., the data matrix before TP). TP only eliminates time correlation, ensuring that the observations become i.i.d. The spatial information in the data matrix is preserved, allowing the beamformer constructed from the GEVD on the preprocessed matrix pencil to effectively reduce noise.

As a result, performing only one GEVD on the data matrix preprocessed by TP is sufficient to obtain the beamformer and estimate the rank.

6.3 Implementation Detail

Fig. 6.1 illustrates the process of speech enhancement using our threshold-based rank detection method for each time-frequency bin. Initially, the multichannel noisy speech signals in the time domain are transformed into the frequency domain using the STFT. For each frequency bin, the data matrix $\mathbf{X}$ undergoes TP to eliminate the time correlation. The VAD is then applied to classify the speech-plus-noise and noise-only time frames, which are utilized to estimate the covariance matrices. The number of samples used for estimating the noisy speech and noise covariance matrices are N_a and N_b respectively.

Next, the GEVD is performed on the matrix pencil $(\hat{\mathbf{R}}_{X'}, \hat{\mathbf{R}}_{N'})$ to obtain the generalized eigenvectors $\mathbf{F}$ and eigenvalues $\hat{\lambda}_i$, for $i = 1, \dots, M$. The rank $\hat{d}$ is estimated using the threshold method based on the matrix dimensions (M, N_a, N_b) and the generalized eigenvalues. The beamformer takes the original noisy STFT coefficients $\mathbf{X}$, eigenvectors, and the estimated rank as inputs, producing the estimated clean speech STFT coefficients $\hat{Y}_1$ at the reference microphone. The estimated clean speech signal in the time domain can then be obtained by applying the inverse STFT.

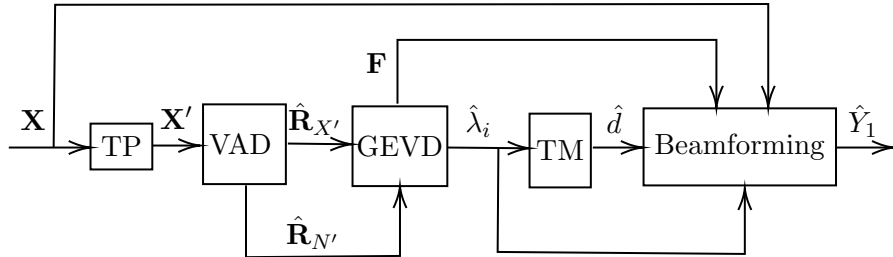


Figure 6.1: Implementation diagram of threshold method in speech enhancement.

In this implementation, two important remarks are worth considering. Firstly, **the assumption of a prefect VAD is essential**. One might question the necessity of using VAD when we can utilize the threshold method to directly detect noise-only time-frequency bins. Why do we need VAD in this context? It is important to note that the threshold method relies on accurate estimation of the pure noise covariance matrix and the data covariance matrix. The noise covariance matrix plays a critical role and requires precise estimation.

If the threshold method misclassifies a data time-frequency bin as a noise sample and includes it in the estimation of the noise covariance matrix, the effectiveness of the prewhitening process in the threshold method may be compromised, and the noise eigenvalues may not follow the Tracy-Widom distribution. Ultimately, this can result in the breakdown of the threshold method. Therefore, we employ VAD to ensure the accurate estimation of the noise covariance matrix. Investigating the problem of rank estimation without relying on VAD in the threshold method could be an interesting topic for future research.

Secondly, **the ghost sources need to be considered**. For the speech signal model, the spectral leakage in STFT would introduce spatially spread and uncalibrated ghost sources. This leads to the situation where the number of eigenvalues greater than the noise threshold is higher than the actual rank [37]. As a consequence, the threshold method may provide incorrect estimations and overestimate the rank. This phenomenon becomes more prominent at high SNR, as the power of ghost sources increases proportionally with the source power [38]. Window functions with lower side lobes in the frequency response are more resistant to spectral leakage and can mitigate the impact of ghost sources [22].

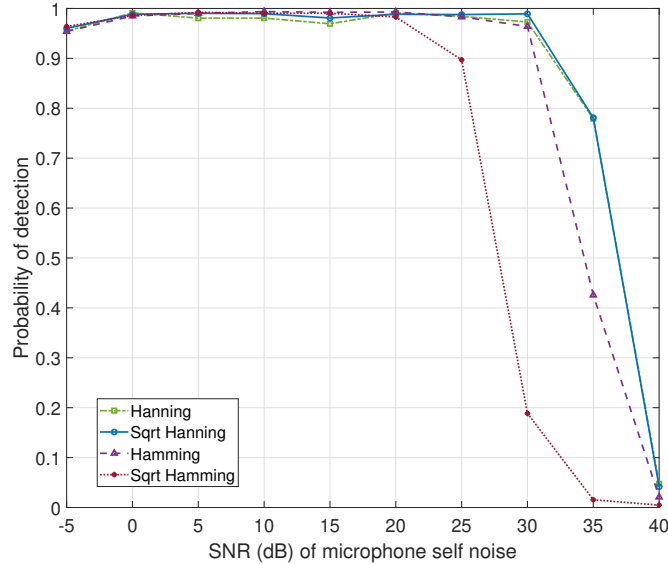


Figure 6.2: Probability of detection as a function of SNR of microphone self-noise for different window functions.

To demonstrate the influence of ghost sources on the threshold method's perfor-

mance, we conducted a simple simulation. We consider a scenario where a Gaussian source with rank 1 is corrupted by Gaussian white noise. By simulating RIRs [39] using the image method [40] and convolving the source signals with the RIRs, we obtain the received signals at the microphone array. Subsequently, the STFT is applied to the received signals and TP is implemented to eliminate the time correlation. Finally, we employ the threshold method on the processed STFT coefficients to estimate the rank. In Fig. 6.2, we present the probability of detection for different window functions. It is evident that at extremely high SNR (>30 dB), the threshold method may encounter difficulties in accurately detecting the rank. The performance of the square-root Hamming window is particularly sensitive to the SNR due to its relatively high side lobe and limited effectiveness in reducing spectral leakage.

To avoid the effect of the ghost sources, we set a relatively low SNR (20 dB) for the microphone self noise in the following simulations. To fully address the issue of ghost sources is beyond the scope of this thesis. Some related methods can be found in [37, 38].

6.4 Simulations

6.4.1 Simulation Setup

The room geometry we consider for the speech enhancement simulations is depicted in Fig. 6.3. The room has dimensions of $5m \times 4m \times 4m$ and all microphones and sources are placed at a height of $z = 2m$. We employ a linear microphone array consisting of 6 microphones evenly spaced 2 cm apart. The leftmost microphone serves as the reference microphone. Our experiments involve the presence of one or two target sources, along with two noise interferences.

The parameters used in the simulations are summarized in Table 6.1. To generate the received speech signals, the target and noise sources are convolved in the time domain using RIRs generated with the image method [40], implemented using the MATLAB toolbox [39]. The microphones are omnidirectional with a 0-degree orientation, and their sensor self-noise is set to 20 dB. In order to ensure a low-rank approximation of the covariance matrix, a reverberation time T_{60} of 10 ms is employed. The sampling rate is set to $f_s = 16$ kHz. For processing the received signals in the frequency domain, the STFT is performed using a square-root Hanning window of length $L = 512$, with a hop size of $H = L/2$, and an FFT point of $K = 512$. For the rank detection based on the threshold method, the false alarm rate P_{FA} is set to 1%. When it comes to the beamformers, we set the trade-off parameter μ to 1.5.

The two target speech signals consist of recordings from a male and a female speaker, obtained from the TIMIT dataset [41]. Both signals have a sampling rate of 16 kHz. Two types of background noise are considered: Gaussian noise and babble noise. To evaluate the performance of the threshold method in detecting the rank, the true rank $d_{m,k}$ in each time-frequency bin must be known. For this purpose, we assume a perfect VAD for each target speech signal separately. The perfect VAD is obtained using the Audio Labeling tool in MATLAB. The speech signals, along with the corresponding VAD labels, are depicted in Fig. 6.4.

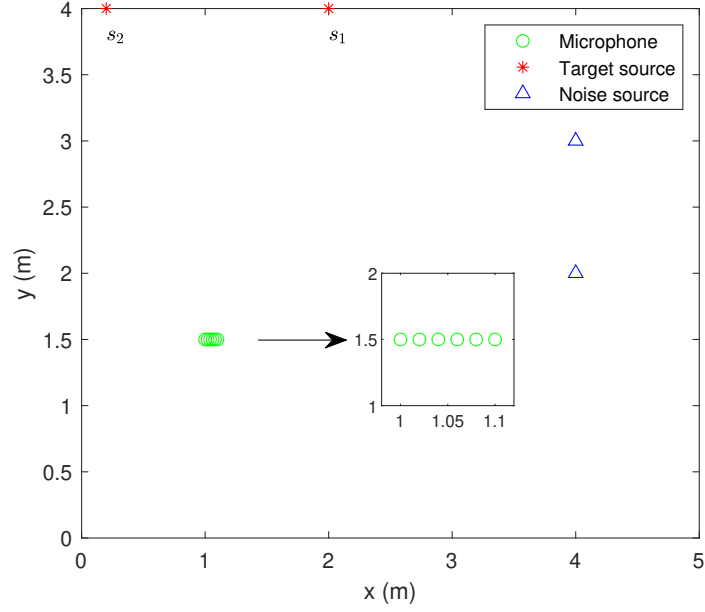
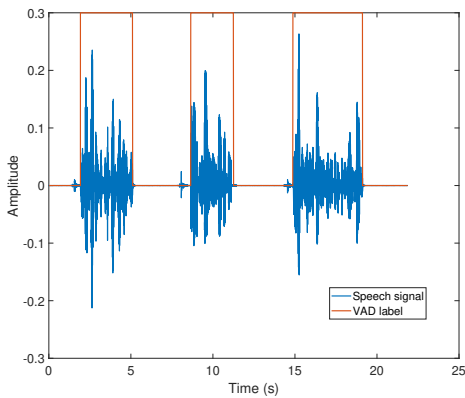


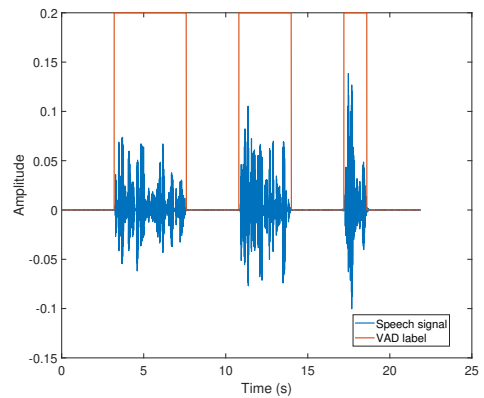
Figure 6.3: The room geometry for simulations with $M = 6$ microphones and $d = 2$ sources.

Table 6.1: Simulation parameters

Parameter	Value
The number of microphones M	6
Simulation realizations	10
Sampling rate f_s	16KHz
Microphone self noise	20 dB
FFT point K	512
Time frame length L	512
Window function	Square-root Hanning window
Hop size H	256
The reverberation time T_{60}	10 ms
False alarm rate of threshold method P_{FA}	1%
Beamforming trade-off parameter μ	1.5



(a) Speech signal s_1 of a female.



(b) Speech signal s_2 of a male.

Figure 6.4: Plots of speech signals and their VAD labels used in the simulations.

In Fig. 6.5, the spectrograms of the clean speech signals are depicted. It is notable that the speech signals align well with the VAD across all time frames. However, the distribution of the speech signals is not uniform across all frequency bins. Specifically, certain frequency bins, such as $k < 10$ and $k > 200$, exhibit relatively low signal power. This non-uniform distribution poses challenges in accurately determining the true rank for each time-frequency bin. Given this uncertainty in rank determination, evaluating the accuracy of rank detection using our proposed method may not yield meaningful results. For simplicity, we focus on assessing the rank detection accuracy for a frequency bin that exhibits higher signal power, such as the bin at $k = 20$. In this specific frequency bin, we can determine the true rank for each target speech signal in every time-frequency bin based on the VAD results. In the case of multiple sources, the true rank is obtained by summing the ranks of each individual source.

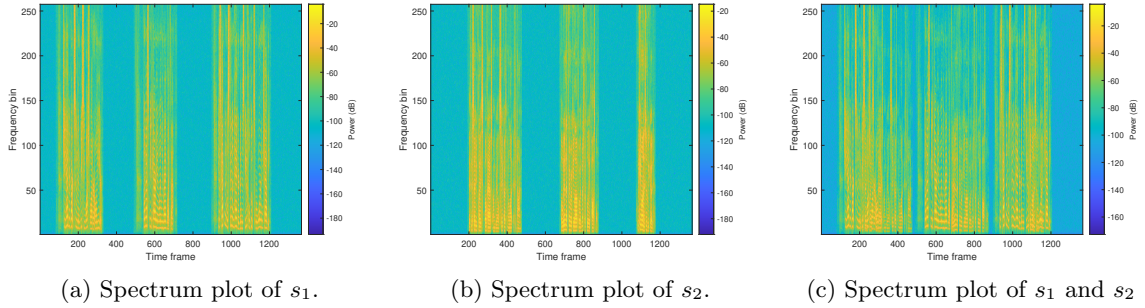


Figure 6.5: Spectrum plots of target speech signals at the reference microphone, when the noise is not presented.

The speech enhancement process is implemented based on the diagram shown in Fig. 6.1. To estimate the covariance matrices, the noisy speech covariance matrix and the noise covariance matrix, we employ the SCM approach, as described in Eq. (6.7) and Eq. (6.9). When applying the threshold method for speech enhancement, it is essential to take into account that the STFT coefficients at the first and last frequency bins are real numbers. Therefore, it is advisable to use the Tracy-Widom model specifically designed for the real case in such situations. The corresponding expressions and mathematical details can be found in Appendix A.

6.4.2 Evaluation Method

To assess the effectiveness of the threshold method in speech enhancement, we consider three evaluation metrics: the probability of rank detection, the output SNR, and the Short-Time Objective Intelligibility (STOI). These metrics are computed over the input SNR, which is defined as the ratio of the power of the target source to the power of the noise in the time domain at the reference microphone:

$$\text{Input SNR} = 10 \log \frac{\sum_{i=1}^N y_1^2[i]}{\sum_{i=1}^N n_1^2[i]}. \quad (6.20)$$

To measure the rank detection accuracy of the threshold method for each time-frequency bin, we focus on computing the probability of detection within the speech-present time-frequency bins. Let $\mathcal{D} = \{d_{m,k} : d_{m,k} \neq 0\}$ represent the set containing the true ranks for speech-present time-frequency bins. The set $\hat{\mathcal{D}} = \{\hat{d}_{m,k} : \hat{d}_{m,k} = d_{m,k} \wedge d_{m,k} \neq 0\}$ contains the correctly estimated ranks. The probability of detection is then computed as the ratio of the sizes of these two sets:

$$\text{Probability of detection} = \frac{|\hat{\mathcal{D}}|}{|\mathcal{D}|}. \quad (6.21)$$

The output SNR and STOI are selected as performance metrics to evaluate the effectiveness of speech enhancement. The output SNR is calculated as the ratio of the power of the target source to the remaining noise power after enhancement, which is given by

$$\text{Output SNR} = 10 \log \frac{\sum_{i=1}^N y_1^2[i]}{\sum_{i=1}^N (y_1[i] - \hat{y}_1[i])^2}. \quad (6.22)$$

The STOI takes inputs of clean and enhanced speech signals and measures the speech intelligibility [42]. It produces a value between -1 and 1, where a value closer to 1 indicates higher intelligibility. However, it is important to note that the STOI value can approach 1 even in scenarios where the noise level is relatively high.

In order to evaluate the performance of the speech enhancement, we compare the results obtained using the rank estimated by the threshold method with those obtained using the maximum rank d_{max} . The maximum rank corresponds to the known number of sources.

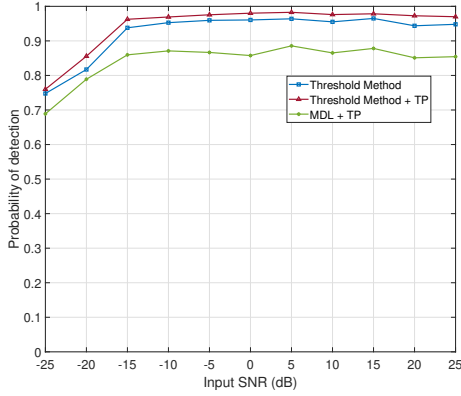
6.4.3 Rank Detection Accuracy of Threshold Method

To assess the rank detection accuracy of the threshold method, we examine the detection results for one target source and two target sources at frequency bin $k = 20$, as shown in Fig. 6.6. The figures depict the averaged detection results over 100 experiments. In Fig. 6.7, the rank estimation results are compared with the true rank at frequency bin 20, with an input SNR of 25 dB.

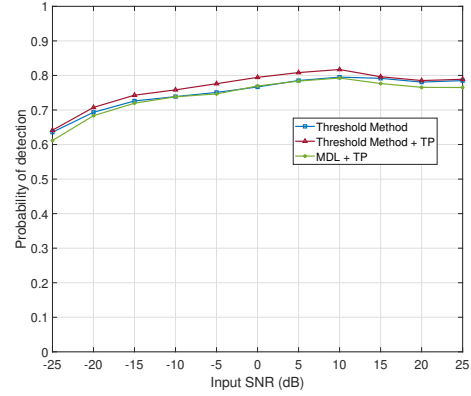
Fig. 6.6a demonstrates that when there is a single target source, the detection performance can almost achieve the desired false alarm rate after applying TP. However, some incorrect estimations may occur for the first time-frequency bins at the beginning of the detection process, as seen in Fig. 6.7a. This could be attributed to insufficient data samples, resulting in relatively less accurate results.

The application of TP improves the detection accuracy, albeit to a small extent, as the time correlation introduced by the square-root Hanning window with 50% overlap is relatively low. Moreover, the MDL, when combined with TP, achieves a high accuracy of approximately 87% at high SNR. However, it is important to note that the MDL method only functions properly when the number of data samples exceeds $M = 6$.

However, the rank detection accuracy declines rapidly when dealing with multiple sources, as depicted in Fig. 6.6b. The threshold method tends to make mistakes at

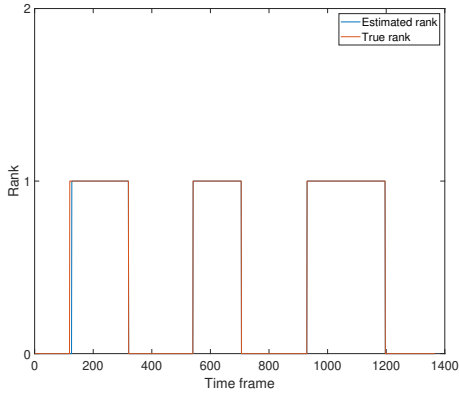


(a) Detection probability for $N_a = N_b = 100, d = 1$.

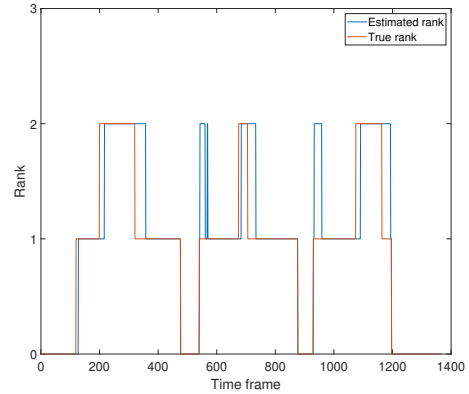


(b) Detection probability for $N_a = 40, N_b = 100, d = 2$.

Figure 6.6: Rank detection probability using threshold method for (a) one target source and (b) two target sources.



(a) Rank estimation results for $N_a = N_b = 100, d = 1$.



(b) Rank estimation results for $N_a = 40, N_b = 100, d = 2$.

Figure 6.7: Rank estimation results for one or two target sources at frequency bin 20, input SNR 25 dB.

the boundaries when the rank is shifting. For instance, when the rank transitions from 2 to 1, the threshold method may still estimate the rank as 2 for the consecutive time-frequency bins. This discrepancy might occur because the collected data samples exhibit a higher number of time-frequency bins with rank 2 compared to rank 1.

Furthermore, the threshold method tends to overestimate the rank when the transition occurs from 0 to 1, as observed at time frames around 550 and 950. This phenomenon can be attributed to the presence of two sources in the collected data samples.

To improve the accuracy of rank detection, one approach is to use a smaller number of data samples, which corresponds to a smaller detection window size. This allows for better time resolution. For instance, we can use $N_a = 20$ signal data samples. The

results of rank detection accuracy are shown in Fig. 6.8, and the corresponding rank estimation results are displayed in Fig. 6.9.

Using a smaller number of data samples can indeed lead to higher detection accuracy at high SNR levels but lower accuracy at low SNR levels. However, this approach may also result in noisier estimations and an increased number of underestimations due to the limitations of estimating the covariance matrix with small data samples. In the context of speech enhancement, underestimation of the rank is more problematic than overestimation [36]. Therefore, while using smaller data samples can improve rank detection accuracy at high SNR, it may not be favorable for overall speech enhancement performance.

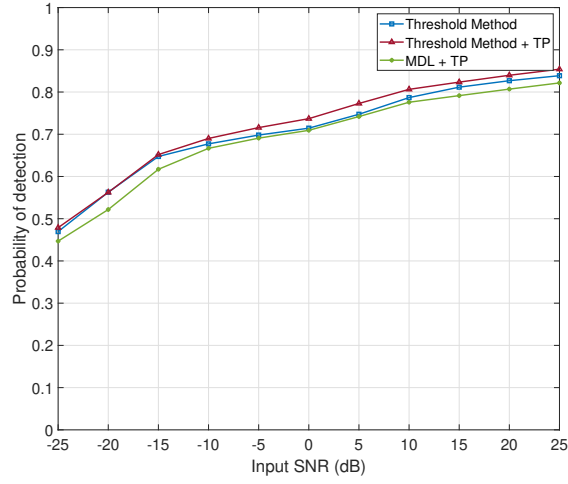


Figure 6.8: Detection probability for two target sources at frequency bin 20, input SNR 25 dB, $N_a = 20$, $N_b = 100$.

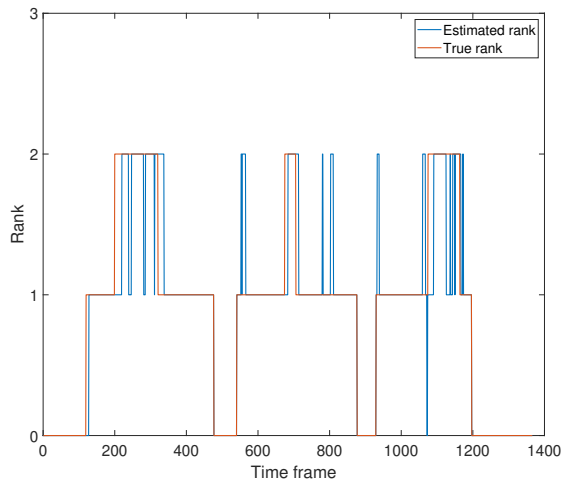


Figure 6.9: Rank estimation results for two target sources at frequency bin 20, input SNR 25 dB, $N_a = 20$, $N_b = 100$.

6.4.4 Speech Enhancement Performance

First, we will examine the speech enhancement performance using the threshold method for a single target source. Fig. 6.10 illustrates the changes in output SNR and STOI as a function of input SNR. It is observed that employing the estimated rank obtained from the threshold method results in a slight improvement in both output SNR and STOI. This improvement is particularly noticeable at low SNR levels. Although the threshold method may initially make some mistakes in rank estimation, these errors have minimal impact on the overall speech enhancement performance. Furthermore, in certain time-frequency bins with low signal power, the threshold method may estimate a rank of 0, thereby excluding these bins from the enhancement process. Surprisingly, disregarding these low-power bins does not significantly affect the speech enhancement performance. On the other hand, when using the maximum rank, more noise components may be included, resulting in lower output SNR and STOI.

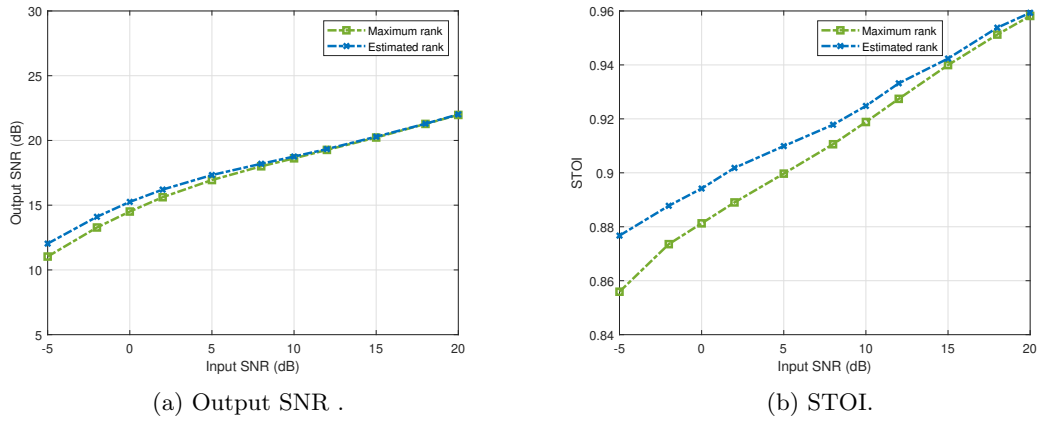
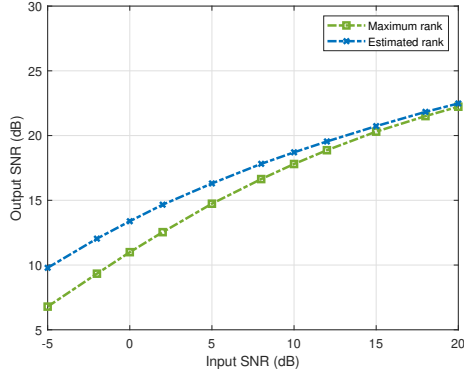
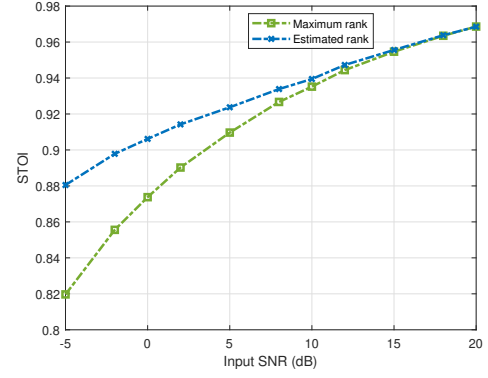


Figure 6.10: Speech enhancement performance using the maximum rank and the estimated rank of the threshold method, for one target source.

Similar results can be observed for the case of two target sources, as shown in Fig. 6.11. However, it is worth noting that the discrepancy between the curves of the true maximum rank and the estimated rank becomes more pronounced at low input SNR levels. This phenomenon may be attributed to the fact that employing the maximum rank includes more noise components for time-frequency bins with actual lower ranks. Conversely, despite the potential errors made by the threshold method in certain time-frequency bins, it still manages to give relatively better speech enhancement performance.



(a) Output SNR



(b) STOI

Figure 6.11: Speech enhancement performance using the maximum rank and the estimated rank of the threshold method, for two target sources.

6.5 Chapter Conclusion

In this chapter, we implemented the threshold-based rank detection method in speech enhancement to estimate the rank of each time-frequency bin. The threshold method shows promising results in speech enhancement, both for single and multiple target sources. Despite potential errors in rank detection at the beginning and in certain time-frequency bins, the threshold method still leads to improved output SNR and STOI compared to using the true maximum rank.

Conclusion and Future Work

7.1 Conclusion

Recalling the main research question posed in Chapter 1, our objective was to achieve precise rank detection based on the threshold method in the presence of colored or correlated noise while considering the finite number of data samples and noise samples. Through our research, we have taken several steps to address this question and provide solutions.

Firstly, we investigated the distribution of the largest eigenvalue of the prewhitened noise and discovered that it follows the Tracy-Widom distribution. This finding allowed us to determine the threshold necessary to achieve a specific false alarm rate. Furthermore, we presented a more accurate threshold when considering the signal-plus-noise case. We made modifications to account for the matrix dimensions when calculating the scaling and centering constants.

Building upon the threshold determination, we developed a rank detecting algorithm using a sequential test approach. Through theoretical analysis and simulations, we demonstrated that our algorithm achieves the desired false alarm rate and more accurate rank detection performance and compared to MDL and AIC methods. Importantly, our proposed method is more robust when dealing with limited data samples and noise samples.

Considering that many real-world signals are non-stationary and often analyzed in the STFT domain, we extended our rank detection method to this domain. However, the correlations among the time-frequency bins introduced by STFT impact the distribution of prewhitened noise. To address this, we investigated methods to remove the correlations and derived expressions for the correlations in STFT. To handle signals that are not evenly distributed across frequency bins, such as speech and audio signals, we proposed a temporal prewhitening technique. Additionally, we developed a more efficient algorithm that considers the special structure of the correlation matrix for achieving temporal prewhitening. Through simulations, we demonstrated that by removing the correlation, the threshold-based rank detection method can achieve the desired false alarm rate.

Finally, we applied the rank detection method to speech enhancement, specifically detecting the rank in every time-frequency bin. Our results showed that the rank detection method integrates well with GEVD-based beamformers, as both rely on the GEVD. By employing our method, we observed improvements in output SNR and STOI compared to using the known number of signals as the rank. Even in cases where the rank detection method made mistakes on some time-frequency bins, it still excluded more noise components than using the number of signals as the rank.

7.2 Future Work

Based on the work presented in this thesis, future studies can benefit from the following ideas, aimed at further improving the current work, overcoming limitations, and extending the proposed method to other applications. These ideas can be categorized into two aspects: theoretical advancements and practical applications.

7.2.1 Theoretical Advancements

The proof for the distribution of the largest eigenvalue arising from the noise in the signal-plus-noise scenario needs improvement. In our case, we simplified the prewhitened noise as Gaussian white noise and relied on an existing proof for that assumption. While the empirical results support the correctness of the distribution, it is crucial to provide a more explicit proof to strengthen the argument.

Similarly, in the performance analysis of the rank detection algorithm, we introduced modifications to the distribution of the eigenvalue originating from the signal. Although simulations were conducted to assess the effectiveness of these modifications, it is essential to provide theoretical proof to establish the validity of the modifications.

The determination of the threshold in the rank detection algorithm can be further improved by using adaptive thresholding techniques. Based on dynamic variations in noise statistics, the threshold can be automatically adjusted to adapt to changing noise conditions and improve the overall performance of the rank detection method.

7.2.2 Practical Applications

In the STFT domain, for applications that involve processing multiple frequency bins and time frames simultaneously, it is necessary to perform time-frequency prewhitening to remove time-frequency correlation. Currently, we employ a sliding window approach to mitigate computation and memory issues. However, more efficient algorithms should be investigated to effectively remove time-frequency correlation.

For applications that exhibit signal models similar to the speech signal model, the presence of ghost sources introduced by spectral leakage in STFT can affect the rank decision of the rank detection method. To mitigate the effects, we currently use a relatively small SNR for spatially uncorrelated noise. To fully resolve this issue, we can refer to works such as [37, 38] for inspiration.

Additionally, in speech enhancement applications, we assume the VAD to help us select speech-present and noise-only time frames. An advanced approach would involve using our threshold method to detect voice activity, eliminating the need for a separate VAD and allowing us to estimate the rank for every time-frequency bin more efficiently.

The threshold method developed for rank detection has potential applications beyond speech enhancement. By exploring its effectiveness and adaptability in other domains, such as DoA estimation in radar systems, imaging denoising, and other applications that require rank estimation, we can discover new way to use it and further expand its potential impact.

Tracy-Widom Distribution in the Real Case



The largest eigenvalue of the Gaussian white noise in the real case can be approximated by the order 1 Tracy-Widom [14]:

$$\frac{N_a \hat{\lambda}_1 - \rho_{M, N_a}}{v_{M, N_a}} \xrightarrow{\mathcal{D}} \mathcal{TW}_1, \quad (\text{A.1})$$

with centering and scaling constants as

$$\begin{aligned} \rho_{M, N_a} &= \left(\sqrt{N_a - \frac{1}{2}} + \sqrt{M - \frac{1}{2}} \right)^2, \\ v_{M, N_a} &= \sqrt{\rho_{M, N_a}} \left(\frac{1}{\sqrt{N_a - \frac{1}{2}}} + \frac{1}{\sqrt{M - \frac{1}{2}}} \right)^{\frac{1}{3}}. \end{aligned}$$

For the prewhitening case, the logarithm of the largest eigenvalue $\hat{\lambda}_1$ of the real F type matrix $\hat{\mathbf{R}}_B^{-1} \hat{\mathbf{R}}_A$ can be approximated to order $O(n^{-2/3})$ by the Tracy-Widom law with proper centering and scaling [18]:

$$\frac{\ln \left(\frac{N_a}{N_b} \hat{\lambda}_1 \right) - \rho_{M, N_a, N_b}}{v_{M, N_a, N_b}} \xrightarrow{\mathcal{D}} \mathcal{TW}_1. \quad (\text{A.2})$$

The centering and scaling constants are given as follows:

$$\rho_{M, N_a, N_b} = 2 \ln \tan \frac{\gamma + \phi}{2}, \quad (\text{A.3})$$

$$v_{M, N_a, N_b}^3 = \frac{16}{(N_a + N_b - 1)^2} \frac{1}{\sin^2(\gamma + \phi) \sin(\gamma) \sin(\phi)}, \quad (\text{A.4})$$

where

$$\begin{aligned} \sin^2\left(\frac{\gamma}{2}\right) &= \frac{\min(N_a, M) - 1/2}{N_a + N_b - 1}, \\ \sin^2\left(\frac{\phi}{2}\right) &= \frac{\max(N_a, M) - 1/2}{N_a + N_b - 1}. \end{aligned}$$

Temporal Prewhitening Algorithm

B

B.1 General Temporal Prewhitening

Let c_τ denote $\mathbb{E}[V_m[k]V_{m+\tau}^*[k]]$. The time correlation matrix, denoted as $\mathbf{C}$, is a banded Toeplitz matrix with dimensions $T_f \times T_f$. This matrix captures the correlations between $V_m[k]$ values for $1 \leq m \leq T_f$. The structure of $\mathbf{C}$ is as follows:

$$\mathbf{C} = \begin{bmatrix} c_0 & \cdots & c_D & & \mathbf{0} \\ \vdots & \ddots & & \ddots & \\ c_D^* & & \ddots & & \\ & \ddots & & \ddots & c_D \\ \mathbf{0} & & & c_D^* & \cdots & c_0 \end{bmatrix} \quad (\text{B.1})$$

The time correlation matrix $\mathbf{C}$ can also be interpreted as the autocorrelation matrix of the sequence $V_m[k]$. Let $\mathbf{v}_k \in \mathbb{C}^{T_f \times 1}$ denote the STFT coefficients at frequency bin k . To remove the time correlation, we use $\mathbf{C}^{-1/2}$ to whiten the vector $\mathbf{v}_k$ along the time axis. That is

$$\mathbf{v}'_k = \mathbf{C}^{-1/2} \mathbf{v}_k \quad (\text{B.2})$$

where $\mathbf{v}'_k$ represents the prewhitened STFT coefficients at frequency bin k , with its elements being uncorrelated across time. $\mathbf{C}^{1/2}$ can be obtained by the Cholesky decomposition of $\mathbf{C}$, i.e., $\mathbf{C} = \mathbf{L}\mathbf{L}^H$, where $\mathbf{L}$ is a lower triangular matrix. For simplicity, we can omit the notation of the frequency bin. Hence, we have

$$\mathbf{v}' = \mathbf{L}^{-1} \mathbf{v} \quad (\text{B.3})$$

In general, obtaining the prewhitened vector $\mathbf{v}'$ involves calculating the Cholesky decomposition of matrix $\mathbf{C}$, which has a time complexity of $O(n^3)$. Additionally, computing the inverse of the lower triangular matrix $\mathbf{L}$ also has a time complexity of $O(n^3)$. However, using MATLAB's 'chol' and 'inv' functions would require recomputation from scratch for each update of the data vector, making it the least time-efficient approach. To address this, we will now present a more efficient algorithm for computing the Cholesky decomposition and its inverse.

B.2 Cholesky-Forward-Substitution Algorithm

First, we consider how to improve the efficiency of the Cholesky decomposition of the matrix $\mathbf{C}$. The general Cholesky decomposition algorithm is given by [43]

$$\begin{aligned} L_{i,i} &= \sqrt{C_{i,i} - \sum_{n=1}^{i-1} L_{i,n} L_{i,n}^*} \\ L_{j,i} &= \frac{1}{L_{i,i}} (C_{j,i} - \sum_{n=1}^{i-1} L_{j,n} L_{i,n}^*), j = i + 1, \dots, T_f \end{aligned} \quad (\text{B.4})$$

Since the $\mathbf{C}$ matrix is a banded Toeplitz matrix, the resulting Cholesky matrix $\mathbf{L}$ is a banded lower triangular matrix and has the following structure:

$$\mathbf{L} = \begin{bmatrix} L_{1,1} & & & & \\ \vdots & & & & \\ L_{D+1,1} & & \ddots & & \\ & \ddots & & & \\ \mathbf{0} & & L_{T_f, T_f-D} & \cdots & L_{T_f, T_f} \end{bmatrix} \quad (\text{B.5})$$

where each row contains a maximum of $D + 1$ non-zero entries. By considering this characteristic, we can effectively reduce complexity. Additionally, we can utilize a vector representation for the matrix $\mathbf{C}$. Let $\mathbf{c} = [c_0, \dots, c_D]^T$ be the correlation vector. The correlation matrix $\mathbf{C}$ can then be expressed as $\mathbf{C} = \text{Toeplitz}(\mathbf{c}, \mathbf{0}_{(T_f-D-1) \times 1})$, where $\mathbf{0}_{(T_f-D-1) \times 1}$ represents a zero-vector of length $T_f - D - 1$. Rather than constructing the entire matrix, we can store only the correlation vector, resulting in significant memory savings. The modified Cholesky decomposition algorithm is as follows:

$$\begin{aligned} L_{i,i} &= \sqrt{c_0 - \sum_{n=\max(1, i-D)}^{i-1} L_{i,n} L_{i,n}^*} \\ L_{j,i} &= \frac{1}{L_{i,i}} (c_{j-i}^* - \sum_{n=\max(1, j-D)}^{i-1} L_{j,n} L_{i,n}^*), j = i + 1, \dots, \min(i + D, T_f) \end{aligned} \quad (\text{B.6})$$

Then, we can effectively reduce the time complexity to approximately $O(D^2 n)$, with $D \ll n$.

Once we have obtained the Cholesky matrix $\mathbf{L}$, the next step is to calculate its inverse. Rather than directly computing the inverse, we can instead solve the linear equation:

$$\mathbf{L} \mathbf{v}' = \mathbf{v} \quad (\text{B.7})$$

The forward substitution algorithm is employed in this case, which typically has a complexity of $O(n^2)$. However, due to the special structure of $\mathbf{L}$, the complexity can

be reduced to $O(Dn)$. The modified algorithm is derived from the original forward substitution algorithm [44], and is presented as follows:

$$\begin{aligned} v'_1 &= \frac{v_1}{L_{1,1}} \\ v'_i &= \frac{1}{L_{i,i}} \left(v_i - \sum_{n=\max(1, i-D)}^{i-1} L_{i,n} v'_n \right), \quad i > 1 \end{aligned} \quad (\text{B.8})$$

where v_i represents the i -th element in the vector $\mathbf{v}$, v'_i is the i -th element in the vector $\mathbf{v}'$.

Since, we combine the Cholesky decomposition and the forward substitution steps, the algorithm is called the Cholesky-Forward-Substitution algorithm. However, in its current form, the algorithm is not adaptive. For each new data sample, it requires recalculating the Cholesky matrix $\mathbf{L}$ and solving the linear equation from scratch. In order to address this limitation, we will explore making the algorithm adaptive. We will start by considering a special case when $D = 1$.

B.2.1 Algorithm for $D = 1$

In the case when $D = 1$ (i.e., using less than or equal to 50% overlap), the correlation vector can be represented as $\mathbf{c} = [c_0, c_1]^T$. Explicit expressions can be derived for the elements of the Cholesky matrix $\mathbf{L}$, and the time complexity for computing $\mathbf{L}$ will be $O(n)$.

- If $c_0 = 2|c_1|$, it indicates the usage of rectangular window function. The derivation in this case is straightforward by using mathematical induction. The expressions of the entries in $\mathbf{L}$ are given by

$$\begin{aligned} L_{1,1} &= \sqrt{c_0}, \\ L_{i,i} &= \sqrt{\frac{c_0(i+1)}{2i}}, \\ L_{i,i-1} &= \frac{c_1}{L_{i,i}}, \quad \text{for } i = 2, \dots, T_f \end{aligned} \quad (\text{B.9})$$

- If $c_0 > 2|c_1|$, it indicates the usage of other window functions. The entries in $\mathbf{L}$ can be computed as [45]

$$\begin{aligned} L_{1,1} &= \sqrt{c_0}, \\ L_{i,i} &= \sqrt{\frac{\alpha_1^{i+1} - \alpha_2^{i+1}}{\alpha_1^i - \alpha_2^i}}, \\ L_{i,i-1} &= \frac{c_1}{L_{i,i}}, \quad \text{for } i = 2, \dots, T_f \end{aligned} \quad (\text{B.10})$$

where $\alpha_1 = \frac{c_0 + \sqrt{c_0^2 - 4c_1^2}}{2}$, $\alpha_2 = \frac{c_0 - \sqrt{c_0^2 - 4c_1^2}}{2}$

In both cases, the entries of the Cholesky matrix $\mathbf{L}$ depend on the position index, enabling efficient adaptive updates. For instance, when incorporating a new data sample and updating the matrix $\mathbf{L}$, it is unnecessary to recalculate all entries from the beginning. Instead, only the entries L_{T_f+1, T_f+1} and L_{T_f+1, T_f} need to be computed, resulting in a time complexity of just $O(1)$. This significantly reduces the computational complexity and improves efficiency.

The forward substitution algorithm can be further reduced to

$$\begin{aligned} v'_1 &= \frac{v_1}{L_{1,1}} \\ v'_i &= \frac{1}{L_{i,i}}(v_i - L_{i,i-1}v'_{i-1}), \quad i > 1 \end{aligned} \tag{B.11}$$

and the time complexity is also $O(1)$ for a single update.

For multiple channels of inputs, the algorithm is required to be performed independently for each channel.

B.2.2 Algorithm for $D > 1$

For general cases where $D > 1$, the Cholesky decomposition algorithm in Eq. (B.6) can also be written in a recursive way, resulting in a complexity of $O(D^2)$ for updating a new incoming sample.

Consider the scenario where we have knowledge of the previous t rows of the $\mathbf{L}$ matrix, where $t \geq D + 1$. Our objective is to determine the $t + 1$ -st row of the $\mathbf{L}$ matrix. Specifically, there will be $D + 1$ elements in the $t + 1$ -st row denoted by $L_{t+1, t-D+1}, \dots, L_{t+1, t+1}$. Based on Eq. (B.6), we now show the calculation of every element in the $t + 1$ -st row.

First, $L_{t+1, t-D+1}$ is calculated based on $L_{t-D+1, t-D+1}$:

$$\begin{bmatrix} \cdots & \times & L_{t-D+1, t-D+1} & 0 & \cdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \cdots & 0 & L_{t+1, t-D+1} & \cdots & L_{t+1, t+1} \end{bmatrix} \tag{B.12}$$

$L_{t+1, t-D+2}$ is determined by the elements $L_{t-D+2, t-D+1}$, $L_{t-D+2, t-D+2}$ and $L_{t+1, t-D+1}$:

$$\begin{bmatrix} \cdots & \times & L_{t-D+1, t-D+1} & 0 & \cdots & \cdots \\ \cdots & \times & L_{t-D+2, t-D+1} & L_{t-D+2, t-D+2} & 0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \cdots & 0 & L_{t+1, t-D+1} & L_{t+1, t-D+2} & \cdots & L_{t+1, t+1} \end{bmatrix} \tag{B.13}$$

Continuing the process, $L_{t+1, t}$ then is determined by:

$$\begin{bmatrix}
\cdots & \times & L_{t-D+1,t-D+1} & 0 & \cdots & \cdots & \cdots \\
\cdots & \times & L_{t-D+2,t-D+1} & L_{t-D+2,t-D+2} & 0 & \cdots & \cdots \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
\cdots & \times & L_{t,t-D+1} & \cdots & L_{t,t-1} & L_{t,t} & 0 \\
\cdots & 0 & L_{t+1,t-D+1} & \cdots & L_{t+1,t-1} & L_{t+1,t} & L_{t+1,t+1}
\end{bmatrix} \quad (\text{B.14})$$

For the last element, we have

$$\left[\cdots \quad 0 \quad L_{t+1,t-D+1} \quad \cdots \quad L_{t+1,t-1} \quad L_{t+1,t} \quad L_{t+1,t+1} \right] \quad (\text{B.15})$$

Now, it becomes evident that in order to determine the $t+1$ -st row, we require the previous lower triangular matrix of size $D \times D$, denoted as $\mathbf{L}[t-D+1:t, t-D+1:t]$. Thus, for a single update, the time complexity is approximately $O(D^2)$.

$$\begin{bmatrix}
\cdots & \times & L_{t-D+1,t-D+1} & 0 & \cdots & \cdots & \cdots \\
\cdots & \times & L_{t-D+2,t-D+1} & L_{t-D+2,t-D+2} & 0 & \cdots & \cdots \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
\cdots & \times & L_{t,t-D+1} & \cdots & L_{t,t-1} & L_{t,t} & 0 \\
\cdots & 0 & L_{t+1,t-D+1} & \cdots & L_{t+1,t-1} & L_{t+1,t} & L_{t+1,t+1}
\end{bmatrix} \quad (\text{B.16})$$

After obtaining the elements in the $t+1$ -st row of $\mathbf{L}$, we can utilize these elements and the previous prewhitened data samples to calculate the prewhitened data sample at time index $t+1$, denoted as v'_{t+1} , based on Eq. (B.8). Let $\mathbf{L}_t = \mathbf{L}[t-D+1:t, t-D+1:t]$. Given $\mathbf{L}_t$, v_{t+1} , and the previous prewhitened data vector $\mathbf{g}_t = [v'_{t-D+1}, \dots, v'_t]^T$, our goal is to determine v'_{t+1} . The algorithm is summarized as follows:

Algorithm 2 Adaptive Cholesky-Forward-Substitution Algorithm

Input: Correlation vector $\mathbf{c}$, $\mathbf{L}_t$, v_{t+1} , and previous $\mathbf{g}_t$

Initialization: $\mathbf{l}_{t+1} = \mathbf{0}_{(D+1) \times 1}$ to store the non-zero elements in $t+1$ -st row of $\mathbf{L}$

for $j = 1 : D$ **do**

$$l_j = \frac{1}{L_t[j,j]} (c_{D-j+1}^* - \sum_{n=1}^{j-1} l_n L_t^*[j, n])$$

end for

$$l_{D+1} = \sqrt{c_0 - \sum_{n=1}^D l_n l_n^*}$$

$$v'_{t+1} = \frac{1}{l_{D+1}} (v_{t+1} - \sum_{n=1}^D l_n g_t[n])$$

Update the new submatrix $\mathbf{L}_{t+1}$

Update the prewhitened vector $\mathbf{g}_{t+1}$

Output: v'_{t+1}

Bibliography

- [1] R. Roy and T. Kailath, “Esprit-estimation of signal parameters via rotational invariance techniques,” *IEEE Transactions on acoustics, speech, and signal processing*, vol. 37, no. 7, pp. 984–995, 1989.
- [2] R. O. Schmidt, *A signal subspace approach to multiple emitter location and spectral estimation*. Stanford University, 1982.
- [3] M. Wax and T. Kailath, “Detection of signals by information theoretic criteria,” *IEEE Transactions on acoustics, speech, and signal processing*, vol. 33, no. 2, pp. 387–392, 1985.
- [4] D. Williams, “Comparison of aic and mdl to the minimum probability of error criterion,” in *[1992] IEEE Sixth SP Workshop on Statistical Signal and Array Processing*, pp. 114–117, IEEE, 1992.
- [5] M. Chiani and M. Z. Win, “Estimating the number of signals observed by multiple sensors,” in *2010 2nd International Workshop on Cognitive Information Processing*, pp. 156–161, IEEE, 2010.
- [6] E. Fishler and H. Messer, “On the use of order statistics for improved detection of signals by the mdl criterion,” *IEEE Transactions on Signal Processing*, vol. 48, no. 8, pp. 2242–2247, 2000.
- [7] S. Kritchman and B. Nadler, “Non-parametric detection of the number of signals: Hypothesis testing and random matrix theory,” *IEEE Transactions on Signal Processing*, vol. 57, no. 10, pp. 3930–3941, 2009.
- [8] Z. Zhu, S. Haykin, and X. Huang, “Estimating the number of signals using reference noise samples,” *IEEE transactions on aerospace and electronic systems*, vol. 27, no. 3, pp. 575–579, 1991.
- [9] L. Zhao, P. R. Krishnaiah, and Z. Bai, “On detection of the number of signals when the noise covariance matrix is arbitrary,” *Journal of Multivariate Analysis*, vol. 20, no. 1, pp. 26–49, 1986.
- [10] G. Vazquez-Vilar, D. Ramírez, R. López-Valcarce, J. Vía, and I. Santamaría, “Spatial rank estimation in cognitive radio networks with uncalibrated multiple antennas,” in *Proceedings of the 4th International Conference on Cognitive Radio and Advanced Spectrum Management*, pp. 1–5, 2011.
- [11] G. H. Golub and C. F. Van Loan, *Matrix computations*. JHU press, 2013.
- [12] Z.-D. Bai and Y.-Q. Yin, “Limit of the smallest eigenvalue of a large dimensional sample covariance matrix,” in *Advances In Statistics*, pp. 108–127, World Scientific, 2008.

- [13] I. M. Johnstone, “On the distribution of the largest eigenvalue in principal components analysis,” *The Annals of statistics*, vol. 29, no. 2, pp. 295–327, 2001.
- [14] M. Chiani, “Distribution of the largest eigenvalue for real wishart and gaussian random matrices and a simple approximation for the tracy–widom distribution,” *Journal of Multivariate Analysis*, vol. 129, pp. 69–81, 2014.
- [15] M. Zhou and A.-J. van der Veen, “Blind separation of partially overlapping data packets,” *Digital Signal Processing*, vol. 68, pp. 154–166, 2017.
- [16] A.-J. van der Veen, J. Romme, and Y. Cui, “Rank detection thresholds for hankel or toeplitz data matrices,” in *2020 28th European Signal Processing Conference (EUSIPCO)*, pp. 1911–1915, IEEE, 2021.
- [17] R. R. Nadakuditi and J. W. Silverstein, “Fundamental limit of sample generalized eigenvalue based detection of signals in noise using relatively few signal-bearing and noise-only samples,” *IEEE Journal of selected topics in Signal Processing*, vol. 4, no. 3, pp. 468–480, 2010.
- [18] I. M. Johnstone, “Multivariate analysis and jacobi ensembles: Largest eigenvalue, tracy–widom limits and rates of convergence,” *Annals of statistics*, vol. 36, no. 6, p. 2638, 2008.
- [19] X. Han, G. Pan, and B. Zhang, “The tracy–widom law for the largest eigenvalue of f type matrices,” 2016.
- [20] P. Dharmawansa, I. M. Johnstone, and A. Onatski, “Local asymptotic normality of the spectrum of high-dimensional spiked f-ratios,” *arXiv preprint arXiv:1411.3875*, 2014.
- [21] D. Lawley, “Tests of significance for the latent roots of covariance and correlation matrices,” *biometrika*, vol. 43, no. 1/2, pp. 128–136, 1956.
- [22] G. Heinzel, A. Rüdiger, and R. Schilling, “Spectrum and spectral density estimation by the discrete fourier transform (dft), including a comprehensive list of window functions and some new at-top windows,” 2002.
- [23] F. Millioz and N. Martin, “Estimation of a white gaussian noise in the short time fourier transform based on the spectral kurtosis of the minimal statistics: Application to underwater noise,” in *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 5638–5641, IEEE, 2010.
- [24] G. Parisi, A. Coluccia, and A. Fascista, “On time-frequency correlation in spectrogram samples with application to target detection,” *Signal Processing*, vol. 200, p. 108648, 2022.
- [25] F. J. Harris, “On the use of windows for harmonic analysis with the discrete fourier transform,” *Proceedings of the IEEE*, vol. 66, no. 1, pp. 51–83, 1978.

- [26] X. Li, R. Horaud, L. Girin, and S. Gannot, “Local relative transfer function for sound source localization,” in *2015 23rd European Signal Processing Conference (EUSIPCO)*, pp. 399–403, IEEE, 2015.
- [27] J. Zhang, H. Chen, L.-R. Dai, and R. C. Hendriks, “A study on reference microphone selection for multi-microphone speech enhancement,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 671–683, 2020.
- [28] M. Souden, J. Benesty, and S. Affes, “A study of the lcmv and mvdr noise reduction filters,” *IEEE Transactions on Signal Processing*, vol. 58, no. 9, pp. 4925–4935, 2010.
- [29] O. L. Frost, “An algorithm for linearly constrained adaptive array processing,” *Proceedings of the IEEE*, vol. 60, no. 8, pp. 926–935, 1972.
- [30] Y. Ephraim and H. L. Van Trees, “A signal subspace approach for speech enhancement,” *IEEE Transactions on speech and audio processing*, vol. 3, no. 4, pp. 251–266, 1995.
- [31] S. Doclo and M. Moonen, “Gsvd-based optimal filtering for single and multimicrophone speech enhancement,” *IEEE Transactions on signal processing*, vol. 50, no. 9, pp. 2230–2244, 2002.
- [32] Y. Hu and P. C. Loizou, “A generalized subspace approach for enhancing speech corrupted by colored noise,” *IEEE transactions on speech and audio processing*, vol. 11, no. 4, pp. 334–341, 2003.
- [33] S. Markovich, S. Gannot, and I. Cohen, “Multichannel eigenspace beamforming in a reverberant noisy environment with multiple interfering speech signals,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 6, pp. 1071–1086, 2009.
- [34] J. R. Jensen, J. Benesty, and M. G. Christensen, “Noise reduction with optimal variable span linear filters,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 4, pp. 631–644, 2015.
- [35] R. Serizel, M. Moonen, B. Van Dijk, and J. Wouters, “Low-rank approximation based multichannel wiener filter algorithms for noise reduction with application in cochlear implants,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 4, pp. 785–799, 2014.
- [36] A. I. Koutrouvelis, R. C. Hendriks, R. Heusdens, and J. Jensen, “Estimation of sensor array signal model parameters using factor analysis,” in *2019 27th European Signal Processing Conference (EUSIPCO)*, pp. 1–5, IEEE, 2019.
- [37] E. D. Di Claudio, R. Parisi, and G. Jacovitti, “Space time music: Consistent signal subspace estimation for wideband sensor arrays,” *IEEE Transactions on Signal Processing*, vol. 66, no. 10, pp. 2685–2699, 2018.

- [38] E. D. Di Claudio and G. Jacovitti, “Wideband source localization by space-time music subspace estimation,” in *2013 8th International Symposium on Image and Signal Processing and Analysis (ISPA)*, pp. 331–336, IEEE, 2013.
- [39] E. Habets, “RIR Generator.” <https://www.audiolabs-erlangen.de/fau/professor/habets/software/rir-generator>, 2020. Accessed: 2023-5-29.
- [40] J. B. Allen and D. A. Berkley, “Image method for efficiently simulating small-room acoustics,” *The Journal of the Acoustical Society of America*, vol. 65, no. 4, pp. 943–950, 1979.
- [41] J. S. Garofolo, “Timit acoustic phonetic continuous speech corpus,” *Linguistic Data Consortium, 1993*, 1993.
- [42] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, “An algorithm for intelligibility prediction of time–frequency weighted noisy speech,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 7, pp. 2125–2136, 2011.
- [43] A. Krishnamoorthy and D. Menon, “Matrix inversion using cholesky decomposition,” in *2013 signal processing: Algorithms, architectures, arrangements, and applications (SPA)*, pp. 70–72, IEEE, 2013.
- [44] L. Beilina, E. Karchevskii, and M. Karchevskii, *Numerical linear algebra: Theory and applications*. Springer, 2017.
- [45] P. Mayorga, A. Estudillo, A. Medina-Santiago, F. Ramos, *et al.*, “Fast approximation for toeplitz, tridiagonal, symmetric and positive definite linear systems that grow over time,” *International Journal of Advanced Computer Science and Applications*, vol. 7, no. 11, 2016.