Signal Processing for Large Arrays: Convolutional Beamspace, Hybrid Analog and Digital Processing, and Distributed Algorithms

Thesis by
Po-Chih Chen

In Partial Fulfillment of the Requirements for the
Degree of
Doctor of Philosophy

CALIFORNIA INSTITUTE OF TECHNOLOGY
Pasadena, California

2024
Defended May 8th, 2024

Po-Chih Chen
ORCID: 0000-0003-1637-9329

ACKNOWLEDGEMENTS

First of all, I would like to thank my advisor, Prof. P. P. Vaidyanathan, for his constant guidance during my PhD study at the California Institute of Technology (Caltech). In the technical aspect, he is professional, knowledgeable, and insightful. I have learned a lot from him on how to do research, write papers, and give presentations. Moreover, he is very supportive and accommodating. It is unfortunate that the COVID era accounts for a great portion of my PhD study, but his flexibility in the format and time of meetings alleviated the burden. It has been a nice journey with Prof. P. P. Vaidyanathan at Caltech.

I would like to thank the members of my defense and candidacy committees: Prof. Yaser Abu-Mostafa, Prof. Victoria Kostina, and Dr. Andre Tkacenko. I am grateful for their comments and feedback on my research and presentations. It is my pleasure to have them on the committees. Also, I would like to thank all the professors at Caltech who offered many great courses.

Generous support from the Office of Naval Research (ONR), the National Science Foundation (NSF), and Caltech is gratefully acknowledged.

I would like to thank past and present members of my group, Prof. See-May Phoong, Prof. Yuan-Pei Lin, Prof. Borching Su, Prof. Piya Pal, Prof. Chun-Lin Liu, Dr. Srikanth Tenneti, Dr. Oguzhan Teke, and Pranav Kulkarni, for their kind support, inspiring works, and valuable discussions. Also, I would like to thank Caroline Murphy, Tanya Owen, and Christine Garske for their administrative assistance.

I would like to thank Benjamin Chang, Chung-Yi Lin, Wen Chao, Yu-Hung Lai, Jake Chou, Shao-Min Hung, and many others in the Association of Caltech Taiwanese, for their friendship. I will miss the times we spent playing badminton, having meals together, and playing board games on weekends.

Finally, I would like to express my sincere gratitude to my father Shao-Wen Chen, my mother Mei-Hui Kao, and and my sister Si-Ying Chen, for their continuous support in all aspects.

ABSTRACT

The estimation of the directions of arrival (DOAs) of incoming waves for a passive antenna array has long been an important topic in array signal processing. Meanwhile, the estimation of the MIMO channel between a transmit antenna array and a receive antenna array is a key problem in wireless communications. In many recent works on these array processing tasks, people consider millimeter waves (mmWaves) due to their potential to offer more bandwidth than the already highly occupied lower-frequency bands. However, new challenges like strong path loss at the high frequencies of mmWaves arise. To compensate for the path loss, large arrays, or massive MIMO, are used to get large beamforming gain. It is practical due to the small sizes of mmWave antennas. When large arrays are used, it is important to develop efficient estimation algorithms with low computational and hardware complexity.

The main contribution of this thesis is to propose low-complexity DOA and channel estimation methods that are especially effective for large arrays. To achieve low complexity, three main aspects are explored: beamspace methods, hybrid analog and digital processing, and distributed algorithms. First, a new beamspace method, convolutional beamspace (CBS), is proposed for DOA estimation based on passive arrays. In CBS, the array output is spatially filtered, followed by uniform decimation (downsampling) to achieve dimensionality reduction. No DOA ambiguity occurs since the filter output is represented only by the passband sources. CBS enjoys the advantages of classical beamspace such as lower computational complexity, increased parallelism of subband processing, and improved resolution threshold for DOA estimation. Moreover, unlike classical beamspace methods, it allows root-MUSIC and ESPRIT to be performed directly for uniform linear arrays without additional preparation since the Vandermonde structure is preserved under the CBS transformation. The method produces more accurate DOA estimates than classical beamspace, and for correlated sources, better estimates than element-space.

The idea of hybrid analog and digital processing is then incorporated into CBS, leading to hybrid CBS for DOA estimation. In hybrid processing, an analog combiner is used to reduce the number of radio frequency (RF) chains and thus hardware complexity. Also for lowering hardware cost, the analog combiner is designed as a phase shifter network with unit-modulus entries. It is shown that any general (arbitrary coefficient) CBS filter can be implemented despite the unit-modulus constraints.

Moreover, a new scheme of CBS is proposed based on nonuniform decimation and difference coarray method. This allows us to identify more sources than RF chains. The retained samples correspond to the sensor locations of a virtual sparse array, dilated by an integer factor, which results in larger coarray aperture and thus better estimation performance. Besides, with the use of random or deterministic filter delays that vary with snapshots, a new method is proposed to decorrelate sources for the coarray method to work.

Next, a 2-dimensional (2-D) hybrid CBS method is developed for mmWave MIMO channel estimation. Since mmWave channel estimation problems can be formulated as 2-D direction-of-departure (DOD) and DOA estimation, benefits of CBS such as low complexity are applicable here. The receiver operation is again filtering followed by decimation. A key novelty is the use of a proper counterpart of CBS at the transmitter—expansion (upsampling) followed by filtering—to reduce RF chains. The expansion and decimation can be either uniform or nonuniform. The nonuniform scheme is used with 2-D coarray method and requires fewer RF chains to achieve the same estimation performance as the uniform scheme. A method based on the introduction of filter delays is also proposed to decorrelate path gains, which is crucial to the success of coarray methods. It is shown that given fixed pilot overhead, 2-D hybrid CBS can yield more accurate channel estimates than previous methods.

Finally, distributed (decentralized) algorithms for array signal processing are studied. With the potential of reducing computation and communication complexity, distributed estimation of covariance, and distributed principal component analysis have been introduced and studied in the signal processing community in recent years. Applications in array processing have been also indicated in some detail. In this thesis, distributed algorithms are further developed for several well-known methods for DOA estimation and beamforming. New distributed algorithms are proposed for DOA estimation methods like root-MUSIC, total least squares ESPRIT, and FOCUSS. Other contributions include distributed design of the Capon beamformer from data, distributed implementation of the spatial smoothing method for coherent sources, and distributed realization of CBS. The proposed algorithms are fully distributed since average consensus (AC) is used to avoid the need for a fusion center. The algorithms are based on a finite-time version of AC which converges to the exact solution in a finite number of iterations. This enables the proposed distributed algorithms to achieve the same performance as the centralized

counterparts, as demonstrated by simulations.

PUBLISHED CONTENT AND CONTRIBUTIONS

All the papers below resulted from research carried out by Po-Chih Chen and Professor Palghat P. Vaidyanathan. Po-Chih Chen was the primary contributor of these papers. He proposed the main results, proved theorems and lemmas, conducted experimental validations, wrote papers, and corresponded with editors and reviewers. Professor Palghat P. Vaidyanathan applied for research grants, supervised research projects, discussed research topics with Po-Chih Chen, and reviewed the main results, paper drafts, and the correspondences with reviewers.

[1] P.-C. Chen and P. P. Vaidyanathan, "Convolutional beamspace for linear arrays," *IEEE Trans. Signal Process.*, vol. 68, pp. 5395–5410, 2020. DOI: 10.1109/TSP.2020.3021670.
The content of this paper is described in Chapter 2 (Sec. 2.2–2.4).

[2] P. P. Vaidyanathan and P.-C. Chen, "Convolutional beamspace for array signal processing," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2020, pp. 4707-4711. DOI: 10.1109/ICASSP40776.2020.9054051.
The content of this paper is the preliminary version of Chapter 2 (Sec. 2.2).

[3] P. P. Vaidyanathan and P.-C. Chen, "Convolutional beamspace and sparse signal recovery for linear arrays," *Asilomar Conf. on Signal, Syst., Comput.*, 2020, pp. 929–933. DOI: 10.1109/IEEECONF51394.2020.9443522.
The content of this paper is the preliminary version of Chapter 2 (Sec. 2.4).

[4] P.-C. Chen and P. P. Vaidyanathan, "Sliding-Capon based convolutional beamspace for linear arrays," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2021, pp. 4565-4569. DOI: 10.1109/ICASSP39728.2021.9414022.
The content of this paper is described in Chapter 2 (Sec. 2.5).

[5] P.-C. Chen and P. P. Vaidyanathan, "Convolutional beamspace using IIR filters," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2022, pp. 5003-5007. DOI: 10.1109/ICASSP43922.2022.9746961.
The content of this paper is described in Chapter 2 (Sec. 2.6).

[6] P.-C. Chen and P. P. Vaidyanathan, "DOA estimation for mmWave passive arrays with hybrid beamspace processing," *IEEE Access*, under review.
The content of this paper is described in Chapter 3.

[7] P.-C. Chen and P. P. Vaidyanathan, "Hybrid convolutional beamspace for DOA estimation of millimeter wave sources," *Asilomar Conf. on Signal, Syst., Comput.*, 2022, pp. 86-90. DOI: 10.1109/IEEECONF56349.2022.10051878.
The content of this paper is the preliminary version of Chapter 3.

[8] P.-C. Chen and P. P. Vaidyanathan, "Channel estimation for mmWave using the convolutional beamspace approach," *IEEE Trans. Signal Process.*, 2024 (to appear). DOI: 10.1109/TSP.2024.3379859.
The content of this paper is described in Chapter 4.

[9] P.-C. Chen and P. P. Vaidyanathan, "Hybrid convolutional beamspace method for mmWave MIMO channel estimation," *Asilomar Conf. on Signal, Syst., Comput.*, 2023, pp. 1293-1297. DOI: 10.1109/IEEECONF59524.2023.10477082.
The content of this paper is the preliminary version of Chapter 4.

[10] P.-C. Chen and P. P. Vaidyanathan, "Error analysis of convolutional beamspace algorithms," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2023, pp. 1-5. DOI: 10.1109/ICASSP49357.2023.10095614.
The content of this paper is the preliminary version of Chapter 5 (Sec. 5.3).

[11] P.-C. Chen and P. P. Vaidyanathan, "Cramér–Rao bound for convolutional beamspace method," *Asilomar Conf. on Signal, Syst., Comput.*, 2023, pp. 1288-1292. DOI: 10.1109/IEEECONF59524.2023.10477074.
The content of this paper is the preliminary version of Chapter 5 (Sec. 5.4).

[12] P.-C. Chen and P. P. Vaidyanathan, "Distributed algorithms for array signal processing," *IEEE Trans. Signal Process.*, vol. 69, pp. 4607-4622, 2021. DOI: 10.1109/TSP.2021.3101015.
The content of this paper is described in Chapter 6.

[13] P. P. Vaidyanathan and P.-C. Chen, "Distributed root-MUSIC using finite-time average consensus," *Asilomar Conf. on Signal, Syst., Comput.*, 2021, pp. 539-543. DOI: 10.1109/IEEECONF53345.2021.9723302.
The content of this paper is the preliminary version of Chapter 6.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

Chapter 1

INTRODUCTION

Antenna arrays have long been used for spatial sampling of incoming waves in many applications, such as radar, sonar, seismology, and communications [8]. The use of antenna arrays facilitates the design of algorithms for many important problems in signal processing, including direction-of-arrival (DOA) estimation and beamforming. Recently, large antenna arrays, or massive MIMO, have received much attention in wireless communications, such as in 5G cellular communications and beyond 5G [9]. In 5G and beyond, people consider millimeter waves (mmWaves) due to their potential to offer more bandwidth than the lower-frequency bands, which are already highly occupied. However, there are also new challenges, such as strong path loss at the high frequencies of mmWaves [10]. To compensate for the path loss, large arrays are used, which are practical due to the small sizes of mmWave antennas, to get large beamforming gain [11]. When large arrays are used, it is important to design efficient estimation algorithms with low computational and hardware complexity.

The use of beamforming prior to DOA estimation, referred to as *beamspace* processing [8, 12–15], can help computational complexity reduction. Given an N-sensor uniform linear array (ULA) with output $\mathbf{x} \in \mathbb{C}^N$, the idea of beamspace is to compute a transformation $\mathbf{y} = \mathbf{T}\mathbf{x} \in \mathbb{C}^B$, where $B < N$, and estimate the DOAs based on $\mathbf{y}$ using, e.g., subspace methods. Due to dimensionality reduction ($B < N$), the $B \times B$ covariance of $\mathbf{y}$ has smaller size than that of $\mathbf{x}$. Thus, the complexity of the eigenspace computation $O(B^3)$, is much smaller than $O(N^3)$, which is the complexity when using *element-space* (estimation based on $\mathbf{x}$). If $\mathbf{T}$ is properly designed, the DOAs which fall outside a chosen beam or subband are attenuated by $\mathbf{T}$, so there are typically much fewer DOAs represented by $\mathbf{y}$, compared to $\mathbf{x}$. We can use a bank of transformations $\{\mathbf{T}_i\}$, which can be operated *in parallel*, to cover the DOAs in all subbands. Besides low computation and parallelism, beamspace methods tend to have higher DOA resolution [8, 16, 17] and smaller bias [2] than the element-space method. However, in classical beamspace methods, $\mathbf{y}$ is no longer represented by a Vandermonde matrix as $\mathbf{x}$ is, so elaborate steps have to be taken to do standard DOA estimation methods including root multiple signal classification (root-MUSIC) [2] and estimation of signal parameters via rotational invariance techniques (ESPRIT) [18].

To reduce hardware complexity for large mmWave arrays, a new line of research topics has emerged, where people design hybrid analog and digital transceivers [4, 5, 9, 11, 19]. The use of analog precoder at the transmitter and analog combiner at the receiver can achieve dimensionality reduction. Hence, fewer radio frequency (RF) chains are required, which reduces hardware cost and power consumption. The analog precoder and combiner are typically designed as networks of phase shifters, so they should have unit-modulus entries. Such constraints lead to new design problems to solve. Meanwhile, the sparse nature of an mmWave MIMO channel due to limited scattering allows it to be modeled by a few paths, each with a direction of departure (DOD) at the transmitter and DOA at the receiver [11]. Hence, the classical DOA estimation problem has also gained renewed interests since techniques for 1-dimensional (1-D) DOA estimation may be extended to mmWave MIMO channel estimation.

Traditionally, DOA estimation and beamforming algorithms require data collection and centralized computation at a fusion center. However, following the pioneering work in [7], *distributed* (*decentralized*) algorithms for DOA estimation and beamforming have gained more research interest. In these algorithms, the sensor array is partitioned into subarrays. The data in each subarray is available only to the processor in that subarray, and between the processors there is some minimal exchange of intermediate results, in order to implement an average consensus [20–22]. Based on such local computations and limited data exchange between processors, we do usual array processing tasks like DOA estimation and beamforming. Thus, such arrays work without the help of a central processor or fusion center. The communication and computation bottleneck that can occur for large arrays with a central processor is thus mitigated in these distributed systems.

With the aforementioned backgrounds, the main theme of this thesis is to *design DOA and channel estimation methods particularly useful to large arrays*, and the main contributions are summarized as follows:

1. *Beamspace DOA estimation*: We propose a beamspace method, called *convolutional beamspace (CBS)* [23, 24], for DOA estimation based on passive arrays. The array output is processed by spatial *filtering* followed by *uniform decimation* (*downsampling*). It achieves all the same advantages as classical beamspace methods [2, 18], including lower computational complexity, increased parallelism of subband processing, and smaller bias. Moreover, unlike classical beamspace, it allows root-MUSIC [25, 26] and ESPRIT [27] to be

performed directly for ULAs without additional preparation since the Vandermonde structure is preserved under the CBS transformation. It can also yield smaller estimation errors for correlated sources compared to element-space.

2. *Hybrid mmWave MIMO channel estimation*: We propose *hybrid convolutional beamspace* [28, 29] for mmWave MIMO channel estimation. The method is obtained by realizing the original CBS [23] by a hybrid analog and digital implementation [30, 31] to reduce hardware cost, and extending it to the 2-dimensional (2-D) MIMO case. We design a proper counterpart of CBS at the transmitter: *uniform expansion* (*upsampling*) followed by spatial filtering. Besides, we propose a new scheme of CBS where *nonuniform* expansion and decimation are used with *difference coarray* method [32] to further reduce the number of required RF chains to achieve the same estimation performance. We show that the proposed CBS precoder and combiner with any filter coefficients can be implemented in a hybrid way under the unit-modulus constraints of phase shifters. Hybrid CBS yields smaller estimation errors than previous channel estimation methods [4, 19] when having the same pilot overhead.

3. *Distributed array processing*: We propose distributed algorithms for a number of well-known methods for DOA estimation and beamforming [33, 34]. We develop distributed algorithms for DOA estimation methods such as root-MUSIC [26] and total least squares (TLS) ESPRIT [27] (known to be more accurate than least squares (LS) ESPRIT [35]). We derive distributed versions of the Capon beamformer [36] and focal underdetermined system solver (FOCUSS) method [37, 38] for sparse-solver based DOA estimation. We also develop distributed realizations of convolutional beamspace [23]. Finally, we propose a distributed algorithm for spatial smoothing [39], which is a technique used for DOA estimation when there are coherent or correlated sources.

Chapter outline: The passive array output model is reviewed in Sec. 1.1. Then, the millimeter-wave MIMO channel model is reviewed in Sec. 1.2. The use of beamspace methods for hybrid analog and digital processing is introduced in Sec. 1.3. The model for distributed array processing is presented in Sec. 1.4. The scope and outline of the thesis is given in Sec. 1.5. Finally, the notations used in this thesis are defined in Sec. 1.6, and a list of acronyms are given in Sec. 1.7.

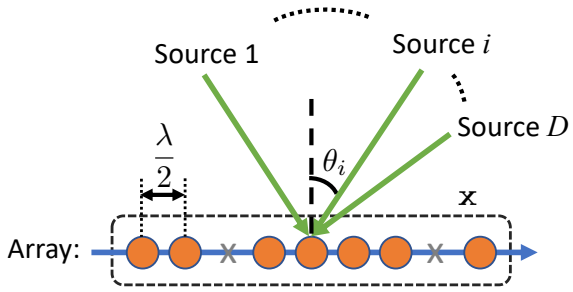


Figure 1.1: Model of a passive array receiving plane waves from D monochromatic far-field sources with wavelength λ and DOAs θ_i. Filled circles represent antennas, and crosses represent empty space.

1.1 Passive Array Output Model

We consider a passive linear array of N antennas (or sensors) as shown in Fig. 1.1. There are D monochromatic sources emitting electromagnetic waves with wavelength λ. These sources are assumed to be in the far field [8] so that the waves received by the array can be well approximated by plane waves. We assume all the sources and the array lie in a plane, and that all the sources are on one side of the array. By definition, the direction of arrival (DOA) $\theta_i \in [-\pi/2, \pi/2)$ of each source i is measured from the normal to the line of array, $i = 1, \ldots, D$. The antenna locations are on a uniform grid with unit spacing d. That is, the antenna locations are $n_l d$ for integers n_l, $l = 0, \ldots, N-1$. Without loss of generality, we assume $n_0 = 0$. Then, the array output can be expressed as [8]

$$\mathbf{x} = [x(0)\ x(1) \cdots x(N-1)]^T = \mathbf{A}\mathbf{c} + \mathbf{e}, \tag{1.1}$$

where $\mathbf{c}$ contains source amplitudes c_i, $\mathbf{e}$ contains additive noise terms, and the array manifold matrix

$$\mathbf{A} = [\mathbf{a}(\omega_1)\ \mathbf{a}(\omega_2)\ \cdots\ \mathbf{a}(\omega_D)] \tag{1.2}$$

with $\omega_i = 2\pi d \sin\theta_i/\lambda$ and the steering vector

$$\mathbf{a}(\omega) = [1\ e^{j\omega n_1}\ e^{j\omega n_2}\ \cdots\ e^{j\omega n_{N-1}}]^T \tag{1.3}$$

for any real number ω. If $d > \lambda/2$, then there exist multiple DOAs that give the same array output, so there is DOA ambiguity [8]. Thus, in this thesis, we assume the classical antenna spacing $d = \lambda/2$ is used. In this case,

$$\omega_i = \pi \sin\theta_i \in [-\pi, \pi). \tag{1.4}$$

In this thesis, we also call ω_i a DOA since there is one-to-one correspondence between ω_i and θ_i. We assume the source amplitudes and noise are random such that $\mathrm{E}[\mathbf{c}] = \mathbf{0}$, $\mathrm{E}[\mathbf{e}] = \mathbf{0}$, $\mathrm{E}[\mathbf{e}\mathbf{e}^H] = \sigma_e^2\mathbf{I}$, and $\mathrm{E}[\mathbf{c}\mathbf{e}^H] = \mathbf{0}$. Then, our goal is to estimate the DOAs ω_i based on a finite number, say K, of independent and identically distributed snapshots of the array output

$$\mathbf{x}[k] = \mathbf{A}\mathbf{c}[k] + \mathbf{e}[k], k = 1, \ldots, K. \tag{1.5}$$

In this model, the DOAs are assumed fixed during the measurements of the K snapshots.

1.1.1 Difference Coarray

For any linear array $\mathcal{N} = \{n_l\}$ with antenna locations n_l, $l = 0, \ldots, N-1$, its difference coarray $\mathcal{C}$ is defined as [32]

$$\mathcal{C} = \{n_k - n_l \mid 0 \le k, l \le N-1\}. \tag{1.6}$$

In other words, $\mathcal{C}$ contains the differences between all pairs of the antenna locations. In the (difference) coarray method [32], second-order statistics of the array output of $\mathcal{N}$ is used to construct a virtual signal on $\mathcal{C}$. This can potentially increase the number of identifiable DOAs if $\mathcal{N}$ is a well-designed sparse array like nested array [32], coprime array [40], or minimum redundancy array (MRA) [3]. The idea of coarrays will be incorporated into the CBS scheme where nonuniform decimation is used.

1.2 Millimeter-Wave MIMO Channel Model

We consider an mmWave MIMO transceiver system as shown in Fig. 1.2. The transmit array is assumed to be an N_t-antenna ULA, and the receive array is an N_r-antenna ULA. There are two phases of the transceiver operation: a training phase and a transmission (communication) phase. In this thesis, we focus on channel estimation, which is done in the training phase. The MIMO channel will be estimated based on the received array outputs due to a finite number of training blocks sent from the transmitter. Due to limited scattering, the $N_\mathrm{r} \times N_\mathrm{t}$ mmWave narrowband MIMO channel matrix can be modeled by [5, 11, 19]

$$\mathbf{H}_{\alpha,k} = \sum_{i=1}^{D} \alpha_{k,i}\mathbf{a}_{N_\mathrm{r}}(\omega_{\mathrm{r},i})\mathbf{a}_{N_\mathrm{t}}^H(\omega_{\mathrm{t},i}) = \mathbf{A}_\mathrm{r}\mathbf{D}_{\alpha,k}\mathbf{A}_\mathrm{t}^H, \tag{1.7}$$

where $\mathbf{A}_p = [\mathbf{a}_{N_p}(\omega_{p,1}) \ \cdots \ \mathbf{a}_{N_p}(\omega_{p,D})]$, $p \in \{\mathrm{t}, \mathrm{r}\}$,

$$\mathbf{a}_N(\omega) = [1 \ e^{j\omega} \ e^{j2\omega} \ \cdots \ e^{j(N-1)\omega}]^T \tag{1.8}$$

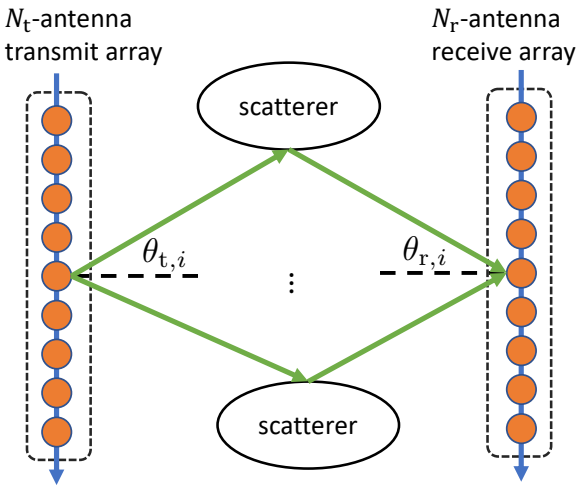


Figure 1.2: Model of an mmWave MIMO channel due to a few paths with DODs $\theta_{t,i}$ and DOAs $\theta_{r,i}$.

for any real number ω and positive integer N, and $\mathbf{D}_{\alpha,k} = \text{diag}(\alpha_{k,1}, \ldots, \alpha_{k,D})$. Here, k is the training block index. In this channel model, D denotes the number of paths, and $\alpha_{k,i}$, $\omega_{t,i}$, and $\omega_{r,i}$ are the path gain, direction of departure (DOD), and direction of arrival (DOA) for the ith path, respectively. As in Sec. 1.1, the classical antenna spacing $\lambda/2$ is assumed, where λ is the signal wavelength. Then, each DOD or DOA $\omega_{p,i} = \pi \sin\theta_{p,i} \in [-\pi, \pi)$, where $\theta_{p,i} \in [-\pi/2, \pi/2)$ is the physical DOD or DOA measured from the normal to the line of transmit or receive array. We assume that the DODs and DOAs remain the same throughout the training process, and that the path gains $\alpha_{k,i}$ vary from block to block. The model (1.7) indicates that estimation of the channel can be achieved by 2-D DOD and DOA estimation followed by path gain estimation.

1.3 Beamspace Methods for Hybrid Analog and Digital Processing

We consider a *hybrid analog and digital processing* model [11] of an mmWave passive array for DOA estimation as shown in Fig. 1.3. The output of the N-antenna receive ULA is first processed by an analog combiner, resulting in an $\bar{N}$-dimensional output, where $\bar{N} \ll N$ typically. It is then passed through $\bar{N}$ analog-to-digital RF chains and processed by a digital combiner. Thus, the hybrid combiner output is

$$\mathbf{v} = \mathbf{F}_d\mathbf{F}_a\mathbf{x} = \mathbf{F}_d\mathbf{F}_a\mathbf{A}\mathbf{c} + \mathbf{F}_d\mathbf{F}_a\mathbf{e}, \tag{1.9}$$

where $\mathbf{x}$ is as defined in (1.1) with the ULA assumption, i.e., $n_l = l$. Finally, we can do whatever digital processing is needed to estimate the DOAs after the digital

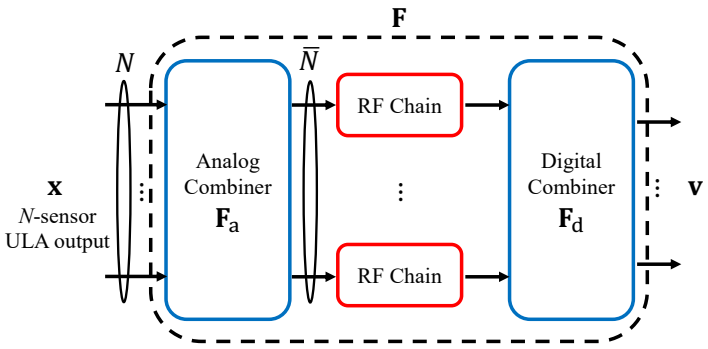


Figure 1.3: Hybrid analog and digital processing model of an mmWave passive array for DOA estimation.

combiner. In this hybrid model, the goal of the analog combiner is to achieve dimensionality reduction so that we need only a small number $\bar{N}$ of RF chains for a large number of N antennas. With fewer RF chains, we reduce hardware cost and power consumption. Also, in this implementation, the analog combiner is effectively a *beamspace processing matrix* as in classical beamspace methods [2, 8, 18]. The combiner has unit-modulus entries since it is implemented by phase shifter networks to reduce hardware cost [11]. Thus, the main problem here is how to design the analog combiner under unit-modulus constraints in order to obtain good DOA estimates. In this thesis, we propose a new beamspace method, *convolutional beamspace (CBS)*, in Chapter 2. Then we show how to design analog and digital combiners based on it in Chapter 3, where the method is named *hybrid CBS*.

Then, we consider a hybrid analog and digital processing model of an mmWave MIMO transceiver system [11] as shown in Fig. 1.4. The receiver structure is the same as in Fig. 1.3, except that N and $\bar{N}$ are replaced by N_r and $\bar{N}_\mathrm{r}$, respectively, to denote "receiver." The analog and digital building blocks of the transmitter are reversely ordered compared to the receiver. In particular, at each time instant, a training vector is processed by a digital precoder. The digital precoder output vector, composed of $\bar{N}_\mathrm{t}$ digital symbols, is then passed through $\bar{N}_\mathrm{t}$ digital-to-analog RF chains. Then the output is processed by an analog precoder which generates the transmitted signals for the N_t-antenna transmit ULA. Again, $\bar{N}_\mathrm{t} \ll N_\mathrm{t}$ typically, so we need only a small number $\bar{N}_\mathrm{t}$ of RF chains for a large number of N_t antennas. If we let $\mathbf{S}_k$ denote the kth training block, which contains multiple training vectors as its columns, then the kth received block will be

$$\mathbf{Y}_k = \mathbf{F}_{\mathrm{r,d}}\mathbf{F}_{\mathrm{r,a}}\mathbf{H}_{\alpha,k}\mathbf{F}_{\mathrm{t,a}}^H\mathbf{F}_{\mathrm{t,d}}^H\mathbf{S}_k + \mathbf{F}_{\mathrm{r,d}}\mathbf{F}_{\mathrm{r,a}}\mathbf{E}_k, \tag{1.10}$$

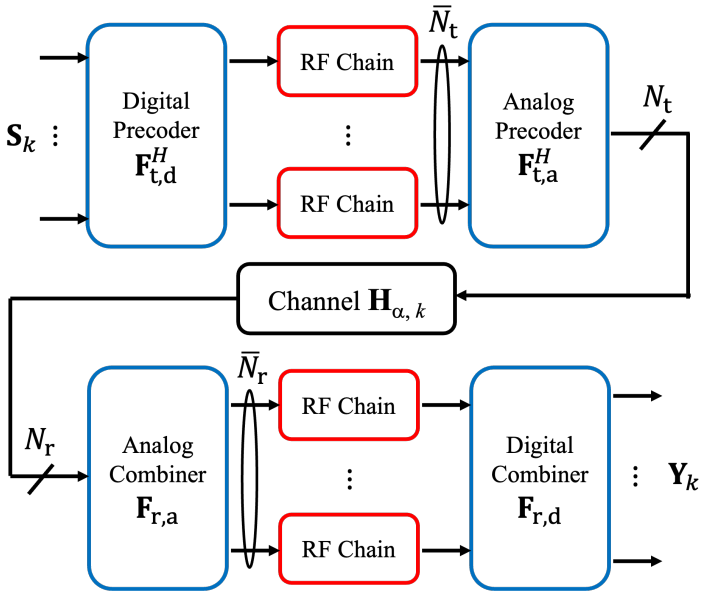


Figure 1.4: Hybrid analog and digital processing model of an mmWave MIMO transceiver.

where the channel matrix $\mathbf{H}_{\alpha,k}$ is defined in (1.7), and $\mathbf{E}_k$ is additive white noise such that $\mathrm{E}[\mathbf{E}_k] = \mathbf{0}$ and $\mathrm{E}[\mathrm{vec}(\mathbf{E}_k)\mathrm{vec}^H(\mathbf{E}_l)] = \sigma_{\mathrm{e}}^2\delta_{kl}\mathbf{I}$ for all k and l. The analog precoder and analog combiner similarly have unit-modulus entries as they are phase shifter networks [11]. Thus, the main problem becomes how to design the analog precoder and analog combiner under unit-modulus constraints in order to obtain good channel estimates. With the model (1.7), estimation of the channel pertains to estimation of the DODs and DOAs (and path gains, which can be estimated after we obtain DOD and DOA estimates, as shown in Sec. 4.4.3). In this thesis, we extend the proposed *hybrid CBS* to this 2-D case and show how to design analog processors based on it, as explained in Chapter 4. The digital precoder and digital combiner in general can be any matrices as they are in the digital domain. They are there for generality, but for channel estimation with CBS, we later set them to be identity matrices (see Chapter 4). In the transmission phase, they are often optimized for other things, such as maximizing rate or minimizing MSE [11, 41].

1.4 Distributed Algorithms for Array Signal Processing

Beamspace methods and hybrid processing are used to reduce computational and hardware complexity for large arrays in Sec. 1.3, where we assume that a central processor can get the entire array data and do any computation. In another research area of this thesis, we consider distributed array processing algorithms so that a

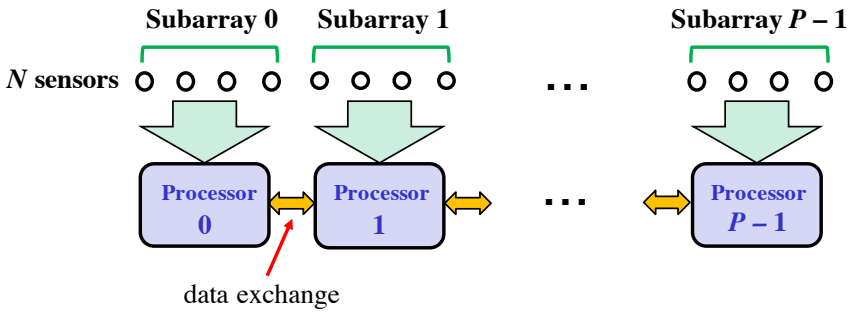


Figure 1.5: Schematic of distributed array processing. The N-sensor linear array is divided into P subarrays. The sensor data from subarray p is directly available only to processor p, corresponding to node p in a communication network modeled by an undirected graph $\mathcal{G}$. Between the processors there is some minimal exchange of intermediate results, in order to implement an average consensus.

communication or computation bottleneck that may occur when there is a central processor can be avoided [42]. The system model is shown in Fig. 1.5. This is a network composed of P nodes, each of which is a Q-sensor linear subarray [7, 35]. For ease of presentation, we assume that each subarray has the same number of sensors, but many of the proposed distributed algorithms in this thesis can be readily extended to subarrays with different numbers of sensors. The sensor data from subarray p is directly available only to a local processor at node p. Between the processors there is some minimal exchange of intermediate results, in order to implement an average consensus (AC) [20–22]. The goal of AC methods is to compute in a distributed way the average of some values stored across the network. The communication network is modeled by an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of the P nodes, and $\mathcal{E}$ is the set of edges. Each node represents a subarray and the edges represent the communication links. If there is an edge between two nodes, then two-way communication is allowed between these nodes (for AC and so forth). The P subarrays, which do not have overlapping sensors, collectively form a linear array with $N = PQ$ sensors. That is, all the subarrays are on the same vertical positions. Assume all the subarrays are located not too far away, and they receive the same set of D source amplitudes as explained in Sec. 1.1. Then similar to (1.1), the array output is

$$\mathbf{x} = [\mathbf{x}_0^T \; \mathbf{x}_1^T \; \cdots \; \mathbf{x}_{P-1}^T]^T = \mathbf{A}\mathbf{c} + \mathbf{e}, \tag{1.11}$$

where $\mathbf{x}_p \in \mathbb{C}^Q$ is the output of the pth subarray, $p = 0, \ldots, P-1$. Then, our main goal is to design distributed algorithms where local processing of $\mathbf{x}_p$ is done at each

subarray p, and there is some minimal exchange of intermediate results derived from $\mathbf{x}_p$ between the P local processors. With these distributed algorithms, we should achieve usual array processing tasks, including DOA estimation and beamforming, as shown in Chapter 6.

1.5 Scope and Outline of the Thesis

In this thesis, low-complexity DOA and channel estimation methods particularly useful to large arrays are developed. To this end, three main approaches are taken: *beamspace methods*, *hybrid analog and digital processing*, and *distributed algorithms*. First, a new beamspace DOA estimation method, *convolutional beamspace (CBS)*, is proposed in Chapter 2. Its hybrid analog and digital implementation, named *hybrid CBS*, is developed in Chapter 3. A 2-D extension of hybrid CBS to mmWave MIMO channel estimation is then presented in Chapter 4. The MSE performance and Cramér–Rao bound (CRB) [43, 44] of 1-D CBS for DOA estimation is analyzed theoretically in Chapter 5. *Distributed* algorithms for several well-known DOA estimation and beamforming methods are proposed in Chapter 6.

1.5.1 Basics of Convolutional Beamspace for DOA Estimation (Chapter 2)

In Chapter 2, a new beamspace method called *convolutional beamspace (CBS)* is introduced. It enjoys the advantages of classical beamspace such as lower computational complexity, increased parallelism of subband processing, and improved resolution threshold for DOA estimation. Furthermore, unlike classical beamspace methods, it allows root-MUSIC and ESPRIT to be performed directly for ULAs without additional preparation since the Vandermonde structure and the shift-invariance are preserved under the CBS transformation. The computational benefits of the proposed methods are quantified and demonstrated. Unlike classical beamspace, CBS is based on the use of filtering with a finite-impulse-response (FIR) filter $H(z)$, followed by *uniform downsampling* (*decimation*) to achieve dimensionality reduction. For large arrays, which are becoming increasingly important [9, 45], the filter can be proportionately longer, offering very effective attenuation of stopband sources. Thus, the filter output is represented only by passband sources, and the uniform decimation after filtering does not cause DOA ambiguity. CBS produces more accurate DOA estimates than classical beamspace, and for correlated sources it often produces better estimates than element-space as well. Some variants of CBS are also proposed, including CBS using infinite-impulse-response (IIR) filters [46] and Capon-CBS [47]. An IIR filter often requires a much lower order than an FIR filter

for the same set of magnitude response specifications. Hence, the computational complexity of IIR-CBS can be smaller than FIR-CBS. In Capon-CBS, the filter is designed to be a sliding Capon beamformer. Such design takes input statistics into account, so it can do a better job of suppressing the sources that fall in the stopband. Thus, Capon-CBS can offer higher DOA resolution and smaller estimation errors than CBS in some cases.

1.5.2 Hybrid Convolutional Beamspace for DOA Estimation (Chapter 3)

The CBS method proposed in Chapter 2 is implemented fully in the digital domain. Such implementation requires that one RF chain per antenna be used. A hybrid analog and digital implementation of CBS is developed in Chapter 3, in order to reduce the required number of RF chains. The analog combiner achieves dimensionality reduction so that fewer RF chains than the number of antennas are needed. This lowers hardware cost and power consumption. As mentioned earlier, the analog combiner is often implemented by phase shifters with unit-modulus entries. A standard digital filter typically does not have constant-modulus coefficients, so the resulting beamspace matrix of CBS cannot be directly implemented in the analog domain. Interestingly, according to [48], any complex vector can be expressed as a linear combination of two vectors with unit-modulus entries. With this, it is shown that any CBS filter coefficients can be implemented under the unit-modulus constraint. Besides CBS based on uniform decimation, a new scheme of CBS using nonuniform decimation is proposed. Then the *difference coarray* method [32] allows the estimation of more sources than RF chains (not achievable by classical beamspace [2] or uniform hybrid CBS). The retained samples correspond to the sensor locations of a virtual sparse array, dilated by an integer factor M, which results in M times larger coarray aperture (than when there is no dilation). The aperture of an array is the difference between its extreme elements, and a larger aperture typically gives better estimation performance [49, 50]. The virtual sparse array can be based on commonly used sparse arrays such as the nested array [32], coprime array [40], and minimum redundancy array (MRA) [3]. Their difference coarrays have large ULA segments and can be used to get good DOA estimates. Given a fixed number of RF chains, the nonuniform scheme can achieve large array aperture and thus better estimation performance than uniform scheme. Or, nonuniform scheme requires fewer RF chains to achieve the same estimation performance. Finally, with the use of random or deterministic filter delays that vary with snapshots, a new method is proposed to decorrelate sources for the coarray method to work.

1.5.3 Hybrid Convolutional Beamspace for mmWave MIMO Channel Estimation (Chapter 4)

The hybrid CBS method presented in Chapter 3 is for receive array only. An extension to both transmit and receive arrays is given in Chapter 4, i.e., hybrid CBS for mmWave MIMO channel estimation. As elaborated in Sec. 1.2, an mmWave MIMO channel can be modeled by a few paths with DODs at the transmitter and DOAs at the receiver. Thus, an mmWave MIMO channel estimation problem can be formulated as a 2-D DOD and DOA estimation problem (followed by path gain estimation). In Chapter 4, a hybrid CBS method is proposed to this 2-D estimation problem. One main novelty is the development of an appropriate counterpart of CBS at the transmitter, an *expander* (*upsampler*) followed by a filter, as opposed to a filter followed by a decimator (downsampler) at the receiver. The designed hybrid CBS can reduce the number of RF chains at the transmitter, just like at the receiver. Moreover, unlike classical beamspace, the 2-D Vandermonde structure due to transmit and receive ULAs is preserved, so standard 2-D angle estimation methods like 2-D unitary ESPRIT [51] can be readily used. Similar to the decimation at the receiver, the expansion at the transmitter can be uniform or nonuniform. Given a fixed number of RF chains, 2-D coarray method [5] can be used in the nonuniform scheme to estimate more sources than the uniform scheme. More importantly, the nonuniform scheme can yield larger array apertures and thus better estimation performance than the uniform scheme. Like in Chapter 3, the proposed CBS precoder and combiner with any filter coefficients can be implemented in a hybrid way under the unit-modulus constraints of phase shifters. Also, a new method is proposed to decorrelate path gains for the coarray method to work.

1.5.4 Theoretical Analysis of 1-D Convolutional Beamspace (Chapter 5)

In Chapter 5, theoretical analysis of 1-D CBS for DOA estimation is given. The analysis is applicable to both the digital CBS in Chapter 2 and hybrid CBS in Chapter 3 since they perform the same. In particular, the MSE performance of CBS is analyzed when MUSIC [52] or root-MUSIC [26] is used for DOA estimation. The error variance is derived from the asymptotic probability distribution of the eigenvectors of an average finite-snapshot covariance matrix. Meanwhile, the bias due to the filtered stopband sources is given by a first-order perturbation analysis. Advantages of CBS mentioned in Chapters 2 and 3 are confirmed by the MSE analysis. Moreover, the Cramér–Rao bound (CRB) [43, 44] for CBS is also derived. Conventionally, the CRB is a lower bound for unbiased estimators. A lower bound

on the variances of the biased CBS estimator is obtained and shown to be typically well approximated by the classical CRB for unbiased estimators. Two forms of CRB expressions are derived, and they offer different insights as explained in Chapter 5. All the results also apply to element-space since element-space is a special case of CBS. Finally, the theoretical results are verified by simulations.

1.5.5 Distributed Algorithms for Array Signal Processing (Chapter 6)

Although there are several inspiring previous papers [7, 35, 42, 53] on distributed algorithms for array processing, distributed algorithms have not yet been reported for a number of well-known methods for DOA estimation and beamforming. In Chapter 6, distributed algorithms for DOA estimation methods such as root-MUSIC [26], and total least squares (TLS) ESPRIT which is known to be more accurate than LS-ESPRIT [35], are developed. Distributed versions of the Capon beamformer [36] and the well-known FOCUSS method [37, 38] for sparse-solver based DOA estimation are also derived. Besides, distributed algorithms for CBS are proposed. It is shown that distributed DOA estimation algorithms, including root-MUSIC, TLS-ESPRIT, and FOCUSS, can be applied either directly to the original array domain, i.e., element-space, or in series with a beamspace method like CBS. A distributed algorithm is also proposed for spatial smoothing [39], which is a technique used for DOA estimation when there are coherent or correlated sources. All the proposed algorithms are *fully distributed* in that a fusion center is not required. The novelties of the proposed algorithms mainly lie in finding a way to implementing the algorithm so that all the data exchange among subarrays can be realized using *average consensus (AC)*. There are two families of AC methods, *asymptotic convergence* [20] and the recently introduced *finite-time convergence* methods [21, 22]. Asymptotic AC is used for the distributed DOA estimation algorithms in previous works [7, 35, 42, 53]. In these methods, one uses only finite but sufficiently many iterations to approximate asymptotic behaviors, so additional estimation errors arise from the use of asymptotic AC. By contrast, finite-time AC offers exact convergence in a finite number of iterations, so no additional estimation errors are introduced. Hence, finite-time AC is used in this thesis, and this is why the proposed distributed algorithms can achieve the same performance as the centralized counterparts.

1.6 Notations

The notations used in this thesis are defined in this section. The imaginary unit is given by $j = \sqrt{-1}$. Boldfaced capital letters denote matrices, and boldfaced

lowercase letters are reserved for column vectors. For a vector $\mathbf{v}$, we use $[\mathbf{v}]_i$, $\|\mathbf{v}\|_2$ and $\|\mathbf{v}\|_\infty$ to denote its ith entry, L^2-norm, and L^∞-norm, respectively. The subscript 2 is sometimes omitted for L^2-norm. For a matrix $\mathbf{A}$, we use $[\mathbf{A}]_{i,k}$ and $[\mathbf{A}]_{:,k}$ to indicate its (i,k)-entry and kth column, respectively. We also use $\|\mathbf{A}\|$ to denote its spectral norm, i.e., maximum singular value, $\|\mathbf{A}\|_F$ its Frobenius norm, and $\|\mathbf{A}\|_{1,2} = \sum_m \sqrt{\sum_n |[\mathbf{A}]_{m,n}|^2}$. We use $\mathbf{A}^T$, $\mathbf{A}^H$, and $\mathbf{A}^+$ to denote its transpose, conjugate transpose, and pseudoinverse, respectively. Also, $\mathbf{A}^*$ is the element-wise complex conjugate of $\mathbf{A}$. The vectorization of a matrix $\mathbf{A} = [\mathbf{a}_1\ \mathbf{a}_2\ \cdots\ \mathbf{a}_N]$ is defined as

$$\operatorname{vec}(\mathbf{A}) = \operatorname{vec}([\mathbf{a}_1\ \mathbf{a}_2\ \cdots\ \mathbf{a}_N]) = \begin{bmatrix} \mathbf{a}_1 \\ \mathbf{a}_2 \\ \vdots \\ \mathbf{a}_N \end{bmatrix}. \tag{1.12}$$

The Kronecker product, Khatri–Rao product, and Hadamard product of two matrices $\mathbf{A}$ and $\mathbf{B}$ are denoted by $\mathbf{A} \otimes \mathbf{B}$, $\mathbf{A} \odot \mathbf{B}$, and $\mathbf{A} \circ \mathbf{B}$, respectively. For any two Hermitian symmetric matrices $\mathbf{A}$ and $\mathbf{B}$, we use $\mathbf{A} \succeq \mathbf{B}$ and $\mathbf{B} \preceq \mathbf{A}$ to denote that $\mathbf{A} - \mathbf{B}$ is positive semidefinite. For square matrices $\mathbf{A}_1, \ldots, \mathbf{A}_n$, we use $\operatorname{diag}(\mathbf{A}_1, \ldots, \mathbf{A}_n)$ to denote the block diagonal matrix having $\mathbf{A}_1, \ldots, \mathbf{A}_n$ in the diagonal. For a matrix $\mathbf{A}$ with full column rank, the matrices

$$\mathbf{\Pi}_{\mathbf{A}} = \mathbf{A}(\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}, \tag{1.13}$$

$$\mathbf{\Pi}_{\mathbf{A}}^{\perp} = \mathbf{I} - \mathbf{A}(\mathbf{A}^H\mathbf{A})^{-1}\mathbf{A}^H \tag{1.14}$$

denote the orthogonal projections onto the column space of $\mathbf{A}$ and onto the null space of $\mathbf{A}^H$, respectively.

The ith standard basis vector for the k-dimensional space is denoted by $\boldsymbol{\delta}_i^{(k)}$. We use $\mathbf{I}_n$ to denote the $n \times n$ identity matrix, and $\mathbf{0}_{m\times n}$ or $\mathbf{O}_{m,n}$ to denote the $m \times n$ zero matrix (the subscripts m and n may be dropped if the dimensions are clear from the context), and $\mathrm{E}[\cdot]$ is the expectation operator. For a set $\mathcal{S}$, $|\mathcal{S}|$ denotes its cardinality. Finally, δ_{ik} denotes the Kronecker delta, i.e., $\delta_{ik} = 1$ if $i = k$ and $\delta_{ik} = 0$ if $i \neq k$, and we also define $\delta(n) = \delta_{n0}$.

1.7 List of Acronyms

Acronyms used in this thesis are listed below for easy reference.

- 1-D: 1-dimensional

- 2-D: 2-dimensional
- CBS: convolutional beamspace
- CRB: Cramér–Rao bound
- DFT: discrete Fourier transform
- DOA: direction of arrival
- DOD: direction of departure
- ESPRIT: estimation of signal parameters via rotational invariance techniques
- EVD: eigenvalue decomposition
- FFT: fast Fourier transform
- FIR: finite implulse response
- FOCUSS: focal underdetermined system solver
- IIR: infinite implulse response
- MIMO: multiple input multiple output
- mmWave: millimeter wave
- MRA: minimum redundancy array
- MSE: mean square error
- MUSIC: multiple signal classification
- RF: radio frequency
- RMSE: root mean square error
- ULA: uniform linear array
- URA: uniform rectangular array

Chapter 2

BASICS OF CONVOLUTIONAL BEAMSPACE FOR DOA ESTIMATION

2.1 Introduction

In array signal processing, the use of beamforming prior to high-resolution estimation of directions of arrival (DOA), referred to as *beamspace* processing, is well-known in the literature [8, 12–15], and continues to be of current research interest [19, 54, 55]. Given an N-sensor array with output $\mathbf{x} \in \mathbb{C}^N$, the idea of beamspace is to compute a transformation $\mathbf{y} = \mathbf{T}\mathbf{x} \in \mathbb{C}^B$, where $B < N$, and estimate the DOAs using $\mathbf{y}$. For example, the covariance of $\mathbf{y}$ can be estimated from its snapshots, and its signal and noise eigenspaces are analyzed to perform DOA estimation as in MUSIC [52], root-MUSIC [26], or ESPRIT [27].

One of the major advantages of beamspace processing is complexity reduction. Due to dimensionality reduction ($B < N$), the $B \times B$ covariance of $\mathbf{y}$ has smaller size than that of $\mathbf{x}$. So the complexity of the eigenspace computation $O(B^3)$, is much smaller than $O(N^3)$, which is the complexity when using *element-space* ($\mathbf{T} = \mathbf{I}$) directly. If $\mathbf{T}$ is carefully chosen, then the DOAs which fall outside a chosen subband in $[-\pi/2, \pi/2)$ are attenuated by $\mathbf{T}$, so there are typically much fewer DOAs represented by $\mathbf{y}$, compared to $\mathbf{x}$. One often uses a bank of transformations $\{\mathbf{T}_i\}$, which can be operated *in parallel*, to cover all DOAs in $[-\pi/2, \pi/2)$.

Besides low computation and parallelism, there are other advantages for beamspace. Beamspace methods tend to have smaller SNR threshold for resolution of closely spaced sources [8, 16, 17]. Beamspace estimates typically have smaller bias (and about the same mean square error) when compared with element-space estimates [2].

However, to successfully perform root-MUSIC after the classical beamspace transformation, one has to take elaborate steps [2]. Specifically, rows of $\mathbf{T}$ are chosen to be columns of the DFT matrix, producing beams exhibiting common out-of-band nulls. These nulls result in spurious roots outside the subband, independent of the true in-band DOAs. The spurious roots need to be factored out to reduce the degree of the polynomial to be rooted so as to lower the complexity. Moreover, this method suffers from numerical sensitivity issues [2] for even moderately large array size N,

such as $N = 48$. Similarly, classical beamspace transformation also compromises the shift-invariance structure required by ESPRIT. One has to choose $\mathbf{T}$ to have the same shift-invariance structure so that the lost shift-invariance structure can be restored with a modified ESPRIT algorithm [18].

2.1.1 Contributions of This Chapter

In this chapter, we introduce a new approach called the *convolutional beamspace (CBS)* approach. It enjoys the advantages of classical beamspace such as lower computational complexity, increased parallelism of subband processing, and improved resolution threshold for DOA estimation. Furthermore, unlike classical beamspace methods, it allows root-MUSIC and ESPRIT to be performed directly for uniform linear arrays (ULAs) without additional preparation since the Vandermonde structure and the shift-invariance are preserved under the CBS transformation. The computational benefits of the proposed methods are quantified and demonstrated throughout.

Unlike classical beamspace, CBS is based on the use of filtering with a finite-impulse-response (FIR) filter $H(z)$ followed by uniform downsampling (decimation) by an appropriate integer M. For large arrays, which are becoming increasingly important [9, 45], the filter can be proportionately longer, offering very effective attenuation of out-of-band DOAs. CBS produces more accurate DOA estimates than classical beamspace, and for correlated sources it often produces better estimates than element-space as well. Crucial to the CBS method is the extraction of a steady-state component from the convolutional layer, as we shall see.

We also provide a basic approximate error analysis of the CBS method, based on error analysis for MUSIC [56] (Sec. 2.2.6). For uncorrelated sources, the error variance of CBS estimates is shown to be close to that of the element-space. However, for correlated sources, CBS can be significantly better. These conclusions are also verified with simulations. Note that a more rigorous error analysis will be given in Chapter 5.

The CBS method in Sec. 2.2 is for the ULA, but we show in Sec. 2.3 that it can be extended to sparse arrays by appealing to the difference coarray of the original array. Difference coarrays of sparse arrays such as the MRA [3], nested array [32], coprime array [40], and their generalizations [57, 58] contain a large ULA segment. Since the autocorrelation of measured data can be estimated for all lags on the coarray, one can perform FIR filtering of the correlation supported on the ULA part of the

coarray to produce a convenient CBS. This coarray CBS offers great computational reduction. The advantage of sparse arrays, namely the ability to identify $O(N^2)$ uncorrelated sources with N sensors [32, 40], can be harnessed even while taking advantage of the benefits of the CBS transformation.

We also show how CBS ideas can be used in the context of sparse signal representation with dictionaries. The use of sparse representation techniques for DOA estimation has been studied in [59] where a dictionary of steering vectors (corresponding to a dense grid of potential DOAs) is used to represent the array output. The sparse solution to this representation problem reveals the DOAs. We show how this problem can be simplified computationally by use of CBS techniques. Besides its significant computational advantage, the method also produces more accurate DOA estimates. Since only the filter responses $H(e^{j\omega_k})$ at the discrete frequencies on the dictionary grid are relevant at the convolutional layer, we also address the interesting problem of designing discrete-frequency FIR filters for CBS dictionaries. Eigen-based methods such as MUSIC and dictionary-based sparse recovery methods are both well-known DOA estimation algorithms in the literature. In this chapter, we show that CBS can be applied to both methods, and to coarray based methods.

Besides, a new variant of CBS, called Capon-CBS, is proposed. The idea is to design the CBS filter to be a sliding Capon beamformer. Such design takes input statistics into account, so it can do a better job of suppressing the sources that fall in the stopband. Capon-CBS can offer higher probability of resolution and smaller mean square error for DOA estimation, as demonstrated in the simulations. Moreover, like traditional CBS, Capon-CBS also has the advantage of low computational complexity.

Finally, we introduce another variant of CBS which uses infinite-impulse-response (IIR) instead of FIR filters in the convolutional layer. In CBS, the sources falling in the stopband are assumed to be sufficiently attenuated so that we can identify the passband DOAs. An IIR filter often requires a much lower order for the same set of magnitude response specifications. Hence, the computational complexity of IIR-CBS can be smaller than FIR-CBS. Moreover, IIR-CBS can even give smaller DOA estimation errors than FIR-CBS because the longer FIR filter length means shorter steady-state filter output length, which leads to larger estimation errors. The advantages of IIR-CBS are verified by numerical examples.

2.1.2 Related Past Work

The use of convolution (digital filtering) prior to frequency estimation for time-domain sum-of-sinusoids was introduced many years ago by Silverstein, et al. [60], and studied in detail in [61]. But these methods, and many of the details in [61], are not directly applicable to spatial arrays. The purpose of this chapter is to develop the appropriate formulation for spatial arrays, and provide several extensions, such as extensions to coarrays and to dictionary methods.

In [62], an alternating low-rank decomposition (ALRD) approach is proposed. Each row of the beamspace transformation matrix $\mathbf{T}$ contains a basis vector to be optimized. A constrained optimization problem based on Capon's minimum-variance criterion is tackled by alternately solving for the basis vectors and the beamforming weight vector using the recursive least squares (RLS) method. A modified-ALRD (MALRD) scheme [62], where only a single basis vector is used, is also proposed to reduce computational complexity. In the MALRD scheme, the transformation matrix $\mathbf{T}$ is equivalent to doing convolution followed by uniform decimation. However, CBS differs from MALRD in that only CBS uses the idea of digital filtering so that standard filter design methods such as the minimax or equiripple method, the window method, and so on [63], can be applied. Moreover, only CBS exploits the Vandermonde structure of the convolutional steady state so that root-MUSIC and ESPRIT can be applied directly. Also note that MALRD is a special case of the joint iterative optimization (JIO) algorithms [64, 65]. Besides the uniform decimation used in MALRD, non-uniform decimation is also considered in [65]. But to preserve the Vandermonde structure of ULAs so that root-MUSIC and ESPRIT can be readily applied after decimation, we always use uniform decimation for CBS in this chapter. (Nonuniform decimation, which should be used with difference coarray method [32], is considered in Chapter 3.) Moreover, we will show by simulation (Fig. 2.7) that there is no loss of performance in using uniform decimation since it achieves the same error variance as not doing decimation (i.e., keeping all samples) does.

Chapter outline: The basic idea of CBS for ULAs is introduced in Sec. 2.2, and details of dimension reduction using uniform decimation, noise whitening in the reduced space using Nyquist filter design, computational complexity and error analysis are also presented. The extension to sparse arrays based on difference coarrays is addressed in Sec. 2.3. CBS for dictionary-based sparse signal recovery is then discussed in Sec. 2.4. Capon-CBS is introduced in Sec. 2.5. CBS based

on IIR filters is proposed in Sec. 2.6. Simulations are also given in Secs. 2.2, 2.3, 2.4, 2.5, and 2.6 to demonstrate the performance of the new methods. Sec. 2.7 concludes the chapter.

2.2 Convolutional Beamspace for Uniform Linear Arrays

We consider an N-sensor ULA with sensor spacing $\lambda/2$, and monochromatic plane waves of wavelength λ arriving from D directions. As explained in Sec. 1.1, the array output equation is

$$\mathbf{x} = \mathbf{A}\mathbf{c} + \mathbf{e}, \tag{2.1}$$

where $\mathbf{c}$ contains source amplitudes c_i, $\mathbf{e}$ contains additive noise terms, and

$$\mathbf{A} = [\mathbf{a}_N(\omega_1)\ \mathbf{a}_N(\omega_2) \cdots \mathbf{a}_N(\omega_D)] \tag{2.2}$$

with $\mathbf{a}_N(\omega) = [1\ e^{j\omega}\ e^{j2\omega}\ \cdots\ e^{j(N-1)\omega}]^T$, so that $\mathbf{A}$ is a Vandermonde matrix. Here $\omega = \pi \sin\theta$, with DOA $\theta \in [-\pi/2, \pi/2)$ measured from the normal to the line of array. We assume $\mathrm{E}[\mathbf{c}] = \mathbf{0}$, $\mathrm{E}[\mathbf{e}] = \mathbf{0}$, $\mathrm{E}[\mathbf{e}\mathbf{e}^H] = \sigma_e^2\mathbf{I}$, and $\mathrm{E}[\mathbf{c}\mathbf{e}^H] = \mathbf{0}$.

The main results of this section are as follows. In Sec. 2.2.1, we show that the convolutional steady state of the CBS output bears the same structure as the ULA output (2.1), as shown in (2.9). In Sec. 2.2.2, we show how to use uniform decimation to reduce computational complexity. The decimated output (2.14) still has the Vandermonde structure of the ULA output (2.1). Hence, root-MUSIC and ESPRIT can be directly applied to the decimated covariance (2.19) without further adjustment, as explained in Sec. 2.2.3. In Sec. 2.2.4, it is shown that by choosing the filter used in CBS to be a spectral factor of Nyquist filters as in (2.25), we can whiten the noise term after decimation. Then, the computational complexity of CBS is compared to various methods in the literature in Sec. 2.2.5, and a basic approximate error analysis is given in Sec. 2.2.6. Finally, simulations are presented in Sec. 2.2.7.

2.2.1 The Convolutional Steady State

Let $x(n), 0 \le n \le N-1$ be the output of the N-sensor ULA. We convolve this sequence with an FIR filter with transfer function

$$H(z) = \sum_{n=0}^{L-1} h(n) z^{-n} \tag{2.3}$$

with $L < N$ to get the possibly nonzero output samples $y(n), 0 \le n \le N + L - 2$. Of these, only

$$y(L-1), y(L), \cdots, y(N-1) \tag{2.4}$$

involve all the L filter coefficients, and can be considered *steady state* output samples:

$$\mathbf{y} \triangleq \begin{bmatrix} y(L-1) \\ y(L) \\ \vdots \\ y(N-1) \end{bmatrix} = \mathbf{H} \begin{bmatrix} x(0) \\ x(1) \\ \vdots \\ x(N-1) \end{bmatrix} = \mathbf{H}\mathbf{x}, \tag{2.5}$$

where $\mathbf{H}$ is a $(N - L + 1) \times N$ banded Toeplitz matrix:

$$\mathbf{H} = \begin{bmatrix} h(L-1) & \cdots & h(0) & 0 & \cdots & 0 \\ 0 & h(L-1) & \cdots & h(0) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & h(L-1) & \cdots & h(0) \end{bmatrix}. \tag{2.6}$$

For example, $y(L-2)$ does not contain $h(L-1)$ (as $x(-1) = 0$) and $y(N)$ does not contain $h(0)$ (as $x(N) = 0$). So these are not part of the steady state output (2.5). The steady state samples $\mathbf{y}$ are obtained by sliding the reversed weights $h(k)$ from left to right uniformly, as shown in Fig. 2.1. We call $\mathbf{y}$ the *convolutional beamspace signal.* In contrast, in classical beamspace $\mathbf{y} = \mathbf{T}\mathbf{x}$, $\mathbf{T}$ is a fat $B \times N$ matrix, but without any Toeplitz structure. For instance, a popular choice is to let $\mathbf{T}$ contain B consecutive rows of the $N \times N$ DFT matrix [2]. As we shall see, the banded Toeplitz structure of $\mathbf{H}$ is essential to obtain a Vandermonde structure in $\mathbf{y}$.

Now assume we have a signal arriving from DOA $\bar{\theta}$ so that $x(n) = e^{j\omega n}, 0 \le n \le N-1$ (up to some scale, which we ignore), where $\omega = \pi \sin\bar{\theta}$. Then from the steady state equation (2.5), ignoring noise, we have

$$\mathbf{y} = e^{j(L-1)\omega} H(e^{j\omega}) \begin{bmatrix} 1 & e^{j\omega} & \cdots & e^{j(N-L)\omega} \end{bmatrix}^T, \tag{2.7}$$

where $H(z) = \sum_{n=0}^{L-1} h(n) z^{-n}$. So the CBS signal $\mathbf{y}$ in response to a single DOA is a *Vandermonde vector* just like the array output vector $\mathbf{x} = \mathbf{a}_N(\omega) = [1 \; e^{j\omega} \; e^{j2\omega} \; \cdots \; e^{j(N-1)\omega}]^T$. Moreover, $\mathbf{y}$ is scaled by the filter frequency response $H(e^{j\omega})$. Thus if there are D sources with DOAs ω_k, then since $x(n) = \sum_{k=1}^{D} c_k e^{j\omega_k n}$, we have

$$\mathbf{y} = \sum_{k=1}^{D} c_k e^{j(L-1)\omega_k} H(e^{j\omega_k}) \mathbf{a}_{N-L+1}(\omega_k) + \mathbf{H}\mathbf{e}. \tag{2.8}$$

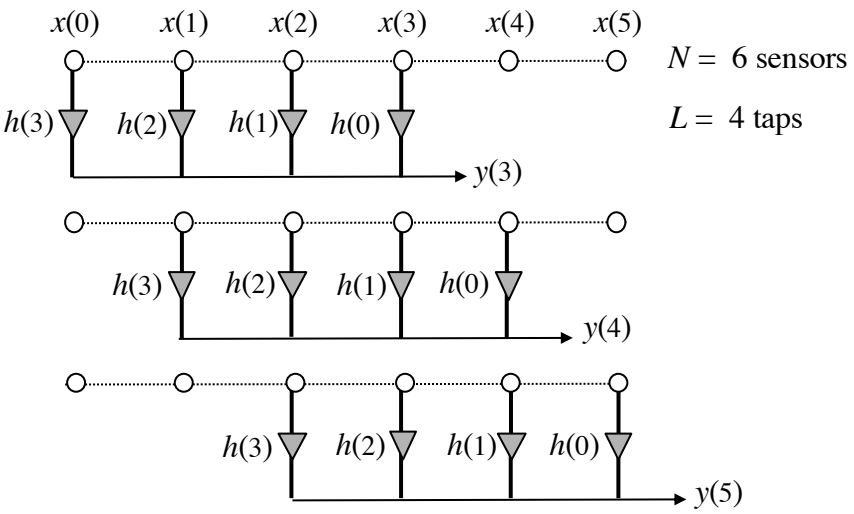


Figure 2.1: The steady state CBS signal $\mathbf{y} = [y(3)\ y(4)\ y(5)]^T$ generated by sliding the weights $h(k)$ over the sensors.

The arriving signals with DOAs ω_k are therefore filtered by the response $H(e^{j\omega})$. Thus the array equation (2.1) is replaced with

$$\mathbf{y} = \mathbf{A}_L\mathbf{d} + \mathbf{He}, \tag{2.9}$$

where $\mathbf{A}_L$ is a Vandermonde matrix obtained from $\mathbf{A}$ by keeping the first $N - L + 1$ rows, and $\mathbf{d}$ has elements $d_k = c_k e^{j(L-1)\omega_k} H(e^{j\omega_k})$. While the development is valid for any ULA, for large arrays (large N), which are getting more attention recently [5, 9, 11], we can make L large and design a sharp-cutoff filter with good stopband. Assuming signals in the stopband are not too strong so that $\mathbf{y}$ contains only those DOAs that fall in the passband of $H(e^{j\omega})$, we have

$$\mathbf{y} \approx \mathbf{A}_{L,0}\mathbf{d}_0 + \mathbf{He}. \tag{2.10}$$

Here $\mathbf{A}_{L,0}$ has D_0 columns of $\mathbf{A}_L$ corresponding to the D_0 sources that fall in the passband of $H(e^{j\omega})$, and $\mathbf{d}_0$ has the corresponding D_0 rows of $\mathbf{d}$. Fig. 2.2 shows a typical filter response, with two out of six DOAs falling in the passband. Since $\omega = \pi \sin\theta$, the DOA range $-\pi/2 \le \theta < \pi/2$ corresponds to $-\pi \le \omega < \pi$; it can equivalently be taken as $0 \le \omega < 2\pi$, as $H(e^{j\omega})$ has period 2π. The FIR filter $H(z)$ can be designed by any standard method such as the minimax or equiripple method, the window method, and so on [63]. If the filter does not have sharp cutoff, it is likely that a DOA falls in the transition band, which requires more careful consideration.

Note that we can process the array output $x(n)$ with an entire *filter bank* $H_i(e^{j\omega}), 0 \le i \le M - 1$ to cover the full DOA range $0 \le \omega < 2\pi$, as in Fig. 2.3. The outputs of

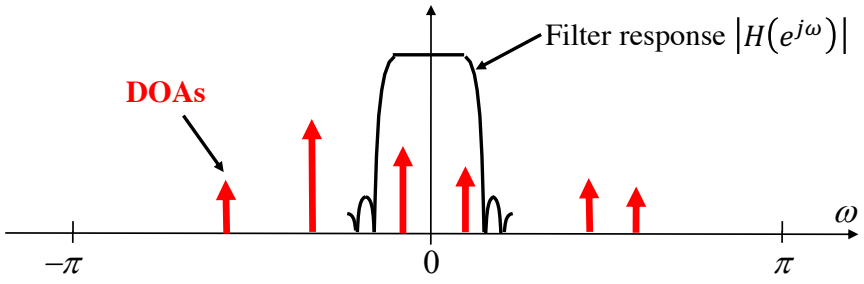


Figure 2.2: Typical magnitude response $|H(e^{j\omega})|$, and example of DOA locations (red arrows). Two of the six DOAs are in the passband.

filters can be processed *in parallel* to estimate all D DOAs. The DOA estimation procedure would be to first estimate the number of DOAs D_0 from $\mathbf{y}$, and identify these D_0 DOAs using standard methods. Since the filter output $\mathbf{y}$ is represented in terms of the Vandermonde matrix $\mathbf{A}_L$ just like the original array output $\mathbf{x}$, we can use root-MUSIC or ESPRIT without any further adjustment or processing to the data. This is an advantage of the proposed CBS method compared to classical beamspace methods, for which root-MUSIC requires some preprocessing [2] (due to loss of Vandermonde structure), and so does ESPRIT [18] (due to loss of shift-invariance). The method, as presented, works best for large ULAs, but can be extended to sparse arrays with relatively few sensor elements, as we shall see in Sec. 2.3.

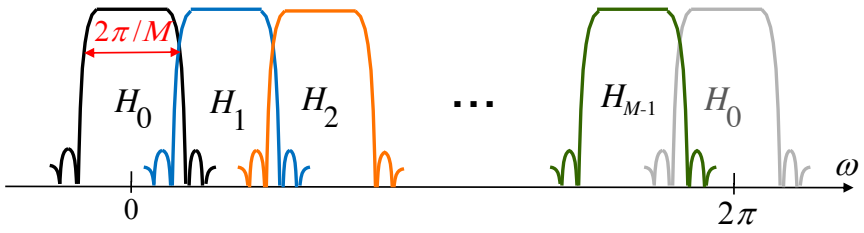


Figure 2.3: Typical beamspace filter bank (magnitude responses).

2.2.2 Decimating the Filter Output

In classical beamspace methods, the complexity advantage is obtained because $B \ll N$. However, for CBS described in the Sec. 2.2.1, $N - L + 1 \approx N$ since $L \ll N$ in practice. To achieve the complexity reduction of beamspace methods, we simply *decimate* $y(n)$ with a uniform downsampler. Since the passband of $H(z)$ has width $\approx 2\pi/M$ (Fig. 2.3), we can decimate $y(n)$ by the integer M. For larger arrays, L can be large, and the filters can be designed with sharp cutoff

and good stopband attenuation to minimize aliasing due to decimation [66]. Let $v(n) = y(n + L - 1)$ so that $\mathbf{y} = [v(0)\ v(1)\ \cdots\ v(N - L)]^T$. Define the decimated version $v_0(n) = v(Mn)$. The vector $\mathbf{y}$ is then replaced by the decimated vector $\bar{\mathbf{v}}_0 = [v(0)\ v(M) \cdots v(J_0 M)]^T$, where $J_0 = \lceil (N - L + 1)/M \rceil$. We can estimate the $J_0 \times J_0$ covariance of $\bar{\mathbf{v}}_0$ from snapshots and estimate the D_0 DOAs in the passband, if $D_0 < J_0$. The complexity of eigenspace computation is now

$$O(J_0^3) \ll O(N^3). \tag{2.11}$$

One might think that decimation leads to "waste" of hard-earned data, but we can make good use of essentially *all* data while estimating a $J \times J$ covariance, where $J = \lfloor (N - L + 1)/M \rfloor$. Consider shifted versions $v(n + l)$ for $0 \le l \le M - 1$ and define their decimated versions $v_l(n) = v(Mn + l)$. These are the *polyphase components* of $v(n)$ [66]. Let $\mathbf{v}_l = [v_l(0)\ v_l(1)\ \cdots v_l(J - 1)]^T$, that is,

$$\mathbf{v}_l = [v(l)\ v(l + M)\ \cdots\ v(l + (J - 1)M)\ ]^T. \tag{2.12}$$

We will estimate $J \times J$ covariances of $\mathbf{v}_l$ and average over all l to obtain a "coherent" estimate of the $J \times J$ covariance of decimated CBS data. Note that we took the floor function of J to accommodate the shorter polyphase components.

Remark: We use *uniform decimation* [67, Sec. II] in the basic form of CBS in this chapter, whereas *nonuniform decimation* will be also considered in Chapter 3. See Sec. 3.2 for more details.

2.2.3 Decimated Covariance and DOA Estimation

In the following, we develop the decimated covariance matrix. Let

$$\mathbf{D}_l = [\boldsymbol{\delta}_l\ \boldsymbol{\delta}_{l+M}\ \ldots\ \boldsymbol{\delta}_{l+(J-1)M}]^T \tag{2.13}$$

be a decimation matrix, where $\boldsymbol{\delta}_l$ is the lth standard basis vector for the $(N - L + 1)$-dimensional space. Then we can write $\mathbf{v}_l = \mathbf{D}_l \mathbf{y}$. From (2.9) we have $\mathbf{v}_l = \mathbf{D}_l \mathbf{y} = \mathbf{D}_l \mathbf{A}_L \mathbf{d} + \mathbf{D}_l \mathbf{H} \mathbf{e}$. It can be verified that this simplifies to

$$\mathbf{v}_l = \mathbf{A}_{\text{dec}} \mathbf{d}_l + \mathbf{D}_l \mathbf{H} \mathbf{e}, \tag{2.14}$$

where

$$\mathbf{A}_{\text{dec}} = [\mathbf{a}_J(M\omega_1)\ \mathbf{a}_J(M\omega_2)\ \cdots\ \mathbf{a}_J(M\omega_D)] \tag{2.15}$$

with

$$\mathbf{d}_l = \left[c_1 e^{j(L-1+l)\omega_1} H(e^{j\omega_1})\ \cdots\ c_D e^{j(L-1+l)\omega_D} H(e^{j\omega_D}) \right]^T.$$

Thus with $\mathbf{R}_{d_l} = \mathrm{E}[\mathbf{d}_l\mathbf{d}_l^H]$, the covariance of $\mathbf{v}_l$ is

$$\mathbf{R}_{v_l} = \mathrm{E}[\mathbf{v}_l\mathbf{v}_l^H] = \mathbf{A}_{\mathrm{dec}}\mathbf{R}_{d_l}\mathbf{A}_{\mathrm{dec}}^H + \sigma_e^2\mathbf{D}_l\mathbf{H}\mathbf{H}^H\mathbf{D}_l^H. \tag{2.16}$$

It can be verified that the "decimated" matrix

$$\mathbf{G}_{\mathrm{dec}} \triangleq \mathbf{D}_l\mathbf{H}\mathbf{H}^H\mathbf{D}_l \tag{2.17}$$

is *independent* of l (see Sec. 2.2.4 for details), so

$$\mathbf{R}_{v_l} = \mathbf{A}_{\mathrm{dec}}\mathbf{R}_{d_l}\mathbf{A}_{\mathrm{dec}}^H + \sigma_e^2\mathbf{G}_{\mathrm{dec}}. \tag{2.18}$$

The dependency of the first term of (2.18) on l can be averaged out:

$$\mathbf{R}_{\mathrm{ave}} = \frac{1}{M}\sum_{l=0}^{M-1}\mathbf{R}_{v_l} = \mathbf{A}_{\mathrm{dec}}\breve{\mathbf{R}}_d\mathbf{A}_{\mathrm{dec}}^H + \sigma_e^2\mathbf{G}_{\mathrm{dec}}, \tag{2.19}$$

where $\breve{\mathbf{R}}_d$ is $\mathbf{R}_{d_l}$ averaged over l. In practice, we estimate $\mathbf{R}_{v_l}$ from snapshots for each l, and then estimate $\mathbf{R}_{\mathrm{ave}}$. This is the estimated $J \times J$ covariance to be used for estimating DOAs in the filter passband. Since $\mathbf{v}_l$ for all l are used, *all* the $N - L + 1$ components of the CBS signal $\mathbf{y}$ are exploited (if $N - L + 1$ is a multiple of M), and no data is wasted. In (2.19), the Vandermonde structure is preserved, so we can directly use root-MUSIC. ESPRIT is also applicable as shift invariance is retained.

Note that since the columns of $\mathbf{A}_{\mathrm{dec}}$ are $\mathbf{a}_J(M\omega_i)$ rather than $\mathbf{a}_J(\omega_i)$, we can only estimate $M\omega_i \bmod 2\pi$, or equivalently

$$\omega_i + 2\pi s_i/M, \tag{2.20}$$

where the integers s_i are unknown, creating *ambiguity*. But since ω_i are known to be in the passband of $H(e^{j\omega})$ which has width $2\pi/M$, the ambiguities s_i can be resolved because for each DOA, there is only one integer s_i such that $\omega_i + 2\pi s_i/M$ is within the passband. We will show by simulation the effectiveness of this decimation method, as it achieves almost the same performance as that obtained when we do eigenspace computation directly on $\mathbf{y}$, which is of much higher complexity $O(N^3)$.

Remark: One merit of (2.19) is that the decimated signals are combined "*coherently*." Note that the ith diagonal element of $\breve{\mathbf{R}}_d$ is

$$[\breve{\mathbf{R}}_d]_{ii} = \frac{1}{M}\sum_{l=0}^{M-1}\mathrm{E}[|c_i e^{j(L-1+l)\omega_i}H(e^{j\omega_i})|^2] \tag{2.21}$$

$$= \frac{1}{M}\sum_{l=0}^{M-1}p_i|H(e^{j\omega_i})|^2 = p_i|H(e^{j\omega_i})|^2, \tag{2.22}$$

where $p_i = \mathrm{E}[|c_i|^2]$ is the power of the ith source. This explains the "coherent" property of the method as the signal powers $[\breve{\mathbf{R}}_{d_l}]_{ii} = p_i|H(e^{j\omega_i})|^2 \geq 0$, $\forall\ l$ are combined coherently. Actually, when the sources are uncorrelated, $\breve{\mathbf{R}}_{d_l}$ is the same diagonal matrix for all l so that $\breve{\mathbf{R}}_d = \breve{\mathbf{R}}_{d_l}$ for all l, and $\mathbf{R}_{\text{ave}}$ in (2.19) is exactly equal to $\mathbf{R}_{v_l}$ in (2.18) for all l. Ideally, if we have the exact autocorrelation matrices $\mathbf{R}_{v_l}$, then there is no need to do this average to obtain the same matrix. However, in practice, we only have estimates of $\mathbf{R}_{v_l}$ from snapshots, so this average is helpful to performance. We will show by simulation that this coherent method outperforms the method using only one polyphase component $\mathbf{v}_0$. We will also see that it is also better than the *noncoherent* method, which just computes the average of DOA estimates obtained separately from eigenspace method using each $\mathbf{v}_l$.

2.2.4 Spectral Factors of Nyquist Filters to Whiten Noise

The undecimated output of convolution (2.9) has covariance $\mathbf{R}_{\mathbf{yy}} = \mathbf{A}_L\mathbf{R}_d\mathbf{A}_L^H + \sigma_e^2\mathbf{G}$, where $\mathbf{R}_d = \mathrm{E}[\mathbf{d}\mathbf{d}^H]$ and $\mathbf{G} = \mathbf{H}\mathbf{H}^H$ is Hermitian and Toeplitz with first row $[g(0)\ g^*(1)\ g^*(2)\ \cdots\ g^*(N-L)]$, where

$$g(k) = \sum_n h(n)h^*(n-k) \tag{2.23}$$

is the deterministic autocorrelation of $h(n)$. The noise term $\sigma_e^2\mathbf{G}$ cannot be a diagonal matrix unless the filter has the trivial form $H(z) = cz^{-n_0}$ [66]. But the decimated output (2.14) has covariance (2.18) for all l. It can be verified that $\mathbf{G}_{\text{dec}} = \mathbf{D}_l\mathbf{H}\mathbf{H}^H\mathbf{D}_l$ is $J \times J$ Hermitian and Toeplitz with first row

$$[g(0)\ g^*(M)\ g^*(2M)\ \cdots\ g^*((J-1)M)], \tag{2.24}$$

which is *independent* of l. Thus, whereas $\mathbf{G}$ is the autocorrelation matrix of $h(n)$, the matrix $\mathbf{G}_{\text{dec}}$ is constructed from the decimated autocorrelation $g(Mk)$, and does not depend on l. So the corresponding noise term can be whitened by making $\mathbf{G}_{\text{dec}} = \mathbf{I}$, or equivalently

$$g(Mk) = \delta(k), \tag{2.25}$$

where $g(k)$ is as in (2.23). Eq. (2.25) is called the *Nyquist*(M) property of $g(k)$. Since $|H(e^{j\omega})|^2$ is the Fourier transform of $g(k)$, we say that $H(z)$ is a *spectral factor* of the Nyquist(M) filter $|H(e^{j\omega})|^2$. In short, by designing the FIR filter $H(z)$ to be a spectral factor of an FIR Nyquist(M) filter $G(z)$ with $G(e^{j\omega}) \geq 0$, we can ensure that the noise terms in the decimated versions $v(Mn+l)$ are white for all l. So $\mathbf{R}_{\text{ave}}$

becomes $\mathbf{R}_{\text{ave}} = \mathbf{A}_{\text{dec}}\check{\mathbf{R}}_d\mathbf{A}_{\text{dec}}^H + \sigma_e^2\mathbf{I}$ where $\mathbf{A}_{\text{dec}}$ is as in (2.15). This makes it easy to find the noise eigenspace by computing eigenvectors of $\mathbf{R}_{\text{ave}}$, which is what we do in simulations. Spectral factors of Nyquist filters arise in digital communications [68] and in filter bank theory [66]. There are many ways to design such filters [69–72]. In fact, any filter $H_k(e^{j\omega})$ in an *orthonormal* (equivalently paraunitary) filter bank is automatically a spectral factor of a Nyquist filter [66]. Many examples of good FIR designs with this property can be found in the literature [66, 70, 73–76]. In fact, if $H(e^{j\omega})$ is a "good" filter with total passband width $\approx 2\pi/M$ and ripples properly constrained, this Nyquist property (2.25) is approximately satisfied, that is,

$$\sum_{n\neq 0} |g(Mn)| \ll g(0) \quad \text{(nearly-Nyquist property)}. \tag{2.26}$$

For simplicity, this is what we use in simulations.

In classical beamspace transformation $\mathbf{y} = \mathbf{T}\mathbf{x}$, the orthogonality condition $\mathbf{T}\mathbf{T}^H = \mathbf{I}$ is normally imposed [77]. The Nyquist condition (2.25) is analogous to this because we are imposing $\mathbf{G}_{\text{dec}} = \mathbf{T}\mathbf{T}^H = \mathbf{I}$, where $\mathbf{T} = \mathbf{D}_l\mathbf{H}$.

2.2.5 Computational Complexity

The computational complexity of CBS is compared to various methods in the literature in Table 2.1. For element-space root-MUSIC and ESPRIT, the complexity is dominated by the eigenvalue decomposition of the $N \times N$ covariance of the array output $\mathbf{x}$ defined in (2.1). The complexity is $O(N^3)$. For element-space MUSIC, an additional $O(GN^2)$ is required to compute the MUSIC spectrum $P(\omega) = (\mathbf{a}_N^H(\omega)\mathbf{E}_{\text{n}}\mathbf{E}_{\text{n}}^H\mathbf{a}_N(\omega))^{-1}$, where $\mathbf{E}_{\text{n}}$ is the noise subspace, and G is the number of grid points used for grid search of ω. For classical beamspace, the complexity is dominated by the eigenvalue decomposition of the $B \times B$ covariance of $\mathbf{y} = \mathbf{T}\mathbf{x}$, where $\mathbf{T}$ is a $B \times N$ beamspace transformation matrix. The complexity is $O(B^3)$. Similarly, classical beamspace MUSIC requires an additional $O(GB^2)$ to compute the MUSIC spectrum. For CBS, the complexity is dominated by the eigenvalue decomposition of the covariance (2.19) for the decimated signals. Assuming the filter length $L \ll N$ so that $(N - L + 1)/M \approx N/M$, the complexity is $O((N/M)^3)$, where M is the decimation ratio. Similarly, CBS MUSIC requires an additional $O(G(N/M)^2)$ to compute the MUSIC spectrum. In Table 2.1, we give an example of typical numbers by setting $N = 100$, $G = 200$, $M = 4$, and $B = N/M = 25$. Note that CBS and classical beamspace have the same complexity, which is much smaller than that of element-space. A direct comparison of running time for each algorithm is also presented in Table 2.2.

Table 2.1: Comparison of computational complexity

Algorithm	Complexity	Typical numbers
Element-space MUSIC [52]	$O(N^3 + GN^2)$	3×10^6
Element-space root-MUSIC [26]	$O(N^3)$	10^6
Element-space ESPRIT [27]	$O(N^3)$	10^6
Classical beamspace MUSIC [16]	$O(B^3 + GB^2)$	1.41×10^5
Classical beamspace root-MUSIC [2]	$O(B^3)$	1.56×10^4
Classical beamspace ESPRIT [18]	$O(B^3)$	1.56×10^4
CBS MUSIC [this chapter]	$O((N/M)^3 + G(N/M)^2)$	1.41×10^5
CBS root-MUSIC [this chapter]	$O((N/M)^3)$	1.56×10^4
CBS ESPRIT [this chapter]	$O((N/M)^3)$	1.56×10^4

For all expressions listed in Table 2.1, we include complexity only for computations after the covariance matrices have been estimated using snapshots. For ALRD-RLS and MALRD-RLS [62], the snapshots are used in the RLS algorithm, but no covariance is estimated explicitly. Hence, we do not include ALRD-RLS and MALRD-RLS in Table 2.1, but we list their total complexity here: $O(GK(\bar{B}I^2+\bar{B}^2))$ for ALRD-RLS and $O(GK(I^2 + \bar{B}^2))$ for MALRD-RLS, where I is the length of the basis vectors contained in the $\bar{B} \times N$ beamspace transformation matrix $\mathbf{T}$, K is the number of snapshots, and G is the number of grid points used for grid search of ω. Note that we use the notation $\bar{B}$ as it may be different from B for other algorithms. To compare ALRD-RLS and MALRD-RLS with other algorithms, we need to include complexity for estimating the covariance also. For instance, for classical beamspace root-MUSIC, the complexity of $\mathbf{y} = \mathbf{T}\mathbf{x}$ for K snapshots, where $\mathbf{T}$ is a $B \times N$ matrix, is either $O(KNB)$, or $O(KN \log_2 N)$ if rows of $\mathbf{T}$ are chosen to be columns of the DFT matrix so that the FFT algorithm can be used. Estimating the $B \times B$ covariance of $\mathbf{y}$ using K snapshots then requires $O(KB^2)$ computation. Thus, the total complexity is $O(B^3 + KB^2 + KN \min(B, \log_2 N))$. But from this expression, it is more tricky to obtain typical numbers as in Table 2.1 because the constant factors of each term in the big-O notation are hidden.

2.2.6 Error Analysis

Now we analyze the performance of CBS with decimation. This is only an approximate analysis of the mean square error (MSE) of the DOA estimates, but it provides insights. A more rigorous analysis will be given in Chapter 5. We assume MUSIC or root-MUSIC is used to estimate DOAs. According to [26, 56], MUSIC

and root-MUSIC achieve the same MSE performance asymptotically. Let $\hat{\omega}_i$ be the estimate of the ith DOA ω_i, $1 \le i \le D$. When MUSIC or root-MUSIC is performed on an N-sensor ULA with K snapshots in element-space, the estimate $\hat{\omega}_i$ is asymptotically (for large K and N) unbiased and has variance [26, 56]

$$\mathrm{var}_{\mathrm{elm}}(\hat{\omega}_i) = \frac{6\sigma_e^2}{KN^3}[\mathbf{R}_{\mathbf{cc}}^{-1}]_{ii}, \tag{2.27}$$

where $\mathbf{R}_{\mathbf{cc}} = \mathrm{E}[\mathbf{c}\mathbf{c}^H]$ is the autocorrelation of the source amplitudes. In particular, if the sources are uncorrelated with powers σ_i^2, then we have

$$\mathrm{var}_{\mathrm{elm}}(\hat{\omega}_i) = \frac{6\sigma_e^2}{KN^3\sigma_i^2}. \tag{2.28}$$

For simplicity, consider CBS with decimation using only one polyphase component, and suppose the decimation ratio M is a divisor of $(N - L + 1)$, so

$$J = (N - L + 1)/M \tag{2.29}$$

is an integer. Suppose the filter $H(e^{j\omega})$ satisfies the nearly-Nyquist property (2.26) so that $\mathbf{G}_{dec} \approx \mathbf{I}$. Let $H(e^{j\omega})$ be a good filter with passband $[-\pi/M, \pi/M]$, i.e.,

$$|H(e^{j\omega})|^2 \approx \begin{cases} M, & |\omega| < \pi/M \\ 0, & \text{otherwise} \end{cases} \tag{2.30}$$

so that it has unit energy. For CBS with decimation, since the columns of $\mathbf{A}_{dec}$ are $\mathbf{a}_J(M\omega_i)$ rather than $\mathbf{a}_J(\omega_i)$, so in view of (2.18), for an in-band DOA ω_i (DOA in the passband of the filter),

$$\mathrm{var}_{\mathrm{CBS}}(M\hat{\omega}_i) \approx \frac{6\sigma_e^2}{KJ^3|H(e^{j\omega_i})|^2\sigma_i^2}. \tag{2.31}$$

Using (2.29), (2.30), and the fact that $\mathrm{var}(cx) = c^2\mathrm{var}(x)$ for any constant $c > 0$ and random variable x, we therefore obtain

$$\mathrm{var}_{\mathrm{CBS}}(\hat{\omega}_i) \approx \frac{6\sigma_e^2}{K(N - L + 1)^3\sigma_i^2}. \tag{2.32}$$

Moreover, if the filter length $L \ll N$,

$$\mathrm{var}_{\mathrm{CBS}}(\hat{\omega}_i) \approx \frac{6\sigma_e^2}{KN^3\sigma_i^2} = \mathrm{var}_{elm}(\hat{\omega}_i). \tag{2.33}$$

Thus, the error variance of CBS is approximately independent of the decimation ratio M and equal to that of element-space. Hence, one may want to choose a large M

to lower computational complexity. However, there is some price to be paid for this advantage. Since the number of identifiable sources is limited by $J = (N-L+1)/M$, large M means fewer sources can be identified in the passband. Secondly, since a large M implies that the filter has narrower pass and transition bands, the stop band attenuation degrades for large M (for fixed filter length L).

The above analysis is valid as long as all the in-band sources are uncorrelated. Whether an in-band source is correlated with an out-of-band source (source in the stopband) or not does not matter. This also suggests how *CBS can improve performance over element-space*. For illustration, consider an example where there is one in-band source with power 1 and one out-of-band source with power P_{o}, with correlation coefficient $0 < \rho < 1$. Hence,

$$\mathbf{R}_{\mathbf{cc}} = \begin{bmatrix} 1 & \rho\sqrt{P_{\mathrm{o}}} \\ \rho\sqrt{P_{\mathrm{o}}} & P_{\mathrm{o}} \end{bmatrix} \tag{2.34}$$

and

$$\mathbf{R}_{\mathbf{cc}}^{-1} = \frac{1}{(1-\rho^2)P_{\mathrm{o}}} \begin{bmatrix} P_{\mathrm{o}} & -\rho\sqrt{P_{\mathrm{o}}} \\ -\rho\sqrt{P_{\mathrm{o}}} & 1 \end{bmatrix}, \tag{2.35}$$

so it can be derived from (2.27) that

$$\mathrm{var}_{\mathrm{elm}}(\hat{\omega}_1) = \frac{6\sigma_e^2}{KN^3(1-\rho^2)}. \tag{2.36}$$

The error variance for element-space gets larger as ρ gets larger, but is independent of the out-of-band power P_{o}. By contrast, for CBS, as long as the out-of-band source is attenuated enough by the filter so that the in-band source is the only effective source after filtering, then it is as if there is only the first source, or its effective $\mathbf{R}_{\mathbf{cc},\mathrm{CBS}}$ is a 1×1 matrix $\mathbf{R}_{\mathbf{cc},\mathrm{CBS}} = 1$. Hence, setting $\sigma_1^2 = 1$ in (2.33), we obtain

$$\mathrm{var}_{\mathrm{CBS}}(\hat{\omega}_1) \approx \frac{6\sigma_e^2}{KN^3} < \mathrm{var}_{elm}(\hat{\omega}_1). \tag{2.37}$$

It is important to note that this does not contradict the analysis for beamspace MUSIC in [77, 78] because therein the signal subspace dimension in the beamspace is assumed to be the same as that in the element-space, but for CBS, signal subspace dimension after filtering can be smaller. That is, in [77, 78], all sources, including those in the stopband if any, still have to be estimated in the beamspace, while in our case we only have to estimate in-band ones. Besides, (2.37) does not contradict the fact that the beamspace Cramér–Rao bound (CRB) cannot be smaller than the

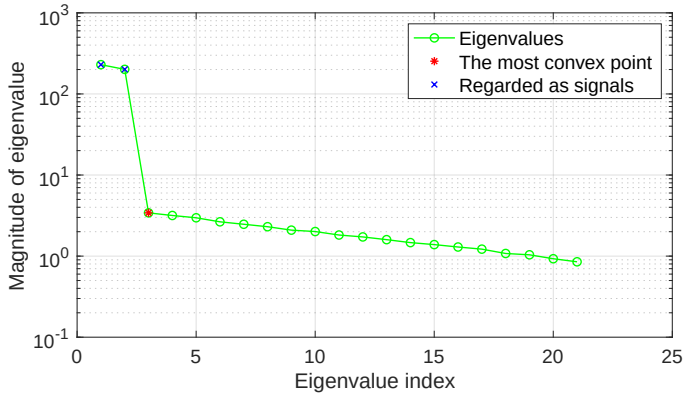
element-space CRB [78] because the gap between the error variance of a practical algorithm and the CRB can differ in different situations. The foregoing example shows that CBS can improve MUSIC MSE performance in some cases when there are correlated sources.

Remark: A more rigorous analysis will have error variance expressions involving the filter coefficients $h(n)$ or frequency response $H(e^{j\omega})$. The filtered out-of-band sources contribute to error terms that perturb the eigenvectors of the covariance matrix. Such an error term is not white and may be correlated with in-band sources. This makes a rigorous analysis complicated, and we will present it in Chapter 5. From a practical point of view, observe that if $H(e^{j\omega})$ is designed to have good stopband attenuation, then the error term due to filtered out-of-band sources is much smaller than the white noise term, and the analyses in this subsection can be good approximations.

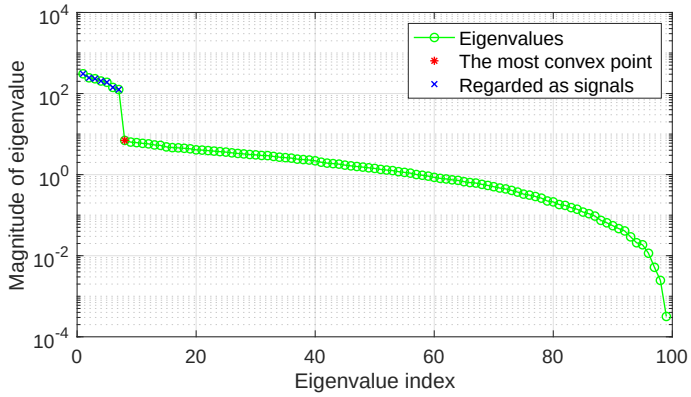
2.2.7 Simulations

In all simulation examples in this section, we assume the number of DOAs is unknown. For CBS, the number of in-band DOAs D_0 has to estimated. For element-space, the number of all DOAs D has to estimated. For CBS method to estimate D_0, we plot the distribution of eigenvalues of the covariance $\mathbf{R}_{\text{ave}}$ in descending order in log scale, and the most convex point (maximum of the second difference) of the curve is regarded as the first noise subspace eigenvalue. Then, the number of eigenvalues larger than this is the estimated D_0. For element-space to estimate D, the same method is used, with $\mathbf{R}_{\text{ave}}$ replaced by the covariance of the original array output $\mathbf{x}$. See Fig. 2.4(a)-(b) for a numerical demonstration (details of this plot will be described below). To compare with CBS using a filter $H(z)$, for element-space, we just consider DOA estimates in the passband of $H(z)$ and ignore those in the stopband. The number of in-band DOAs obtained in this way is also viewed as the estimate of D_0 for element-space.

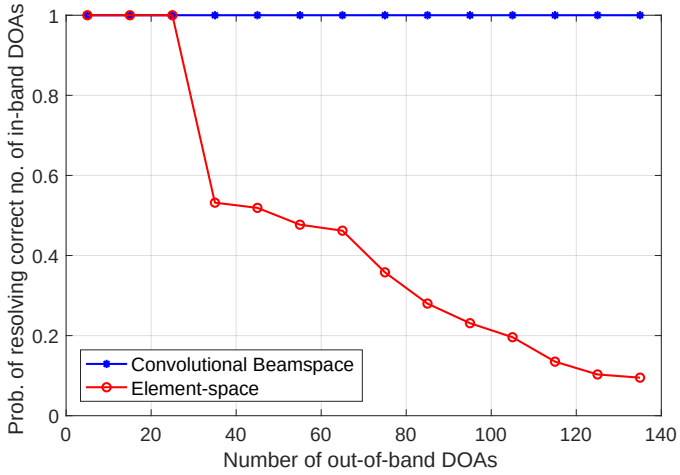
Whenever we mention root mean square errors (RMSE) in detected in-band source angles, we refer to averaging square errors measured in ω over all in-band DOAs and over those Monte Carlo runs that obtain the correct number of in-band DOAs. Similarly, since the stochastic Cramér–Rao bounds (CRBs) [44] depend on DOAs, we have averaged over in-band DOAs in the plots. Noise variance $\sigma_e^2 = 1$ is used. If not specified particularly, the following settings are used for each example. First, all sources are uncorrelated with equal powers $p_k = 1$. Second, the coherent method



(a)



(b)



(c)

Figure 2.4: Performance of CBS and element-space when there are many out-of-band DOAs. (a) Typical eigenvalue distribution for CBS. (b) Typical eigenvalue distribution for element-space. (c) Probability of resolution.

(2.19) is used for CBS with decimation. Third, $H(z)$ is designed to be lowpass using the Parks-McClellan algorithm [63], with passband edge $\pi/2M$ and stopband

edge $3\pi/2M$. Here their average π/M is viewed as the filter *cutoff*, and M is the decimation ratio. These Parks-McClellan filters satisfy (2.26).

Probability of resolution. We first consider a scenario where there are many out-of-band DOAs. In this case, CBS is especially advantageous over element-space in terms of probability of resolution. Consider a ULA with $N = 99$ sensors. The filter length is $L = 16$, and the decimation ratio is $M = 4$. There are two in-band DOAs, which are at angles $\theta = -5°, 5°$. We vary the number of out-of-band DOAs $\bar{D}$, while they are uniformly placed in the range $\omega \in [0.5\pi, 0.98\pi]$, i.e., $\omega = 0.5\pi, 0.5\pi + \delta, 0.5\pi + 2\delta, \ldots, 0.98\pi$ with $\delta = 0.48\pi/(\bar{D} - 1)$. (Recall $\omega = \pi \sin\theta$.) Fig. 2.4(c) shows the probability of resolving the correct number of in-band DOAs using the method of finding the most convex point of eigenvalue distribution. Typical eigenvalue distributions for a Monte Carlo run are shown in Fig. 2.4(a)-(b) when there are 5 out-of-band DOAs. Note that 2 and 7 eigenvalues are regarded as signals for CBS and element-space, respectively, corresponding to the number of in-band DOAs and all DOAs. Covariance estimates are obtained by using 100 snapshots, and we average 1000 Monte Carlo runs to get the plot. As expected, for CBS, the number of out-of-band DOAs does not affect the probability of resolution, which is always 1 in this example, because they are attenuated by the filter. Even when the number of out-of-band DOAs is greater than the number of sensors $N = 99$, CBS can still recover the in-band DOAs with probability one. But for element-space, the probability of resolution decreases significantly as the number of out-of-band DOAs increases.

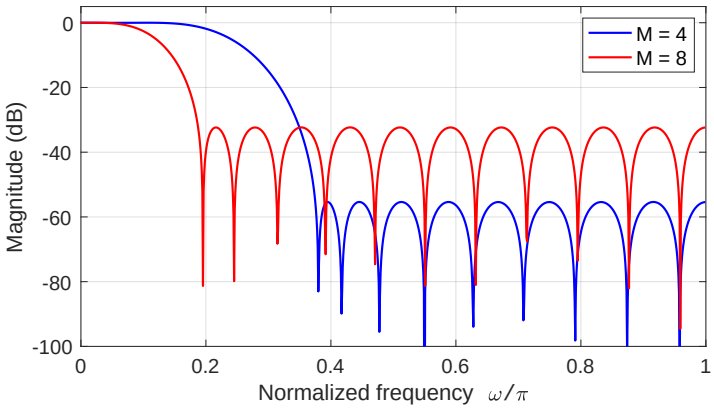
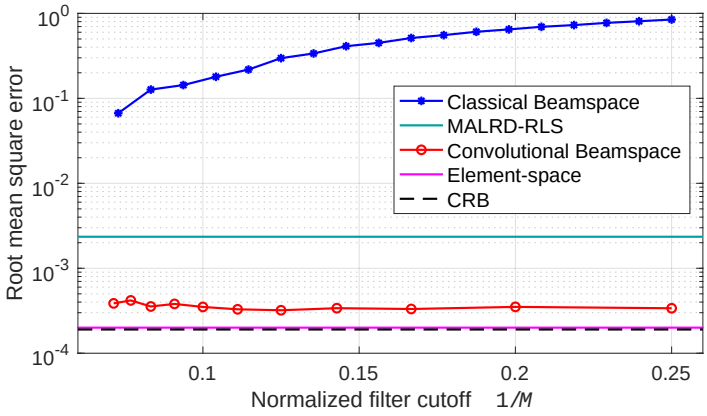
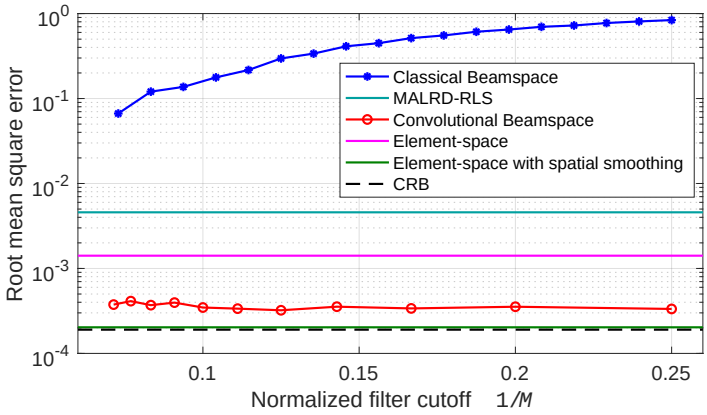


Figure 2.5: Responses of typical filters used for the example in Fig. 2.6.

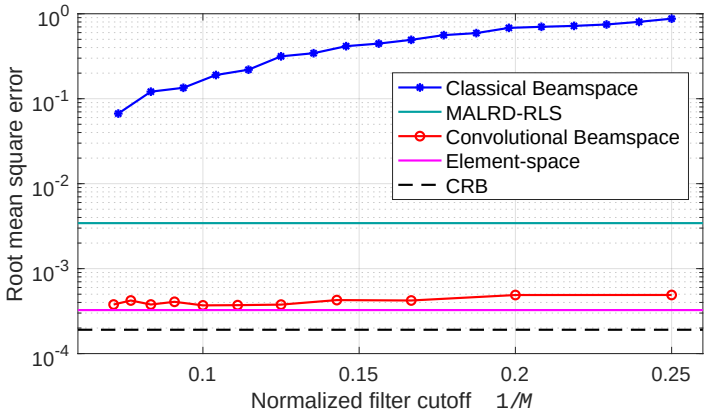
Estimation errors. We now compare the estimation errors of classical beamspace [2], MALRD-RLS [62], CBS, and element-space. Consider a ULA with $N = 96$



(a)



(b)



(c)

Figure 2.6: RMSE of classical beamspace, MALRD-RLS, CBS, and element-space. (a) RMSE for uncorrelated sources. (b) RMSE for in-band sources correlated with out-of-band sources. (c) RMSE for correlated in-band sources.

sensors receiving 6 sources at angles $-3°, 1.5°, 3°, 40°$, $60°$, and $80°$. The filter length is $L = 25$, and the decimation ratio M is varied. Filter responses for some M

are shown in Fig. 2.5. For all filters used, there are three sources in the passband ($-3°, 1.5°$ and $3°$) and three in the stopband ($40°, 60°$, and $80°$). All sources are first assumed uncorrelated. For MALRD-RLS, the parameters (as mentioned in the last paragraph of Sec. 2.2.5) are reasonably chosen as per [62]. Specifically, we set $I = 12$, $\bar{B} = N/I$, and the forgetting factor $\alpha = 0.998$ as in [62]. The number of grid points for grid search of ω is chosen as $G = 1000$ to keep a balance between performance and complexity. For MALRD-RLS, the number of DOAs and their locations are estimated together based on the *output power spectrum* $P(\omega)$ [62]. We declare that there is a source at $\bar{\omega}$ if $P(\bar{\omega})$ is a local maximum with prominence greater than 0.4 (with the spectrum normalized to have a maximum value 1), where prominence is defined as in `findpeaks` of MATLAB®. In all cases experimented except MALRD-RLS, the probability of resolving the correct number of in-band DOAs is always 1, so it is not plotted. MALRD-RLS failed to resolve only once in 500 trials, so the probability of success is nearly unity as well. We turn to study the RMSE in detected in-band source angles using root-MUSIC, for various values of $1/M$ (filter cutoff normalized by π), as shown in Fig. 2.6(a). Covariance estimates are obtained by using 200 snapshots, and 500 Monte Carlo runs are used. Note that, for MALRD-RLS and element-space, the notion of filter cutoff is not relevant, so their plots are constant. CBS outperforms classical beamspace [2], the poor performance of the latter being consistent with numerical sensitivity issues mentioned in [2] as the number of "beams" B (i.e., passband width in our notion) increases. CBS also outperforms MALRD-RLS. Note that element-space performs slightly better, and that the RMSE is almost independent of M for CBS, consistent with (2.28) and (2.32). Fig. 2.6(b) shows the performance when there are in-band sources *correlated* with out-of-band sources: sources n and $n+3$ have a correlation coefficient $\rho = 0.85$ for $n = 1, 2, 3$. In this case, CBS outperforms element-space significantly (without spatial smoothing), consistent with (2.37). Again, this does not contradict [77, 78] as we explained for (2.37). Note that we also show the RMSE of element-space using spatial smoothing [39]. We divide the array into 6 overlapping subarrays of size 91: $\{1, \ldots, 91\}$, $\{2, \ldots, 92\}$, $\ldots$, $\{6, \ldots, 96\}$, and then do spatial smoothing. Although spatial smoothing can achieve performance improvement for correlated sources, only CBS can achieve both performance improvement and complexity reduction (see Table 2.2). This is because spatial smoothing does not do dimensionality reduction (unlike CBS). Stochastic (element-space) CRBs [44] are also shown in Fig. 2.6(a)-(b). While both CBS and element-space come close to the CRB for uncorrelated sources, only CBS comes close to the CRB for correlated

sources. CBS fills in the gap between element-space (without spatial smoothing) and CRB for correlated sources. The running time per Monte Carlo run for each algorithm in Fig. 2.6(b) is shown in Table 2.2. The computational complexity of CBS is comparable to that of classical beamspace and more than 10 times lower than that of element-space (with or without spatial smoothing). Running time of MALRD-RLS depends on the number of grid points G for grid search of ω. As mentioned before, we have chosen $G = 1000$, and the complexity of MALRD-RLS is the highest. This choice of G was necessary to obtain the reasonable performances shown in Fig. 2.6. On the other hand, performances cannot be significantly improved by a larger G. For instance, the RMSE for Fig. 2.6(a) would change from 0.0023 only to 0.0021 if G is changed from 1000 to 10000, but the complexity gets much higher. Finally, we consider the case when there are in-band correlated sources in Fig. 2.6(c). Here, sources 1 and 2 are correlated with $\rho = 0.85$ and all others are uncorrelated. CBS again outperforms classical beamspace and MALRD-RLS. Although CBS does not outperform element-space as in Fig. 2.6(b), it is only slightly worse than element-space, and both of them are reasonably close to the CRB.

Table 2.2: Running time per Monte Carlo run for Fig. 2.6(b) when $M = 4$ for beamspace methods

Algorithm	Running time (sec.)
Classical beamspace	0.00385
MALRD-RLS	0.272
CBS	0.00835
Element-space	0.0840
Element-space with spatial smoothing	0.0945

Truncation versus decimation. Next, we show that decimating the filter output is indeed an effective method. To this end, we compare CBS with decimation (2.19), CBS with *truncation*, and element-space. "CBS with truncation" means we keep only the first $N_{bs} \leq N-L+1$ samples of the filter output, $v(0), v(1), \ldots, v(N_{bs}-1)$. Consider a ULA with $N = 99$ sensors receiving 2 in-band sources at angles $-5°, 5°$, and 1 out-of-band source at angle $40°$. The filter length is $L = 16$, and the decimation ratio is $M = 4$. Fig. 2.7 shows the RMSE in detected in-band source angles using root-MUSIC. Covariance estimates are obtained by using 100 snapshots, and 500 Monte Carlo runs are used. As expected, for the truncated CBS, RMSE decreases

as N_{bs} increases. Remarkably, the RMSE at $N_{bs} = 84$, corresponding to keeping all the steady-state samples of the filter output, is about the same as the RMSE of CBS with decimation by $M = 4$. Moreover, CBS with decimation gives a RMSE almost the same as element-space, as suggested by (2.33). So in the decimation method, the only loss is due to discarding the transient part of the filter output, which is insignificant if $L \ll N$. Hence, decimation reduces the complexity of eigenspace computation by a factor of $O(M^3)$ without compromising the RMSE performance!

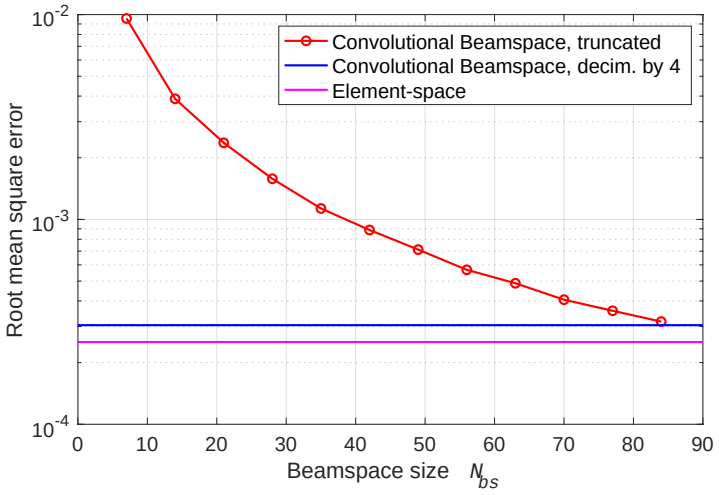


Figure 2.7: RMSE of truncated CBS, decimated CBS, and element-space.

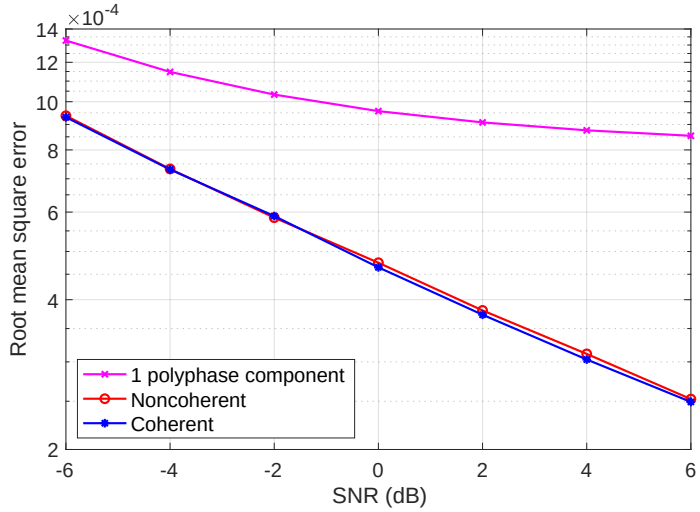


Figure 2.8: RMSE of CBS using all polyphase components coherently and noncoherently, versus using only one polyphase component.

Coherent versus noncoherent. Finally we compare the coherent method (2.19)

with the noncoherent method, and with the method which uses only one polyphase component. Consider a ULA with $N = 99$ sensors. The filter length is $L = 16$, and the decimation ratio is $M = 12$. Two in-band sources are at angles $\theta = -0.5°, 0.5°$, and 30 out-of-band sources are uniformly placed in the range $\omega \in [0.5\pi, 0.98\pi]$. Fig. 2.8 shows the RMSE in detected in-band source angles using root-MUSIC. Covariance estimates are obtained by using 100 snapshots, and 5000 Monte Carlo runs are used. Indeed, the RMSE of the coherent method is slightly smaller than that of the noncoherent method, and is much smaller than that of using only one polyphase component.

2.3 Convolutional Beamspace for Sparse Arrays

In this section, we show that CBS can be also applied to sparse arrays. In Sec. 2.3.1, the idea of difference coarrays for sparse arrays is reviewed. In Sec. 2.3.2, we show how to do CBS in the coarray domain, as depicted in (2.44) or (2.45). In Sec. 2.3.3, we show that we can again use uniform decimation to reduce computational complexity, and the decimated output (2.48) still has the Vandermonde structure of the ULA output. Hence, root-MUSIC and ESPRIT can be directly applied to the filtered and decimated coarray output without further adjustment. A brief discussion of computational complexity is given in Sec. 2.3.4. Finally, simulations are presented in Sec. 2.3.5.

Consider linear arrays for which the sensor locations are at $n\lambda/2$ where $n \in \mathcal{N} = \{n_0, n_1, \cdots, n_{N-1}\}$. The integer set $\mathcal{N}$ defines the array. For a ULA, we have $\mathcal{N} = \{0, 1, 2, \cdots, N-1\}$. More generally $\mathcal{N}$ can be a sparse array like the nested array [32], coprime array [40], or minimum redundancy array (MRA) [3]. One advantage of sparse arrays over ULAs is that it is possible to estimate $O(N^2)$ DOAs using an N-sensor sparse array [32].

Let $x(n_i), 0 \le i \le N-1$ be the array output. One can still use a filter $h(i)$ to perform a "convolution" as before, that is, $y(m) = \sum_i x(n_i)h(m-i)$. But this is not useful because the array output in response to a single DOA has the form $x(n_i) = ce^{j\omega n_i} \neq e^{j\omega i}$ (except for a ULA). So the vector $\mathbf{x} = [x(n_0) \ \cdots \ x(n_{N-1})]$ is not Vandermonde, and Eqs. (2.7) and (2.8) are not true. So a filtering effect like $H(e^{j\omega_k})\mathbf{a}_{N-L+1}(\omega_k)$ cannot be achieved in this way. But we will have better luck working in the difference coarray domain. This is similar in principle to the idea in [1] for classical beamspace.

2.3.1 Difference Coarrays

With the array output

$$x(n_i) = \sum_{k=1}^{D} c_k e^{j\omega_k n_i} + e(n_i), \quad n_i \in \mathcal{N}, \tag{2.38}$$

the cross-correlation between outputs of sensors n_i and n_m is

$$\mathrm{E}[x(n_i)x^*(n_m)] = \sum_{k=1}^{D}\sum_{l=1}^{D} \mathrm{E}[c_k c_l^*] e^{j(\omega_k n_i - \omega_l n_m)} + \sigma_e^2 \delta(n_i - n_m) \tag{2.39}$$

under standard statistical assumptions mentioned in Sec. 2.2. For zero-mean uncorrelated sources, $\mathrm{E}[c_k c_l^*] = p_k \delta(k-l)$, where $p_k = \mathrm{E}[|c_k|^2]$ is the power of the kth source. So

$$\begin{aligned} R(n_i - n_m) &\triangleq \mathrm{E}[x(n_i)x^*(n_m)] \\ &= \sum_{k=1}^{D} p_k e^{j\omega_k(n_i - n_m)} + \sigma_e^2 \delta(n_i - n_m), \end{aligned}$$

which depends only on the difference $n_i - n_m$ between sensor locations, hence the notation $R(n_i - n_m)$. The *difference coarray* $\mathcal{C}$ of the array $\mathcal{N} = \{n_i\}$ is the set of all possible differences $n_i - n_m$ between sensor locations. By estimating $\mathrm{E}[x(n_i)x^*(n_m)]$ using snapshot averages, we can estimate

$$R(l) = \sum_{k=1}^{D} p_k e^{j\omega_k l} + \sigma_e^2 \delta(l) \tag{2.40}$$

for all $l \in \mathcal{C}$. The difference coarray is symmetric in the sense that if $l \in \mathcal{C}$, then $-l \in \mathcal{C}$. Let the largest element in $\mathcal{C}$ be Z, and let U be the largest integer such that the uniform region $-(U-1) \le l \le U-1$ is in $\mathcal{C}$. Then $-(U-1) \le l \le U-1$ is called the *central ULA segment* of the difference coarray. If $U < Z$, the region $U \le l \le Z-1$ contains some integers which do not belong in $\mathcal{C}$, called *holes* in the coarray, and in particular U is a hole. Note that for an array with hole-free coarray (like the nested array), the coarray itself is the central ULA segment. Since (2.40) can be estimated for all $l \in \mathcal{C}$, we can in particular estimate $R(l)$ over the central ULA segment $-(U-1) \le l \le U-1$, and define a Hermitian Toeplitz matrix

$$\mathbf{R} = \begin{bmatrix} R(0) & R^*(1) & \cdots & R^*(U-1) \\ R(1) & R(0) & \cdots & R^*(U-2) \\ \vdots & \vdots & \ddots & \vdots \\ R(U-1) & R(U-2) & \cdots & R(0) \end{bmatrix}. \tag{2.41}$$

All elements of the matrix can be estimated by averaging $x(n_i)x^*(n_m)$ over snapshots, and over all n_i, n_m that produce identical difference $l = n_i - n_m$. This estimate of $\mathbf{R}$ was denoted as $\widetilde{\mathbf{R}}$ in [79]. By computing the noise eigenspace of $\widetilde{\mathbf{R}}$ we can estimate the DOAs ω_k using standard methods such as MUSIC. In general, $\widetilde{\mathbf{R}}$ may fail to be positive definite because it is a finite snapshot estimate. But it is shown in [79] that if we order its eigenvalues in terms of their absolute values and define the noise subspace accordingly, this always works.

2.3.2 Convolutional Beamspace in the Coarray Domain

Consider an FIR filter $G(z) = H(z)H^*(1/z^*)$ where $H(z) = \sum_{n=0}^{L-1} h(n)z^{-n}$ so that $G(e^{j\omega}) = |H(e^{j\omega})|^2 \geq 0$. We have

$$G(z) = \sum_{k=-L+1}^{L-1} g(k)z^{-k}, \tag{2.42}$$

where $g(k) = \sum_n h(n)h^*(n-k)$ and $g(k) = g^*(-k)$. Assume $L < U$ and define the finite duration signal

$$R^{(u)}(l) = \begin{cases} R(l), & -(U-1) \leq l \leq U-1 \\ 0, & \text{otherwise} \end{cases}, \tag{2.43}$$

where $R(l)$ is as in (2.40). This is $R(l)$ restricted to the central ULA portion of the coarray, hence the superscript "(u)". Now consider the convolution

$$R_0(n) = \sum_{l=-(U-1)}^{U-1} R^{(u)}(l)g(n-l), \tag{2.44}$$

which can be nonzero in $-(U+L-2) \leq n \leq U+L-2$. In the restricted range $\mathcal{S} = \{n \mid -(U-L) \leq n \leq U-L\}$, we have

$$\underbrace{\begin{bmatrix} R_0(-(U-L)) \\ R_0(-(U-L-1)) \\ \vdots \\ R_0(U-L) \end{bmatrix}}_{\mathbf{r}_0} = \mathbf{G}_{\text{co}} \underbrace{\begin{bmatrix} R(-(U-1)) \\ R(-(U-2)) \\ \vdots \\ R(U-1) \end{bmatrix}}_{\mathbf{r}}, \tag{2.45}$$

where $\mathbf{G}_{\text{co}}$ is the banded Toeplitz matrix

$$\begin{bmatrix} g(L-1) & \cdots & g(-(L-1)) & 0 & \cdots & 0 \\ 0 & \ddots & & \ddots & \ddots & \vdots \\ \vdots & \ddots & & & & 0 \\ & & & & & \\ 0 & \cdots & 0 & g(L-1) & \cdots & g(-(L-1)) \end{bmatrix}$$

of size $(2(U-L)+1)\times(2U-1)$. Thus, all the $2L-1$ filter coefficients $g(k)$ are involved in the computation of $R_0(n)$ in $n\in\mathcal{S}$, so that $\mathbf{r}_0$ in (2.45) defines the *steady state* portion of the output. We refer to $\mathbf{r}_0$ as the *coarray convolutional beamspace* signal generated from the ULA segment $\mathbf{r}$ of the coarray. In this steady state, we can readily verify that

$$R_0(n)=\sum_{k=1}^{D}p_kG(e^{j\omega_k})e^{j\omega_k n}+\sigma_e^2 g(n),\quad n\in\mathcal{S}. \tag{2.46}$$

Since $p_kG(e^{j\omega})\geq 0$, the quantity $R_0(n)$ in (2.46) resembles the autocorrelation of a sum of sinusoids buried in noise with autocorrelation $\sigma_e^2 g(n)$, and in particular $R_0(n)=R_0^*(-n)$. The DOA ω_k is filtered by $G(e^{j\omega_k})$, so if $G(e^{j\omega})$ is a good lowpass filter with passband width $\approx 2\pi/M$, then

$$R_0(n)\approx\sum_{k=1}^{D_0}p_kG(e^{j\omega_k})e^{j\omega_k n}+\sigma_e^2 g(n),\quad n\in\mathcal{S}, \tag{2.47}$$

where D_0 is the number of sources falling within the passband of the filter $G(e^{j\omega})$.

In summary, the DOAs in a narrow subband can be isolated by this coarray filtering. Restricting the filtering to the ULA portion of the coarray (as in (2.43)) and considering only the outputs in the steady state $\mathcal{S}$, we ensure the exact relation (2.46). Since the filter, by design, satisfies $G(e^{j\omega})\geq 0$, (2.46) is still a valid autocorrelation truncated to $\mathcal{S}$. Since the steady state domain $\mathcal{S}$ still looks like a ULA (i.e., (2.46) has the Vandermonde structure with respect to $n\in\mathcal{S}$), we can directly apply root-MUSIC or ESPRIT without any additional steps. As U is typically as large as $O(N^2)$, we can design a sharp-cutoff filter with good stopband attenuation by using a relatively long filter length L.

2.3.3 Decimating in the Coarray Domain

Since the filter $G(e^{j\omega})$ has passband width $\approx 2\pi/M$, we can work with the uniformly decimated version

$$R_0(Mn)=\sum_{k=1}^{D}p_kG(e^{j\omega_k})e^{jM\omega_k n}+\sigma_e^2 g(Mn) \tag{2.48}$$

with n restricted such that $Mn\in\mathcal{S}$. Decimation achieves the dimensionality reduction needed to reduce the computational complexity in the subbands, such reduction being an integral part of any beamspace processing. If $G(e^{j\omega})$ is *Nyquist*(M), then

$g(Mn) = \delta(n)$ and

$$R_0(Mn) = \sum_{k=1}^{D} p_k G(e^{j\omega_k}) e^{jM\omega_k n} + \sigma_e^2 \delta(n)$$
$$\approx \sum_{k=1}^{D_0} p_k G(e^{j\omega_k}) e^{jM\omega_k n} + \sigma_e^2 \delta(n). \tag{2.49}$$

Since the noise term has $\delta(n)$ and $p_k G(e^{j\omega_k}) \geq 0$, Eq. (2.49) represents the autocorrelation of a sum of sinusoids buried in *white* noise. We now define a $q \times q$ Hermitian Toeplitz matrix (where q is the largest integer with $(q-1)M \in \mathcal{S}$):

$$\begin{bmatrix} R_0(0) & R_0^*(M) & \cdots & R_0^*((q-1)M) \\ R_0(M) & R_0(0) & \cdots & R_0^*((q-2)M) \\ \vdots & \vdots & \ddots & \vdots \\ R_0((q-1)M) & R_0((q-2)M) & \cdots & R_0(0) \end{bmatrix}.$$

Its eigendecomposition reveals the noise subspace from which the frequencies $M\omega_k$ (mod 2π) can be identified (when $D_0 < q$). From these we can identify ω_k without ambiguity as before, by using the fact that only those ω_k that are in the passband of $G(e^{j\omega})$ contribute significantly to $R_0(n)$. Again, for finite snapshots, we order eigenvalues in terms of their absolute values to define noise subspace [79].

As before, instead of using one filter $G(e^{j\omega}) \geq 0$, we can use a filter bank $G_i(e^{j\omega}), 0 \leq i \leq M-1$ with filter responses covering the range $0 \leq \omega < 2\pi$ as in Fig. 2.3. We now constrain $G_i(e^{j\omega})$ to be Nyquist(M) with $G_i(e^{j\omega}) \geq 0$. Using these we obtain the *decimated coarray CBS* signals $R_i(Mn), 0 \leq i \leq M-1$. Analysis of the ith signal $R_i(Mn)$ reveals the D_i DOAs falling into the ith subband. The union of these DOAs gives the set of all D DOAs.

2.3.4 Computational Complexity

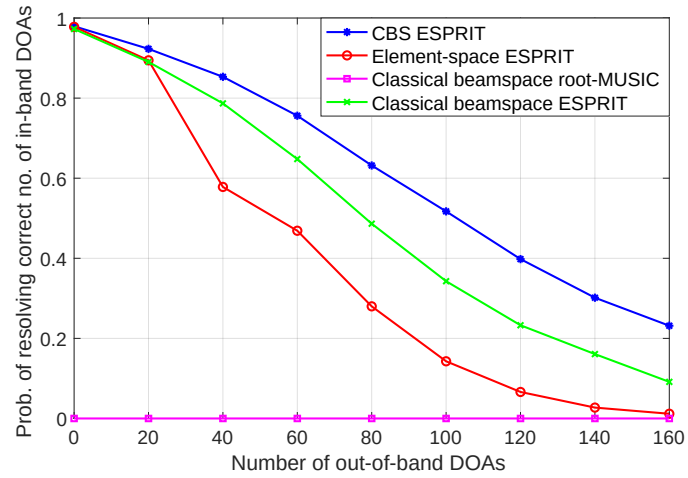
The computational advantage of CBS for sparse arrays is similar to CBS for ULAs. With $-(U-1) \leq l \leq U-1$ being the central ULA segment of the difference coarray, the complexity of eigenspace computation for CBS is $O((U/M)^3)$, which is much smaller than $O(U^3)$, the complexity of eigenspace computation for element-space. (Here, we work in the coarray domain for both CBS and element-space. So, in this section element-space means "coarray domain without filtering.") A direct comparison of running time for CBS, element-space, and classical beamspace is also presented in Table 2.3.

Table 2.3: Running time per Monte Carlo Run for coarray based CBS for sparse arrays (Fig. 2.9) when there are 80 out-of-band DOAs

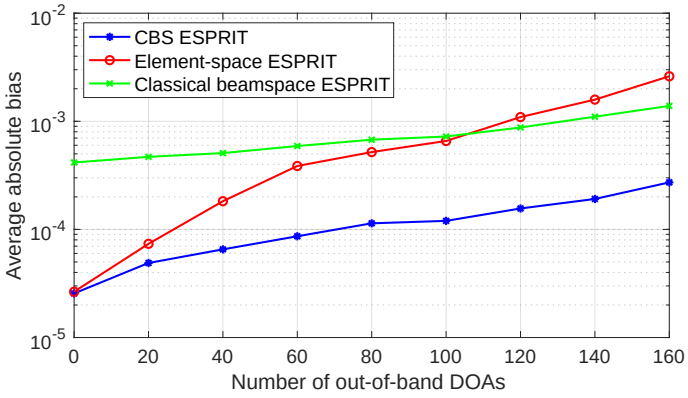
Algorithm	Running time (sec.)
CBS ESPRIT	0.00646
Element-space ESPRIT	0.0447
Classical beamspace root-MUSIC	0.0109
Classical beamspace ESPRIT	0.00796

2.3.5 Simulations

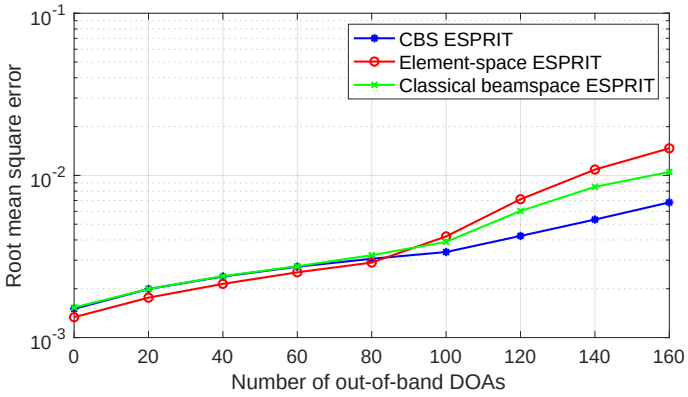
We consider an example where the number of in-band DOAs is greater than the number of sensors, which is a scenario that makes sparse arrays stand out. A two-level nested array [32] with each level having 12 sensors is considered, a total of 24 sensors. CBS ESPRIT is compared with element-space ESPRIT, classical beamspace root-MUSIC [2], and classical beamspace ESPRIT [18]. We follow the procedure in [1] to do classical beamspace in the coarray domain. The covariance in the beamspace is computed by $\mathbf{TRT}^H$, where $\mathbf{T}$ is the beamspace transformation matrix, and $\mathbf{R}$ is defined in (2.41). For CBS, a Parks-McClellan filter $H(z)$ of length $L = 16$ and with cutoff π/M is used, where $M = 4$ is the decimation ratio. So the filter $G(z)$ in (2.42) has length 31. There are 25 in-band DOAs, uniformly placed in the range $\omega \in [-0.2\pi, 0.2\pi]$. We vary the number of out-of-band DOAs, while they are uniformly placed in $\omega \in [0.5\pi, 0.98\pi]$. Covariance estimates are obtained by using 1000 snapshots, and 5000 Monte Carlo runs are used. The number of DOAs is assumed unknown and estimated based on the distribution of eigenvalues of the covariance (see Fig. 2.4). Fig. 2.9(a) shows the probability of resolving the correct number of in-band DOAs. CBS consistently offers a larger probability of resolution than element-space and classical beamspace root-MUSIC and ESPRIT, as the number of out-of-band DOAs varies. Moreover, as shown in Fig. 2.9(b), CBS has the smallest average of absolute values of bias in detected in-band source angles. Finally, Fig. 2.9(c) shows that compared to element-space and classical beamspace ESPRIT, CBS has similar RMSE in detected in-band source angles when there are fewer out-of-band DOAs. But, CBS has significantly smaller RMSE when there are more out-of-band DOAs. Note that classical beamspace root-MUSIC always has zero probability of resolution due to numerical sensitivity issues [2] as we have a large coarray size $U = 156$; so we do not plot its bias and RMSE in Fig. 2.9(b)-(c). The running time per Monte Carlo run for each algorithm is shown in Table 2.3. The



(a)



(b)



(c)

Figure 2.9: Performance of CBS in coarray domain, and element-space in coarray domain for sparse arrays using ESPRIT. The classical beamspace method in the coarray domain [1] is also compared using both root-MUSIC and ESPRIT. (a) Probability of resolution. (b) Average absolute bias. (c) RMSE.

computational complexity of CBS is a little lower than that of classical beamspace

and about 7 times lower than that of element-space. This example shows that for the typical scenario of resolving more sources than sensors using sparse arrays, many benefits of CBS can still be realized. Also, according to our simulations, ESPRIT typically yields better estimates for CBS in the coarray domain than root-MUSIC.

2.4 Convolutional Beamspace and Sparse Recovery

In this section, we show that CBS can also be used in conjunction with sparse signal recovery. In Sec. 2.4.1, we show how to integrate CBS into the basic single-snapshot sparse recovery problem, as depicted in Problem (2.56). In Sec. 2.4.2, we show that we can again use uniform decimation to reduce computational complexity as in Problem (2.58). In Sec. 2.4.3, the multiple-snapshot version is also presented, as depicted in Problem (2.60) (without decimation) and Problem (2.61) (with decimation). The comparison of computational complexity between CBS and element-space is summarized in Sec. 2.4.4. As only the filter responses at discrete frequencies on the dictionary grid are relevant, the problem of designing discrete-frequency FIR filters is addressed in Sec. 2.4.5. Finally, some remarks are given in Sec. 2.4.6, and simulations are presented in Sec. 2.4.7.

Sparse signal representation techniques for DOA estimation have been studied in the literature [59]. In this context, a dictionary $\mathbf{D}$ of steering vectors $\mathbf{a}(\omega_i)$ on a grid of potential DOAs $\{\omega_i\}_{i=1}^{d}$ is considered, and the goal is to find a sparse signal $\mathbf{q} = [q_1\ q_2\ \cdots\ q_d]^T$ that well represents the ULA output $\mathbf{x}$:

$$\mathbf{x} = \underbrace{[\mathbf{a}_N(\omega_1)\ \mathbf{a}_N(\omega_2)\ \cdots\ \mathbf{a}_N(\omega_d)]}_{\text{dictionary } \mathbf{D}}\mathbf{q} + \mathbf{e}, \tag{2.50}$$

where the error term $\mathbf{e}$ should be "small." The number of dictionary atoms d is typically much larger than D, the number of sources. A popular technique to obtain the sparse vector $\mathbf{q}$ is the Lasso method [80] that solves the following problem:

$$\min_{\mathbf{q}\in\mathbb{C}^d} \quad \|\mathbf{q}\|_1 \tag{2.51a}$$

$$\text{subject to} \quad \|\mathbf{x} - \mathbf{D}\mathbf{q}\|_2^2 \le \beta, \tag{2.51b}$$

where $\beta > 0$ is a parameter. The l_1-norm objective (2.51a) serves as a surrogate for sparsity, and the l_2-norm constraint (2.51b) limits the search space to where the noise term is small.

2.4.1 Convolutional Beamspace and Dictionaries

As in (2.5), we convolve the sequence $x(n), 0 \le n \le N-1$ with an FIR filter $h(n), 0 \le n \le L-1$ with $L < N$, and extract the steady state samples:

$$\mathbf{y} = \mathbf{H}\mathbf{x}. \tag{2.52}$$

As in (2.7), the response to a single DOA is (ignoring noise)

$$\mathbf{y} = \mathbf{H}\mathbf{a}_N(\omega) = e^{j(L-1)\omega} H(e^{j\omega}) \mathbf{a}_{N-L+1}(\omega). \tag{2.53}$$

Thus, (2.50) and (2.52) yield

$$\mathbf{y} = \left[\mathbf{a}_{N-L+1}(\omega_1) \ \cdots \ \mathbf{a}_{N-L+1}(\omega_d)\right] \mathbf{\Lambda}_h \mathbf{q} + \mathbf{H}\mathbf{e}, \tag{2.54}$$

where $\mathbf{\Lambda}_h$ is a diagonal matrix with ith diagonal element $[\mathbf{\Lambda}_h]_{ii} = e^{j(L-1)\omega_i} H(e^{j\omega_i})$. In other words, the diagonal elements are the frequency responses of $h(n)$ evaluated at the dictionary frequencies (with some phase shift). If $H(e^{j\omega})$ is a good narrowband lowpass filter, then

$$\mathbf{y} \approx \underbrace{\left[\mathbf{a}_{N-L+1}(\omega_1) \ \cdots \ \mathbf{a}_{N-L+1}(\omega_{d_0})\right]}_{\text{dictionary } \mathbf{D}_L} \mathbf{q}_0 + \mathbf{H}\mathbf{e}, \tag{2.55}$$

where $\omega_1, \omega_2, \ldots, \omega_{d_0}$ are the frequencies within the passband of $H(e^{j\omega})$, and $\mathbf{q}_0 \in \mathbb{C}^{d_0}$ is a significantly shorter vector than $\mathbf{q}$. Thus, a Lasso problem can be formulated for the CBS signal $\mathbf{y}$ as

$$\min_{\mathbf{q}_0 \in \mathbb{C}^{d_0}} \quad \|\mathbf{q}_0\|_1 \tag{2.56a}$$

$$\text{subject to} \quad \|\mathbf{y} - \mathbf{D}_L \mathbf{q}_0\|_2^2 \le \beta. \tag{2.56b}$$

Here, the original dictionary $\mathbf{D}$ in (2.51b) is replaced by the dictionary $\mathbf{D}_L$ for CBS.

2.4.2 Decimation for Dictionaries

To reduce computational complexity by dimensionality reduction, we decimate the CBS signal $\mathbf{y}$ by M if $H(e^{j\omega})$ is a good filter with passband width $\approx 2\pi/M$. Let $v(n) = y(n+L-1)$ so that $\mathbf{y} = [v(0) \ v(1) \ \cdots \ v(N-L)]^T$. Let $v_0(n) = v(Mn)$ and $\mathbf{v}_0 = [v_0(0) \ v_0(1) \ \cdots \ v_0(J_0-1)]^T$, where $J_0 = \lceil (N-L+1)/M \rceil$, so the decimated version

$$\mathbf{v}_0 = [v(0) \ v(M) \ \cdots \ v((J_0-1)M) \ ]^T. \tag{2.57}$$

Then, we obtain the complexity-reduced problem

$$\min_{\mathbf{q}_0 \in \mathbb{C}^{d_0}} \quad \|\mathbf{q}_0\|_1 \tag{2.58a}$$

$$\text{subject to} \quad \|\mathbf{v}_0 - \mathbf{D}_{L,0}\mathbf{q}_0\|_2^2 \leq \beta, \tag{2.58b}$$

where $\mathbf{D}_{L,0}$ is the matrix obtained by retaining the rows $0, M, \ldots, (J_0 - 1)M$ of $\mathbf{D}_L$.

2.4.3 Multiple Snapshots for Dictionaries

The previous formulation is for a single snapshot. For multiple snapshots, we adopt the ℓ_1-SVD method proposed in [59]. Suppose we have K snapshots, $\mathbf{X} = [\mathbf{x}(1)\ \mathbf{x}(2)\ \cdots\ \mathbf{x}(K)]$. To reduce dimensionality, we take the SVD $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^H$ and retain a $N \times k$ matrix containing most of the signal power: $\mathbf{X}_{\text{SV}} = \mathbf{U}\mathbf{\Sigma}\mathbf{J}_k = \mathbf{X}\mathbf{V}\mathbf{J}_k$, where $\mathbf{J}_k = [\mathbf{I}_k\ \mathbf{0}]^T$. We often take $k < K$ to be roughly the number of sources, and the original formulation of the ℓ_1-SVD method is then [59]

$$\min_{\mathbf{Q} \in \mathbb{C}^{d \times k}} \quad \|\mathbf{Q}\|_{1,2} \tag{2.59a}$$

$$\text{subject to} \quad \|\mathbf{X}_{\text{SV}} - \mathbf{D}\mathbf{Q}\|_F^2 \leq \beta, \tag{2.59b}$$

where $\|\mathbf{Q}\|_{1,2} = \sum_m \sqrt{\sum_n |[\mathbf{Q}]_{m,n}|^2}$. That is, the ℓ_2-norm across singular vector samples is first computed for each spatial index, and then the ℓ_1-norm is computed across spatial samples for sparsity.

CBS can also be applied to the multiple snapshot scheme based on the ℓ_1-SVD method. We first convolve the spatial samples of each snapshot with a filter $h(n)$ of length L and extract the steady state samples: $\mathbf{Y} = \mathbf{H}\mathbf{X}$, similar to (2.52). Then, we take the SVD $\mathbf{Y} = \mathbf{U}_Y\mathbf{\Sigma}_Y\mathbf{V}_Y^H$ and retain a $(N - L + 1) \times k_0$ matrix containing most of the signal power: $\mathbf{Y}_{\text{SV}} = \mathbf{U}_Y\mathbf{\Sigma}_Y\mathbf{J}_{k_0} = \mathbf{Y}\mathbf{V}_Y\mathbf{J}_{k_0}$. A multiple-snapshot version of Problem (2.56) can then be formulated as

$$\min_{\mathbf{Q}_0 \in \mathbb{C}^{d_0 \times k_0}} \quad \|\mathbf{Q}_0\|_{1,2} \tag{2.60a}$$

$$\text{subject to} \quad \|\mathbf{Y}_{\text{SV}} - \mathbf{D}_L\mathbf{Q}_0\|_F^2 \leq \beta. \tag{2.60b}$$

Likewise, a decimated version can be considered. Let $\mathbf{V}_0 = [\mathbf{v}_0(1)\ \mathbf{v}_0(2) \cdots \mathbf{v}_0(K)]$ be the multiple-snapshot counterpart of (2.57). Then, we take the SVD $\mathbf{V}_0 = \mathbf{U}_V\mathbf{\Sigma}_V\mathbf{V}_V^H$ and retain a $J_0 \times k_0$ matrix containing most of the signal power: $\mathbf{V}_{\text{SV}} = \mathbf{U}_V\mathbf{\Sigma}_V\mathbf{J}_{k_0} = \mathbf{V}_0\mathbf{V}_V\mathbf{J}_{k_0}$. Then, a multiple-snapshot version of Problem (2.58) can be formulated as

$$\min_{\mathbf{Q}_0 \in \mathbb{C}^{d_0 \times k_0}} \quad \|\mathbf{Q}_0\|_{1,2} \tag{2.61a}$$

$$\text{subject to} \quad \|\mathbf{V}_{\text{SV}} - \mathbf{D}_{L,0}\mathbf{Q}_0\|_F^2 \leq \beta. \tag{2.61b}$$

Problem (2.61) is expected to yield better performance than Problem (2.59) because fewer DOAs are to be recovered in (2.61). This is indeed the case as we shall see in simulations (Sec. 2.4.7).

2.4.4 Computational Complexity

An important advantage of CBS is reduced computational complexity. The comparison of computational complexity between CBS and element-space is summarized in Table 2.4. Assume that the filter $H(e^{j\omega})$ used in CBS has passband width $\approx 2\pi/M$, and that the dictionary grid ω_i is a uniform grid of frequencies in $[0, 2\pi)$. Then, d in Problem (2.51) and d_0 in Problems (2.56) and (2.58) are related by $d_0 \approx d/M$. That is, the number of optimization variables of CBS is only $1/M$ of that of element-space for the single-snapshot case.

Table 2.4: Comparison of computational complexity of sparse recovery methods

Algorithm	Number of optimization variables
Element-space (single snapshot)	d
Element-space ℓ_1-SVD (multiple-snapshot)	dk
CBS (single snapshot)	$d_0 \approx d/M$
CBS ℓ_1-SVD (multiple-snapshot)	$d_0 k_0 \approx dk/M^2$

Next, consider the multiple-snapshot case. According to [59], to get adequate performance, we have to take k to be roughly the number of sources D in Problem (2.59). As only the D_0 sources in the passband are effective after filtering, we can take $k_0 \approx D_0$ in Problem (2.60). If the sources are roughly uniformly distributed, then $D_0 \approx D/M$, so $k_0 \approx D/M$. Then, dk in Problem (2.59) and $d_0 k_0$ in Problems (2.60) and (2.61) are related by $d_0 k_0 \approx dk/M^2$. That is, the number of optimization variables of CBS is only $1/M^2$ of that of element-space for the multiple-snapshot case using the ℓ_1-SVD method. This can be a very significant complexity reduction. A direct comparison of running time for CBS and element-space is also presented in Table 2.5.

2.4.5 Filters Designed for Dictionaries

An interesting question that arises for CBS dictionaries is how to design the filter $H(e^{j\omega})$. One can directly adopt standard methods such as the Parks-McClellan algorithm, the window method, and so on [63], but these standard filters are designed to be optimal or sub-optimal over continuous frequencies ω. The fact that only the

response $H(e^{j\omega_i})$ at the *discrete* frequencies ω_i, $i = 1, \ldots, d$ are relevant to CBS dictionaries, as in (2.54), can be leveraged to design better filters for dictionaries.

Consider a Type I [63, 66] linear-phase FIR filter $h(n), 0 \leq n \leq L-1$ such that L is odd and that $h(n)$ is even symmetric, i.e., $h(L-1-n) = h(n)$. Type I is just for illustration, and other types of linear-phase FIR filters can be designed similarly. It can be shown that $H(e^{j\omega}) = e^{-j\omega\frac{L-1}{2}}A(\omega)$, where $A(\omega) = \sum_{m=0}^{M} b_m \cos(m\omega)$ where $M = (L-1)/2$, $b_0 = h(M)$, and $b_m = 2h(M-m)$ otherwise. Note that $|A(\omega)|$ is the magnitude response. Then, a standard *minimax* filter design problem can be formulated as [63]

$$\min_{h(\frac{L-1}{2}),\ldots,h(L-1)} \max_{\omega\in\Omega_p\cup\Omega_s} |W(\omega)(A(\omega) - A_d(\omega))|, \tag{2.62}$$

where Ω_p is the passband, Ω_s is the stopband,

$$A_d(\omega) = \begin{cases} 1, & \omega \in \Omega_p \\ 0, & \omega \in \Omega_s \end{cases} \tag{2.63}$$

is the desired magnitude response, and $W(\omega)$ is a weighting function such that

$$W(\omega) = \begin{cases} 1, & \omega \in \Omega_p \\ \lambda, & \omega \in \Omega_s \end{cases} \tag{2.64}$$

for some design parameter $\lambda > 0$. Note that $W(\omega)(A(\omega) - A_d(\omega))$ is affine in the variables $h(\frac{L-1}{2}), \ldots, h(L-1)$, and that the absolute function is a convex function. Hence, since the composition of a convex function with an affine mapping is still convex, $|W(\omega)(A(\omega) - A_d(\omega))|$ is convex. Moreover, the pointwise supremum of a collection of convex functions is still convex, so we conclude that Problem (2.62) is a convex problem. Traditionally, the passband Ω_p and stopband Ω_s are continuous. However, as described earlier, only the frequency response $H(e^{j\omega_i})$ at the discrete frequency grid $\{\omega_i\}_{i=1}^{d}$ are relevant, so instead of (2.62), we can consider

$$\min_{h(\frac{L-1}{2}),\ldots,h(L-1)} \max_{\omega\in\Omega_p'\cup\Omega_s'} |W(\omega)(A(\omega) - A_d(\omega))|, \tag{2.65}$$

where $\Omega_p' = \{\omega_i, 1 \leq i \leq d \mid \omega_i \in \Omega_p\}$ and $\Omega_s' = \{\omega_i, 1 \leq i \leq d \mid \omega_i \in \Omega_s\}$. We may also choose a sparser frequency grid (e.g., only include even i's) for Problem (2.65) to obtain a potentially better filter. See Fig. 2.11(a) for an example. Problem (2.65) can be readily solved by any numerical convex program solver such as `CVX` [81]. Such a filter should have a better response than a standard filter in the literature (e.g., `firpm` of MATLAB®) over the discrete frequencies $\omega_1, \ldots, \omega_d$.

2.4.6 Remarks

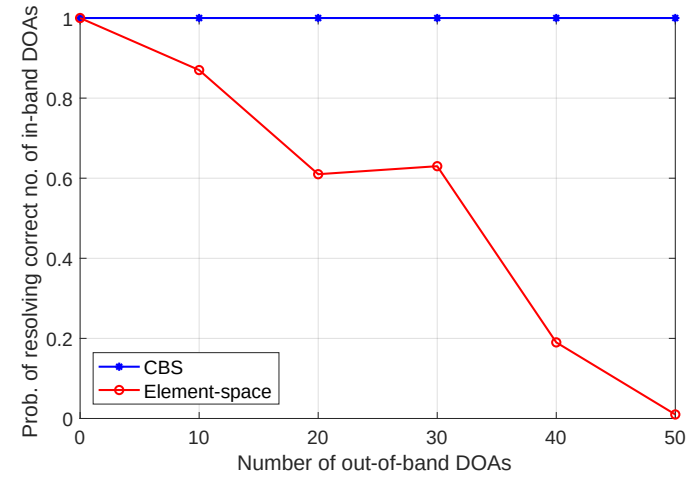
1) The choice of the grid of potential DOAs $\{\omega_i\}_{i=1}^d$ is in the designer's hands. One way is to choose a *uniform* grid in ω. Another is to choose a grid uniform in the physical DOA θ. Recall that $\omega = \pi \sin\theta$. Hence, the second way leads to a *nonuniform* grid in ω, with denser points in high-frequency part and sparser points in low-frequency part. It is hard to say which type, uniform or nonuniform, is better because it depends on the actual locations of the DOAs. Yet, if the DOAs are expected to be uniformly distributed over the physical angle θ, it makes sense to choose the nonuniform type. An interesting observation is that the type of the grid also affects the result of filter design. For example, suppose we want to design a lowpass filter $H(e^{j\omega})$. If we choose the nonuniform type of grid, with denser points in high-frequency part, then the resulting filter will have better attenuation for the high-frequency stopband. That is, the density of the grid points induces a weighting effect across different frequency bands. See Fig. 2.12 for a numerical demonstration.

2) Instead of using a filter $H(e^{j\omega})$, we can use a filter bank $H_i(e^{j\omega}), 0 \le i \le M-1$ to cover the full range $0 \le \omega < 2\pi$ as in Fig. 2.3. Using these, we obtain the CBS signals $\mathbf{y}_i, 0 \le i \le M-1$. Solving an optimization problem as in (2.56) for each $\mathbf{y}_i$ in parallel reveals the D_i DOAs falling into the ith subband. The union of these DOAs gives the set of all D DOAs. Similar idea can be applied to Problems (2.58), (2.60), and (2.61). Note that here we do not confine each filter $H_i(e^{j\omega})$ to be a spectral factor of a Nyquist filter as we did for subspace-based methods such as MUSIC because the property that the noise is white is not used in sparse signal recovery.

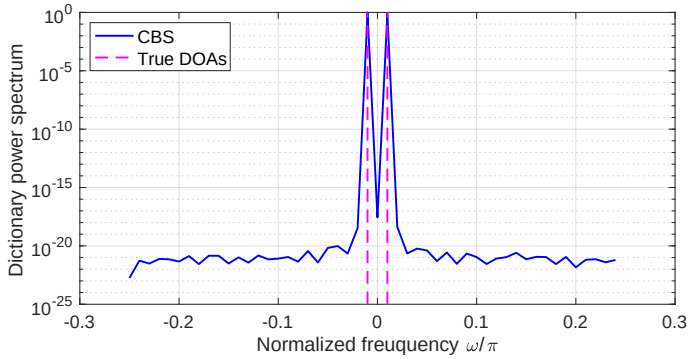
3) One method for selecting the parameter β in the sparse recovery problems (e.g., Problem (2.51)) was proposed in [59]. The method is based on estimating the variance of the left hand side of the constraint (e.g., $\|\mathbf{x} - \mathbf{D}\mathbf{q}\|_2^2$ in (2.51b)). It was originally proposed for element-space, but we can extend the method to CBS. In the simulations, we will follow this method to choose β.

2.4.7 Simulations

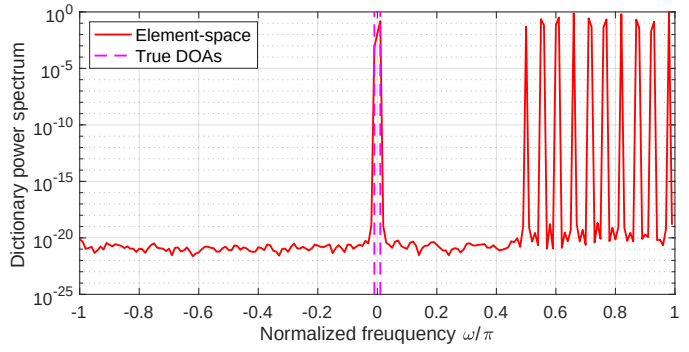
Unlike in subspace-based methods such as MUSIC, where we estimate the number of DOAs and their locations separately, for dictionaries, we estimate them together. Specifically, for element-space, after getting the optimal solution $\hat{\mathbf{Q}}$ for Problem (2.59), we plot the *dictionary power spectrum* $P(\omega_i) = \sum_n |\hat{\mathbf{Q}}_{in}|^2$ for $1 \le i \le d$.



(a)



(b)



(c)

Figure 2.10: Performance of CBS and element-space dictionaries when there are many out-of-band DOAs. (a) Probability of resolution. (b) Typical dictionary power spectrum of CBS. (c) Typical dictionary power spectrum of element-space.

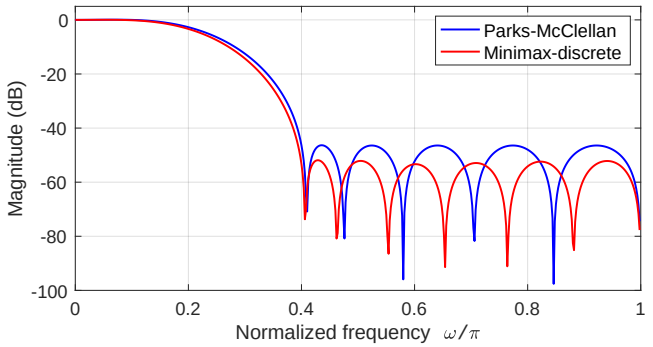
Then, we declare that there is a source at ω_i if there is a peak (local maximum) that is larger than a particular threshold: $P(\omega_i) \geq \epsilon$. The same method is used for CBS. See Fig. 2.10(a)-(b) for an example (described below). In all examples, we use $\lambda = 1$ in (2.64).

We first compare the most complexity-reduced version of CBS (2.61) with the element-space (2.59) under multiple snapshots. Consider a ULA with $N = 99$ sensors. First assume a Parks-McClellan filter $H(z)$ of length $L = 16$ and with cutoff π/M is used, where $M = 4$ is the decimation ratio. A dictionary of 200 points uniform in ω is used. There are two in-band DOAs at angles $-0.573°$ and $0.573°$, which are exactly on the grid of the dictionary for simplicity. We vary the number of out-of-band DOAs, while they are uniformly placed in the range $\omega \in [0.5\pi, 0.98\pi]$. The reduced dimension for the ℓ_1-SVD method in (2.59) and (2.61) is chosen as $k = k_0 = 3$. The peak threshold $\epsilon = 0.1$ is used (with the spectrum normalized to have a maximum value 1). We choose $\beta = 641.7$ in (2.59b) and $\beta = 221.7$ in (2.61b) according to the method in [59], as described in Remark 3 above. Fig. 2.10(a) shows the probability of resolving the correct number of in-band DOAs. We use $K = 100$ snapshots and 100 Monte Carlo runs to get the plot. As the number of out-of-band DOAs increases, the probability of resolution decreases significantly for element-space, while the probability of resolution is always one for CBS due to good stopband attenuation. Note that there is a trade-off between the performance and the reduced dimension k for the ℓ_1-SVD method for element-space. A large k can lead to better performance, but it requires higher computational complexity. Typical dictionary power spectra of CBS and element-space for a Monte Carlo run with 10 out-of-band DOAs are shown in Fig. 2.10(b)-(c). For CBS, only the passband part is plotted. The two in-band DOAs are clearly distinguished by CBS, but they cannot be resolved by element-space. The running time per Monte Carlo run for CBS and element-space is shown in Table 2.5. The computational complexity of CBS is about 9 times lower than that of element-space.

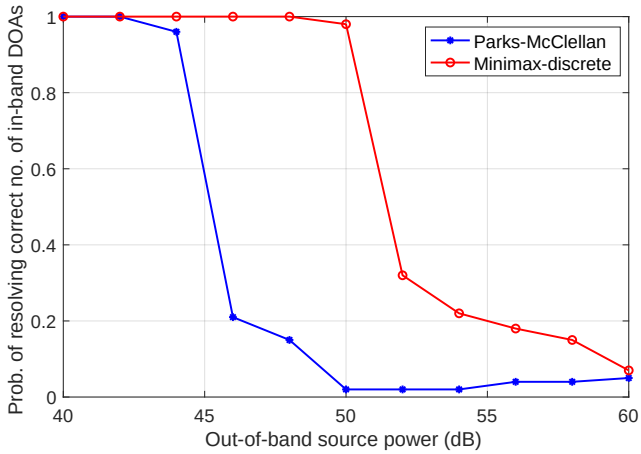
Table 2.5: Running time per Monte Carlo run for dictionary methods (Fig. 2.10) when there are 10 out-of-band DOAs

Algorithm	Running time (sec.)
CBS	0.458
Element-space	4.13

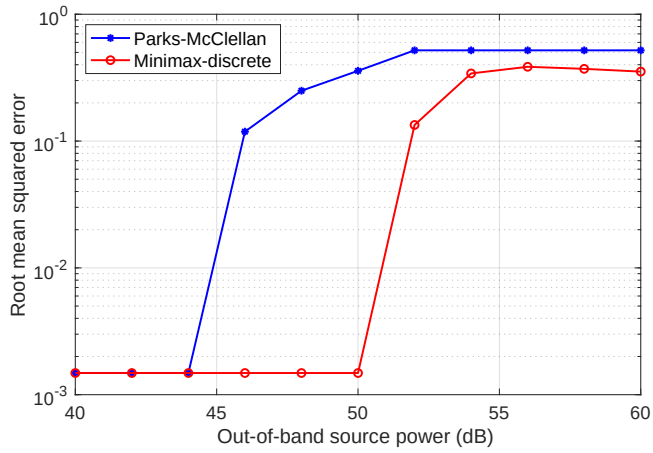
Next, the impact of a well-designed *minimax-discrete* type filter described in Sec. 2.4.5 (instead of the Parks-McClellan filter) is studied. Consider a ULA with $N = 99$ sensors. Problem (2.61) is to be solved again. A dictionary of 200 points uniform in ω is used. We compare lowpass FIR filters with length $L = 16$ designed using the Parks-McClellan algorithm [63] and our minimax-discrete optimization problem (2.65). The passband is $\Omega_p = [0, 0.1\pi]$, and the stopband is $\Omega_s = [0.4\pi, \pi]$. For



(a)



(b)



(c)

Figure 2.11: Performance of minimax-discrete and Parks-McClellan filters for CBS dictionaries when there are powerful out-of-band sources. (a) Filter responses. (b) Probability of resolution. (c) RMSE.

the filter design problem (2.65), we use a grid of 50 points uniform in ω, which is sparser than the dictionary grid. Two in-band DOAs are at angles $-0.6°, 0.6°$

with equal powers $p_k = 1$, and 10 out-of-band DOAs are uniformly placed in the range $\omega \in [0.5\pi, 0.98\pi]$ with varying equal powers. Here all the DOAs are not on the dictionary grid, in contrast to the previous example. The decimation ratio is $M = 4$. The reduced dimension for the ℓ_1-SVD method is chosen as $k_0 = 2$. The same peak threshold $\epsilon = 0.1$ is used, and we choose $\beta = 1338$ in (2.61b) according to the method in [59], as described in Remark 3 in Sec. 2.4.6. In Fig. 2.11(a), the magnitude responses of the filters are plotted. Although the minimax-discrete type filter is optimized only over a discrete grid, it has better attenuation over the whole continuous stopband. We experimentally found that if we use a grid of 200 rather than 50 points for filter design, the resulting minimax-discrete filter is almost the same as the Park-McClellan filter. This is expected because the denser the grid, the closer the discrete type to the continuous type. Due to the better stopband attenuation, as we vary the power of the out-of-band sources, the minimax-discrete filter has larger probability of resolving the correct number of in-band DOAs and smaller RMSE in detected in-band source angles, as shown in Fig. 2.11(b)-(c). Here we use $K = 100$ snapshots and 100 Monte Carlo runs. Due to the powerful out-of-band DOAs, element-space always has zero probability of resolution, so is not plotted in Fig. 2.11. For smaller out-of-band source power in Fig. 2.11(c), the minimum possible RMSE ($\approx 10^{-3}$), which is the distance between each true DOA and the closest grid point, is achieved. This example shows that designing discrete-frequency filters specifically for CBS dictionaries makes a difference.

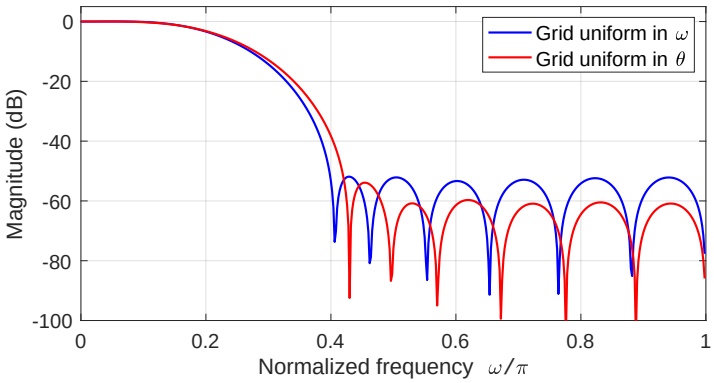


Figure 2.12: Responses of minimax-discrete filters designed based on different types of grids.

Finally, Fig. 2.12 shows how the design result of minimax-discrete type filers is affected by the type of grid. The filter length is $L = 16$. The passband is $\Omega_p = [0, 0.1\pi]$, and the stopband is $\Omega_s = [0.4\pi, \pi]$. Grids of 50 points uniform in

ω and uniform in θ are compared. Due to the denser grid points in high frequencies, the latter has a better attenuation in the high-frequency range at the expense of a worse response around the transition band. Hence, if there are powerful out-of-band sources far from the transition band, the latter is better, but if they are near the transition band, the former is better.

2.5 Sliding-Capon Based Convolutional Beamspace

In this section, we propose a new method to design the CBS FIR filter $H(z)$ based on the technique of Capon beamforming [36]. In Sec. 2.2, the filter was designed by standard methods such as the equiripple method or the window method [63]. These methods yield a reasonably well-behaved lowpass filter, and sources falling in the stopband are assumed to be sufficiently attenuated, so we can identify the passband DOAs. However, such a filter was not designed by taking input statistics into account. In this section, we show that by designing the filter as a sliding (convolutional) Capon beamformer, the sources that fall in the stopband can be better suppressed, as they are treated as "interference" in the Capon method. This new method is called *Capon-CBS*. DOA estimation using subspace methods following Capon-CBS performs better than those using traditional CBS. Capon preprocessing was done earlier in the context of atomic norm minimization [82], but our focus here is *convolutional* Capon preprocessing.

Unlike in beamforming where we have an assumed look direction ω_0, in DOA estimation under CBS, we require a flat passband in which the DOAs of interest are located as shown in Sec. 2.2. However, the Capon beamformer typically does not yield a flat passband in the neighborhood of ω_0. To make the passband look like a conventional flat passband, we can use a robust Capon beamforming method [83–85]. In this section, we use the method in [83]. Even with the robust method, the overall computational complexity of Capon-CBS is still much lower than element-space. Capon-CBS can offer higher probability of resolution and smaller mean square error for DOA estimation, compared to traditional CBS.

The outline of this section is as follows. The design of the CBS filter based on a sliding Capon beamformer is introduced in Sec. 2.5.1. The use of robust Capon beamforming and the detailed steps of Capon-CBS for DOA estimation are elaborated in Sec. 2.5.2. Simulations in Sec. 2.5.3 demonstrate the performance of Capon-CBS.

2.5.1 Sliding Capon Beamformer for CBS

In traditional beamforming, we consider the array output

$$\mathbf{x} = c_0\mathbf{a}_N(\omega_0) + \mathbf{u}, \tag{2.66}$$

where $\mathbf{u} = \mathbf{Ac}+\mathbf{e}$ is the interference plus noise. The quantity ω_0 represents the "look direction," i.e., the direction in which we want to point the beam, c_0 is the amplitude of the signal coming from that direction, and $\mathbf{A}$, $\mathbf{c}$, $\mathbf{e}$ are as defined in (2.1). The output of the beamformer can be expressed as $\mathbf{h}^H\mathbf{x}$, where $\mathbf{h} = [h(0) \ \cdots \ h(N-1)]$ is a complex weighting vector. The output signal-to-interference-plus-noise ratio (SINR) of the beamformer is defined as

$$\text{SINR} = \frac{\text{E}[|c_0\mathbf{h}^H\mathbf{a}_N(\omega_0)|^2]}{\text{E}[|\mathbf{h}^H\mathbf{u}|^2]} = \frac{p_0|\mathbf{h}^H\mathbf{a}_N(\omega_0)|^2}{\mathbf{h}^H\mathbf{R}_{\mathbf{uu}}\mathbf{h}}, \tag{2.67}$$

where $p_0 = \text{E}[|c_0|^2]$ and $\mathbf{R}_{\mathbf{uu}} = \text{E}[\mathbf{uu}^H]$. When the signal is uncorrelated with the interference, the Capon beamformer [36], which is the solution to the optimization problem

$$\begin{aligned} \min_{\mathbf{h}} \quad & \mathbf{h}^H\mathbf{R}_{\mathbf{xx}}\mathbf{h} \\ \text{subject to} \quad & \mathbf{h}^H\mathbf{a}_N(\omega_0) = 1, \end{aligned} \tag{2.68}$$

maximizes the SINR, where $\mathbf{R}_{\mathbf{xx}} = \text{E}[\mathbf{xx}^H]$. The solution to this problem is given by

$$\mathbf{h} = \frac{\mathbf{R}_{\mathbf{xx}}^{-1}\mathbf{a}_N(\omega_0)}{\mathbf{a}_N^H(\omega_0)\mathbf{R}_{\mathbf{xx}}^{-1}\mathbf{a}_N(\omega_0)}. \tag{2.69}$$

In practice, $\mathbf{R}_{\mathbf{xx}}$ is replaced by its estimate $\widehat{\mathbf{R}}_{\mathbf{xx}} = \sum_{k=1}^{K} \mathbf{x}[k]\mathbf{x}^H[k]/K$, where K is the number of snapshots.

Now consider Fig. 2.13(a) where we have an N-sensor array, with an L-tap beamformer, with $L < N$. Assume that the taps are chosen as in the optimal Capon [36] beamformer for the first L sensors. Now, if we shift the taps to the right as shown in Fig. 2.13(b), then this continues to be the optimal beamformer for the next L sensors, provided the $L \times L$ covariance matrix for the first L sensor outputs is the same as that for the second L sensor outputs. This happens if there is spatial wide sense stationarity (WSS) in the sensor output signals. So, as the taps slide along from extreme left to extreme right as in Fig. 2.13, the outputs $y(L-1), y(L), \ldots, y(N-1)$ can be regarded as optimal beamformer outputs, generated from successive subarrays of

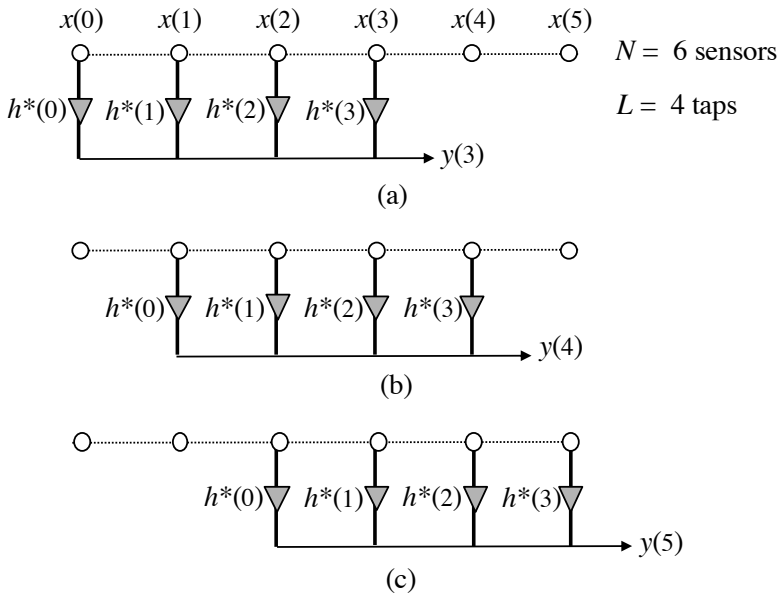


Figure 2.13: (a)-(c) An N-sensor array with an L-sensor sliding beamformer.

the original array. The spatial WSS property is achieved under the usual assumption that all the sources c_i are uncorrelated because in this case,

$$\mathbf{R_{xx}} = \sum_{i=0}^{D} p_i \mathbf{a}_N(\omega_i)\mathbf{a}_N^H(\omega_i) + \sigma_e^2 \mathbf{I} \tag{2.70}$$

is a Toeplitz matrix, where $p_i = \mathrm{E}[|c_i|^2]$.

Now we compare the sliding beamformer in Fig. 2.13 with the convolution stage of CBS (2.5) or Fig. 2.1. The two systems are the same except that the taps in one are reversed and conjugated with respect to the other. So in the CBS stage, if we choose the filter coefficients to be the reverse-conjugates of the L-tap Capon beamformer, then the CBS signals $y(n)$ are optimal beamformer outputs and furthermore belong in the "passband" of the beamformer! The CBS stage designed in this way will be called *sliding Capon-CBS* or just Capon-CBS.

In traditional CBS (Sec. 2.2), the filter $H(z)$ was designed as a standard (say, a Parks-McClellan) lowpass filter. Sources falling in the stopband are assumed to be sufficiently attenuated, so we can identify the passband DOAs. But the filter was not designed by taking input statistics into account. The Capon-CBS filter above is designed based on data, so it should do a better job of suppressing the sources that fall in the stopband, as they are treated as "interference" in the Capon method.

2.5.2 Capon-CBS for DOA Estimation

The sliding Capon beamformer produces the $N - L + 1$ samples $y(n)$ in (2.5). We can use these to perform DOA estimation similar to Sec. 2.2, or we can perform a second stage of beamforming using these $y(n)$. In this section, we focus on DOA estimation. Unlike for beamforming where we have an assumed look direction ω_0, for DOA estimation following the CBS stage, we require a flat passband in which the DOAs of interest are located. Yet the Capon beamformer (2.69) typically does not yield a flat passband in the neighborhood of ω_0, although $|H(e^{j\omega_0})| = 1$. To make the passband look like a conventional flat passband, we can adopt any robust Capon beamforming method [83–85] so that $|H(e^{j\omega})| \approx 1$ for all ω in the passband. Here we use the method in [83] so that instead of (2.68) (with N replaced by L), the L-tap robust beamformer is the solution to the problem

$$\begin{aligned} \min_{\mathbf{h}} \quad & \mathbf{h}^H \mathbf{R}_{\mathbf{xx}} \mathbf{h} \\ \text{subject to} \quad & \mathrm{Re}\{\mathbf{h}^H \mathbf{a}\} \geq 1 \ \ \forall \mathbf{a} \in \mathcal{E}, \end{aligned} \tag{2.71}$$

where Re denotes the real part, and $\mathcal{E}$ is an L-dimensional ellipsoid that covers the range of values of $\mathbf{a}_L(\omega)$ in the passband. It is shown in [83] that this problem can be solved by Lagrange multiplier methods, and the computational complexity is still $O(L^3)$, the same as that for computing the Capon beamformer. This is important because one significant advantage of CBS over element-space is the lower computational complexity, which should not be compromised. We will show by simulations that this advantage is still preserved when the robust beamforming method in [83] is used.

For purposes of computation [83], the beamformer will be expressed explicitly in terms of real and imaginary components: $[\mathrm{Re}\{\mathbf{h}^T\} \ \mathrm{Im}\{\mathbf{h}^T\}]^T \triangleq \widetilde{\mathbf{h}}$ where $\mathbf{h} = [h^*(L-1) \ h^*(L-2) \ \cdots \ h^*(0)]^T$. In summary, Capon-CBS design for DOA estimation involves the following steps:

1. Design the CBS filter coefficients $\widetilde{\mathbf{h}}$ as the solution to

$$\begin{aligned} \min_{\widetilde{\mathbf{h}}} \quad & \widetilde{\mathbf{h}}^H \widetilde{\mathbf{R}}_L \widetilde{\mathbf{h}} \\ \text{subject to} \quad & \mathrm{Re}\{\widetilde{\mathbf{h}}^H \mathbf{a}\} \geq 1 \ \ \forall \mathbf{a} \in \mathcal{E}, \end{aligned} \tag{2.72}$$

   where $\widetilde{\mathbf{R}}_L = \sum_{i=0}^{N-L} \mathrm{E}[\widetilde{\mathbf{x}}_{L,i} \widetilde{\mathbf{x}}_{L,i}^H]/(N-L+1)$ and $\widetilde{\mathbf{x}}_{L,i} = [\mathrm{Re}\{\mathbf{x}_{L,i}^T\} \ \mathrm{Im}\{\mathbf{x}_{L,i}^T\}]^T$ with $\mathbf{x}_{L,i} = [x(i) \ x(i+1) \ \cdots \ x(i+L-1)]^T$.

2. Obtain the filtered and decimated outputs $\mathbf{v}_l$ of the Capon-CBS stage, as defined in (2.14).

3. Use $\mathbf{R}_{\text{ave}}$ in (2.19) to do root-MUSIC or ESPRIT.

In (2.72), the ellipsoid $\mathcal{E}$ can be designed from r equally spaced samples of the array response in the passband [83]. Assume the passband of $H(e^{j\omega})$ is $(-\pi/2M, \pi/2M)$. Let $\bar{\omega}_i = -\pi/2M + \pi i/(r-1)M$, $i = 0, \ldots, r-1$. Then we choose

$$\mathcal{E} = \{\mathbf{P}^{1/2}\mathbf{u} + \mathbf{c} \mid \|\mathbf{u}\| \leq 1\}, \tag{2.73}$$

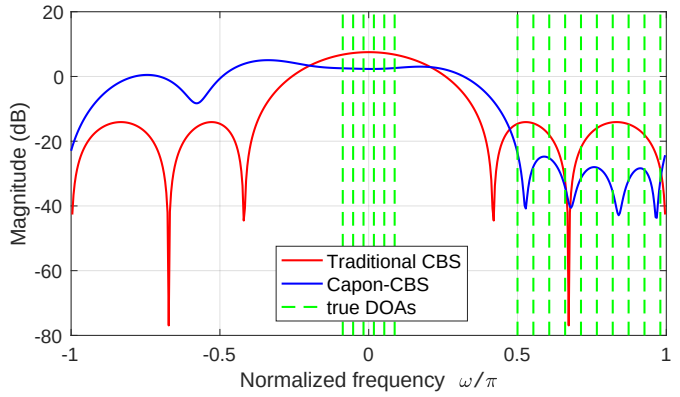
where $\mathbf{c} = \sum_{i=0}^{r-1} \widetilde{\mathbf{a}}(\bar{\omega}_i)/r$, $\mathbf{P}' = \sum_{i=0}^{r-1} (\widetilde{\mathbf{a}}(\bar{\omega}_i) - \mathbf{c})(\widetilde{\mathbf{a}}(\bar{\omega}_i) - \mathbf{c})^H$, and $\mathbf{P} = \alpha\mathbf{P}'$ with

$$\alpha = \max_i \, (\widetilde{\mathbf{a}}(\bar{\omega}_i) - \mathbf{c})^H (\mathbf{P}')^+ (\widetilde{\mathbf{a}}(\bar{\omega}_i) - \mathbf{c}), \tag{2.74}$$

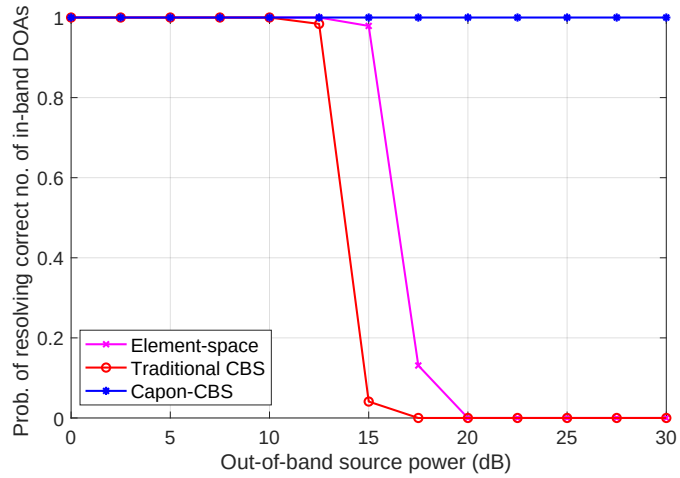
and $\widetilde{\mathbf{a}}(\omega) = [\text{Re}\{\mathbf{a}_L^T(\omega)\} \;\; \text{Im}\{\mathbf{a}_L^T(\omega)\}]^T$. Here $(\cdot)^+$ denotes the pseudoinverse.

2.5.3 Simulations

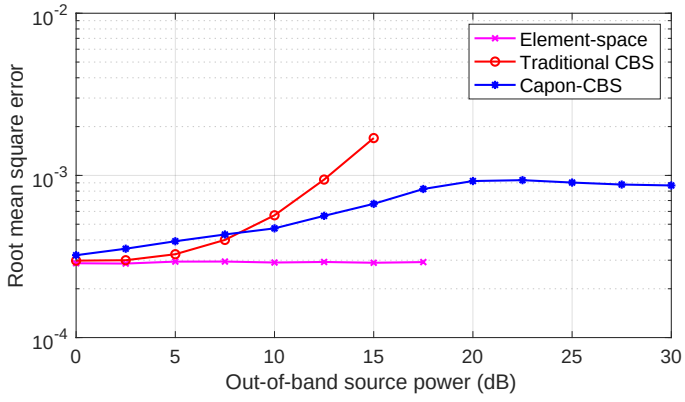
We compare Capon-CBS for DOA estimation with element-space, and traditional CBS. Consider a ULA with $N = 99$ sensors. There are 6 in-band DOAs, at $\theta = -5°, -3°, -1°, 1°, 3°, 5°$. There are 10 uniformly spaced out-of-band DOAs at $\omega = 0.5\pi, 0.5\pi + \delta, 0.5\pi + 2\delta, \ldots, 0.98\pi$ with $\delta = 0.48\pi/9$. Each in-band source has power 1, the noise variance is $\sigma_e^2 = 1$, and the out-of-band sources have equal powers, but varying through the experiment. All sources are uncorrelated. The filter length is $L = 8$, and the decimation ratio is $M = 4$. The Capon-CBS filter is designed using (2.72) and (2.73) with $r = 10$, while for traditional CBS, $H(z)$ is designed to be a lowpass Parks-McClellan filter [63], with passband edge $\pi/2M$ and stopband edge $3\pi/2M$. Fig. 2.14(a) shows that the Capon-CBS filter indeed has a flat passband while suppressing the out-of-band sources. Using the Capon-CBS output, we perform DOA estimation using root-MUSIC. Covariance estimates are obtained by using 100 snapshots, and we average 1000 Monte Carlo runs to get the plots. The number of DOAs is assumed unknown and estimated based on the distribution of eigenvalues of the covariance (see Fig. 2.4). As shown in Fig. 2.14(b), the probability of resolution of element-space and traditional CBS drops to zero as out-of-band source power increases, but that of Capon-CBS is unity throughout (it starts to decrease only after the power gets larger than about 45 dB, which is uncommonly large). Fig. 2.14(c) shows root mean square error (RMSE), obtained by averaging square errors measured in ω over all in-band DOAs and over



(a)



(b)

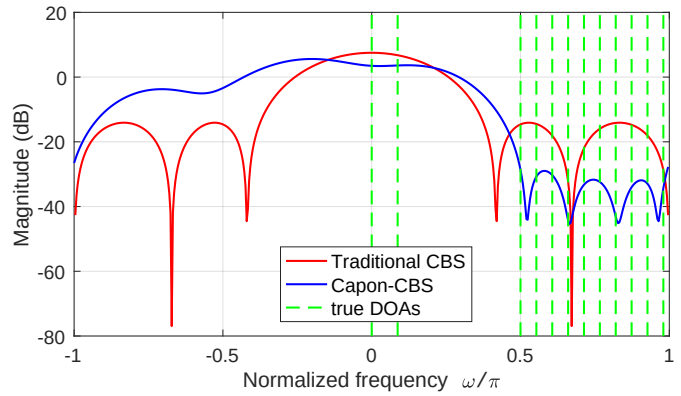


(c)

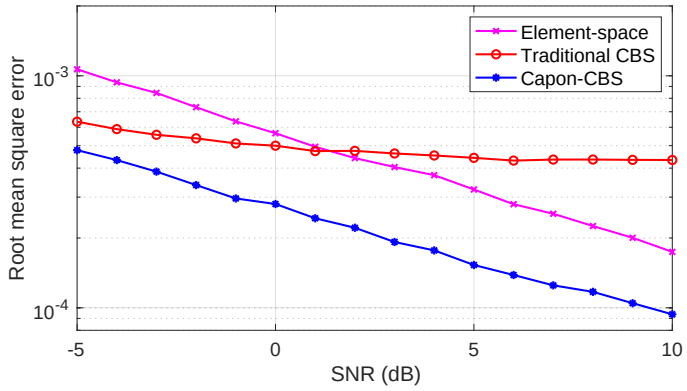
Figure 2.14: Performance of CBS and element-space methods when there are strong out-of-band sources. (a) Filter responses when the out-of-band source power is 15 dB. (b) Probability of resolution. (c) RMSE in DOA estimation.

those Monte Carlo runs that obtain the correct number of in-band DOAs. For smaller out-of-band source power, both CBS methods have similar RMSE, but the RMSE of

traditional CBS gets larger for middle out-of-band source power (10 – 15 dB). For larger out-of-band source power, we cannot plot element-space and traditional CBS because they have zero probability of resolution, but Capon-CBS still has reasonably good RMSE. In this example, CBS cannot beat element-space in terms of RMSE since the sources are uncorrelated [23]. The running time per Monte Carlo run for element-space, traditional CBS, and Capon-CBS is 0.054, 0.0026, and 0.0048 seconds, respectively. Both CBS methods have an order of magnitude reduction in complexity.



(a)



(b)

Figure 2.15: Performance of CBS and element-space methods when there are correlated sources. (a) Filter responses when SNR is 0 dB. (b) RMSE in DOA estimation.

In the second example, we consider correlated sources. Although the spatial WSS property may not be satisfied in this case, we can still follow the proposed steps of Capon-CBS. All parameters not mentioned here are the same as the first example. There are 2 in-band DOAs at angles $\theta = 0°, 5°$, and the 10 out-of-band DOAs are the same as in the first example. Each in-band source has power 1, and each out-of-band

source has power 15 dB. Each pair of the 12 sources have correlation coefficient 0.6, except that the two in-band sources are uncorrelated. The number of DOAs is assumed known. Fig. 2.15(a) shows that the Capon-CBS filter has better stopband attenuation than traditional CBS. Capon-CBS filters keep a balance between attenuating interference and noise, which is the benefit of design based on data. We then consider DOA estimation using root-MUSIC, with the number of DOAs assumed known. Fig. 2.15(b) shows that the RMSE of Capon-CBS is significantly smaller than element-space and traditional CBS. The constant gap between element-space and Capon-CBS is because the in-band sources are correlated with the out-of-band sources. This correlation makes the RMSE of element-space larger as we saw in Sec. 2.2, but Capon-CBS can effectively null all the out-of-band sources, so the RMSE of Capon-CBS remains small as if there are no correlated sources. This example shows the clear advantage of Capon-CBS even when there are correlated sources.

2.6 Convolutional Beamspace Using IIR Filters

In Sec. 2.2, CBS uses a spatial finite-impulse-response (FIR) filter $H(z)$ to restrict the ULA output to an angular sector. The "CBS trick" is to retain only the *steady-state* part of the output, whereby the Vandermonde structure is preserved. Uniform decimation is then used for dimensionality and complexity reduction, after which we can readily estimate DOAs using subspace methods.

A natural question is whether we can use infinite-impulse-response (IIR) filters instead of FIR. We explore this in this section. While the idea of using IIR filters instead of FIR might seem minor, there are some nontrivial details as we shall see. Also, IIR filtering has not been widely used in array processing in the past. With large arrays becoming more prevalent today [9, 45], perhaps the time has come for more widespread use of IIR methods. They are certainly very promising for CBS-based DOA estimation as we shall see.

In CBS, even for large arrays the FIR filter length L has to be small compared to the number of sensors N. The reason is that the MSE of DOA estimates is approximately proportional to $(N - L + 1)^{-3}$ as shown in (2.32). (This is because the effective output length is $N - L + 1$ after discarding the transient samples.) So FIR-CBS has larger estimation errors as L gets closer to N. On the other hand, with small L we cannot produce very sharp transition bands. This is where IIR filters help. They not only produce beamspace filters with sharper transition bands, they are also more

economic than FIR filters (i.e., a much smaller filter order is required). That is, IIR-CBS has lower computational complexity. Furthermore, IIR-CBS can achieve smaller DOA estimation errors than FIR-CBS. This is because the region where the transient component of IIR filters is significant is typically smaller than the transient duration in the FIR case as we shall see. These advantages of IIR-CBS will be verified by numerical examples.

In the following, the new method IIR-CBS is introduced in Sec. 2.6.1. Then, Sec. 2.6.2 demonstrates the advantage of IIR-CBS using simulations.

2.6.1 CBS Using IIR Filters

As explained in Sec. 2.2.1, in traditional CBS, the ULA output $x(n), 0 \le n \le N-1$ is filtered with an FIR filter $H(z) = \sum_{n=0}^{L-1} h(n)z^{-n}$ to obtain the output $y(n)$. That is, the z-transform of $y(n)$ can be computed as [63]

$$Y(z) = H(z)X(z), \tag{2.75}$$

where $X(z) = \sum_{n=0}^{N-1} x(n)z^{-n}$. Here $L < N$ and we retain only the *steady-state* part $\mathbf{y} = [y(L-1)\ y(L)\ \cdots\ y(N-1)]^T$, so that $\mathbf{y}$ is still represented by a Vandermonde matrix $\mathbf{A}_L$ as in (2.9). In this section, we propose to use an IIR filter

$$H(z) = \frac{P(z)}{D(z)} = \sum_{n=0}^{\infty} h(n)z^{-n} \tag{2.76}$$

in (2.75), where $P(z)$ and $D(z)$ are degree-R polynomials in z^{-1}. The motivation is that an IIR filter often requires a much lower order for the same set of magnitude response specifications [66]. This can lead to lower computational complexity and better DOA estimation performance, as we shall see later. The filter $H(z)$ can be designed as any standard IIR filter such as a (discrete-time) Chebyshev filter or elliptic filter [63, 66]. In the traditional FIR-CBS, there are only a finite number $L-1$ of transient output samples $y(0), \ldots, y(L-2)$ before reaching steady state, so we simply discard these samples in order to preserve the Vandermonde structure of the ULA. Now in IIR-CBS, since the filter is infinitely long, strictly speaking, there is no steady state. However, typically $|h(n)|$ decays as n increases. Suppose it decays to a negligible level after $n \ge L_{\mathrm{I}}$. Then we can similarly discard the $L_{\mathrm{I}}-1$ output samples $y(0), \ldots, y(L_{\mathrm{I}}-2)$ and define $\mathbf{y} = [y(L_{\mathrm{I}}-1)\ y(L_{\mathrm{I}}) \cdots y(N-1)]^T$ as the IIR-CBS output signal. Then the Vandermonde structure is approximately preserved because for $L_{\mathrm{I}}-1 \le n \le N-1$, we have

$$y(n) = \sum_{i=1}^{D} c_i H(e^{j\omega_i})e^{j\omega_i n} + q(n) + \sum_{k=0}^{n} h(k)e(n-k), \tag{2.77}$$

where $q(n) = \sum_{k=n+1}^{\infty} h(k) \sum_{i=1}^{D} c_i e^{j\omega_i(n-k)}$ is negligible. Thus, ignoring the noise term, we have

$$\mathbf{y} \approx \mathbf{A}_{L_\mathrm{I}} \mathbf{d}', \tag{2.78}$$

where $\mathbf{d}'$ has elements $d'_k = c_k e^{j(L_\mathrm{I}-1)\omega_k} H(e^{j\omega_k})$. This is similar to (2.9). Hence, we can decimate $\mathbf{y}$ and estimate DOAs based on the average covariance $\mathbf{R}_{\mathrm{ave}}$ of the decimated outputs using standard methods like root-MUSIC.

Now we compare the computational complexity of IIR-CBS and FIR-CBS. Suppose root-MUSIC is used to estimate DOAs. Then we need to do filtering, eigenvalue decomposition of $\mathbf{R}_{\mathrm{ave}}$, and root-finding. Given K snapshots of the N-sensor ULA output, Table 2.6 shows the total complexity for IIR-CBS with filter order R and for FIR-CBS with filter length L. In the expressions in Table 2.6, the first term (KNR or KNL) is the filtering complexity, and the second term is the eigendecomposition complexity. The complexity for root-finding is $O(n^2 \log n)$ [86], where $n = (N - L_\mathrm{I} + 1)/M$ for IIR-CBS and $n = (N - L + 1)/M$ for FIR-CBS, so it is ignored in the big-O notation. As mentioned earlier, an IIR filter often requires a much lower order, i.e.,

$$R \ll L, \tag{2.79}$$

for the same set of magnitude response specifications [66]. Hence, the complexity of IIR-CBS is typically much smaller than that of FIR-CBS, especially for large arrays (large N), which are becoming increasingly important [9, 45]. In Table 2.6, we also show numerical values by setting the parameters as in Examples 1 and 2 in Sec. 2.6.2. In both examples, $K = 500$, $N = 99$, $R = 5$, and $M = 4$. In Example 1, $L_\mathrm{I} = 12$ and $L = 20$. In Example 2, $L_\mathrm{I} = 16$ and $L = 56$. The numbers are computed exactly from the expressions in Table 2.6 without considering the hidden constant factors in the big-O notation. IIR-CBS indeed has much smaller complexity. In Sec. 2.6.2, we show that with less complexity, IIR-CBS can achieve similar or even better DOA estimation performance than FIR-CBS.

Table 2.6: Complexity of IIR-CBS and FIR-CBS

	Complexity	Example 1	Example 2
IIR-CBS	$O\left(KNR + \left(\frac{N-L_\mathrm{I}+1}{M}\right)^3\right)$	258,148	256,761
FIR-CBS	$O\left(KNL + \left(\frac{N-L+1}{M}\right)^3\right)$	998,000	2,773,331

IIR-CBS is particularly advantageous over FIR-CBS when we want to design a filter with a narrow transition band. This reduces the probability of a DOA appearing in the transition band, so it is of practical interest. An FIR filter requires a very large filter length L to have good stopband attenuation in this case, which means the effective filter output length $N - L + 1$ will be small. Since the MSE of DOA estimates is approximately proportional to $(N - L + 1)^{-3}$ as shown in (2.32), FIR-CBS will suffer large estimation error. By contrast, an IIR filter requires a relatively small order R, and we can also choose a reasonably small L_{I} to get good estimation performance. Such advantage will be verified by a numerical example in Sec. 2.6.2. The nonlinear phase response of IIR filters does not have a significant effect on the MSE of DOA estimates because the MSE mainly depends on the power of the source but not phase of the source amplitude (see Sec. 2.2.6).

Remark 1: The IIR filter $H(z) = P(z)/D(z)$ can be implemented with R multipliers, as explained below. A wide family of IIR filters, including Butterworth, Chebyshev, and elliptic filters, can be represented as

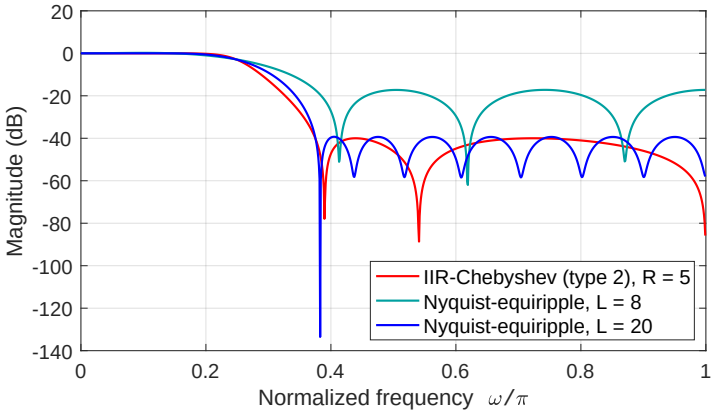
$$H(z) = [A_0(z) + A_1(z)]/2, \tag{2.80}$$

where $A_0(z)$ and $A_1(z)$ are stable unit-magnitude allpass filters [66, 87, 88], and their orders r_0 and r_1 satisfy $r_0 + r_1 = R$. Moreover, if the filter order R of $H(z)$ is odd, then $A_0(z)$ and $A_1(z)$ have real coefficients. Since a real coefficient allpass filter with order r can be implemented with r multipliers [66], we can implement (2.80) with R multipliers. We always use an odd R in this section.

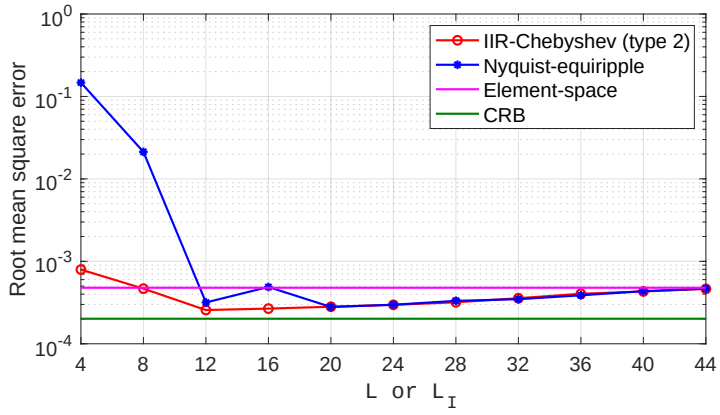
Remark 2: In Sec. 2.2.4, it was shown that the noise term after filtering and decimation can be whitened if $H(z)$ is a spectral factor of a Nyquist(M) filter $G(z)$ (i.e., the impulse response of $G(z)$ satisfies $g(Mn) = \delta(n)$ where $G(z)$ is such that $G(e^{j\omega}) = |H(e^{j\omega})|^2$). This makes it easy to find the noise eigenspace by computing eigenvectors of $\mathbf{R}_{\mathrm{ave}}$, which is what we do in simulations. We can also design an IIR filter $H(z)$ to be a spectral factor of an IIR Nyquist(M) filter to whiten the noise. There are many ways to design such filters [48, 89–91]. But in many cases, if $H(z)$ is a "good" IIR filter with total passband width $\approx 2\pi/M$ and ripples properly constrained, this Nyquist property is approximately satisfied, that is,

$$\max_{n \neq 0} |g(Mn)| \ll g(0) \quad \text{(nearly-Nyquist property)}. \tag{2.81}$$

For simplicity, this is what we use in simulations.



(a)



(b)

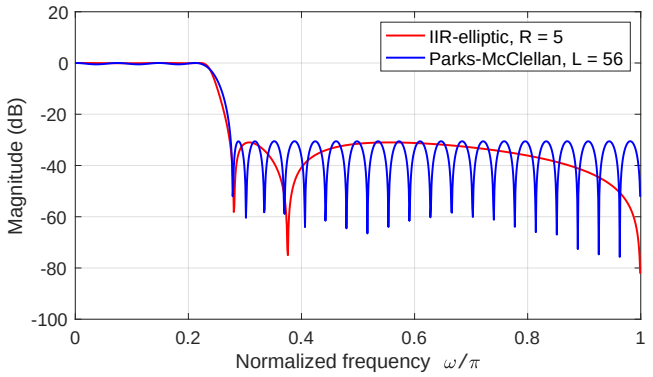
Figure 2.16: Performance of IIR-CBS using Chebyshev (type 2) filter for various L_{I}, FIR-CBS using Nyquist-equiripple filter for various L, and element-space. The IIR filter order is fixed at $R = 5$. (a) Responses of IIR and FIR filters used. The transition band for each filter is $\omega \in [0.125\pi, 0.375\pi]$. (b) RMSE of DOA estimates as we vary L for FIR-CBS (or L_{I} for IIR-CBS). Element-space CRB is also plotted.

2.6.2 Simulations

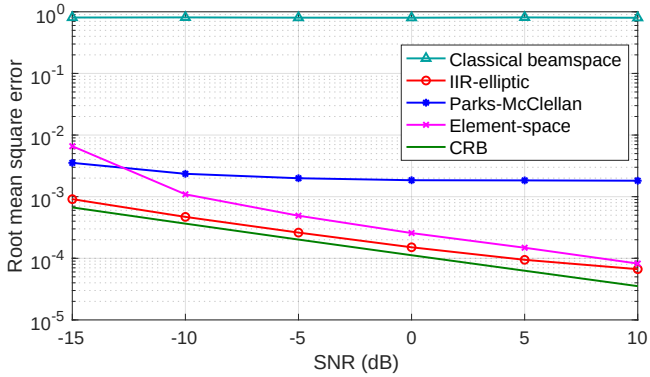
Example 1: We compare IIR-CBS for DOA estimation with FIR-CBS and element-space (i.e., directly using $\mathbf{x}$ to estimate DOAs). Consider a ULA with $N = 99$ sensors. There are 3 in-band (passband) sources with DOAs $\theta = -5°, 0°, 5°$, and 3 out-of-band (stopband) sources with DOAs $\theta = 40°, 60°, 80°$. Each in-band source has power 0 dB, each out-of-band source has power 20 dB, and the noise variance is $\sigma_e^2 = 5$ dB. Sources n and $n+3$ have a correlation coefficient $\rho = 0.9$ for $n = 1, 2, 3$. The decimation ratio is $M = 4$. For IIR-CBS, $H(z)$ is designed to be a lowpass Chebyshev (type 2) filter with order $R = 5$. For this filter, $\max_{n\neq 0} |g(Mn)| \approx 0.001 g(0)$, so it indeed satisfies the nearly-Nyquist property (2.81). For FIR-CBS,

$H(z)$ is designed to be a spectral factor of a lowpass Nyquist-equiripple filter [71]. The transition band for each filter is $\omega \in [0.125\pi, 0.375\pi]$. The parameter L_{I} for IIR-CBS and the FIR filter length L are varied in this experiment. Magnitude responses of the IIR filter and FIR filters with some typical lengths L used are shown in Fig. 2.16(a). The stopband attenuation of the IIR filter with order 5 is as good as that of the FIR filter with length 20. The stopband attenuation of the FIR filter with a shorter length, e.g. 8, is not so good. Then we perform DOA estimation using root-MUSIC. The number of DOAs is assumed known. Covariance estimates are obtained by using 500 snapshots, and we average 500 Monte Carlo runs to get the plot. Fig. 2.16(b) shows root mean square error (RMSE) obtained by averaging square errors measured in ω over all in-band DOAs, as we vary L or L_{I}. For IIR-CBS, there is an optimal L_{I}, 12 in this case, that gives the smallest RMSE. When L_{I} is too small, RMSE increases because the first few samples suffer from a significant transient effect due to the typically large $|h(n)|$ for small n. When L_{I} is too large, RMSE increases because we discard too many filter output samples and the effective output length $N - L_{\mathrm{I}} + 1$ is too small. For FIR-CBS, there is also an optimal L, 20 in this case, that gives the smallest RMSE. When L is too small, RMSE increases because the filter stopband attenuation is not good enough to suppress the out-of-band sources. When L is too large, RMSE increases because the effective filter output length is small. In this example, the RMSE of IIR-CBS is a lower bound for the RMSE of FIR-CBS. Moreover, for each simulation point when $L_{\mathrm{I}} = L \geq 12$, the computational complexity of IIR-CBS is smaller than that of FIR-CBS. That is, IIR-CBS is better in terms of both complexity and estimation performance. Meanwhile, when L_{I} and L are properly chosen, the RMSEs of both IIR-CBS and FIR-CBS are smaller than element-space and close to the element-space Cramér–Rao bound (CRB) [56] (i.e., CRB based on the original array output $\mathbf{x}$). Element-space typically has larger errors for correlated sources as shown in Sec. 2.2. Even though element-space CRB cannot be larger than beamspace CRB, it is possible for element-space MSE to be larger than beamspace MSE (see Sec. 2.2.6).

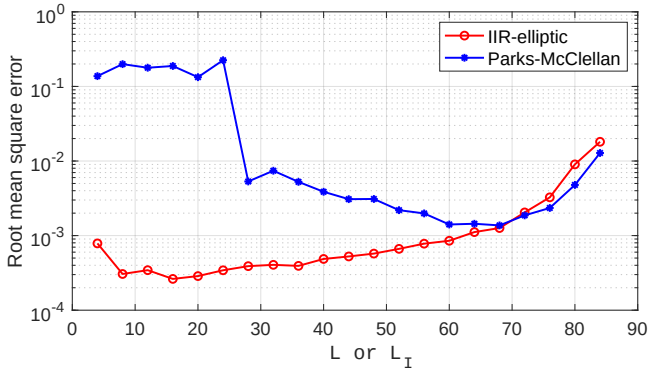
Example 2: We consider an example where CBS filters have a narrow transition band, $\omega \in [0.225\pi, 0.275\pi]$. All simulation parameters are the same as in the first example unless mentioned otherwise. For IIR-CBS, $H(z)$ is designed to be a lowpass elliptic filter with order $R = 5$. For this filter, $\max_{n\neq 0} |g(Mn)| \approx 0.04 g(0)$, so it satisfies (2.81). For FIR-CBS, $H(z)$ is designed to be a Parks-McClellan filter [63]. Magnitude responses of the IIR and FIR filters used are shown in Fig. 2.17(a). The stopband attenuation of the IIR filter with order 5 is as good as that of the FIR



(a)



(b)



(c)

Figure 2.17: Performance of IIR-CBS using elliptic filter, FIR-CBS using Parks-McClellan filter, classical beamspace, and element-space. The IIR filter order is fixed at $R = 5$. (a) Responses of IIR and FIR filters used. A narrow transition band $\omega \in [0.225\pi, 0.275\pi]$ is used for each filter. (b) RMSE of DOA estimates as we vary SNR. The IIR and FIR filters in (a) are used. We set $L_{\mathrm{I}} = 16$ for the IIR filter. Element-space CRB is also plotted. (c) RMSE of DOA estimates as we vary L for FIR-CBS (or L_{I} for IIR-CBS).

filter with length 56. IIR-CBS can thus obtain significant complexity reduction. Then we perform DOA estimation using root-MUSIC. Fig. 2.17(b) shows RMSE

of in-band DOA estimates for CBS using the IIR and FIR filters in Fig. 2.17(a) and element-space, as we vary SNR ($= 1/\sigma_e^2$). We set $L_{\mathrm{I}} = 16$ for the IIR filter. Here we also show the RMSE of classical beamspace using a 99×25 DFT matrix beamformer [2]. (Note that $25 \approx 99/M$.) The poor performance of classical beamspace is due to numerical sensitivity issues for a moderately large N as mentioned in [2]. RMSE of FIR-CBS is relatively large because the large $L = 56$ means the effective filter output length $N-L+1$ is small. However, the RMSE of IIR-CBS is the smallest and close to the element-space CRB. The gap between the IIR-CBS RMSE and CRB gets a bit larger as SNR increases because the fixed error term $q(n)$ in (2.77) does not depend on the noise. This example shows the benefit of introducing IIR filters in CBS because FIR-CBS does not work better than element-space, but IIR-CBS does. In fact, FIR-CBS has larger RMSE than IIR-CBS for a wide range of L. We vary L or L_{I} in Fig. 2.17(c) while fixing $\sigma_e^2 = 5$ dB. Since we want a narrow transition band in this example, a much larger length L is required for the FIR filter to have enough stopband attenuation to suppress the out-of-band sources, but a large L results in a small effective filter output length and large RMSE. By contrast, an IIR filter needs only a small order to have enough stopband attenuation, so it is possible to get much smaller RMSE.

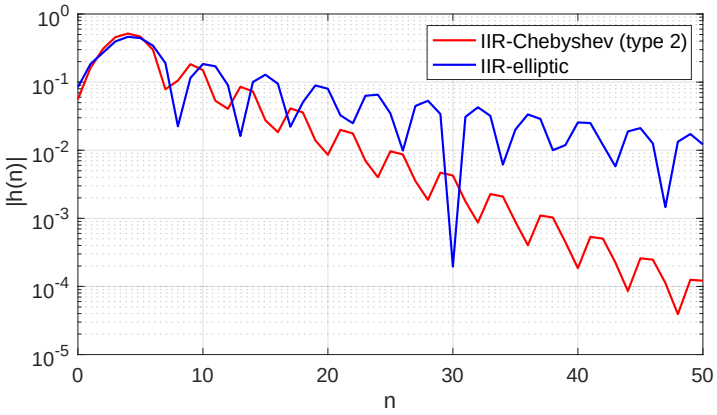


Figure 2.18: Magnitude of impulse responses of the Chebyshev filter in Fig. 2.16(a) and elliptic filter in Fig. 2.17(a).

Finally, Fig. 2.18 shows the impulse responses of the Chebyshev filter in Fig. 2.16(a) and elliptic filter in Fig. 2.17(a). The impulse response of the Chebyshev filter decays faster than that of the elliptic filter. This explains why the optimal $L_{\mathrm{I}} = 12$ in Fig. 2.16(b) is smaller than the optimal $L_{\mathrm{I}} = 16$ in Fig. 2.17(c). We can discard less output samples for the Chebyshev filter.

2.7 Concluding Remarks

In this chapter, we introduce the convolutional beamspace (CBS) as an alternative to classical beamspace methods of array processing. While enjoying the computational advantages of classical beamspace, CBS also allows the direct use of root-MUSIC and ESPRIT without any complicated preprocessing. A simple error analysis shows that CBS can have better estimation performance when the sources are correlated. We also develop CBS methods for coarrays of sparse arrays, and for dictionary based methods which use dictionaries of steering vectors to obtain sparse representations of array data. Due to dimension reduction and effective filtering of out-of-band sources, many advantages are obtained across all these frameworks, such as lower computational complexity, better DOA estimates, and improved resolution. We also propose a new method to design the CBS filter based on a sliding (convolutional) Capon beamformer. Compared to traditional CBS, the Capon-CBS filter is designed based on data, so it can better suppress the sources that fall in the stopband. We show by simulations that Capon-CBS can offer higher probability of resolution and smaller RMSE for DOA estimation. Besides, a new variant of CBS using IIR filters is introduced. Since an IIR filter typically requires a much lower order for the same set of magnitude response specifications, smaller complexity and better estimation performance can be obtained compared to FIR-CBS, as verified in simulations. The last numerical example for IIR-CBS shows how an optimal L_{I}, which controls the number of discarded filter output samples, is related to the filter impulse response. A topic for future investigation would be to find a more systematic way to choose L_{I}.

For FIR-CBS, we can consider nonlinear phase filters instead of linear phase filters like Parks-McClellan in order to decrease the filter order for comparable magnitude response. Moreover, in this chapter, we consider CBS with dictionary methods, which are grid-based methods and suffer from off-grid errors. Thus, one may also want to consider CBS with gridless sparse recovery methods such as atomic norm minimization method [92]. Atomic norm minimization method suffers from high computational complexity, so it will be beneficial to use CBS with it.

Chapter 3

HYBRID CONVOLUTIONAL BEAMSPACE FOR DOA ESTIMATION

3.1 Introduction

In recent years, array signal processing for millimeter waves (mmWaves) has become an important topic [5, 9–11, 19, 93, 94]. A prominent application is in mmWave MIMO systems in 5G [10, 11]. A main reason for considering mmWaves is the potential to offer more bandwidth than the highly occupied lower-frequency microwave bands, but there are also new challenges, such as strong path loss at the high frequencies of mmWaves [10]. To tackle this, large arrays are used, which are practical due to the small sizes of mmWave antennas, to get large beamforming gain [11]. However, the high power consumption of mixed signal components prevents the use of one RF chain per antenna and purely digital signal processing for large mmWave arrays. Instead, many have proposed to use hybrid analog and digital processing. Given a large receive uniform linear array (ULA), the output is first processed by an analog combiner. The combiner gives a lower-dimensional output, which is then passed through a smaller number of RF chains [11]. Digital processing is finally done on the lower-dimensional RF chain output to achieve the required task like direction-of-arrival (DOA) estimation or MIMO channel estimation. Such dimensionality reduction can lower hardware complexity and power consumption due to reduced number of RF chains. In this chapter, we consider particularly DOA estimation. For this application, the analog combiner is also called a *beamspace* matrix [2, 18]. Its lower-dimensional output contains information only about DOAs in a particular beam or passband.

To reduce hardware cost, the analog combiner is often implemented by a phase shifter network [11]. Such implementation imposes the constraint that all the entries of the analog processing matrix should have unit modulus (i.e., unit absolute value). In classical beamspace methods [2, 18], the combiners are designed as DFT beamformers. DFT beamformers have unit-modulus entries, so they can be directly applied in the hybrid architecture as in [4, 19]. However, only 13 dB attenuation is obtained at the first sidelobe, and the poorly-attenuated stopband sources can cause large estimation errors. Other standard digital filters like Parks-McClellan

filters [63] can be used to achieve larger stopband attenuation. This motivates us to propose a new hybrid beamspace method based on digital filters.

In this chapter, we propose to use the idea of *convolutional beamspace (CBS)* described in Chapter 2 to design the hybrid analog and digital processors for passive arrays, so we call the method *hybrid CBS*. CBS is a beamspace DOA estimation method which achieves advantages including complexity reduction, higher DOA resolution, and smaller bias, just as classical beamspace methods do [2, 8, 17, 18]. In CBS, the beamspace matrix is a banded Toeplitz matrix, so the processing is equivalent to convolving the array output with a digital filter. After filtering, the output is represented only by sources falling in the passband, so we can *decimate* (*downsample*) the output without causing DOA ambiguities. It is such decimation that contributes to significant complexity reduction and that makes CBS appealing for mmWave arrays. With CBS, only a small number of RF chains is needed even for large arrays.

Two decimation schemes will be explained in this chapter: *uniform decimation* and *nonuniform decimation*. Uniform decimation is used in traditional CBS in Chapter 2, while nonuniform decimation is proposed in this chapter. Given any input $x(n)$, a uniform decimator [67, Sec. II] gives output $y(n) = x(Mn)$, where M is a positive integer called decimation ratio. In the new nonuniform scheme, the decimator output samples $y(i)$ correspond to the sensor locations $\{\tilde{n}_i\}$ of a standard sparse (nonuniform) array, dilated by an integer factor M, i.e., $y(i) = x(M\tilde{n}_i)$. We will use the phrase "virtual sparse array" to refer to this array since it is not a physical array. This virtual sparse array can be based on commonly used sparse arrays such as the nested array [32], coprime array [40], and minimum redundancy array (MRA) [3]. The nonuniformly decimated output is viewed as the output of the virtual dilated sparse array in response to the sources which fall in the passband of the CBS filter. Hence, we can use its *difference coarray* [32] to increase the number of identifiable sources. The details of the coarray method are explained in Sec. 3.2.2. Since the sparse array is dilated by M times, we get a central sparse ULA segment with adjacent element spacing M in the difference coarray. Despite the M-sparsity, we show that by using a CBS filter with passband width $2\pi/M$, the DOAs can still be estimated without ambiguities.

As mentioned earlier, the analog combiner is often implemented by phase shifters with unit-modulus entries. A standard digital filter typically does not have constant-modulus coefficients, so the resulting beamspace matrix of CBS cannot be directly

implemented in the analog domain. To tackle the constant-modulus constraints, we use an important result [48] that any complex vector $\mathbf{v}$ can be expressed as a linear combination of two vectors with unit-modulus entries. To be precise, there exist complex numbers a_1 and a_2 and complex vectors $\mathbf{v}_1$ and $\mathbf{v}_2$ satisfying $|[\mathbf{v}_1]_i| = |[\mathbf{v}_2]_i| = 1$ for all i such that $\mathbf{v} = a_1\mathbf{v}_1 + a_2\mathbf{v}_2$. It is even possible to let $a_1 = a_2$. A key fact is that two vectors with unit-modulus entries are sufficient. It is not necessary to include more than two.

The main contributions of this chapter include the following:

- We propose a new hybrid DOA estimation method, called hybrid convolutional beamspace (CBS), for mmWave receive arrays. The combiner is designed to be an FIR filter followed by a decimator (downsampler). This decimation reduces dimension a lot, so we can process the combiner output with a small number of RF chains even for a large array.

- A new scheme of CBS is proposed based on nonuniform decimation and coarray method, which allows the estimation of more sources than RF chains (not achievable by classical beamspace [2] or uniform hybrid CBS). The retained samples correspond to the sensor locations of a virtual sparse array, dilated by an integer factor M, which results in M times larger coarray aperture (than when there is no dilation). The aperture of an array is the difference between its extreme elements, and a larger aperture often gives better estimation performance [49, 50]. Given a fixed number of RF chains, the nonuniform scheme can achieve large array aperture and thus better estimation performance than uniform scheme. Or, nonuniform scheme requires fewer RF chains to achieve the same estimation performance.

- We show how to implement CBS in a hybrid architecture with an analog phase shifter network. In particular, any CBS filter coefficients can be implemented despite the unit-modulus constraint. Moreover, the required number of RF chains is as small as the dimension of the decimated CBS output (as opposed to [48, 95], using twice the number of RF chains). As no approximation is involved in this hybrid analog and digital realization of CBS, no performance loss is incurred for DOA estimation. Similarly, hybrid CBS inherits all the advantages of the traditional purely digital CBS.

- A new method for decorrelating sources is proposed based on designing the CBS filters for different snapshots to be different delayed versions of a proto-

type filter. Since the filters are delayed versions, they have the same magnitude response. The method can be used for both uniform CBS and nonuniform CBS, but it is more crucial to the latter because we assume uncorrelated sources when deriving coarray methods. The delays are chosen to be random or deterministic. We typically achieve similar estimation performance for both, so deterministic delays are recommended as they are easier to implement in the analog processors.

- We derive the Cramér–Rao bound (CRB) [43] for DOA estimates based on the output of any beamspace processor for a passive array. Since CBS is a special case of beamspace methods, we can apply the CRB result and compare to MSE performance via numerical examples. The CRBs of different beamspace methods can also be used to compare their fundamental limits.

- We propose hybrid Capon-CBS. As a variant of CBS, Capon-CBS described in Sec. 2.5 is composed of two stages. In the first stage, we design the CBS filter based on the idea of Capon beamforming [36]. The resulting Capon-CBS filter can better suppress stopband sources because input statistics are taken into account. In the second stage, we simply implement the obtained Capon-CBS filter as in traditional CBS. Hence, the purpose here is to show how to realize the first stage in a hybrid analog and digital architecture with RF chain constraints. The main idea is to use nonuniformly decimated samples and work in the coarray domain, in order to obtain an expression for the Capon-CBS filter's coefficients.

With all the details explained above, we now emphasize that the relevance of mmWaves to the proposed method appears in the following aspects. First, large arrays are often implemented for mmWaves [11], and large arrays allow us to use a large enough filter length to design a good CBS filter. Second, mmWave arrays often require hybrid processing to reduce hardware complexity, and hybrid CBS is a hybrid method which can achieve the same performance as its digital counterpart. Finally, we usually use few RF chains for mmWave arrays to lower hardware cost and power consumption, and the new scheme of hybrid CBS using nonuniform decimation and coarray method enables us to identify more sources than the number of RF chains. The exact value of the mmWave source frequency or wavelength λ is not crucial since we typically design the antenna unit spacing as $\lambda/2$ [40], which makes the array output independent of λ (see Sec. 1.1).

Nonuniformly decimating a convolution output also appears in [5], but their design is not for filtering but for decorrelating sources. Without the filtering effect, the dilation factor M cannot be applied. Thus, the effective aperture of our coarray is M times larger than that of [5]. In hybrid CBS with nonuniform decimation, it is the aperture of the virtual dilated sparse array, i.e., sampling grid aperture, that determines the performance. Hence, our method yields higher DOA resolution and smaller estimation errors than methods like [5], as we shall see in simulations (Figs. 3.5 and 3.6). Meanwhile, we note that the effective aperture can never be larger than the aperture of the original large physical ULA. The method in [5] is for 2-dimensional (2-D) mmWave MIMO channel estimation, whereas we only consider 1-dimensional DOA estimation in this chapter. We can also extend hybrid CBS to the 2-D case, which can be found in Chapter 4. However, the 1-D version is much easier to read and understand, and is important for readers interested only in passive sensing (i.e., passive DOA estimation). This is the motivation for this chapter.

Chapter outline: The details of hybrid CBS for passive arrays are given in Sec. 3.2, and CBS with uniform decimation and with nonuniform decimation are both introduced. We show how to implement them in a hybrid analog and digital architecture in Sec. 3.3. The new method for decorrelating sources based on filter delays in CBS is presented in Sec. 3.4. In Sec. 3.5, we derive the CRB for DOA estimates based on any beamspace-processed passive array output. Then, in Sec. 3.6, we show numerical examples for hybrid CBS and compare the MSE performance to the CRB. We then introduce hybrid Capon-CBS for passive arrays and show some numerical examples in Sec. 3.7. Finally, the conclusion is given in Sec. 3.8.

3.2 Hybrid CBS for Passive Arrays

In this section, we introduce the mathematical models of the two schemes, uniform and nonuniform decimation, of hybrid CBS. Their hybrid implementations will be explained in Sec. 3.3. In the following, the system model of CBS in Sec. 2.2 is reviewed for easy reference. We consider an N-sensor passive ULA with sensor spacing $\lambda/2$. The array receives D monochromatic plane waves of wavelength λ with DOAs $\theta_i \in [-\pi/2, \pi/2)$ measured from the normal to the line of array. Hence, the array output is

$$\mathbf{x} = [x(0)\ x(1)\ \cdots\ x(N-1)]^T = \mathbf{A}\mathbf{c} + \mathbf{e}, \tag{3.1}$$

where $\mathbf{c}$ contains source amplitudes c_i, and $\mathbf{e}$ is additive noise. Here, $\mathbf{A} = [\mathbf{a}_N(\omega_1)\ \mathbf{a}_N(\omega_2)\ \cdots\ \mathbf{a}_N(\omega_D)]$ is the array manifold matrix, where

$$\mathbf{a}_N(\omega) = [1\ e^{j\omega}\ e^{j2\omega}\ \cdots\ e^{j(N-1)\omega}]^T \tag{3.2}$$

and $\omega_i = \pi \sin\theta_i$. We assume $\mathrm{E}[\mathbf{c}] = \mathbf{0}$, $\mathrm{E}[|c_i|^2] = p_i$, $\mathrm{E}[\mathbf{e}] = \mathbf{0}$, $\mathrm{E}[\mathbf{e}\mathbf{e}^H] = \sigma_{\mathrm{e}}^2\mathbf{I}$, and $\mathrm{E}[\mathbf{c}\mathbf{e}^H] = \mathbf{0}$.

In CBS, the ULA output $x(n), 0 \le n \le N-1$ is convolved with an FIR filter $H(z) = \sum_{n=0}^{L-1} h(n)z^{-n}$ to obtain the output $y(n)$, where $L < N$. Then the steady-state samples are collected in a vector

$$\begin{aligned}\mathbf{y} &\triangleq [y(L-1)\ y(L)\ \cdots\ y(N-1)]^T = \mathbf{H}\mathbf{x} \\ &= \mathbf{A}_L\mathbf{d} + \mathbf{H}\mathbf{e},\end{aligned} \tag{3.3}$$

where

$$\mathbf{H} = \begin{bmatrix} h(L-1) & \cdots & h(0) & 0 & \cdots & 0 \\ 0 & h(L-1) & \cdots & h(0) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & h(L-1) & \cdots & h(0) \end{bmatrix} \tag{3.4}$$

is a $(N-L+1) \times N$ banded Toeplitz matrix,

$$\mathbf{A}_L = [\mathbf{a}_{N-L+1}(\omega_1)\ \cdots\ \mathbf{a}_{N-L+1}(\omega_D)], \tag{3.5}$$

and $\mathbf{d}$ has entries

$$[\mathbf{d}]_i = c_i e^{j(L-1)\omega_i} H(e^{j\omega_i}). \tag{3.6}$$

To further reduce computational complexity, we decimate $\mathbf{y}$ before computing covariance and subspace methods. In the following, for easy reference, we first review CBS with uniform decimation, which is first presented in Sec. 2.2. Then we will propose a new form of CBS used with nonuniform decimation. With the review of uniform CBS, it will be easier to compare the two decimation schemes.

3.2.1 CBS With Uniform Decimation

Note from (3.6) that the source amplitudes c_i are filtered by the frequency response $H(e^{j\omega_i})$. We assume signals in the filter stopband are well attenuated so that $\mathbf{y}$

contains only in-band DOAs, i.e., those DOAs appearing in the passband of $H(e^{j\omega})$. Without loss of generality, assume $\omega_1, \ldots, \omega_{D_0}$ are in-band DOAs. Then,

$$\mathbf{y} \approx \mathbf{A}_{L,0}\mathbf{d}_{\mathrm{IB}} + \mathbf{He}, \tag{3.7}$$

where $\mathbf{A}_{L,0}$ has the first D_0 columns of $\mathbf{A}_L$, and $\mathbf{d}_{\mathrm{IB}}$ has the first D_0 entries of $\mathbf{d}$. Due to this filtering effect, we can decimate $\mathbf{y}$ without causing ambiguity. Let

$$y_{\mathrm{s}}(n) = y(n + L - 1) \tag{3.8}$$

denote the steady-state filter output so that

$$\mathbf{y} = [y_{\mathrm{s}}(0)\ y_{\mathrm{s}}(1)\ \cdots\ y_{\mathrm{s}}(N - L)]^T. \tag{3.9}$$

Then in particular, if $H(e^{j\omega})$ has passband width $2\pi/M$ for some integer M, we can uniformly decimate $y_s(n)$ by M and obtain

$$v_l(n) = y_s(Mn + l) \tag{3.10}$$

for $l = 0, 1, \ldots, M - 1$. These are the *M polyphase components* of $y_s(n)$ [66]. For easier presentation later, we can write in the equivalent vector form

$$\mathbf{v}_l = \mathbf{D}_l\mathbf{y} = \mathbf{A}_{\mathrm{dec}}\mathbf{d}_l + \mathbf{D}_l\mathbf{He} \approx \mathbf{A}_{\mathrm{dec},0}\mathbf{d}_{l,\mathrm{IB}} + \mathbf{D}_l\mathbf{He}, \tag{3.11}$$

where

$$\mathbf{D}_l = [\boldsymbol{\delta}_l^{(N-L+1)}\ \boldsymbol{\delta}_{l+M}^{(N-L+1)}\ \ldots\ \boldsymbol{\delta}_{l+(\bar{N}-1)M}^{(N-L+1)}]^T \tag{3.12}$$

is a decimation matrix, $\mathbf{A}_{\mathrm{dec}} = [\mathbf{a}_{\bar{N}}(M\omega_1)\ \cdots\ \mathbf{a}_{\bar{N}}(M\omega_D)]$,

$$\bar{N} = (N - L + 1)/M \tag{3.13}$$

(assumed an integer for simplicity), $\mathbf{d}_l$ has entries $[\mathbf{d}_l]_i = c_i e^{j(L-1+l)\omega_i} H(e^{j\omega_i})$, $\mathbf{A}_{\mathrm{dec},0}$ has the first D_0 columns of $\mathbf{A}_{\mathrm{dec}}$, and $\mathbf{d}_{l,\mathrm{IB}}$ has the first D_0 entries of $\mathbf{d}_l$. According to (3.11), each $\mathbf{v}_l$ resembles an output of a virtual $\bar{N}$-sensor ULA due to DOAs $M\omega_i$. This ULA is called virtual because it is not a physical array. We will estimate only the D_0 passband DOAs based on $\mathbf{v}_l$. We assume $D_0 < \bar{N}$ for MUSIC to identify DOAs without ambiguity [52]. Then we can estimate DOAs based on the covariance

$$\mathbf{R}_{v_l} = \mathrm{E}[\mathbf{v}_l\mathbf{v}_l^H] = \mathbf{A}_{\mathrm{dec}}\mathbf{R}_{d_l}\mathbf{A}_{\mathrm{dec}}^H + \sigma_{\mathrm{e}}^2\mathbf{G}_{\mathrm{dec}} \tag{3.14}$$

for a particular l, say $l = 0$. Here $\mathbf{G}_{\text{dec}} \triangleq \mathbf{D}_l\mathbf{H}\mathbf{H}^H\mathbf{D}_l$ is independent of l, and $\mathbf{R}_{d_l} = \mathrm{E}[\mathbf{d}_l\mathbf{d}_l^H]$. As in Sec. 2.2.4, one can show that $\mathbf{G}_{\text{dec}}$ is Hermitian Toeplitz with $[\mathbf{G}_{\text{dec}}]_{m,l} = g(M(m-l))$, where

$$g(k) = \sum_n h(n)h^*(n-k) \tag{3.15}$$

is the deterministic autocorrelation of $h(n)$. The noise term can be whitened if the CBS filter $H(z)$ is chosen as a spectral factor of a Nyquist(M) filter, which makes $g(Mk) = \delta(k)$ for all integers k and thus $\mathbf{G}_{\text{dec}} = \mathbf{I}$. When there is no constraint on the number of RF chains or when all processing is done in the digital domain, we can obtain all $\mathbf{v}_l$ and estimate DOAs based on

$$\mathbf{R}_{\text{ave}} = \frac{1}{M}\sum_{l=0}^{M-1}\mathbf{R}_{v_l} = \mathbf{A}_{\text{dec}}\breve{\mathbf{R}}_d\mathbf{A}_{\text{dec}}^H + \sigma_{\text{e}}^2\mathbf{I}, \tag{3.16}$$

where $\breve{\mathbf{R}}_d$ is $\mathbf{R}_{d_l}$ averaged over l. Note that $\mathbf{A}_{\text{dec}}$ has columns $\mathbf{a}_{\bar{N}}(M\omega_i)$, so initially we can only identify $M\omega_i$ mod 2π, causing ambiguities. However, since ω_i are known to be in the passband of $H(e^{j\omega})$ which has width $2\pi/M$, the ambiguities can be resolved. In practice, we estimate the DOAs based on a finite-snapshot average

$$\widehat{\mathbf{R}}_{\text{ave}} = \frac{1}{KM}\sum_{k=1}^{K}\sum_{l=0}^{M-1}\mathbf{v}_l[k]\mathbf{v}_l^H[k], \tag{3.17}$$

where k is the snapshot index. Then we can estimate DOAs using root-MUSIC [26] or ESPRIT [27] with computational complexity $O(\bar{N}^3) \ll O(N^3)$.

As explained in Fig. 2.3, we can process the array output $x(n)$ with an entire filter bank $H^{(i)}(e^{j\omega}), 0 \le i \le M-1$ to cover the full DOA range $-\pi \le \omega < \pi$. This allows us to estimate all the DOAs if they are not known a priori to be located in a specific sector. This method of filter banks can also be used for CBS with nonuniform decimation proposed in Sec. 3.2.2.

3.2.2 CBS With Nonuniform Decimation

To reduce the required number of RF chains, which equals the number of decimated output samples, and thus lower hardware complexity, we can consider nonuniform decimation. That is, instead of uniform decimation as in (3.10), we nonuniformly decimate the steady-state filter output to obtain

$$v(i) = y_s(n_i) \tag{3.18}$$

where those n_i's are nonuniformly spaced. Again, for easier presentation later, we write in the equivalent vector form

$$\mathbf{v} = \mathbf{D}\mathbf{y} = \mathbf{D}\mathbf{H}\mathbf{x}, \tag{3.19}$$

where

$$\mathbf{D} = [\boldsymbol{\delta}_{n_0}^{(N-L+1)} \ \boldsymbol{\delta}_{n_1}^{(N-L+1)} \ \ldots \ \boldsymbol{\delta}_{n_{\bar{N}-1}}^{(N-L+1)}]^T. \tag{3.20}$$

In particular, we choose the sampling grid

$$n_i = \tilde{n}_i M, \ 0 \le i \le \bar{N} - 1, \tag{3.21}$$

where M is some positive integer, and the integer set $\mathcal{N} = \{\tilde{n}_0, \ldots, \tilde{n}_{\bar{N}-1}\}$ corresponds to the sensor locations of some standard sparse array like a nested array [32], coprime array [40], or minimum redundancy array (MRA) [3]. Without loss of generality, let $\tilde{n}_0 = 0$. The locations of the decimated output samples correspond to a *dilated* sparse array. For example, we consider a ULA with $N = 24$ sensors and CBS filter length $L = 6$, and after decimation, we retain $\bar{N} = 5$ samples corresponding to a 5-sensor MRA $\mathcal{N} = \{\tilde{n}_0, \ldots, \tilde{n}_4\} = \{0, 1, 4, 7, 9\}$ dilated by $M = 2$. Thus, the outputs of the CBS filter that are retained correspond to a dilated MRA $\mathcal{N}_M = \{M\tilde{n}_0, \ldots, M\tilde{n}_4\} = \{0, 2, 8, 14, 18\}$, with the use of the 5×19 decimation matrix

$$\mathbf{D} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

See Fig. 3.1(a)-(c) for demonstration. As we shall see, the use of M allows us to obtain a virtual sparse array dilated by M times, so its difference coarray [79] will contain a sparse ULA with sensor spacing M. *Normally, such a sparse ULA creates DOA ambiguity, but it is not a problem for us because* $\mathbf{y}$ *is represented only by passband DOAs.* The benefit of introducing M is that larger array aperture can yield smaller estimation errors and higher resolution, which will be verified by simulations in Sec. 3.6.

With the proposed nonuniform decimation, we can derive that

$$\mathbf{v} = \mathbf{A}_s\mathbf{d} + \mathbf{D}\mathbf{H}\mathbf{e}, \tag{3.22}$$

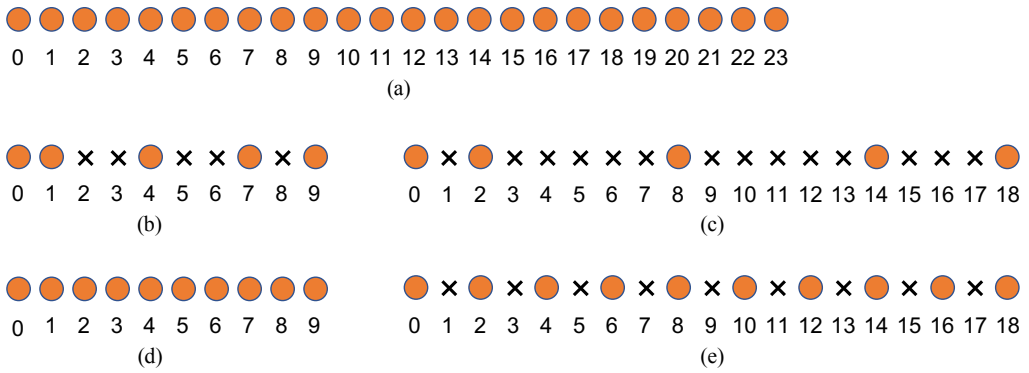


Figure 3.1: An example of nonuniform decimation for hybrid CBS. Filled circles represent sensors, and crosses represent empty space. (a) The physical ULA with $N = 24$ sensors. (b) A 5-sensor MRA $\mathcal{N} = \{\tilde{n}_0, \ldots, \tilde{n}_4\} = \{0, 1, 4, 7, 9\}$. (c) The dilated 5-sensor MRA $\mathcal{N}_M = \{M\tilde{n}_0, \ldots, M\tilde{n}_{\bar{N}-1}\} = \{0, 2, 8, 14, 18\}$, where $M = 2$. (d) The difference coarray $\mathcal{C}$ of the 5-sensor MRA $\mathcal{N}$. (Only the nonnegative portion is shown because coarrays are symmetric.) Note that $\mathcal{C}$ is a ULA here. (e) The difference coarray $\mathcal{C}_M$ of the dilated 5-sensor MRA $\mathcal{N}_M$. Note that $\mathcal{C}_M$ is a sparse ULA of sensor spacing $M = 2$, and it is equivalent to $\mathcal{C}$ dilated by M.

where

$$\mathbf{A}_{\mathrm{s}} = [\mathbf{a}_{\mathrm{s}}(M\omega_1) \ \cdots \ \mathbf{a}_{\mathrm{s}}(M\omega_D)] \tag{3.23}$$

with $\mathbf{a}_{\mathrm{s}}(\omega) = [1 \ e^{j\omega\tilde{n}_1} \ e^{j\omega\tilde{n}_2} \ \cdots \ e^{j\omega\tilde{n}_{\bar{N}-1}}]^T$, and $\mathbf{d}$ is defined in (3.6). In other words, $\mathbf{v}$ corresponds to the output of a sparse array $\mathcal{N}$ with sensor locations $\{\tilde{n}_0, \ldots, \tilde{n}_{\bar{N}-1}\}$ due to the filtered sources with amplitudes $[\mathbf{d}]_i$ and DOAs $M\omega_i$. Equivalently, $\mathbf{v}$ can be viewed as the output of a dilated sparse array $\mathcal{N}_M = \{M\tilde{n}_0, \ldots, M\tilde{n}_{\bar{N}-1}\}$ due to the DOAs ω_i. We use virtual sparse array and virtual dilated sparse array to refer to the arrays $\mathcal{N}$ and $\mathcal{N}_M$ since they are not physical arrays. *Then, the difference coarray method for sparse arrays [79] can be applied to estimate more than $\bar{N}$ sources, in fact $O(\bar{N}^2)$ sources in principle.* This is in contrast to the case of uniform decimation, where the number of identifiable sources is at most $\bar{N} - 1$, where $\bar{N}$ is the number of RF chains.

The difference coarray $\mathcal{C}$ of the sparse array $\mathcal{N} = \{\tilde{n}_i\}$ is the set of all possible differences $\tilde{n}_m - \tilde{n}_l$ between sensor locations. As shown in the example in Fig. 3.1(d), the coarray of a well-designed sparse array like MRA typically contains a large ULA segment. Then in coarray method [79], we use the second-order statistics of the sparse array output to create a virtual signal on the coarray. Then better performance can be obtained by estimating DOAs using the signal on the large ULA segment of the coarray. In the proposed method, we do not have a

physical nonuniform sparse array. Instead, we create a virtual sparse array output $\mathbf{v}$ by nonuniformly decimating the filter output $\mathbf{y}$. Then we apply coarray method to the nonuniformly decimated output $\mathbf{v}$ as explained in the following.

The covariance matrix of the nonuniformly decimated CBS output $\mathbf{v}$ is

$$\mathbf{R_{vv}} = \mathbf{A}_\mathrm{s}\mathbf{R_{dd}}\mathbf{A}_\mathrm{s}^H + \sigma_\mathrm{e}^2\mathbf{D}\mathbf{H}\mathbf{H}^H\mathbf{D}^H. \tag{3.24}$$

Here, we assume the sources are uncorrelated so that $\mathbf{R_{dd}} = \mathrm{E}[\mathbf{d}\mathbf{d}^H]$ is diagonal. (If sources are correlated, we can use the method for decorrelating sources in Sec. 3.4.) Hence, we have the (m,l)-entry

$$[\mathbf{R_{vv}}]_{m,l} = \sum_{i=1}^{D} p_i' e^{jM\omega_i(\tilde{n}_m - \tilde{n}_l)} + \sigma_\mathrm{e}^2 g(M(\tilde{n}_m - \tilde{n}_l)), \tag{3.25}$$

where $p_i' = p_i|H(e^{j\omega_i})|^2$, and $g(k)$ is as in (3.15). Again, we let $H(z)$ be a spectral factor of a Nyquist(M) filter so that $g(Mk) = \delta(k)$. Now (3.25) depends only on the difference $\tilde{n}_m - \tilde{n}_l$ between the virtual sensor locations, so we can write

$$\begin{aligned} R_M(\tilde{n}_m - \tilde{n}_l) &\triangleq [\mathbf{R_{vv}}]_{m,l} && (3.26)\\ &= \sum_{i=1}^{D} p_i' e^{jM\omega_i(\tilde{n}_m - \tilde{n}_l)} + \sigma_\mathrm{e}^2\delta(\tilde{n}_m - \tilde{n}_l) && (3.27)\\ &\approx \sum_{i=1}^{D_0} p_i' e^{jM\omega_i(\tilde{n}_m - \tilde{n}_l)} + \sigma_\mathrm{e}^2\delta(\tilde{n}_m - \tilde{n}_l), && (3.28) \end{aligned}$$

where we again assume that only the first D_0 sources fall in the passband of $H(e^{j\omega})$. Then, by estimating $R_M(\tilde{n}_m - \tilde{n}_l) = \mathrm{E}[[\mathbf{v}]_m[\mathbf{v}]_l^*]$ using snapshot averages over all $\tilde{n}_m, \tilde{n}_l$ that produce identical difference $k = \tilde{n}_m - \tilde{n}_l$, we can estimate

$$R_M(k) \approx \sum_{i=1}^{D_0} p_i' e^{jM\omega_i k} + \sigma_\mathrm{e}^2\delta(k), \tag{3.29}$$

for all $k \in C$, where C is the difference coarray of the array $\mathcal{N} = \{\tilde{n}_i\}$. Note that $R_M(k)$ resembles a single snapshot of the output of the array C due to the D_0 DOAs $M\omega_i$. Equivalently, $R_M(k)$ can be viewed as a single-snapshot output of the array C_M due to the D_0 DOAs ω_i, where C_M is obtained by dilating C by M (i.e., $k \in C$ if and only if $Mk \in C_M$), as shown in Fig. 3.1(e). In fact, C_M is the difference coarray of the virtual dilated sparse array $\mathcal{N}_M$. To estimate the DOAs using subspace methods, we apply the method in [79], which is an improved way of doing spatial smoothing with lower computational complexity. The difference coarray is symmetric in the

sense that if $k \in C$, then $-k \in C$. Let U be the largest integer such that the uniform region $-(U-1) \le k \le U-1$ is in C. Then $-(U-1) \le k \le U-1$ is called the central ULA segment of the difference coarray. Correspondingly, we have a central sparse ULA segment $\{Mk \mid -(U-1) \le k \le U-1\}$ in the dilated coarray C_M. This M times larger aperture can lead to higher DOA resolution and smaller estimation errors, as we shall see in Sec. 3.6. For many sparse arrays, including nested arrays, coprime arrays, and MRAs, $U = O(\bar{N}^2)$, where $\bar{N}$ is the number of sensors of the sparse array (i.e., number of RF chains). We then define a Hermitian Toeplitz matrix

$$\mathbf{R} = \begin{bmatrix} R_M(0) & R_M^*(1) & \cdots & R_M^*(U-1) \\ R_M(1) & R_M(0) & \cdots & R_M^*(U-2) \\ \vdots & \vdots & \ddots & \vdots \\ R_M(U-1) & R_M(U-2) & \cdots & R_M(0) \end{bmatrix}. \tag{3.30}$$

All elements $R_M(k)$ of the matrix can be estimated by averaging $[\mathbf{v}]_m[\mathbf{v}]_l^*$ over snapshots, and over all $\tilde{n}_m, \tilde{n}_l$ that produce identical difference $k = \tilde{n}_m - \tilde{n}_l$. Let $\widetilde{\mathbf{R}}$ denote the estimated matrix. Then, the DOAs can be estimated using subspace methods if we order the eigenvalues of $\widetilde{\mathbf{R}}$ in terms of their absolute values and find the signal and noise subspaces accordingly [79]. In theory, as many as $U - 1 = O(\bar{N}^2)$ DOAs can be identified using MUSIC [52] or ESPRIT [27]. Compared to the uniform scheme (where decimated output corresponds to a sparse ULA with spacing M) in Sec. 3.2.1, the nonuniform scheme can achieve the same effective array aperture based on the dilated coarray C_M. Thus, nonuniform scheme requires fewer RF chains and thus lower hardware complexity to achieve the same estimation performance. Or, nonuniform scheme gives larger effective array aperture and better estimation performance than uniform scheme given a fixed number of RF chains.

Similar to the case of uniform decimation in Sec. 3.2.1, because of the presence of $M\omega_i$ in (3.29), initially we can only identify $M\omega_i$ mod 2π, causing ambiguities. Yet we know that after filtering, we only have ω_i in the passband of $H(e^{j\omega})$ which has width $2\pi/M$, so the ambiguities can be resolved. This shows that the filtering effect of CBS is crucial for the proposed nonuniform decimation scheme to work. Because of the presence of $M > 1$ in $n_i = \tilde{n}_i M$ and the sparsity of the virtual sparse array $\mathcal{N} = \{\tilde{n}_0, \ldots, \tilde{n}_{\bar{N}-1}\}$, we can make $\bar{N} \ll N$. In other words, it is practical to use a small number $\bar{N}$ of RF chains even for a large number N of antennas. Moreover, we can still estimate $O(\bar{N}^2)$ DOAs.

3.3 Hybrid Analog and Digital Implementation of CBS

As described in the introduction (Sec. 3.1), to avoid large hardware cost, a hybrid of analog and digital processing is used for mmWave arrays. In particular, we implement the beamspace matrix of CBS in the analog domain, and the resulting lower-dimensional output is passed through a smaller number $\bar{N}$ of RF chains. Since the analog combiner is realized by a phase shifter network, each entry of the beamspace matrix should have unit modulus. However, a standard digital filter typically does not have constant-modulus coefficients. To overcome this issue, we use an important result that any complex vector $\mathbf{v}$ can be expressed as a linear combination of two vectors with unit-modulus entries:

Lemma 3.1 ([48, 96]) *For any complex vector $\mathbf{v} \in \mathbb{C}^n$, there exist complex numbers a_1 and a_2 and complex vectors $\mathbf{v}_1 \in \mathbb{C}^n$ and $\mathbf{v}_2 \in \mathbb{C}^n$ whose entries $v_{1,i}$ and $v_{2,i}$ satisfy $|v_{1,i}| = |v_{2,i}| = 1$ for all i such that*

$$\mathbf{v} = a_1\mathbf{v}_1 + a_2\mathbf{v}_2. \tag{3.31}$$

In particular, to satisfy (3.31), we can choose

$$a_1 = a_2 = \|\mathbf{v}\|_\infty/2, \tag{3.32}$$

$$v_{1,i} = e^{j(\arg(v_i)+\cos^{-1}(|v_i|/\|\mathbf{v}\|_\infty))}, \tag{3.33}$$

$$v_{2,i} = e^{j(\arg(v_i)-\cos^{-1}(|v_i|/\|\mathbf{v}\|_\infty))} \tag{3.34}$$

for all i, where v_i is the ith entry of $\mathbf{v}$.

Proof: It can be directly verified that $\mathbf{v} = a_1\mathbf{v}_1 + a_2\mathbf{v}_2$ given (3.32), (3.33), and (3.34). ■

Remark: It is not necessary to choose $a_1 = a_2$ to satisfy (3.31). However, it is advantageous to do so in terms of hardware implementation, as we shall explain. The merit of this lemma is that, just two vectors $\mathbf{v}_1, \mathbf{v}_2$ are sufficient. There is no need to consider the linear combination of more than two vectors.

We first illustrate the implementation of CBS with nonuniform decimation, as shown in Fig. 3.2. Consider an arbitrary CBS filter $h(n)$, $0 \le n \le L-1$ and its vector form $\mathbf{h} = [h(L-1)\ h(L-2)\ \cdots\ h(0)]^T$. According to Lemma 3.1, there exist $a_1, a_2 \in \mathbb{C}$ and $\mathbf{h}_1, \mathbf{h}_2 \in \mathbb{C}^L$ whose entries are unit-modulus such that

$$\mathbf{h} = a_1\mathbf{h}_1 + a_2\mathbf{h}_2. \tag{3.35}$$

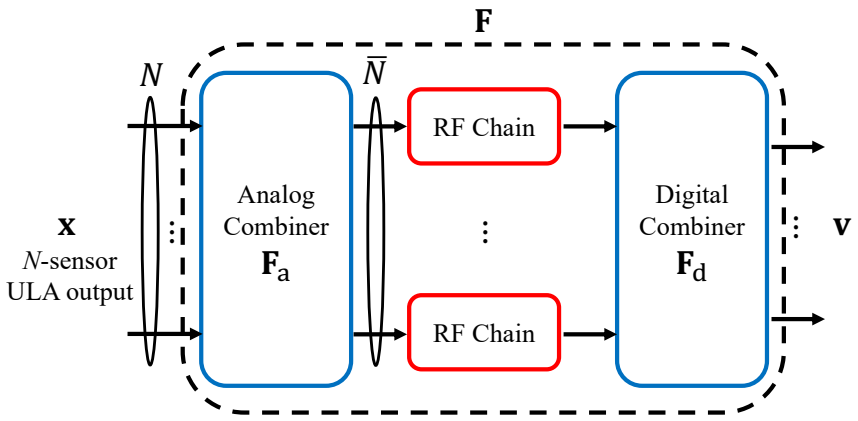


Figure 3.2: System model of hybrid CBS for DOA estimation based on passive arrays. The analog combiner can only have unit-magnitude multipliers.

Hence, for the operation $\mathbf{v} = \mathbf{DHx}$ in (3.19), the overall combiner $\mathbf{F}$ can be written as

$$\mathbf{F} = \mathbf{DH} = \widetilde{\mathbf{H}}\overline{\mathbf{H}}, \tag{3.36}$$

where

$$\widetilde{\mathbf{H}} = \begin{bmatrix} a_1 & a_2 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & a_1 & a_2 & \cdots & 0 & 0 \\ \vdots & \vdots & & & \ddots & & \\ 0 & 0 & 0 & 0 & \cdots & a_1 & a_2 \end{bmatrix} \in \mathbb{C}^{\bar{N}\times 2\bar{N}}, \tag{3.37}$$

$$\overline{\mathbf{H}} = [\overline{\mathbf{h}}_1^{(0)} \ \overline{\mathbf{h}}_2^{(0)} \ \overline{\mathbf{h}}_1^{(n_1)} \ \overline{\mathbf{h}}_2^{(n_1)} \ \cdots \ \overline{\mathbf{h}}_1^{(n_{\bar{N}-1})} \ \overline{\mathbf{h}}_2^{(n_{\bar{N}-1})}]^T \tag{3.38}$$

with $\overline{\mathbf{h}}_m^{(k)} = [\mathbf{0}_{1\times k} \ \mathbf{h}_m^T \ \mathbf{0}_{1\times(N-L-k)}]^T$ for $m = 1, 2$ and nonnegative integers k. Since $\mathbf{h}_1$ and $\mathbf{h}_2$ have unit modulus entries (thanks to Lemma 3.1), we can implement $\overline{\mathbf{H}}$ with $2\bar{N}L$ phase shifting operations and $2\bar{N}(L-1)$ additions. In general, $a_1 \neq a_2$, and they do not have the same modulus. Thus, we cannot implement $\widetilde{\mathbf{H}}$ with phase shifters. Instead, we can do analog-to-digital conversion on $\overline{\mathbf{H}}\mathbf{x}$ and implement $\widetilde{\mathbf{H}}$ in the digital domain so that the analog combiner $\mathbf{F}_a = \overline{\mathbf{H}}$ and the digital combiner $\mathbf{F}_d = \widetilde{\mathbf{H}}$. Such implementation requires $2\bar{N}$ RF chains. Alternatively, if we choose $a_1 = a_2$ as in (3.32), then

$$\widetilde{\mathbf{H}} = a_1\mathbf{B}, \tag{3.39}$$

where

$$\mathbf{B} = \begin{bmatrix} 1 & 1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & & & \ddots & & \\ 0 & 0 & 0 & 0 & \cdots & 1 & 1 \end{bmatrix} \in \mathbb{C}^{\bar{N} \times 2\bar{N}} \tag{3.40}$$

can be implemented with $\bar{N}$ additions in the analog domain. The constant scalar a_1 can be implemented in the digital domain. Or, it may be even ignored because the performance of most DOA estimation methods such as MUSIC [52] or ESPRIT [27] is not affected by scaling the array output vector by a constant. In summary, the analog combiner is designed as

$$\mathbf{F}_{\mathrm{a}} = \mathbf{B}\overline{\mathbf{H}}, \tag{3.41}$$

realized with $2\bar{N}L$ phase shifting operations, and we need only $\bar{N}$ RF chains to process the combiner output. This is half of the number of RF chains required for $|a_1| \neq |a_2|$. The digital combiner is designed as $\mathbf{F}_{\mathrm{d}} = a_1\mathbf{I}$ or $\mathbf{F}_{\mathrm{d}} = \mathbf{I}$ (i.e., no digital combiner). All of the remaining steps for coarray method and subspace algorithms described in Sec. 3.2.2 can then be realized since any digital processing can be done after we acquire the vector $\mathbf{v}$ in the digital domain.

For CBS with uniform decimation (3.11), we can implement $\mathbf{v}_l = \mathbf{D}_l\mathbf{H}\mathbf{x}$ in the same way as we do for the nonuniform case $\mathbf{v} = \mathbf{D}\mathbf{H}\mathbf{x}$. In fact, the former can be viewed as a special case of the latter if we let the virtual sensor locations $n_i = iM$. Thus, we need $\bar{N} = (N - L + 1)/M$ RF chains. In Sec. 3.2.1, it is shown that we can average over $\mathbf{R}_{v_l}$ from all polyphase components $\mathbf{v}_l$ to get a better covariance estimate when we work purely in the digital domain. However, in hybrid CBS, if we have only $\bar{N}$ RF chains, then we can do analog-to-digital conversion for only one polyphase component, say $\mathbf{v}_0$, at each time instant. Hence, we will estimate DOAs based on the covariance $\mathbf{R}_{v_0}$. Although the other polyphase components are discarded, the resulting estimation performance is still close to the ideal benchmark, element-space (i.e., no beamspace processing) without RF chain constraint, as demonstrated in Sec. 3.6.

3.4 Filters With Delays for Decorrelating Sources in CBS

In this section, we propose a new method for decorrelating sources based on CBS filters with delays. In all derivations so far, the banded Toeplitz matrix $\mathbf{H}$ is assumed fixed for all K snapshots of the array output $\mathbf{x}[k]$, i.e., $\mathbf{y}[k] = \mathbf{H}\mathbf{x}[k]$ for $k =$

$1, \ldots, K$. However, we can also use different banded Toeplitz matrices for different snapshots. That is,

$$\mathbf{y}[k] = \mathbf{H}_k \mathbf{x}[k], \tag{3.42}$$

where $\mathbf{H}_k$ is the $(N - L + 1) \times N$ banded Toeplitz matrix:

$$\begin{bmatrix} h_k(L-1) & \cdots & h_k(0) & 0 & \cdots & 0 \\ 0 & h_k(L-1) & \cdots & h_k(0) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & h_k(L-1) & \cdots & h_k(0) \end{bmatrix}$$

for $k = 1, \ldots, K$. In the following, we show that if we design $h_k(n)$ to be different delayed versions of a prototype filter $h(n)$, the sources c_i can to some degree be decorrelated (if they were correlated, to begin with). This technique can be used for both CBS with uniform decimation (Sec. 3.2.1) and CBS with nonuniform decimation (Sec. 3.2.2), but it is particularly important to the latter because we assume uncorrelated sources when deriving coarrays methods. Thus, we will illustrate the idea with nonuniform decimation.

We design the filter $h_k(n)$ for the kth snapshot as a delayed version of some prototype filter $h(n)$ with length $L' < L$. Let $q_k \in [0, L - L']$ denote the delay. Then the filter $H_k(z) = \sum_{n=0}^{L-1} h_k(n) z^{-n}$ is related to the prototype by

$$H_k(z) = H(z) z^{-q_k}, \tag{3.43}$$

where $H(z) = \sum_{n=0}^{L'-1} h(n) z^{-n}$. Note that each filter $h_k(n)$ only has at most L' nonzero coefficients. Now replacing $\mathbf{H}$ by $\mathbf{H}_k$ in (3.19), we have

$$\mathbf{v}[k] = \mathbf{D}\mathbf{H}_k \mathbf{x}[k] = \mathbf{A}_{\mathrm{s}} \mathbf{d}[k] + \mathbf{D}\mathbf{H}_k \mathbf{e}[k], \tag{3.44}$$

where $\mathbf{A}_{\mathrm{s}}$ is defined in (3.23), and $\mathbf{d}[k]$ has entries

$$[\mathbf{d}[k]]_i = c_i[k] e^{j(L-1)\omega_i} H_k(e^{j\omega_i}) \tag{3.45}$$

$$= c_i[k] e^{j(L-1)\omega_i} H(e^{j\omega_i}) e^{-j\omega_i q_k}. \tag{3.46}$$

Here $c_i[k]$ is the ith source amplitude in the kth snapshot. Thus,

$$\mathbf{R}_{\mathbf{vv}} = \mathrm{E}[\mathbf{v}[k] \mathbf{v}^H[k]] \tag{3.47}$$

$$= \mathbf{A}_{\mathrm{s}} \mathbf{R}_{\mathbf{dd}} \mathbf{A}_{\mathrm{s}}^H + \sigma_{\mathrm{e}}^2 \, \mathrm{E}[\mathbf{D}\mathbf{H}_k \mathbf{H}_k^H \mathbf{D}^H], \tag{3.48}$$

where $\mathbf{R_{dd}} = \mathrm{E}[\mathbf{d}[k]\mathbf{d}^H[k]]$. We can show that in (3.48), $\mathrm{E}[\mathbf{D}\mathbf{H}_k\mathbf{H}_k^H\mathbf{D}^H] = \mathbf{D}\mathbf{H}\mathbf{H}^H\mathbf{D}^H$ regardless of the delay q_k, where $\mathbf{H}$ is as defined in (3.4) with the understanding that $h(n) = 0$ for $n = L', \ldots, L-1$. Thus, the noise term can be whitened as in (3.24) if the CBS filter $H(z)$ is a spectral factor of a Nyquist(M) filter. Meanwhile,

$$[\mathbf{R_{dd}}]_{i,l} = \mathrm{E}[c_i[k]c_l^*[k]]e^{j(L-1)(\omega_i-\omega_l)}H(e^{j\omega_i})H^*(e^{j\omega_l})\,\mathrm{E}[e^{-j(\omega_i-\omega_l)q_k}]. \quad (3.49)$$

3.4.1 Case of Random Delays

Suppose q_k are independent and identically distributed (i.i.d.) with probability mass function $f(q)$. Since the discrete-time Fourier transform of $f(q)$ is

$$F(e^{j\omega}) = \sum_{q=0}^{L-L'} f(q)e^{-j\omega q} = \mathrm{E}[e^{-j\omega q_k}], \quad (3.50)$$

we have that

$$\mathrm{E}[e^{-j(\omega_i-\omega_l)q_k}] = F(e^{j(\omega_i-\omega_l)}). \quad (3.51)$$

Thus, by designing $F(e^{j\omega})$ as a lowpass filter and assuming all the DOAs ω_i are not too close to one another (so that $\omega_i - \omega_l$ is in the passband), we can make $\mathrm{E}[e^{-j(\omega_i-\omega_l)q_k}] \approx 0$ for all $i \neq l$. This further makes $\mathbf{R_{dd}}$ approximately diagonal. That is, the sources are decorrelated. This technique allows us to use the coarray method as in Sec. 3.2.2 even when the source amplitudes c_i are correlated. Standard windows with nonnegative coefficients, such as rectangular, Hann, and Hamming windows [63], can be used to design $f(q)$.

We note that the factor $H(e^{j\omega_i})H^*(e^{j\omega_l})$ is still present in (3.49), so the filtering effect of traditional CBS on the DOAs is achieved by designing $H(e^{j\omega})$ to be a standard lowpass filter. The effective filter lengths of $h(n)$ and $f(q)$ are L' and $L - L' + 1$, respectively. Hence, there is a tradeoff between the filtering effect on the DOAs and the decorrelating effect on the sources when we choose L' given a fixed L. Here is where another advantage of large arrays comes in: for large arrays (large N), we can choose a large L and design the system to have both good filtering ability and good decorrelating ability. We also note that $F(e^{j\omega})$ can be designed to have a narrower passband than $H(e^{j\omega})$. In this way, correlated passband sources (passband in terms of $H(e^{j\omega})$) can still be decorrelated. By contrast, traditional CBS in Sec. 2.2 can only handle correlation between passband and stopband sources, but suffers if two passband sources are correlated.

3.4.2 Case of Deterministic Delays

In deriving (3.49) and (3.51), we have assumed that the delays q_k are random and i.i.d. with probability mass function $f(q)$. Instead of random delays, we can also use deterministic delays. In practice, we use K snapshots to estimate $\mathbf{R_{vv}}$ and obtain

$$\widehat{\mathbf{R}}_{\mathbf{vv}} = \frac{1}{K}\sum_{k=1}^{K} \mathbf{v}[k]\mathbf{v}^H[k]. \tag{3.52}$$

We let $\hat{f}(q)$ denote the number of snapshots with delay q, divided by K, i.e.,

$$\hat{f}(q) = \frac{1}{K}\Big|\big\{1 \le k \le K \mid q_k = q\big\}\Big|. \tag{3.53}$$

The discrete-time Fourier transform of $\hat{f}(q)$ is given by

$$\widehat{F}(e^{j\omega}) = \sum_{q=0}^{L-L'} \hat{f}(q)e^{-j\omega q} = \frac{1}{K}\sum_{k=1}^{K} e^{-j\omega q_k}, \tag{3.54}$$

which can be viewed as an empirical counterpart of (3.50). Then using (3.44) and (3.52), we can obtain

$$\mathrm{E}[\widehat{\mathbf{R}}_{\mathbf{vv}}] = \mathbf{A}_{\mathrm{s}}\,\mathrm{E}[\widehat{\mathbf{R}}_{\mathbf{dd}}]\mathbf{A}_{\mathrm{s}}^H + \sigma_{\mathrm{e}}^2\mathbf{D}\mathbf{H}\mathbf{H}^H\mathbf{D}^H, \tag{3.55}$$

where

$$\mathrm{E}[\widehat{\mathbf{R}}_{\mathbf{dd}}] = \frac{1}{K}\sum_{k=1}^{K} \mathrm{E}[\mathbf{d}[k]\mathbf{d}^H[k]]. \tag{3.56}$$

In view of (3.46) and (3.54), $\mathrm{E}[\widehat{\mathbf{R}}_{\mathbf{dd}}]$ has entries

$$\begin{aligned}[\mathrm{E}[\widehat{\mathbf{R}}_{\mathbf{dd}}]]_{i,l} = \mathrm{E}[c_i[k]c_l^*[k]]e^{j(L-1)(\omega_i-\omega_l)}H(e^{j\omega_i})H^*(e^{j\omega_l}) \\ \cdot \widehat{F}(e^{j(\omega_i-\omega_l)}).\end{aligned} \tag{3.57}$$

Hence, by designing $\hat{f}(q)$ and thus $\widehat{F}(e^{j\omega})$ to approximate the desired lowpass filter $F(e^{j\omega})$, we can similarly make $\widehat{F}(e^{j(\omega_i-\omega_l)}) \approx 0$ for all $i \neq l$. This further makes $\mathrm{E}[\widehat{\mathbf{R}}_{\mathbf{dd}}]$ approximately diagonal and decorrelates the sources. There is a simple way to designing $\hat{f}(q)$ to approximate the probability mass function $f(q)$. We consider the cumulative distribution function $f_{\mathrm{c}}(q) = \sum_{i=-\infty}^{q} f(i)$. Then out of the K snapshots, we use

$$r(f_{\mathrm{c}}(q)K)) - r(f_{\mathrm{c}}(q-1)K) \tag{3.58}$$

snapshots with delays q, where $r(x)$ rounds x to the nearest integer. With this choice, one can verify that the total number of snapshots is indeed K. In the simulations

later, we will compare the performance of random delays with deterministic delays designed in this way. It is found that the two types of delays result in similar DOA estimation errors. Hence, deterministic delays are more recommended since they are easier to implement than random delays in the analog combiner.

3.5 Cramér–Rao Bound Based on Beamspace Outputs

In this section, we derive the Cramér–Rao Bound (CRB) [43] for the DOA estimates based on the beamspace output

$$\mathbf{v} = \mathbf{W}^H \mathbf{x} \tag{3.59}$$

under the stochastic model [44, 49, 50]. Here $\mathbf{x}$ is the ULA output in (3.1), and $\mathbf{W}$ is any fixed beamspace matrix with full column rank. We assume that the sources are known to be uncorrelated a priori, and we have K snapshots.

Theorem 3.1 *The CRB for the DOAs $\boldsymbol{\omega} = [\omega_1 \ \cdots \ \omega_D]^T$ based on the beamspace output $\mathbf{v}$ in (3.59) is*

$$\mathrm{CRB}(\boldsymbol{\omega}) = \frac{1}{K}\left(\mathbf{G}^H \boldsymbol{\Pi}_{\boldsymbol{\Delta}}^{\perp} \mathbf{G}\right)^{-1}, \tag{3.60}$$

where $\boldsymbol{\Pi}_{\boldsymbol{\Delta}}^{\perp} = \mathbf{I} - \boldsymbol{\Delta}(\boldsymbol{\Delta}^H\boldsymbol{\Delta})^{-1}\boldsymbol{\Delta}^H$,

$$\mathbf{G} = \left(\mathbf{V}^T \otimes \mathbf{V}\right)^{1/2} \left(\dot{\mathbf{A}}^* \odot (\mathbf{AP}) + (\mathbf{AP})^* \odot \dot{\mathbf{A}}\right), \tag{3.61}$$

$$\boldsymbol{\Delta} = \left(\mathbf{V}^T \otimes \mathbf{V}\right)^{1/2} \left[\mathbf{A}^* \odot \mathbf{A} \quad \mathrm{vec}(\mathbf{I})\right], \tag{3.62}$$

$$\mathbf{V} = \mathbf{W}(\mathbf{W}^H \mathbf{R}_{\mathbf{xx}} \mathbf{W})^{-1} \mathbf{W}^H. \tag{3.63}$$

Here $\dot{\mathbf{A}} = [\dot{\mathbf{a}}_N(\omega_1) \cdots \dot{\mathbf{a}}_N(\omega_D)]$ with $\dot{\mathbf{a}}_N(\omega) = \frac{\mathrm{d}}{\mathrm{d}\omega}\mathbf{a}_N(\omega)$ and $\mathbf{P} = \mathrm{diag}(p_1, \ldots, p_D)$. Also, we assume $\mathbf{R}_{\mathbf{xx}} = \mathrm{E}[\mathbf{x}\mathbf{x}^H]$ is positive definite so that $\mathbf{V}$ exists, and that $(\mathbf{V}^T \otimes \mathbf{V})$ is positive semidefinite. Then $(\mathbf{V}^T \otimes \mathbf{V})^{1/2}$ denotes its positive semidefinite square root.

Proof: The parameter vector for the uncorrelated model is

$$\boldsymbol{\alpha} = \left[[\omega_i]_{i=1}^D \ \ [p_i]_{i=1}^D \ \ \sigma_{\mathrm{e}}^2\right]^T = \left[\boldsymbol{\omega}^T \ \ [p_i]_{i=1}^D \ \ \sigma_{\mathrm{e}}^2\right]^T. \tag{3.64}$$

The Fisher information matrix $\boldsymbol{\mathcal{I}}(\boldsymbol{\alpha})$ for the model can be derived as [49]

$$\boldsymbol{\mathcal{I}}(\boldsymbol{\alpha}) = K \left[\frac{\partial \mathbf{r}_{\mathbf{vv}}}{\partial \boldsymbol{\alpha}}\right]^H \left(\mathbf{R}_{\mathbf{vv}}^T \otimes \mathbf{R}_{\mathbf{vv}}\right)^{-1} \frac{\partial \mathbf{r}_{\mathbf{vv}}}{\partial \boldsymbol{\alpha}}, \tag{3.65}$$

where $\mathbf{r_{vv}} = \text{vec}(\mathbf{R_{vv}})$, $\mathbf{R_{vv}} = \text{E}[\mathbf{v}\mathbf{v}^H] = \mathbf{W}^H\mathbf{R_{xx}}\mathbf{W}$, and $\frac{\partial \mathbf{r_{vv}}}{\partial \boldsymbol{\alpha}}$ is the Jacobian matrix with entries

$$\left[\frac{\partial \mathbf{r_{vv}}}{\partial \boldsymbol{\alpha}}\right]_{i,k} = \frac{\partial [\mathbf{r_{vv}}]_i}{\partial [\boldsymbol{\alpha}]_k}. \tag{3.66}$$

Using the identities $\text{vec}(\mathbf{ABC}) = (\mathbf{C}^T \otimes \mathbf{A})\text{vec}(\mathbf{B})$, $(\mathbf{AC} \otimes \mathbf{BD}) = (\mathbf{A} \otimes \mathbf{B})(\mathbf{C} \otimes \mathbf{D})$, and $(\mathbf{E} \otimes \mathbf{F})^{-1} = (\mathbf{E}^{-1} \otimes \mathbf{F}^{-1})$ for any matrices $\mathbf{A}, \mathbf{B}, \ldots, \mathbf{F}$ with proper dimensions (and $\mathbf{E}$, $\mathbf{F}$ invertible), we can derive that

$$\boldsymbol{\mathcal{I}}(\boldsymbol{\alpha}) = K \left[\frac{\partial \mathbf{r_{xx}}}{\partial \boldsymbol{\alpha}}\right]^H \left(\mathbf{V}^T \otimes \mathbf{V}\right) \frac{\partial \mathbf{r_{xx}}}{\partial \boldsymbol{\alpha}}, \tag{3.67}$$

where $\mathbf{r_{xx}} = \text{vec}(\mathbf{R_{xx}})$, and $\mathbf{V}$ is defined in (3.63). The detailed expression of the Jacobian matrix $\frac{\partial \mathbf{r_{xx}}}{\partial \boldsymbol{\alpha}}$ can be obtained by directly computing the derivatives in its definition. We note that the CRB for $\boldsymbol{\alpha}$ is the inverse of $\boldsymbol{\mathcal{I}}(\boldsymbol{\alpha})$, but we are interested only in the CRB for $\boldsymbol{\omega}$. Thus, following derivations similar to those in [49, 50], we can use block-wise matrix inversion to obtain (3.60), (3.61), and (3.62). ■

The uniformly and nonuniformly decimated CBS outputs are special cases of the model (3.59). For example, for the nonuniformly decimated CBS output (3.19), we let $\mathbf{W} = \mathbf{H}^H\mathbf{D}^H$. Hence, we can use Theorem 3.1 to compare the MSE performance of the proposed algorithm with the CRB via numerical examples in the following.

3.6 Simulations for Hybrid CBS

In this section, we study the performance of the proposed hybrid CBS for passive arrays via numerical examples. In all the examples, we assume that the number of DOAs is known for element-space methods, and that the number of in-band DOAs is known for CBS. To compare with CBS using a filter $H(z)$, for element-space, we just consider DOA estimates in the passband of $H(z)$ and ignore those in the stopband. Whenever we mention root mean square errors (RMSEs) in detected in-band source angles, we refer to averaging square errors measured in ω over all in-band DOAs and over those Monte Carlo runs that obtain the correct number of in-band DOAs (as element-space can get a wrong number of in-band DOAs). CBS filters are designed to be lowpass filters with passband edge $(1-\beta)\pi/M$ and stopband edge $(1+\beta)\pi/M$, where M is the decimation ratio, and β is the roll-off factor which determines the transition bandwidth. If not mentioned otherwise, all sources c_i have the same power, and the SNR is defined as the ratio of the power of an in-band source to the noise power σ_e^2.

Example 1 (Hybrid CBS with uniform decimation): We consider a ULA with $N = 96$ sensors receiving 2 passband sources at angles $\theta = -2°, 2°$, and 2 stopband

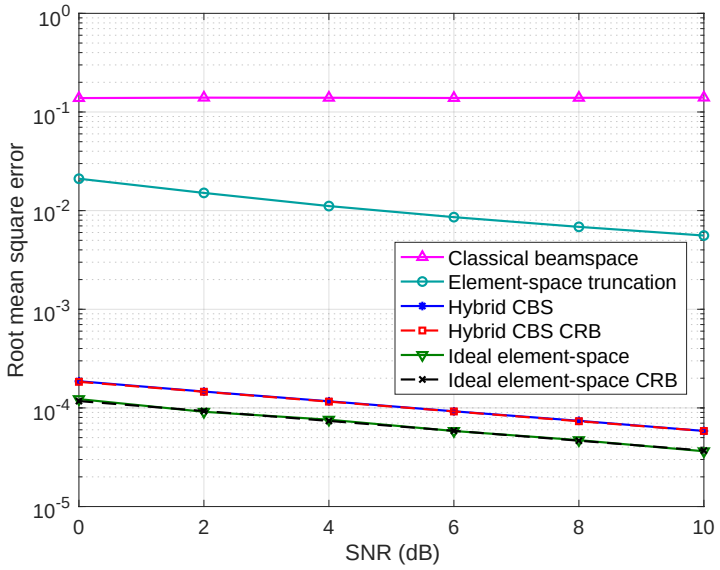


Figure 3.3: Example 1. RMSE of DOA estimates of classical beamspace [2], element-space truncation, hybrid CBS with uniform decimation, and ideal element-space without RF chain constraint. Except ideal element-space, all the other methods can be realized with 9 RF chains in the hybrid model. The ideal element-space CRB based on N-sensor ULA output and hybrid CBS CRB computed from Theorem 3.1 are also shown.

sources at angles $\theta = 40°, 60°$. The decimation ratio is $M = 8$ for hybrid CBS with uniform decimation. The CBS filter is designed to be a spectral factor of a lowpass Nyquist(M)-equiripple filter [71] with length $L = 25$ and roll-off factor $\beta = 0.5$. Thus, the number of RF chains is only $\bar{N} = (N - L + 1)/M = 9$. The DOAs are estimated using root-MUSIC [26]. Covariance estimates are obtained by using 500 snapshots, and we average 500 Monte Carlo runs to get Fig. 3.3. The RMSEs of DOA estimates of hybrid CBS are much smaller than classical beamspace [2]. Classical beamspace can be directly applied in the analog domain since its beamspace matrix is a submatrix of a DFT matrix with constant-modulus entries. The poor performance of classical beamspace is due to the numerical sensitivity issues mentioned in [2]. Just for reference, we also show the RMSE when only the first $\bar{N}$ sensors are used and all other sensors are discarded (element-space truncation). In this case there is no analog combiner, and the outputs of the $\bar{N}$ sensors are processed in the digital domain. Even in this extreme case we see that the performance is better than classical beamspace, but hybrid CBS is much better. Finally, we compare to ideal element-space without RF chain constraint, where the N-sensor ULA output can be processed fully in the digital domain. This

requires impractically high hardware cost and just serves as an ideal benchmark. *It is quite striking that the RMSE of hybrid CBS is close to this ideal case.* We also plot the ideal element-space CRB [44] based on N-sensor ULA output and hybrid CBS CRB computed from Theorem 3.1. The RMSE of each method is close to its CRB. This is not surprising since MUSIC and root-MUSIC are statistically efficient for sufficiently large arrays [26, 56].

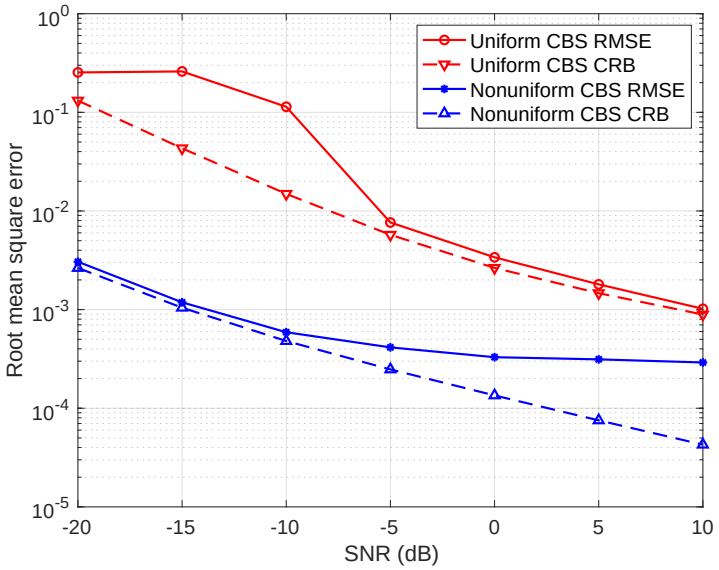


Figure 3.4: Example 2. RMSE and CRB for DOA estimates of hybrid CBS with uniform or nonuniform decimation.

Example 2 (Uniform decimation vs. nonuniform decimation for hybrid CBS): Next, we compare nonuniform decimation (CBS coarray method) to uniform decimation for hybrid CBS. We consider a ULA with $N = 108$ sensors receiving 3 passband sources at angles $\theta = -2°, 0°, 1.5°$, and 2 stopband sources at angles $\theta = 40°, 60°$. The decimation ratio is $M = 4$. The CBS filter is designed to be a spectral factor of a lowpass Nyquist(M)-equiripple filter [71] with length $L = 16$ and roll-off factor $\beta = 0.5$. For CBS coarray method, we do nonuniform decimation after filtering and retain samples at $n_i = \tilde{n}_i M$, $0 \le i \le 7$, where $\{\tilde{n}_i\}$ are the sensor locations of an 8-sensor restricted MRA [3]. For a fair comparison, we retain only 8 samples at $n_i = iM$, $0 \le i \le 7$ for uniform decimation. Hence, each method can be realized with $\bar{N} = 8$ RF chains in the hybrid model. The DOAs are estimated using root-MUSIC. Covariance estimates are obtained by using 1000 snapshots, and we average 500 Monte Carlo runs to get Fig. 3.4. The CRB for each method, which can be computed from Theorem 3.1, is also plotted. The RMSE and CRB for nonuniform

CBS is much smaller than the RMSE and CRB for uniform CBS, respectively. Nonuniform CBS has smaller estimation errors because the coarray aperture of the MRA is larger than the ULA aperture of uniform CBS. The large errors of uniform CBS here is due to the very closely spaced DOAs. Uniform CBS can still be a good method in other cases, e.g., in Example 1.

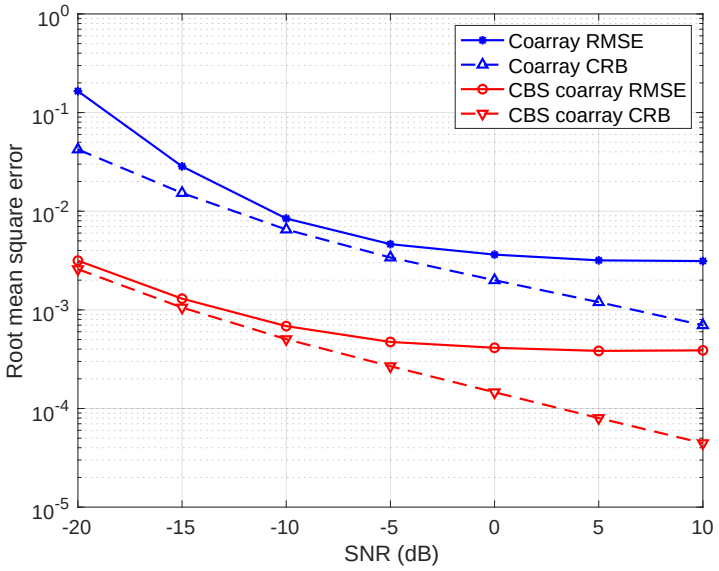


Figure 3.5: Example 3. RMSE and CRB for DOA estimates of direct coarray method in element-space and hybrid CBS coarray method. For CBS coarray method, we do nonuniform decimation after filtering and retain samples at $n_i = \tilde{n}_i M$, $0 \le i \le 7$, where $M = 4$ and $\{\tilde{n}_i\}$ are the sensor locations of an 8-sensor restricted MRA [3]. For direct coarray method, we do nonuniform decimation in element-space and retain samples at $\tilde{n}_i$, $0 \le i \le 7$. Thus, each method can be realized with 8 RF chains in the hybrid model.

Example 3 (Hybrid CBS coarray method vs. direct coarray method (both with nonuniform decimation)): Next we again consider hybrid CBS with nonuniform decimation and coarray method. There are 4 passband sources at angles $\theta = -6°, -2°, 3°, 7°$, and 2 stopband sources at angles $\theta = 40°, 60°$. All the other simulation parameters are the same as in Example 2. For comparison, we also show the performance of direct coarray method in element-space. That is, we do nonuniform decimation in element-space and retain samples at $\tilde{n}_i$, $0 \le i \le 7$, where $\{\tilde{n}_i\}$ are the sensor locations of an 8-sensor restricted MRA [3]. This nonuniform decimation in element-space (compared to beamspace after filtering) also appears in [5]. Hence, each method can be realized with $\bar{N} = 8$ RF chains in the hybrid model. In Fig. 3.5, the RMSE and CRB (computed from Theorem 3.1) for

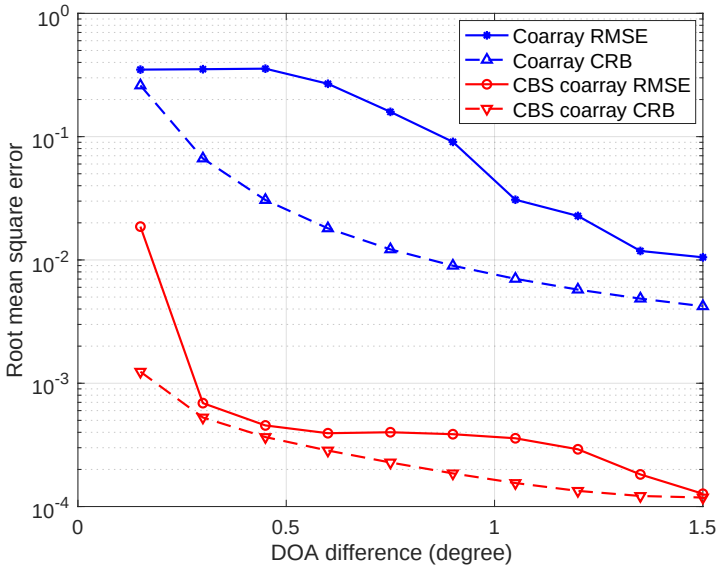


Figure 3.6: Example 3. RMSE and CRB for DOA estimates of hybrid CBS coarray method and direct coarray method in element-space as we vary the DOA difference. The decimation schemes of the two methods are the same as in Fig. 3.5.

CBS coarray method is much smaller than the RMSE and CRB for direct coarray method, respectively. CBS has smaller estimation errors because the effective coarray aperture is $M = 4$ times larger. Notice that the RMSE, but not the CRB saturates as SNR increases. Next, we consider this same example except that the SNR is fixed at 0 dB and that there are only two passband sources at angles $-\theta_0, \theta_0$. RMSEs and CRBs for DOA estimates versus the DOA difference $2\theta_0$ are plotted for hybrid CBS coarray method and direct coarray method in Fig. 3.6. Here we can see that as the two DOAs get closer, the significant increase in RMSE or CRB of direct coarray method happens earlier than those of CBS coarray method. In other words, CBS coarray method also achieves higher DOA resolution than direct coarray method.

Example 4 (Hybrid CBS with different nonuniform decimation schemes, 16 passband DOAs, 8 RF chains): In Fig. 3.7, we compare hybrid CBS coarray method with different nonuniform decimation schemes. Here we retain samples at $n_i = \tilde{n}_i M$, $0 \le i \le 7$, where $M = 4$ and $\{\tilde{n}_i\}$ are the sensor locations of either an 8-sensor nested array [32] with 4 sensors in each layer or an 8-sensor restricted MRA [3]. Each decimation scheme can be realized with $\bar{N} = 8$ RF chains. The difference coarrays of the two sparse arrays are $-19 \le k \le 19$ and $-23 \le k \le 23$, respectively. We consider a ULA with $N = 117$ sensors receiving 16 passband sources at angles

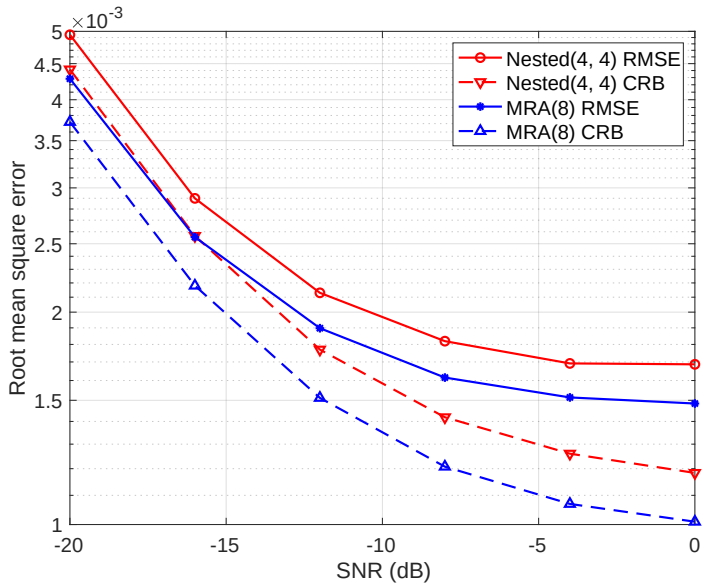


Figure 3.7: Example 4. (16 passband DOAs, 8 RF chains) RMSE and CRB for DOA estimates of hybrid CBS coarray method with nonuniform decimation corresponding to an 8-sensor nested array with 4 sensors in each layer and an 8-sensor MRA.

$\omega = -0.2\pi + i\Delta$, $i = 0, 1, \ldots, 15$, where $\Delta = 0.4\pi/15$ and no stopband sources. (Recall that $\omega = \pi \sin\theta$.) The CBS filter is designed to be a lowpass Parks-McClellan filter [63] with length $L = 25$ and roll-off factor $\beta = 0.2$. The Parks-McClellan filter is approximately a spectral factor of a Nyquist(M) filter [23], so we treat the noise after filtering as white noise and apply root-MUSIC as usual. We again use 1000 snapshots and 500 Monte Carlo runs. From the simulation results, we observe that the RMSE and CRB for the MRA are smaller than the RMSE and CRB for the nested array, respectively. This is because the coarray aperture of the MRA is larger than that of the nested array. Although the RMSE and CRB tend to saturate with SNR (as also observed in [49, 50]), we can identify 16 passband DOAs, more than the number $\bar{N} = 8$ of RF chains. This is not possible for classical beamspace or hybrid CBS with uniform decimation. The direct coarray method in element-space (as defined in Fig. 3.5) is not shown in this example since it cannot resolve those closely spaced DOAs. This is because its coarray aperture is too small without dilation.

Example 5 (Hybrid CBS with different nonuniform decimation schemes, 64 DOAs in 4 subbands, 8 RF chains): Next, we consider the same settings as in Example 4 except that there are 64 DOAs $\omega = -0.2\pi + i\Delta + 2l\pi/M$ (taken modulo 2π to make each $\omega \in [-\pi, \pi)$), $0 \le i \le 15$, $0 \le l \le 3$, where $\Delta = 0.4\pi/15$. We use a bank of 4

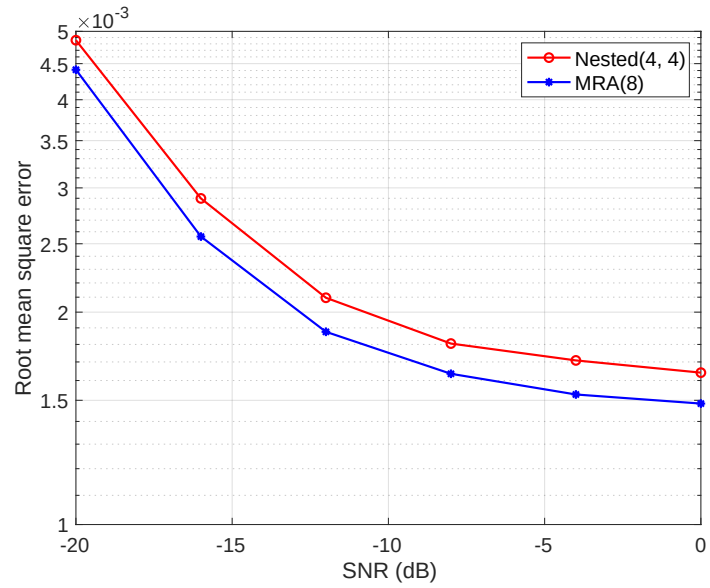


Figure 3.8: Example 5. (64 DOAs in 4 subbands, 8 RF chains) RMSE for DOA estimates of hybrid CBS coarray method with nonuniform decimation.

filters and 4×1000 snapshots to estimate the DOAs in the 4 subbands. As shown in Fig. 3.8, we can identify these 64 DOAs using 8 RF chains. In fact, the RMSE of the DOA estimates for each array in Fig. 3.8 is almost the same as that in Fig. 3.7.

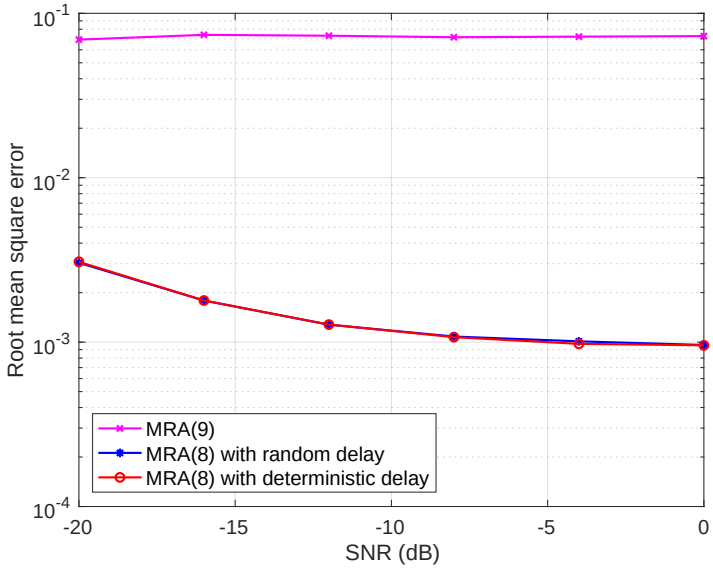


Figure 3.9: Example 6. (10 correlated passband sources, 8 or 9 RF chains) RMSE for DOA estimates of hybrid CBS coarray method with nonuniform decimation and filter delays for decorrelating sources.

Example 6 (Decorrelation method for hybrid CBS with nonuniform decimation, 10 correlated passband sources, 8 RF chains): Finally, we verify the effectiveness of the decorrelation method proposed in Sec. 3.4. We consider a ULA with $N = 132$ sensors receiving 10 passband sources at angles $\omega = -0.125\pi + i\Delta$, $i = 0, 1, \ldots, 9$, where $\Delta = 0.25\pi/9$, and 2 stopband sources at angles $\theta = 40°, 60°$. Passband sources i and $i+5$ have a correlation coefficient $\rho = 0.9$ for $i = 0, \ldots, 4$. Hybrid CBS coarray method with nonuniform decimation is used. We retain samples at $n_i = \tilde{n}_i M$, where $M = 4$ and $\{\tilde{n}_i\}$ are the sensor locations of either a 9-sensor restricted MRA or an 8-sensor restricted MRA [3]. We assume that at most 9 RF chains can be used, so each decimation scheme can be implemented. The CBS prototype filter is designed to be a lowpass Parks-McClellan filter [63] with length $L' = 16$ and roll-off factor $\beta = 0.5$. For the 8-sensor MRA scheme, we introduce random delays q_k uniformly distributed over $[0, Q]$, where $Q = 24$. That is, the decorrelation filter $f(q)$ in (3.50) is a rectangular window with length $L - L' + 1 = 25$. For comparison, we also introduce deterministic delays to approximate the uniform distribution. Since an 8-sensor MRA has aperture $A = 23$ (i.e., the last sensor location $\tilde{n}_7 = 23$), the parameters are consistent as $AM+Q+L' = 23\times4+24+16 = N$. For the 9-sensor MRA scheme, there is no room for introducing delays (i.e., decorrelation cannot be applied) because a 9-sensor MRA has aperture $A = 29$ so that $AM + L' = 29 \times 4 + 16 = N$. The DOAs are estimated using root-MUSIC with $K = 1000$ snapshots, and we average 500 Monte Carlo runs to get Fig. 3.9. The 9-sensor MRA scheme has large RMSE because the coarray method fails when sources are correlated. The 8-sensor MRA scheme with random delays performs similarly to 8-sensor MRA scheme with deterministic delays, and they are much better than the 9-sensor MRA scheme. Thus, deterministic delays are more recommended since they are easier to implement than random delays in the analog combiner.

3.7 Hybrid Capon-CBS for Passive Arrays

In Sec. 2.5, a variant of CBS called Capon-CBS is proposed. The method consists of two stages. In the first stage, we design the CBS filter based on the idea of Capon beamforming [36]. The resulting Capon-CBS filter can do a better job of suppressing out-of-band sources because input statistics are taken into account. Hence, Capon-CBS is especially useful when there are powerful out-of-band sources that are correlated with in-band sources. In the second stage, we simply implement the Capon-CBS filter obtained earlier as in traditional CBS. Thus, the hybrid implementation described in Sec. 3.3 can be readily applied for the second stage. In this

section, we therefore focus on the realization of the first stage (design stage) in the hybrid analog and digital architecture with RF chain constraints.

We consider again the ULA output $\mathbf{x}$ as in (3.1) and aim to design a CBS filter $h(n)$ of length L. As explained in Sec. 2.5, for purposes of computation, the Capon-CBS filter will be expressed explicitly in terms of real and imaginary components: $[\mathrm{Re}\{\mathbf{h}^T\}\ \mathrm{Im}\{\mathbf{h}^T\}]^T \triangleq \widetilde{\mathbf{h}}$ where $\mathbf{h} = [h^*(L-1)\ h^*(L-2)\ \cdots\ h^*(0)]^T$. Then, the CBS filter coefficient vector $\widetilde{\mathbf{h}}$ is obtained from the solution to

$$\begin{aligned} \min_{\widetilde{\mathbf{h}}} \quad & \widetilde{\mathbf{h}}^H \widetilde{\mathbf{R}}_L \widetilde{\mathbf{h}} \\ \text{subject to} \quad & \mathrm{Re}\{\widetilde{\mathbf{h}}^H \mathbf{a}\} \geq 1 \ \ \forall \mathbf{a} \in \mathcal{E}, \end{aligned} \tag{3.68}$$

where

$$\widetilde{\mathbf{R}}_L = \frac{1}{N-L+1} \sum_{i=0}^{N-L} \mathrm{E}[\widetilde{\mathbf{x}}_{L,i} \widetilde{\mathbf{x}}_{L,i}^H], \tag{3.69}$$

$$\widetilde{\mathbf{x}}_{L,i} = [\mathrm{Re}\{\mathbf{x}_{L,i}^T\}\ \mathrm{Im}\{\mathbf{x}_{L,i}^T\}]^T \tag{3.70}$$

with $\mathbf{x}_{L,i} = [x(i)\ x(i+1) \cdots x(i+L-1)]^T$, and $\mathcal{E}$ is a $2L$-dimensional ellipsoid that covers the range of values of $\widetilde{\mathbf{a}}_L(\omega) = [\mathrm{Re}\{\mathbf{a}_L^T(\omega)\}\ \mathrm{Im}\{\mathbf{a}_L^T(\omega)\}]^T$ for ω in the passband. Given $\widetilde{\mathbf{R}}_L$ and $\mathcal{E}$, the problem can be solved by Lagrange multiplier methods as shown in [83]. Hence, the problem that remains is how to obtain $\widetilde{\mathbf{R}}_L$ (or its surrogate) in the hybrid analog and digital architecture, where typically we have only $\bar{N} \ll N$ RF chains, and the size of $\widetilde{\mathbf{R}}_L$ is $2L \times 2L$ often with $L > \bar{N}$. In this case, we are unable to compute $\widetilde{\mathbf{R}}_L$ in the digital domain after the $\bar{N}$ RF chains since we cannot get the complete array output $\mathbf{x}$ needed to construct each $\mathbf{x}_{L,i}$.

To solve this problem, we propose to compute a surrogate of $\widetilde{\mathbf{R}}_L$ by nonuniformly decimating $\mathbf{x}$ and using the coarray method. We consider the integer set $\mathcal{N} = \{n_0, n_1, \ldots, n_{\bar{N}-1}\}$ corresponding to the sensor locations of some standard sparse array. Then we nonuniformly decimate $\mathbf{x}$ and retain

$$\bar{\mathbf{x}} = [x(0)\ x(n_1)\ \cdots\ x(n_{\bar{N}-1})]^T = \overline{\mathbf{A}}\mathbf{c} + \bar{\mathbf{e}}, \tag{3.71}$$

where

$$\overline{\mathbf{A}} = [\bar{\mathbf{a}}(\omega_1)\ \cdots\ \bar{\mathbf{a}}(\omega_D)] \tag{3.72}$$

with $\bar{\mathbf{a}}(\omega) = [1\ e^{j\omega n_1}\ e^{j\omega n_2}\ \cdots\ e^{j\omega n_{\bar{N}-1}}]^T$, and $\bar{\mathbf{e}} = [e(0)\ e(n_1) \cdots e(n_{\bar{N}-1})]^T$ consists of the corresponding decimated noise samples. To proceed further, we

again assume the sources c_i are uncorrelated so that the coarray method can work. (If sources are correlated, we can use the method for decorrelating sources in Sec. 3.4.) The decimated samples in $\bar{\mathbf{x}}$ can be passed through the $\bar{N}$ RF chains. Hence, in the digital domain, we can compute

$$R(n_m - n_l) \triangleq \mathrm{E}[x(n_m)x^*(n_l)] \tag{3.73}$$

$$= \sum_{i=1}^{D} p_i e^{j\omega_i(n_m - n_l)} + \sigma_{\mathrm{e}}^2 \delta(n_m - n_l), \tag{3.74}$$

which depends only on the difference $n_m - n_l$ between the virtual sensor locations, hence the notation $R(n_m - n_l)$. Let $\mathcal{C}$ be the difference coarray of the array $\mathcal{N}$. By averaging $x(n_m)x^*(n_l)$ over snapshots, and over all n_m, n_l that produce identical difference $k = n_m - n_l$, we can estimate

$$R(k) = \sum_{i=1}^{D} p_i e^{j\omega_i k} + \sigma_{\mathrm{e}}^2 \delta(k), \tag{3.75}$$

for all $k \in \mathcal{C}$, and particularly for all k in the central ULA segment, say, $-(U-1) \le k \le U-1$. Note that $R(k)$ resembles a single snapshot of the output of the array $\mathcal{C}$ due to the D DOAs ω_i. Hence, as a surrogate of $\widetilde{\mathbf{R}}_L$, we define

$$\widetilde{\mathbf{R}}'_L = \frac{1}{2U-L} \sum_{i=-U+1}^{U-L} \widetilde{\mathbf{r}}_{L,i} \widetilde{\mathbf{r}}_{L,i}^H, \tag{3.76}$$

$$\widetilde{\mathbf{r}}_{L,i} = [\mathrm{Re}\{\mathbf{r}_{L,i}^T\} \ \mathrm{Im}\{\mathbf{r}_{L,i}^T\}]^T \tag{3.77}$$

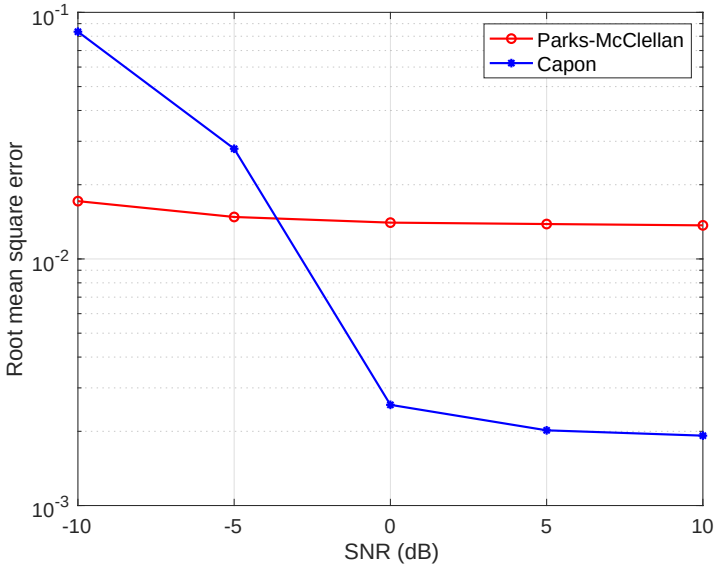
with $\mathbf{r}_{L,i} = [R(i) \ R(i+1) \cdots R(i+L-1)]^T$. We assume that $2U - L \ge 2L$ so that $\widetilde{\mathbf{R}}'_L$ can be positive definite, which is required by the method in [83]. The inequality can be satisfied easily since $U = O(\bar{N}^2)$. Then we can solve the problem (3.68) with $\widetilde{\mathbf{R}}_L$ replaced by $\widetilde{\mathbf{R}}'_L$ and obtain the Capon-CBS filter $h(n)$ in the digital domain. The unit-modulus coefficients $h_1(n)$ and $h_2(n)$ such that $a_1 h_1(n) + a_2 h_2(n) = h(n)$ for some $a_1, a_2 \in \mathbb{C}$ are then found by Lemma 3.1 in Sec. 3.3 and sent back to the analog domain to implement the phase shifters in the second stage of doing CBS filtering.

Remark 1) The application of nonuniform decimation and coarray method in the Capon-CBS design stage is similar to that in the filtering stage described in Sec. 3.2.2. However, the main difference is the decimation ratio M, as we can see by comparing (3.75) to (3.29). In the context of (3.29), there is filtering followed by decimation. In the context of (3.75), there is no filtering yet because it is precisely the filter which is being designed here. Thus, there cannot be decimation and we have to use undecimated information to design the Capon-CBS filter.

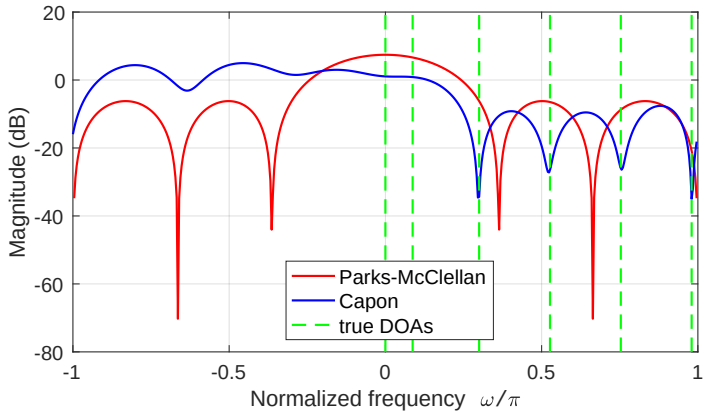
Remark 2) The definition of $\widetilde{\mathbf{R}}'_L$ in (3.76) follows the spatial smoothing method in [32] and consists of fourth-order statistics of the sources. To mimic the second-order statistics in (3.69), one may consider the improved spatial smoothing method in [79]. However, the matrix constructed from the method in [79] is generally not positive semidefinite under finite snapshots. This does not matter in terms of finding signal and noise subspaces in [79], but it does not make sense for us to minimize $\widetilde{\mathbf{h}}^H \widetilde{\mathbf{R}}'_L \widetilde{\mathbf{h}}$ if $\widetilde{\mathbf{R}}'_L$ has negative eigenvalues. This is why we choose the method in [32]. Moreover, whether we use second-order statistics or fourth-order statistics is not vital since in either way, the resulting Capon-CBS filter will tend to put zeros at the locations of stopband DOAs. The intuition is that $\widetilde{\mathbf{R}}'_L$ is like a squared covariance [32]. The eigenvectors of such covariance matrix can be classified into two groups, one corresponding to passband DOAs (signal subspace) and the other corresponding to stopband DOAs plus noise (interference-plus-noise subspace). Squaring the matrix does not alter the eigenvectors, while the eigenvalues are squared. Thus, the goal of the optimization problem is still to suppress the stopband DOAs plus noise, though with a slightly modified weighting function.

3.7.1 Simulations

Now we give a numerical example to compare hybrid Capon-CBS with hybrid CBS. We consider a ULA with $N = 49$ sensors receiving 2 passband sources at angles $\theta = 0°, 5°$, and 4 stopband sources at angles $\omega = 0.3\pi + i\Delta$, $i = 0, 1, 2, 3$, where $\Delta = 0.68\pi/3$. Each in-band source has power 0 dB, and each out-of-band source has power 10 dB. Each pair of the 6 sources has a correlation coefficient $\rho = 0.5$, except that the 2 in-band sources are uncorrelated. The CBS filter is designed to be either a lowpass Parks-McClellan filter [63] with roll-off factor $\beta = 0.5$ or a Capon filter. The filter length is $L = 8$. In the first stage of Capon-CBS design (filter design stage), we do nonuniform decimation and retain samples at n_i, $0 \le i \le 7$, where $\{n_i\}$ are the sensor locations of an 8-sensor restricted MRA [3]. To decorrelate sources, we introduce deterministic delays q_k to approximate a uniform distribution over $[0, Q]$, where $Q = 25$. The ellipsoid $\mathcal{E}$ in (3.68) is designed in the same way as in Sec. 2.5. In the second stage of CBS filtering (filter implementation stage), we do uniform decimation with ratio $M = 4$. We assume that there are only $\bar{N} = 8$ RF chains, so only 8 uniform samples are retained. The DOAs are estimated using root-MUSIC. We use 1000 snapshots and 500 Monte Carlo runs. As shown in Fig. 3.10(a), if the SNR is not too low, Capon-CBS can yield a smaller RMSE of DOA estimates. This is because the Capon-CBS filter can better suppress the powerful



(a)



(b)

Figure 3.10: Performance of hybrid CBS using a Parks-McClellan filter and Capon-CBS. (a) RMSE of DOA estimates. (b) Filter responses when SNR is 0 dB.

out-of-band sources that are correlated with in-band sources, as depicted in Fig. 3.10(b) when SNR is 0 dB. The Capon-CBS filter almost puts zeros at the locations of the stopband DOAs. Hybrid Capon-CBS has a large error at very low SNR since the filter obtained is far from the ideal Capon filter due to large noise. By contrast, the Parks-McClellan filter of hybrid CBS is predefined and not affected by noise.

3.8 Concluding Remarks

In this chapter, hybrid CBS is proposed for mmWave passive arrays. Using the fact that any complex vector can be written as a linear combination of two vectors with

unit-modulus entries, we show that any general CBS filter can be implemented. Two decimation schemes are shown: uniform decimation and nonuniform decimation. Uniform decimation is used in traditional CBS (Chapter 2), while nonuniform decimation is proposed in this chapter. In the new scheme, we use the coarray method to estimate $O(\bar{N}^2)$ DOAs with $\bar{N}$ RF chains. Moreover, the retained samples correspond to the sensor locations of a virtual sparse array, dilated by an integer factor M. This leads to M times larger coarray aperture and thus smaller estimation errors, as shown in simulations. We also propose a new decorrelation method by using random or deterministic filter delays. The two types of delays yield similar estimation performance, so we recommend deterministic delays as they are easier to implement in analog processors. Besides, we derive the CRB for any beamspace-processed passive array output. We also show how to realize the design stage of Capon-CBS filter in the hybrid mode based on nonuniform decimation. We have verified the performance of the proposed methods via simulations.

To deal with the unit-modulus constraints of analog processors, we write each CBS filter coefficient as a linear combination of two unit-modulus numbers, so we need twice the number of phase shifters. Instead, we can consider using deep learning to optimize the analog processor coefficients or the filter coefficients under the unit-modulus constraints without increasing phase shifters. These problems are nonconvex and difficult to solve using classical techniques. Machine learning offers a new way to solving them.

Chapter 4

HYBRID CONVOLUTIONAL BEAMSPACE FOR MMWAVE MIMO CHANNEL ESTIMATION

4.1 Introduction

Array processing for millimeter wave (mmWave) signals has been a popular topic in recent years [4–6, 9–11, 19, 41, 93, 94, 97, 98]. A main reason for considering mmWaves is the potential to offer more bandwidth than the highly occupied lower-frequency microwave bands, but there are also new challenges, such as strong path loss at the high frequencies of mmWaves [10]. To compensate for the path loss, large arrays are used, which are practical due to the small sizes of mmWave antennas, to get large beamforming gain [11]. This makes it impractical to implement one RF chain per antenna in view of hardware complexity. Instead, many have proposed to use hybrid analog and digital processing.

In Chapter 3, DOA estimation based on mmWave passive arrays is studied. In this chapter, we consider mmWave MIMO channel estimation problem [11]. In a hybrid transceiver design, the transmitter has a digital precoder and an analog precoder, and the receiver has an analog combiner and a digital combiner, as shown in Fig. 4.1 (see Sec. 4.2). The analog precoder and combiner constitute an important part of the design. They serve as dimension reducers so that we can use fewer RF chains than the number of antennas. Meanwhile, they should be designed in a way such that we can get good channel estimates from the combiner output. In the literature, there are two typical analog designs [48]. One is based on switches so that the precoder and combiner are selection matrices with entries 1's and 0's. The other is based on phase shifters so that the precoder and combiner are matrices with unit-modulus entries. (The unit-modulus constraint is assumed in many papers [4, 6, 11, 19, 97] but not necessary as some hybrid processors allow gain control [99].) Phase shifter design in general gives better performance than switch design since all antenna outputs can be used simultaneously [48]. Thus, we consider phase shifter design in this chapter.

The sparse nature of an mmWave MIMO channel due to limited scattering allows it to be modeled by a few paths, each with a direction of departure (DOD) at the transmitter and direction of arrival (DOA) at the receiver [11]. Hence, channel estimation can be realized by the estimation of 2-dimensional (2-D) angles DODs

and DOAs followed by path gain estimation, and techniques for 1-dimensional (1-D) DOA estimation may be extended to mmWave MIMO channel estimation. In general, there are two phases in mmWave MIMO channel estimation, the initial beam training phase and the beam tracking phase [100]. In this chapter, we focus on the beam training phase, where we use training data to obtain initial DOD and DOA estimates (whereas the generally time-varying DODs and DOAs should be estimated, or tracked, in the beam tracking phase). To fulfill the unit-modulus constraint, in many papers [4, 6, 11, 19, 97], the columns of the analog precoder and rows of the combiner are designed to be of the form of a steering vector. A hierarchical multi-resolution beamforming codebook is used with compressed sensing in [11]. However, it is a grid-based method, so the DOD and DOA resolution and estimation performance are limited by the density of grid points. To get good performance, we need a dense grid, which requires high computational complexity. Another hierarchical beam search method based on joint subarray and deactivation codebook design [101] is proposed in [6] and shown to require smaller pilot overhead than the method in [11], but it is still a grid-based method. In [4, 19, 97], the analog precoder and combiner are particularly designed as DFT beamformers, and DODs and DOAs are then estimated by high-resolution subspace methods, such as MUltiple SIgnal Classification (MUSIC) [19] and Estimation of Signal Parameters via Rotational Invariance Techniques (ESPRIT) [4, 97]. MUSIC is also a grid-based method, so the performance of [19] is limited by the density of grid points. In [97], although the analog precoder has DFT columns, the digital precoder is designed such that the overall precoder is equivalent to a selection matrix with entries 1's and 0's. Thus, it behaves more like the switch design. ESPRIT is a gridless method which uses the rotational invariance [27] of a uniform linear array (ULA) output to estimate DOAs. DFT beamformers do not preserve this rotational invariance, so elaborate steps are taken in [4] to rebuild a similar invariance.

To make high-resolution gridless subspace methods readily applicable to the combiner output, we propose to use the idea of *convolutional beamspace (CBS)* described in Chapter 2 to design the hybrid precoder and combiner. CBS is a beamspace DOA estimation method originally proposed for passive arrays with purely digital implementation. In CBS, the received array output is first spatially filtered by a finite-impulse-response (FIR) filter. Then uniform *decimation* [67, Sec. II] (also called *downsampling*) is used to reduce dimension. As mathematically defined in (4.10), a uniform decimator keeps one sample every M_{r} samples, where M_{r} is a positive integer called decimation ratio. The decimation does not cause DOA ambiguity

since the filter output is represented only by passband sources. The advantages of CBS include complexity reduction, higher DOA resolution, and smaller bias.

In this chapter, motivated by the digital CBS combiner in Chapter 2 and hybrid CBS combiner in Chapter 3, we design hybrid precoder and combiner for mmWave MIMO channels so that the 2-D Vandermonde structure is preserved at the combiner output. Then, high-resolution gridless subspace methods like 2-D unitary ESPRIT [51] can be used without additional processing. In contrast to decimation, we use *expansion* [67, Sec. II] (also called *upsampling*) in the precoder. As defined in (4.12), a uniform expander inserts $M_{\mathrm{t}} - 1$ zero-valued samples between each pair of adjacent input samples, where M_{t} is a positive integer called expansion ratio. An expander is typically used with a filter to do interpolation [67, Sec. II]. Besides the uniform scheme, a new scheme using nonuniform decimation and expansion, as mathematically defined in (4.20) and (4.21), is also proposed. The new scheme will be used with (difference) coarray method [32]. Details about coarrays are explained in Sec. 4.4.2. For a standard sparse array, like a minimum redundancy array (MRA) [3], nested array [32], or coprime array [40], its coarray has a large central ULA segment, so we can get better DOA estimates.

The main contributions of this chapter are as follows:

- We propose a new hybrid precoding (and combining) and channel estimation method, called 2-D hybrid convolutional beamspace (CBS), for mmWave MIMO transceivers. The precoder is designed as an expander (upsampler) followed by an FIR filter (the two together also called an interpolator [67, Sec. II]). Such expansion allows us to use a small number of RF chains to generate the transmitted signals for a large number of antennas. The combiner is designed as an FIR filter followed by a decimator (downsampler). Such decimation greatly reduces dimension, so the combiner output can be processed by a small number of RF chains despite a large number of receive antennas.

- We propose the first precoding scheme, where uniform expansion and decimation are used. Let M_{t} and M_{r} denote the transmit expansion ratio and receive decimation ratio, and $\bar{N}_{\mathrm{t}}$ and $\bar{N}_{\mathrm{r}}$ denote number of transmit and receive RF chains, respectively. Then we show that the received block will resemble one obtained from a pair of virtual (not physical) $\bar{N}_{\mathrm{t}}$-antenna transmit sparse ULA with antenna spacing M_{t} (measured in multiples of half-wavelength),

and virtual $\bar{N}_{\mathrm{r}}$-antenna receive sparse ULA with antenna spacing M_{r}. *There is no DOD/DOA ambiguity as only in-band paths are present after filtering*, so high-resolution subspace methods like 2-D unitary ESPRIT [51] can be readily used to estimate $O(\bar{N}_{\mathrm{t}}\bar{N}_{\mathrm{r}})$ pairs of DODs and DOAs. Moreover, the effective transmit and receive array apertures are M_{t} and M_{r} times larger. The aperture of a linear array is the difference between its extreme elements, and a larger aperture can give better estimation performance [49, 50].

- We propose the second precoding scheme, where nonuniform expansion and decimation are used. For the expander, the output samples at sample locations $\tilde{n}_{\mathrm{t},i}M_{\mathrm{t}}$ equal the input samples at sample locations i, $0 \le i \le \bar{N}_{\mathrm{t}} - 1$. For the decimator, the output samples at sample locations i equal the input samples at sample locations $\tilde{n}_{\mathrm{r},i}M_{\mathrm{r}}$, $0 \le i \le \bar{N}_{\mathrm{r}} - 1$. (See (4.18), (4.20), and (4.21).) Here each $\mathcal{N}_p = \{\tilde{n}_{p,0}, \ldots, \tilde{n}_{p,\bar{N}_p-1}\}$, $p \in \{\mathrm{t}, \mathrm{r}\}$, is the set of antenna locations of a virtual (not physical) standard sparse array. Thus, the expansion and decimation correspond to standard sparse arrays dilated by M_p times. Then, we show that such nonuniform decimation output resembles a sparse array output, so by applying the 2-D coarray method [5], we can theoretically estimate as many as $O(\bar{N}_{\mathrm{t}}^2\bar{N}_{\mathrm{r}}^2)$ pairs of DODs and DOAs using methods like 2-D unitary ESPRIT [51]. More importantly, given a fixed number of RF chains, the nonuniform scheme yields larger array apertures and thus better estimation performance than the uniform scheme. Besides, because of the dilation factors M_{t} and M_{r}, we get larger effective coarray apertures (compared to a system not using dilation) and thus higher DOD and DOA resolution and smaller estimation errors.

- We show that the proposed CBS precoder and combiner with any filter coefficients can be implemented in a hybrid way under the unit-modulus constraints of phase shifters. Moreover, the required number of transmit RF chains equals the number of transmitted symbols in a training vector, and the number of receive RF chains we need equals the dimension of the decimated CBS output. In other words, no additional cost in terms of RF chains is incurred by the unit-modulus constraints (unlike in [48, 95], using twice the number of RF chains).

- We show that hybrid CBS yields smaller estimation errors than other non-beam-search methods like DFT beamspace MUSIC [19] and DFT beamspace ESPRIT [4] when having the same pilot overhead. Also, since hybrid CBS can

be used with high-resolution gridless subspace methods, it requires smaller pilot overhead to achieve the same estimation error as grid-based beam search methods such as hierarchical beam search [6].

- We propose a method for decorrelating path gains. We design the CBS filters for different snapshots as different delayed versions of a prototype filter. Then, the effective correlations between a pair of path gains will be proportional to the 2-D discrete-time Fourier transform of the joint probability distribution of the transmit and receive filter delays. Thus, by designing that Fourier transform to be a 2-D lowpass filter, we can decorrelate the path gains. The method is particularly useful for CBS with nonuniform decimation because uncorrelated path gains are needed to apply coarray method.

There are two stages of a hybrid MIMO channel estimation method. Stage 1 is hybrid precoding and combining. Stage 2 is channel estimation based on the combiner output. Our main focus is that if stage 1 is designed as proposed, we can use gridless subspace methods (and with coarray method for nonuniform scheme) to get good estimates in stage 2. Such good estimates can be also obtained if one uses other 2-D angle estimation methods, such as state-of-the-art compressed sensing methods like expectation-maximization-based turbo compressed sensing [102] or successive-linear-approximation variational Bayesian inference [103]. However, we will not show simulations particularly for these compressed sensing methods. Instead, we show that hybrid CBS in stage 1 with subspace method in stage 2 performs better than classical beamspace in stage 1 with subspace method in stage 2. This is sufficient to see the merit of hybrid CBS. Also, we compare to hierarchical beam search [6] since like CBS, the design of stage 1 is also a key part of the method.

The proposed method is especially suitable for massive MIMO or large arrays, which are getting popular recently [9, 11], because we can design good CBS filters with large enough filter lengths. Also, in conventional array processing, the M_p-sparsity can cause DOD and DOA ambiguity. However, in our case, only in-band paths remain after CBS filtering. Thus, by designing the transmit and receive CBS filters to have passband widths $2\pi/M_\mathrm{t}$ and $2\pi/M_\mathrm{r}$, the ambiguity can be resolved. By contrast, in [5], nonuniformly decimating a convolution output is also considered, but their design is for decorrelating the path gains instead of for filtering the DODs and DOAs. Without filtering, they cannot use the dilation factors M_p to increase effective coarray aperture and get better estimation performance. The good performance of our method will be shown in simulations (Sec. 4.8).

Chapter outline: The model of mmWave MIMO transceiver systems is presented in Sec. 4.2. Then, 2-D hybrid CBS with uniform decimation and expansion is proposed for mmWave MIMO channel estimation in Sec. 4.3, and 2-D hybrid CBS with nonuniform decimation and expansion is proposed in Sec. 4.4. Their hybrid analog and digital implementation is shown in Sec. 4.5. Time complexity is analyzed in Sec. 4.6, where we compare 2-D hybrid CBS with previous methods in terms of pilot overhead. The new method for decorrelating path gains based on filter delays in 2-D hybrid CBS is presented in Sec. 4.7. Numerical examples for 2-D hybrid CBS are shown in Sec. 4.8. Finally, the conclusion is given in Sec. 4.9.

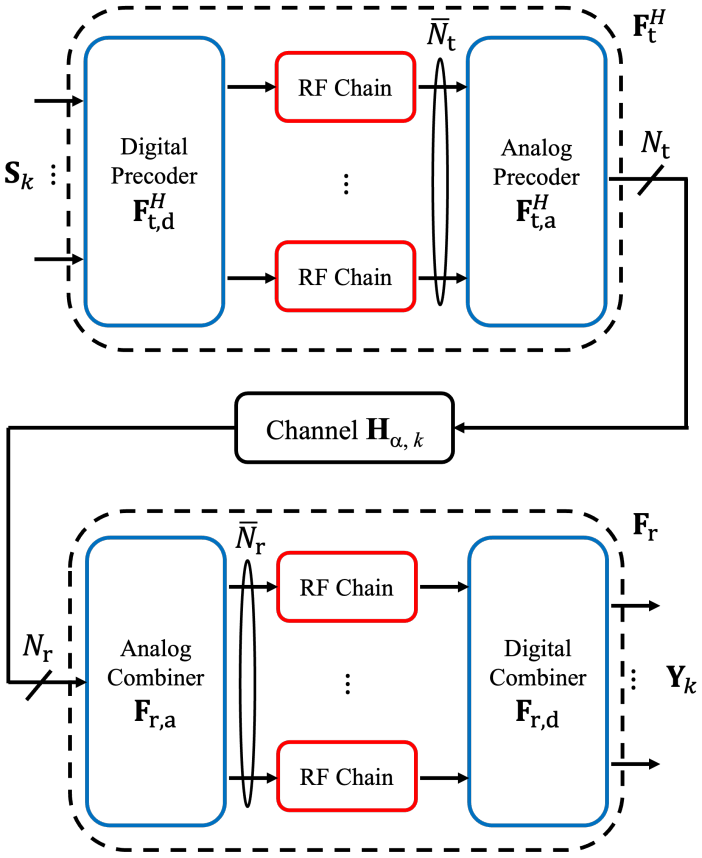


Figure 4.1: System model of hybrid CBS for mmWave MIMO channel estimation. The analog precoder and analog combiner can only have unit-magnitude multipliers. Typically, $N_\mathrm{t} \gg \bar{N}_\mathrm{t}$ and $N_\mathrm{r} \gg \bar{N}_\mathrm{r}$.

4.2 Model of mmWave MIMO Transceiver Systems

We consider an mmWave MIMO transceiver system as shown in Fig. 4.1. At the receiver, the output of the N_r-antenna receive ULA is first processed by an analog combiner, resulting in a $\bar{N}_\mathrm{r}$-dimensional output, where $\bar{N}_\mathrm{r} \ll N_\mathrm{r}$ typically. It is then

passed through $\bar{N}_{\mathrm{r}}$ analog-to-digital RF chains, and finally DOA estimation is done in the digital domain. The analog and digital building blocks of the transmitter are reversely ordered compared to the receiver. In particular, $\bar{N}_{\mathrm{t}}$ digital symbols are passed through $\bar{N}_{\mathrm{t}}$ digital-to-analog RF chains. Then the output is processed by an analog precoder which generates the transmitted signals for the N_{t}-antenna transmit ULA. Again, $\bar{N}_{\mathrm{t}} \ll N_{\mathrm{t}}$ typically. The analog precoder and analog combiner have unit-modulus entries since they are implemented by phase shifter networks. The digital precoder and digital combiner in general can be any matrices as they are in the digital domain. They are there for generality, but for channel estimation problem, we later set them to be identity matrices. As explained in Sec. 1.2, the $N_{\mathrm{r}} \times N_{\mathrm{t}}$ mmWave narrowband MIMO channel matrix can be modeled by [5, 11, 19]

$$\mathbf{H}_{\alpha,k} = \sum_{i=1}^{D} \alpha_{k,i} \mathbf{a}_{N_{\mathrm{r}}}(\omega_{\mathrm{r},i}) \mathbf{a}_{N_{\mathrm{t}}}^{H}(\omega_{\mathrm{t},i}) = \mathbf{A}_{\mathrm{r}} \mathbf{D}_{\alpha,k} \mathbf{A}_{\mathrm{t}}^{H}, \tag{4.1}$$

where $\mathbf{A}_p = [\mathbf{a}_{N_p}(\omega_{p,1}) \ \cdots \mathbf{a}_{N_p}(\omega_{p,D})]$, $p \in \{\mathrm{t}, \mathrm{r}\}$,

$$\mathbf{a}_N(\omega) = [1 \ e^{j\omega} \ e^{j2\omega} \ \cdots \ e^{j(N-1)\omega}]^T \tag{4.2}$$

for any real number ω and positive integer N, and $\mathbf{D}_{\alpha,k} = \mathrm{diag}(\alpha_{k,1}, \ldots, \alpha_{k,D})$. Here, k is the training block index. In this channel model, D denotes the number of paths, and $\alpha_{k,i}$, $\omega_{\mathrm{t},i}$, and $\omega_{\mathrm{r},i}$ are the path gain, direction of departure (DOD), and direction of arrival (DOA) for the ith path, respectively. Also, each DOD or DOA $\omega_{p,i} = \pi \sin\theta_{p,i} \in [-\pi, \pi)$, where $\theta_{p,i} \in [-\pi/2, \pi/2)$ is the physical DOD or DOA measured from the normal to the line of array. We assume that the DODs and DOAs remain the same throughout the training process, and that the path gains $\alpha_{k,i}$ vary from block to block independently such that $\mathrm{E}[\alpha_{k,m}] = 0$ and

$$\mathrm{E}[\alpha_{k,m} \alpha_{i,l}^{*}] = \rho_{m,l} \delta_{ki} \tag{4.3}$$

for all i, k, l, and m. In general, the gains of different paths can be correlated so that $\rho_{m,l} \neq 0$ for some $m \neq l$.

To estimate the MIMO channel, we transmit K training blocks $\mathbf{S}_k$, each of which is composed of $\bar{N}_{\mathrm{t}}$ training vectors $\mathbf{s}_{k,i}$ for $i = 1, \ldots, \bar{N}_{\mathrm{t}}$ and $k = 1, \ldots, K$. In particular, we design the training blocks to be [11]

$$\mathbf{S}_k = [\mathbf{s}_{k,1} \ \mathbf{s}_{k,2} \ \cdots \ \mathbf{s}_{k,\bar{N}_{\mathrm{t}}}] = \sigma_{\mathrm{s}} \mathbf{I}_{\bar{N}_{\mathrm{t}}} \tag{4.4}$$

for all k. The receive array output is perturbed by additive white noise $\mathbf{E}_k$ with $\mathrm{E}[\mathbf{E}_k] = \mathbf{0}$ and $\mathrm{E}[\mathrm{vec}(\mathbf{E}_k)\mathrm{vec}^{H}(\mathbf{E}_l)] = \sigma_{\mathrm{e}}^2 \delta_{kl} \mathbf{I}$ for all k and l. The noise is

also assumed to be uncorrelated with the path gains. In the following, we first consider CBS with uniform decimation (and expansion) and then with nonuniform decimation.

4.3 2-D Hybrid CBS With Uniform Decimation and Expansion

The CBS method described in Chapter 3 is for passive arrays. We now extend the method to both transmit and receive arrays for mmWave MIMO channel estimation. In this section, we design an mmWave MIMO transceiver using CBS with uniform decimation and expansion. The receive combiner is designed in a similar way to CBS for passive arrays in Chapter 3, while the transmit precoder is designed as the Hermitian transpose of a receive combiner. We consider FIR filters $H_p(z) = \sum_{n=0}^{L_p-1} h_p(n)z^{-n}$ of length L_p, where $p \in \{\mathrm{t}, \mathrm{r}\}$ refers to the transmitter and receiver, respectively. Then we define $(N_p - L_p + 1) \times N_p$ banded Toeplitz matrices $\mathbf{H}_p$ as in (2.6) in traditional CBS. That is, $\mathbf{H}_p$ have entries

$$[\mathbf{H}_p]_{i,k} = h_p(L_p - 1 + i - k), \tag{4.5}$$

where we assume $h_p(n) = 0$ for $n < 0$ or $n \geq L_p$. Then, we consider uniform decimation matrices

$$\overline{\mathbf{D}}_p = [\boldsymbol{\delta}_0^{(N_p-L_p+1)} \ \boldsymbol{\delta}_{M_p}^{(N_p-L_p+1)} \ \cdots \ \boldsymbol{\delta}_{(\bar{N}_p-1)M_p}^{(N_p-L_p+1)}]^T, \tag{4.6}$$

where $\bar{N}_p = \lceil (N_p - L_p + 1)/M_p \rceil$, $p \in \{\mathrm{t}, \mathrm{r}\}$. Here M_t and M_r are the expansion ratio for the transmitter and decimation ratio for the receiver, respectively.

Now we design the overall precoder to be

$$\mathbf{F}_\mathrm{t}^H = \mathbf{H}_\mathrm{t}^H \overline{\mathbf{D}}_\mathrm{t}^H \in \mathbb{C}^{N_\mathrm{t} \times \bar{N}_\mathrm{t}} \tag{4.7}$$

and the overall combiner to be

$$\mathbf{F}_\mathrm{r} = \overline{\mathbf{D}}_\mathrm{r} \mathbf{H}_\mathrm{r} \in \mathbb{C}^{\bar{N}_\mathrm{r} \times N_\mathrm{r}}. \tag{4.8}$$

We only consider the overall precoder $\mathbf{F}_\mathrm{t}^H$ and combiner $\mathbf{F}_\mathrm{r}$ here, and the factorization into analog and digital parts for hybrid implementation will be explained in Sec. 4.5. Then, the kth received block is

$$\mathbf{Y}_k = \overline{\mathbf{D}}_\mathrm{r} \mathbf{H}_\mathrm{r} \mathbf{H}_{\alpha,k} \mathbf{H}_\mathrm{t}^H \overline{\mathbf{D}}_\mathrm{t}^H \mathbf{S}_k + \overline{\mathbf{D}}_\mathrm{r} \mathbf{H}_\mathrm{r} \mathbf{E}_k \in \mathbb{C}^{\bar{N}_\mathrm{r} \times \bar{N}_\mathrm{t}}, \tag{4.9}$$

where $\mathbf{E}_k$ is the additive noise as defined in Sec. 4.2. We can observe that there is Hermitian-symmetry between the transmitter and receiver. That is, we define the

overall precoder to be like the Hermitian transpose of a combiner as in (4.7). Since $\mathbf{S}_k$ is (a constant multiple of) identity, each column of $\mathbf{S}_k$ produces a shifted version of the filter as beamformer. These shifts are rendered uniform due to our definition of $\overline{\mathbf{D}}_{\mathrm{t}}$. Also, it is good to understand the operation in terms of basic building blocks of digital signal processing. By definition, the overall combiner is equivalent to the filter $h_{\mathrm{r}}(n)$ followed by a uniform decimator, as in Fig. 4.2. In Fig. 4.2, $x(n)$, $0 \le n \le N_{\mathrm{r}} - 1$ is the receive array output, and $x_1(n)$ is the filter output, given input $x(n)$. The uniform decimator [67, Sec. II] is defined by

$$y(n) = x_1(M_{\mathrm{r}} n) \tag{4.10}$$

for all integers n. For the precoder, if we consider any input $\mathbf{s} = [s(0)\ s(1) \cdots s(\bar{N}_{\mathrm{t}} - 1)]^T$ and output $\mathbf{u} = [u(0)\ u(1) \cdots u(N_{\mathrm{t}} - 1)]^T$ such that

$$\mathbf{u} = \mathbf{F}_{\mathrm{t}}^H \mathbf{s}, \tag{4.11}$$

we can show that the operation is equivalent to a uniform expander followed by the filter $h_{\mathrm{t}}^*(L_{\mathrm{t}} - 1 - n)$, as in Fig. 4.3. The uniform expander [67, Sec. II] is defined by

$$s_1(n) = \sum_{i=-\infty}^{\infty} s(i)\delta(n - iM_{\mathrm{t}}). \tag{4.12}$$

That is, $s_1(n)$ is obtained by inserting $M_{\mathrm{t}} - 1$ zeros between every pair of adjacent samples of $s(n)$. The overall operation in Fig. 4.3 is also called linear interpolation [66]. Now using (4.1) and (4.4), and remembering that $\mathbf{H}_p$, $p \in \{\mathrm{t}, \mathrm{r}\}$ are as in (4.5), we can show that (4.9) simplifies to

$$\mathbf{Y}_k = \overline{\mathbf{A}}_{\mathrm{r}} \widetilde{\mathbf{D}}_k \overline{\mathbf{A}}_{\mathrm{t}}^H + \overline{\mathbf{D}}_{\mathrm{r}} \mathbf{H}_{\mathrm{r}} \mathbf{E}_k, \tag{4.13}$$

where

$$\overline{\mathbf{A}}_p = [\mathbf{a}_{\bar{N}_p}(M_p \omega_{p,1}) \ \cdots \ \mathbf{a}_{\bar{N}_p}(M_p \omega_{p,D})] \tag{4.14}$$

with $\mathbf{a}_N(\omega)$ as defined in (4.2), and $\widetilde{\mathbf{D}}_k$ is diagonal with

$$[\widetilde{\mathbf{D}}_k]_{i,i} = \sigma_{\mathrm{s}} \alpha_{k,i} H_{\mathrm{r}}(e^{j\omega_{\mathrm{r},i}}) e^{j\omega_{\mathrm{r},i}(L_{\mathrm{r}}-1)} H_{\mathrm{t}}^*(e^{j\omega_{\mathrm{t},i}}) e^{-j\omega_{\mathrm{t},i}(L_{\mathrm{t}}-1)}. \tag{4.15}$$

Hence, $\mathbf{Y}_k$ can be viewed as though it is a received block obtained from a pair of virtual $\bar{N}_{\mathrm{t}}$-antenna transmit ULA and virtual $\bar{N}_{\mathrm{r}}$-antenna receive ULA due to paths with filtered gains $\alpha_{k,i} H_{\mathrm{r}}(e^{j\omega_{r,i}}) H_{\mathrm{t}}^*(e^{j\omega_{t,i}})$ (ignoring the unimportant phase shifts), DODs $M_{\mathrm{t}}\omega_{\mathrm{t},i}$, and DOAs $M_{\mathrm{r}}\omega_{\mathrm{r},i}$. The two ULAs are called virtual because they are

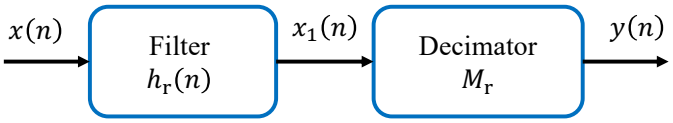


Figure 4.2: Equivalence of the overall uniform CBS combiner to a filter followed by a uniform decimator.

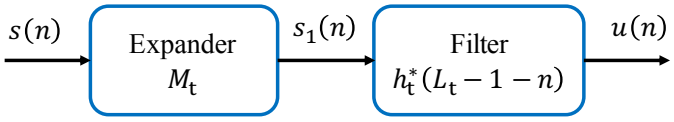


Figure 4.3: Equivalence of the overall uniform CBS precoder to a uniform expander followed by a filter.

not physical arrays. For large transmit and receive ULAs (large N_{t} and N_{r}), which are getting popular recently [5, 9, 11], we can make L_{t} and L_{r} large and design sharp-cutoff filters with good stopband.

Just like the 1-D case of passive arrays, the presence of M_p in (4.14) implies that initially we can only identify $M_p\omega_{p,i}$ mod 2π, causing ambiguities. However, since only those $\omega_{p,i}$ in the passbands of $H_p(e^{j\omega})$ which have width $2\pi/M_p$, $p \in \{\mathrm{t}, \mathrm{r}\}$ will remain, the ambiguities can be resolved (see (2.20)). Without loss of generality, we assume that the first D_0 paths are in-band paths. An in-band path is a path whose DOD is in the passband of $H_{\mathrm{t}}(e^{j\omega})$, and whose DOA is in the passband of $H_{\mathrm{r}}(e^{j\omega})$. A path that is not an in-band path will be called an out-of-band path. Then we have

$$\mathbf{Y}_k \approx \overline{\mathbf{A}}_{\mathrm{r,IB}}\widetilde{\mathbf{D}}_{k,\mathrm{IB}}\overline{\mathbf{A}}_{\mathrm{t,IB}}^H + \overline{\mathbf{D}}_{\mathrm{r}}\mathbf{H}_{\mathrm{r}}\mathbf{E}_k, \tag{4.16}$$

where $\overline{\mathbf{A}}_{p,\mathrm{IB}}$ has the first D_0 columns of $\overline{\mathbf{A}}_p$, and $\widetilde{\mathbf{D}}_{k,\mathrm{IB}}$ is the top left $D_0 \times D_0$ subblock of $\widetilde{\mathbf{D}}_k$.

We note that $\mathbf{Y}_k$ also resembles a received block obtained from a pair of virtual $\bar{N}_{\mathrm{t}}$-antenna transmit sparse ULA (with antenna spacing M_{t}) and virtual $\bar{N}_{\mathrm{r}}$-antenna receive sparse ULA (with antenna spacing M_{r}) due to DODs $\omega_{\mathrm{t},i}$ and DOAs $\omega_{\mathrm{r},i}$. This M_{t} and M_{r} times dilation yields larger effective array apertures at the transmitter and receiver and thus better estimation performance, as we shall see in simulations. The noise term can be whitened by letting the CBS filter $H_{\mathrm{r}}(z)$ be a spectral factor of a Nyquist(M_{r}) filter. (See Sec. 4.4.2 for more details.) The DODs and DOAs can be estimated from (4.16) using standard methods like 2-D unitary ESPRIT [51], and at most $\min(\bar{N}_{\mathrm{t}}(\bar{N}_{\mathrm{r}}-1), \bar{N}_{\mathrm{r}}(\bar{N}_{\mathrm{t}}-1))$ paths can be identified. Also like the 1-D case of passive arrays, we can process the training blocks with two entire filter banks

$H_p^{(i_p)}(e^{j\omega}), 0 \le i_p \le M_p - 1$ to cover the full DOD and DOA ranges $-\pi \le \omega_p < \pi$, $p \in \{\mathrm{t}, \mathrm{r}\}$, similar to Fig. 2.3. This allows us to estimate all the DODs and DOAs if they are not known a priori to be located in specific sectors. This method of filter banks can also be used for CBS with nonuniform decimation and expansion proposed in Sec. 4.4.

4.4 2-D Hybrid CBS With Nonuniform Decimation and Expansion

In this section, we introduce an mmWave MIMO transceiver using CBS with nonuniform decimation and expansion. The new method is a 2-D extension to 1-D hybrid CBS with nonuniform decimation in Sec. 3.2.2. Given a fixed number of RF chains, this yields larger array apertures and thus better estimation performance than the uniform scheme in Sec. 4.3, as we shall explain. Several relevant arrays will appear in this section. We list them here for easy reference:

1. Physical (large) transmit ULA and receive ULA
2. Virtual transmit sparse array $\mathcal{N}_\mathrm{t} = \{\tilde{n}_{\mathrm{t},i}\}$ and receive sparse array $\mathcal{N}_\mathrm{r} = \{\tilde{n}_{\mathrm{r},i}\}$
3. Virtual transmit dilated sparse array $\mathcal{N}_{\mathrm{t},M_\mathrm{t}} = \{\tilde{n}_{\mathrm{t},i} M_\mathrm{t}\}$ and receive dilated sparse array $\mathcal{N}_{\mathrm{r},M_\mathrm{r}} = \{\tilde{n}_{\mathrm{r},i} M_\mathrm{r}\}$
4. Virtual transmit coarray $\mathcal{C}_\mathrm{t} = \{k - l \mid k, l \in \mathcal{N}_\mathrm{t}\}$ and receive coarray $\mathcal{C}_\mathrm{r} = \{k - l \mid k, l \in \mathcal{N}_\mathrm{r}\}$
5. Virtual transmit dilated coarray $\mathcal{C}_{\mathrm{t},M_\mathrm{t}} = \{k - l \mid k, l \in \mathcal{N}_{\mathrm{t},M_\mathrm{t}}\}$ and receive dilated coarray $\mathcal{C}_{\mathrm{r},M_\mathrm{r}} = \{k - l \mid k, l \in \mathcal{N}_{\mathrm{r},M_\mathrm{r}}\}$

Items 2 and 3 are explained Sec. 4.4.1, and Items 4 and 5 are explained Sec. 4.4.2.

4.4.1 Nonuniform Decimation and Expansion

In the definitions (4.7) and (4.8) of the overall precoder and combiner, instead of uniform decimation matrices, we can also consider nonuniform decimation matrices

$$\overline{\mathbf{D}}_p = [\boldsymbol{\delta}_{n_{p,0}}^{(N_p - L_p + 1)} \; \boldsymbol{\delta}_{n_{p,1}}^{(N_p - L_p + 1)} \; \cdots \; \boldsymbol{\delta}_{n_{p,\bar{N}_p - 1}}^{(N_p - L_p + 1)}]^T, \tag{4.17}$$

$p \in \{\mathrm{t}, \mathrm{r}\}$. In particular, we choose

$$n_{p,i} = \tilde{n}_{p,i} M_p, \tag{4.18}$$

where M_p are some positive integers, and the integer sets

$$\mathcal{N}_p = \{\tilde{n}_{p,0}, \ldots, \tilde{n}_{p,\bar{N}_p - 1}\} \tag{4.19}$$

correspond to the antenna locations of some standard sparse arrays. Without loss of generality, let $n_{p,0} = 0$. In the following, we will show that by introducing M_p, we can get a pair of virtual transmit and receive sparse arrays, dilated by M_t and M_r times, respectively. Hence, there will be central sparse ULA segments with antenna spacings M_p in their difference coarrays [79]. (Details for coarrays are explained in Sec. 4.4.2.) These M_p times larger coarray apertures will give better DOD and DOA estimation performance, which will be observed in numerical examples in Sec. 4.8. Conventionally, the M_p-sparsity can create DOD and DOA ambiguity, but we can resolve it because only in-band paths remain after CBS filtering, as explained in Sec. 4.3. Note that coarray method need not be considered in uniform scheme, because the combiner output there resembles a ULA output instead of sparse array output.

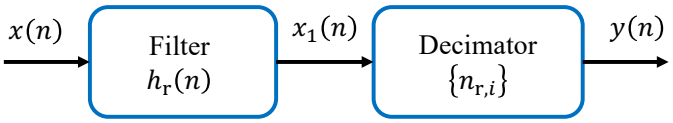


Figure 4.4: Equivalence of the overall nonuniform CBS combiner to a filter followed by a nonuniform combiner.

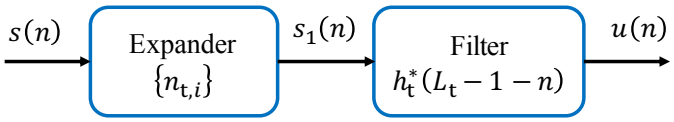


Figure 4.5: Equivalence of the overall nonuniform CBS precoder to a nonuniform expander followed by a filter.

Just like the uniform case, with the relevant matrices defined as in (4.7), (4.8), and (4.17), the overall combiner is equivalent to the filter $h_\mathrm{r}(n)$ followed by a nonuniform decimator, as in Fig. 4.4. The nonuniform decimator is defined by

$$y(i) = x_1(n_{\mathrm{r},i}) \tag{4.20}$$

for all integers i. For the precoder, we can show that the operation (4.11) is equivalent to a nonuniform expander followed by the filter $h_\mathrm{t}^*(L_\mathrm{t} - 1 - n)$, as in Fig. 4.5. The nonuniform expander is defined by

$$s_1(n) = \sum_{i=-\infty}^{\infty} s(i)\delta(n - n_{\mathrm{t},i}). \tag{4.21}$$

That is, $s_1(n_{\mathrm{t},i}) = s(i)$, and $s_1(n)$ is zero otherwise. For instance, at the transmitter, we consider a ULA with $N_\mathrm{t} = 17$ antennas and CBS filter length $L_\mathrm{t} = 5$, and

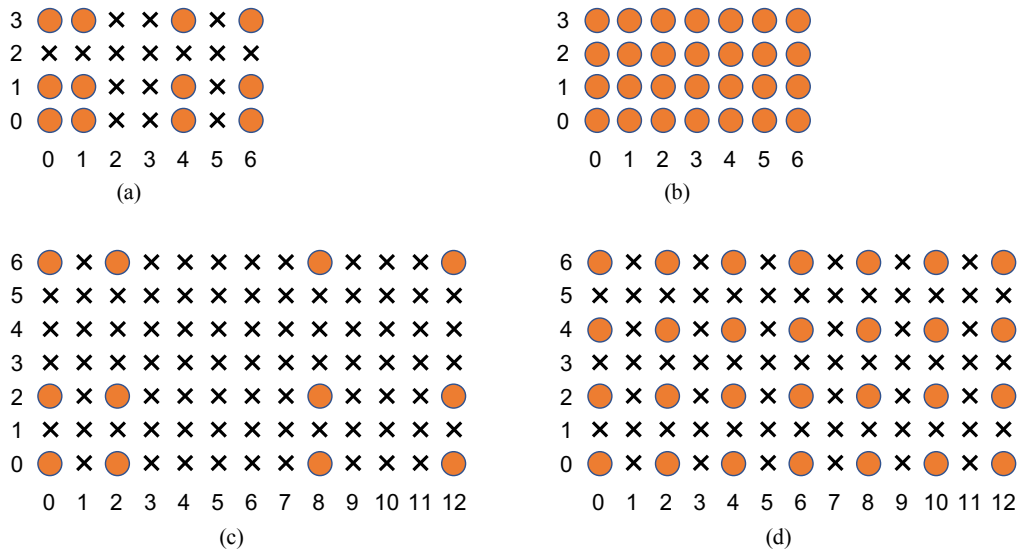


Figure 4.6: An example of nonuniform decimation for 2-D hybrid CBS. The 2-D arrays shown are virtual (not physical) and represent the sample locations we get if we consider the received block as a sum of 2-D complex sinusoids (see (4.28)). Filled circles represent antennas (sample locations), and crosses represent empty space. The array sizes are relatively small here just for ease of presentation. (a) A 2-D rectangular array $\mathcal{N}^{(2)} = \{(\tilde{n}_{\mathrm{t},i_\mathrm{t}}, \tilde{n}_{\mathrm{r},i_\mathrm{r}}) \mid \tilde{n}_{\mathrm{t},i_\mathrm{t}} \in \mathcal{N}_\mathrm{t}, \tilde{n}_{\mathrm{r},i_\mathrm{r}} \in \mathcal{N}_\mathrm{r}\}$ constructed from a 4-antenna transmit MRA $\mathcal{N}_\mathrm{t} = \{\tilde{n}_{\mathrm{t},0}, \ldots, \tilde{n}_{\mathrm{t},3}\} = \{0, 1, 4, 6\}$ and a 3-antenna receive MRA $\mathcal{N}_\mathrm{r} = \{\tilde{n}_{\mathrm{r},0}, \ldots, \tilde{n}_{\mathrm{t},2}\} = \{0, 1, 3\}$. (b) The 2-D difference coarray $\mathcal{C}^{(2)}$ of the 2-D rectangular array $\mathcal{N}^{(2)}$. (Only the nonnegative portion, i.e., quadrant I, is shown because coarrays are symmetric.) Note that $\mathcal{C}^{(2)}$ is a URA here. (c) The dilated rectangular array $\mathcal{N}^{(2)}_{M_\mathrm{t},M_\mathrm{r}} = \{(\tilde{n}'_{\mathrm{t},i_\mathrm{t}}, \tilde{n}'_{\mathrm{r},i_\mathrm{r}}) \mid \tilde{n}'_{\mathrm{t},i_\mathrm{t}} \in \mathcal{N}_{\mathrm{t},M_\mathrm{t}}, \tilde{n}'_{\mathrm{r},i_\mathrm{r}} \in \mathcal{N}_{\mathrm{r},M_\mathrm{r}}\}$ constructed from the dilated 4-antenna transmit MRA $\mathcal{N}_{\mathrm{t},M_\mathrm{t}} = \{M_\mathrm{t}\tilde{n}_{\mathrm{t},0}, \ldots, M_\mathrm{t}\tilde{n}_{\mathrm{t},3}\} = \{0, 2, 8, 12\}$ and the dilated 3-antenna receive MRA $\mathcal{N}_{\mathrm{r},M_\mathrm{r}} = \{M_\mathrm{r}\tilde{n}_{\mathrm{r},0}, \ldots, M_\mathrm{r}\tilde{n}_{\mathrm{r},2}\} = \{0, 2, 6\}$, where $M_\mathrm{t} = M_\mathrm{r} = 2$. (d) The 2-D difference coarray $\mathcal{C}^{(2)}_{M_\mathrm{t},M_\mathrm{r}}$ of the dilated rectangular array $\mathcal{N}^{(2)}_{M_\mathrm{t},M_\mathrm{r}}$. Note that $\mathcal{C}^{(2)}_{M_\mathrm{t},M_\mathrm{r}}$ is a sparse URA of antenna spacings $(M_\mathrm{t}, M_\mathrm{r}) = (2, 2)$, and it is equivalent to $\mathcal{C}^{(2)}$ dilated by M_t and M_r in the two directions.

$\bar{N}_\mathrm{t} = 4$ samples are expanded to locations corresponding to a 4-antenna MRA $\mathcal{N}_\mathrm{t} = \{\tilde{n}_{\mathrm{t},0}, \ldots, \tilde{n}_{\mathrm{t},3}\} = \{0, 1, 4, 6\}$ dilated by $M_\mathrm{t} = 2$. In this case, we have the 4×13 expansion matrix

$$\overline{\mathbf{D}}_\mathrm{t}^T = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}^T \tag{4.22}$$

so that the expanded samples correspond to the dilated 4-antenna MRA $\mathcal{N}_{\mathrm{t},M_\mathrm{t}} = \{M_\mathrm{t}\tilde{n}_{\mathrm{t},0}, \ldots, M_\mathrm{t}\tilde{n}_{\mathrm{t},3}\} = \{0, 2, 8, 12\}$. Meanwhile, at the receiver, we consider a ULA

with $N_{\rm r} = 11$ antennas and CBS filter length $L_{\rm r} = 5$, and we do decimation to retain $\bar{N}_{\rm r} = 3$ samples corresponding to a 3-antenna MRA $\mathcal{N}_{\rm r} = \{\tilde{n}_{{\rm r},0}, \ldots, \tilde{n}_{{\rm r},2}\} = \{0, 1, 3\}$ dilated by $M_{\rm r} = 2$. In this case, we have the 3×7 decimation matrix

$$\overline{\mathbf{D}}_{\rm r} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.23}$$

so that the decimated samples correspond to the dilated 3-antenna MRA $\mathcal{N}_{{\rm r},M_{\rm r}} = \{M_{\rm r}\tilde{n}_{{\rm r},0}, \ldots, M_{\rm r}\tilde{n}_{{\rm r},2}\} = \{0, 2, 6\}$. See Fig. 4.6 for demonstration and visualization. The 2-D arrays shown in Fig. 4.6 are constructed from 1-D transmit and receive arrays. The physical meaning of these 2-D arrays will be explained in Sec. 4.4.2. The array sizes are relatively small in the above example just for ease of presentation.

Applying the nonuniform decimation matrices (4.17), we obtain the kth received block

$$\mathbf{Y}_k = \overline{\mathbf{D}}_{\rm r}\mathbf{H}_{\rm r}\mathbf{H}_{\alpha,k}\mathbf{H}_{\rm t}^H\overline{\mathbf{D}}_{\rm t}^H\mathbf{S}_k + \overline{\mathbf{D}}_{\rm r}\mathbf{H}_{\rm r}\mathbf{E}_k \in \mathbb{C}^{\bar{N}_{\rm r}\times\bar{N}_{\rm t}}. \tag{4.24}$$

Again, as $\mathbf{S}_k$ is identity, each column of $\mathbf{S}_k$ produces a shifted version of the filter as beamformer, but these shifts are now rendered nonuniform. Using (4.1) and (4.5), we can show that this simplifies to

$$\mathbf{Y}_k = \overline{\mathbf{A}}_{\rm r}\widetilde{\mathbf{D}}_k\overline{\mathbf{A}}_{\rm t}^H + \overline{\mathbf{D}}_{\rm r}\mathbf{H}_{\rm r}\mathbf{E}_k, \tag{4.25}$$

where

$$\overline{\mathbf{A}}_p = [\bar{\mathbf{a}}_p(M_p\omega_{p,1}) \ \cdots \ \bar{\mathbf{a}}_p(M_p\omega_{p,D})] \tag{4.26}$$

with $\bar{\mathbf{a}}_p(\omega) = [1 \ e^{j\omega\tilde{n}_{p,1}} \ e^{j\omega\tilde{n}_{p,2}} \ \cdots \ e^{j\omega\tilde{n}_{p,\bar{N}_p-1}}]^T$, $p \in \{{\rm t}, {\rm r}\}$, and $\widetilde{\mathbf{D}}_k$ is diagonal with diagonal entries as in (4.15). From (4.25), we see that $\mathbf{Y}_k$ can be viewed as a received block obtained from a pair of virtual transmit sparse array $\mathcal{N}_{\rm t}$ and receive sparse array $\mathcal{N}_{\rm r}$ (Item 2 in the beginning of this section) due to paths with filtered gains $\alpha_{k,i}H_{\rm r}(e^{j\omega_{r,i}})H_{\rm t}^*(e^{j\omega_{t,i}})$ (ignoring the unimportant phase shifts), DODs $M_{\rm t}\omega_{{\rm t},i}$, and DOAs $M_{\rm r}\omega_{{\rm r},i}$. We note that ambiguity due to M_p can be resolved as explained in Sec. 4.3. Equivalently, $\mathbf{Y}_k$ resembles a received block obtained from a pair of virtual transmit dilated sparse array $\mathcal{N}_{{\rm t},M_{\rm t}}$ and receive dilated sparse array $\mathcal{N}_{{\rm r},M_{\rm r}}$ (Item 3 in the beginning of this section) due to DODs $\omega_{{\rm t},i}$ and DOAs $\omega_{{\rm r},i}$. Notice that *the two dilated sparse arrays merely represent the spatial coordinates in the beamspace domain where RF chains are deployed.*

4.4.2 Coarray Method

The (difference) coarray method [32] can be used to estimate the DODs and DOAs in our nonuniform scheme. In the coarray method, second-order statistics of received data on a sparse array $\mathcal{N}$ is used to construct a signal on its corray $\mathcal{C} = \{k - l \mid k, l \in \mathcal{N}\}$. The aperture of $\mathcal{C}$, which equals the aperture of $\mathcal{N}$, can be much larger than $|\mathcal{N}|$, so typically, $\mathcal{C}$ has a large central ULA segment. In fact, the size of the central ULA segment is often as large as $O(|\mathcal{N}|^2)$. Thus, we can get more accurate DOD and DOA estimates, which is the main advantage of our nonuniform scheme. Though not our main focus, another benefit is that the number of identifiable paths is increased.

In particular, the 2-D coarray method [5] is applied. To begin with, we consider the vectorized received block

$$\mathbf{y}_k = \mathrm{vec}(\mathbf{Y}_k) = (\overline{\mathbf{A}}_{\mathrm{t}}^{*} \odot \overline{\mathbf{A}}_{\mathrm{r}})\tilde{\mathbf{d}}_k + \mathbf{e}_k \in \mathbb{C}^{\bar{N}_{\mathrm{t}}\bar{N}_{\mathrm{r}}}, \tag{4.27}$$

where $\mathbf{e}_k = \mathrm{vec}(\overline{\mathbf{D}}_{\mathrm{r}}\mathbf{H}_{\mathrm{r}}\mathbf{E}_k)$ and $\tilde{\mathbf{d}}_k$ has entries $[\tilde{\mathbf{d}}_k]_i = [\widetilde{\mathbf{D}}_k]_{i,i}$. To aid understanding of the method overall, we note that $\mathbf{y}_k$ has entries (ignoring noise)

$$\begin{aligned}[\mathbf{y}_k]_l = [\mathbf{Y}_k]_{i_r,i_t} &= \sum_{i=1}^{D} [\tilde{\mathbf{d}}_k]_i e^{j(-M_{\mathrm{t}}\omega_{\mathrm{t},i}\tilde{n}_{\mathrm{t},i_{\mathrm{t}}} + M_{\mathrm{r}}\omega_{\mathrm{r},i}\tilde{n}_{\mathrm{r},i_{\mathrm{r}}})} \\ &\triangleq y_{k,M_{\mathrm{t}},M_{\mathrm{r}}}(\tilde{n}_{\mathrm{t},i_{\mathrm{t}}}, \tilde{n}_{\mathrm{r},i_{\mathrm{r}}})\end{aligned} \tag{4.28}$$

with $l = \bar{N}_{\mathrm{r}} i_{\mathrm{t}} + i_{\mathrm{r}}$ for $0 \le i_{\mathrm{t}} < \bar{N}_{\mathrm{t}}$ and $0 \le i_{\mathrm{r}} < \bar{N}_{\mathrm{r}}$. Here, $y_{k,M_{\mathrm{t}},M_{\mathrm{r}}}(\tilde{n}_{\mathrm{t},i_{\mathrm{t}}}, \tilde{n}_{\mathrm{r},i_{\mathrm{r}}})$ is the sum of D complex sinusoids with 2-D frequencies $[-M_{\mathrm{t}}\omega_{\mathrm{t},i}\ M_{\mathrm{r}}\omega_{\mathrm{r},i}]^T$, and we have those samples evaluated at the locations of the 2-D sparse rectangular array

$$\mathcal{N}^{(2)} = \{(\tilde{n}_{\mathrm{t},i_{\mathrm{t}}}, \tilde{n}_{\mathrm{r},i_{\mathrm{r}}}) \mid \tilde{n}_{\mathrm{t},i_{\mathrm{t}}} \in \mathcal{N}_{\mathrm{t}}, \tilde{n}_{\mathrm{r},i_{\mathrm{r}}} \in \mathcal{N}_{\mathrm{r}}\}. \tag{4.29}$$

See Fig. 4.6(a) for an example. Equivalently, we can view $y_{k,M_{\mathrm{t}},M_{\mathrm{r}}}(\tilde{n}_{\mathrm{t},i_{\mathrm{t}}}, \tilde{n}_{\mathrm{r},i_{\mathrm{r}}})$ as the sum of complex sinusoids with 2-D frequencies $[-\omega_{\mathrm{t},i}\ \omega_{\mathrm{r},i}]^T$, and we have those samples evaluated at the locations of the dilated sparse rectangular array

$$\mathcal{N}^{(2)}_{M_{\mathrm{t}},M_{\mathrm{r}}} = \{(\tilde{n}'_{\mathrm{t},i_{\mathrm{t}}}, \tilde{n}'_{\mathrm{r},i_{\mathrm{r}}}) \mid \tilde{n}'_{\mathrm{t},i_{\mathrm{t}}} \in \mathcal{N}_{\mathrm{t},M_{\mathrm{t}}}, \tilde{n}'_{\mathrm{r},i_{\mathrm{r}}} \in \mathcal{N}_{\mathrm{r},M_{\mathrm{r}}}\}.$$

See Fig. 4.6(c) for an example. Then we compute the covariance

$$\mathbf{R}_y = \mathrm{E}[\mathbf{y}_k\mathbf{y}_k^H] = (\overline{\mathbf{A}}_{\mathrm{t}}^{*} \odot \overline{\mathbf{A}}_{\mathrm{r}})\widetilde{\mathbf{R}}_d(\overline{\mathbf{A}}_{\mathrm{t}}^{*} \odot \overline{\mathbf{A}}_{\mathrm{r}})^H + \mathbf{R}_{\mathrm{e}}, \tag{4.30}$$

where $\widetilde{\mathbf{R}}_d = \mathrm{E}[\tilde{\mathbf{d}}_k\tilde{\mathbf{d}}_k^H]$ has entries

$$\begin{aligned}[\widetilde{\mathbf{R}}_d]_{m,l} = \sigma_{\mathrm{s}}^2\rho_{m,l}H_{\mathrm{r}}(e^{j\omega_{\mathrm{r},m}})H_{\mathrm{r}}^{*}(e^{j\omega_{\mathrm{r},l}})e^{j(\omega_{\mathrm{r},m}-\omega_{\mathrm{r},l})(L_{\mathrm{r}}-1)} \\ \cdot H_{\mathrm{t}}^{*}(e^{j\omega_{\mathrm{t},m}})H_{\mathrm{t}}(e^{j\omega_{\mathrm{t},l}})e^{j(\omega_{\mathrm{t},l}-\omega_{\mathrm{t},m})(L_{\mathrm{t}}-1)}\end{aligned} \tag{4.31}$$

and

$$\mathbf{R}_{\mathrm{e}} = \mathrm{E}[\mathbf{e}_k \mathbf{e}_k^H] = \mathrm{diag}(\overline{\mathbf{R}}_{\mathrm{e}}, \overline{\mathbf{R}}_{\mathrm{e}}, \ldots, \overline{\mathbf{R}}_{\mathrm{e}}) \in \mathbb{C}^{\bar{N}_{\mathrm{t}}\bar{N}_{\mathrm{r}} \times \bar{N}_{\mathrm{t}}\bar{N}_{\mathrm{r}}} \tag{4.32}$$

with

$$\overline{\mathbf{R}}_{\mathrm{e}} = \sigma_{\mathrm{e}}^2 \overline{\mathbf{D}}_{\mathrm{r}} \mathbf{H}_{\mathrm{r}} \mathbf{H}_{\mathrm{r}}^H \overline{\mathbf{D}}_{\mathrm{r}}^H \in \mathbb{C}^{\bar{N}_{\mathrm{r}} \times \bar{N}_{\mathrm{r}}}. \tag{4.33}$$

We can show that $\overline{\mathbf{R}}_{\mathrm{e}}$ has entries

$$[\overline{\mathbf{R}}_{\mathrm{e}}]_{m,l} = \sigma_{\mathrm{e}}^2 g_{\mathrm{r}}((\tilde{n}_{\mathrm{r},m} - \tilde{n}_{\mathrm{r},l}) M_{\mathrm{r}}), \tag{4.34}$$

where

$$g_{\mathrm{r}}(k) = \sum_n h_{\mathrm{r}}(n) h_{\mathrm{r}}^*(n-k) \tag{4.35}$$

is the deterministic autocorrelation of $h_{\mathrm{r}}(n)$. Again, we can whiten the noise term by letting the CBS filter $H_{\mathrm{r}}(z)$ be a spectral factor of a Nyquist(M_{r}) filter (see Sec. 2.2.4). This makes $g_{\mathrm{r}}(M_{\mathrm{r}}k) = \delta(k)$ for all integers k and thus $\mathbf{R}_{\mathrm{e}} = \sigma_{\mathrm{e}}^2 \mathbf{I}$. Now we assume the path gains are uncorrelated so that $\rho_{m,l} = 0$ for all $m \neq l$ in (4.3). (If path gains are correlated, we can use the decorrelation method of Sec. 4.7.) This implies that $\widetilde{\mathbf{R}}_d$ is diagonal.

Then we can consider the vectorization

$$\mathbf{r}_y \triangleq \mathrm{vec}(\mathbf{R}_y) = (\overline{\mathbf{A}}_{\mathrm{t}} \odot \overline{\mathbf{A}}_{\mathrm{r}}^* \odot \overline{\mathbf{A}}_{\mathrm{t}}^* \odot \overline{\mathbf{A}}_{\mathrm{r}}) \tilde{\mathbf{d}} + \sigma_{\mathrm{e}}^2 \mathrm{vec}(\mathbf{I}_{\bar{N}_{\mathrm{t}}\bar{N}_{\mathrm{r}}}), \tag{4.36}$$

where $\tilde{\mathbf{d}}$ has entries $[\tilde{\mathbf{d}}]_i = [\widetilde{\mathbf{R}}_d]_{i,i}$. This kind of Khatri–Rao product of 4 manifold matrices often appears when we vectorize the covariance of a 2-D measurement, e.g., as in [104]. Considering the i_1th, i_2th, i_3th, and i_4th rows of $\overline{\mathbf{A}}_{\mathrm{t}}$, $\overline{\mathbf{A}}_{\mathrm{r}}^*$, $\overline{\mathbf{A}}_{\mathrm{t}}^*$, and $\overline{\mathbf{A}}_{\mathrm{r}}$, respectively in the Khatri–Rao product, we have that

$$[\mathbf{r}_y]_l = \sum_{i=1}^{D} [\tilde{\mathbf{d}}]_i e^{j[M_{\mathrm{t}}\omega_{\mathrm{t},i}(\tilde{n}_{\mathrm{t},i_1} - \tilde{n}_{\mathrm{t},i_3}) + M_{\mathrm{r}}\omega_{\mathrm{r},i}(\tilde{n}_{\mathrm{r},i_4} - \tilde{n}_{\mathrm{r},i_2})]} + \sigma_{\mathrm{e}}^2 [\mathbf{I}_{\bar{N}_{\mathrm{t}}\bar{N}_{\mathrm{r}}}]_{\bar{N}_{\mathrm{r}} i_1 + i_2, \bar{N}_{\mathrm{r}} i_3 + i_4} \tag{4.37}$$

with

$$l = \bar{N}_{\mathrm{r}}(\bar{N}_{\mathrm{t}}(\bar{N}_{\mathrm{r}} i_1 + i_2) + i_3) + i_4 \tag{4.38}$$

$$= \bar{N}_{\mathrm{r}}\bar{N}_{\mathrm{t}}(\bar{N}_{\mathrm{r}} i_1 + i_2) + (\bar{N}_{\mathrm{r}} i_3 + i_4) \tag{4.39}$$

for $0 \leq i_1, i_3 < \bar{N}_{\mathrm{t}}$ and $0 \leq i_2, i_4 < \bar{N}_{\mathrm{r}}$. We observe that $[\mathbf{I}_{\bar{N}_{\mathrm{t}}\bar{N}_{\mathrm{r}}}]_{\bar{N}_{\mathrm{r}} i_1 + i_2, \bar{N}_{\mathrm{r}} i_3 + i_4}$ is nonzero only if

$$\bar{N}_{\mathrm{r}} i_1 + i_2 = \bar{N}_{\mathrm{r}} i_3 + i_4 \;\Leftrightarrow\; i_1 = i_3, i_2 = i_4. \tag{4.40}$$

Thus, we can rewrite (4.37) as

$$[\mathbf{r}_y]_l = \sum_{i=1}^{D} [\tilde{\mathbf{d}}]_i e^{j[M_\mathrm{t}\omega_{\mathrm{t},i}(\tilde{n}_{\mathrm{t},i_1}-\tilde{n}_{\mathrm{t},i_3})+M_\mathrm{r}\omega_{\mathrm{r},i}(\tilde{n}_{\mathrm{r},i_4}-\tilde{n}_{\mathrm{r},i_2})]} + \sigma_\mathrm{e}^2\delta(\tilde{n}_{\mathrm{t},i_1}-\tilde{n}_{\mathrm{t},i_3})\delta(\tilde{n}_{\mathrm{r},i_4}-\tilde{n}_{\mathrm{r},i_2}) \tag{4.41}$$

$$\triangleq R_{M_\mathrm{t},M_\mathrm{r}}(\tilde{n}_{\mathrm{t},i_1}-\tilde{n}_{\mathrm{t},i_3}, \tilde{n}_{\mathrm{r},i_4}-\tilde{n}_{\mathrm{r},i_2}) \tag{4.42}$$

since it depends only on the differences $\tilde{n}_{\mathrm{t},i_1} - \tilde{n}_{\mathrm{t},i_3}$ between the virtual transmit antenna locations and the differences $\tilde{n}_{\mathrm{r},i_4} - \tilde{n}_{\mathrm{r},i_2}$ between the virtual receive antenna locations. Suppose the coarrays C_p (Item 4 in the beginning of this section) have central ULA segments $-(U_p-1) \le k_p \le U_p - 1$, $p \in \{\mathrm{t},\mathrm{r}\}$. Then in practice, we can estimate

$$R_{M_\mathrm{t},M_\mathrm{r}}(k_\mathrm{t},k_\mathrm{r}) = \sum_{i=1}^{D} [\tilde{\mathbf{d}}]_i e^{j(M_\mathrm{t}\omega_{\mathrm{t},i}k_\mathrm{t}+M_\mathrm{r}\omega_{\mathrm{r},i}k_\mathrm{r})} + \sigma_\mathrm{e}^2\delta(k_\mathrm{t})\delta(k_\mathrm{r}) \tag{4.43}$$

for all $k_\mathrm{t} \in C_\mathrm{t}$ and $k_\mathrm{r} \in C_\mathrm{r}$ by averaging $[\mathbf{r}_y]_l$ (with l as in (4.38)) over snapshots, and over all $\tilde{n}_{\mathrm{t},i_1} - \tilde{n}_{\mathrm{t},i_3}$ and $\tilde{n}_{\mathrm{r},i_4} - \tilde{n}_{\mathrm{r},i_2}$ that produce identical differences $k_\mathrm{t} = \tilde{n}_{\mathrm{t},i_1} - \tilde{n}_{\mathrm{t},i_3}$ and $k_\mathrm{r} = \tilde{n}_{\mathrm{r},i_4} - \tilde{n}_{\mathrm{r},i_2}$, respectively. We can see the connection between (4.43) and (4.28). To be precise, $R_{M_\mathrm{t},M_\mathrm{r}}(k_\mathrm{t},k_\mathrm{r})$ can be regarded as a 2-D correlation function of the sum of complex sinusoids with 2-D frequencies $[M_\mathrm{t}\omega_{\mathrm{t},i}\ M_\mathrm{r}\omega_{\mathrm{r},i}]^T$, and we have those samples evaluated at the locations of the 2-D coarray

$$C^{(2)} = \{(k_\mathrm{t},k_\mathrm{r}) \mid k_\mathrm{t} \in C_\mathrm{t}, k_\mathrm{r} \in C_\mathrm{r}\}. \tag{4.44}$$

See Fig. 4.6(b) for an example. Equivalently, we can view $R_{M_\mathrm{t},M_\mathrm{r}}(k_\mathrm{t},k_\mathrm{r})$ as a 2-D correlation function $R(k'_\mathrm{t},k'_\mathrm{r})$ of the sum of complex sinusoids with 2-D frequencies $[\omega_{\mathrm{t},i}\ \omega_{\mathrm{r},i}]^T$, and we have those samples evaluated at the locations of the 2-D dilated coarray

$$C^{(2)}_{M_\mathrm{t},M_\mathrm{r}} = \{(k'_\mathrm{t},k'_\mathrm{r}) \mid k'_\mathrm{t} \in C_{\mathrm{t},M_\mathrm{t}}, k'_\mathrm{r} \in C_{\mathrm{r},M_\mathrm{r}}\}, \tag{4.45}$$

where $C_{p,M}$ are as in Item 5 in the beginning of this section, $p \in \{\mathrm{t},\mathrm{r}\}$. That is,

$$R_{M_\mathrm{t},M_\mathrm{r}}(k_\mathrm{t},k_\mathrm{r}) = R(M_\mathrm{t}k_\mathrm{t}, M_\mathrm{r}k_\mathrm{r}), \tag{4.46}$$

where

$$R(k'_\mathrm{t},k'_\mathrm{r}) = \sum_{i=1}^{D} [\tilde{\mathbf{d}}]_i e^{j(\omega_{\mathrm{t},i}k'_\mathrm{t}+\omega_{\mathrm{r},i}k'_\mathrm{r})} + \sigma_\mathrm{e}^2\delta(k'_\mathrm{t})\delta(k'_\mathrm{r}). \tag{4.47}$$

See Fig. 4.6(d) for an example. When $\mathcal{N}_p$, $p \in \{\mathrm{t},\mathrm{r}\}$ are standard sparse arrays like MRAs [3], nested arrays [32], or coprime arrays [40], $C^{(2)}$ will have a large central

uniform rectangular array (URA) segment $\{(k_\mathrm{t}, k_\mathrm{r}) \mid -(U_p - 1) \le k_p \le U_p - 1, p \in \{\mathrm{t}, \mathrm{r}\}\}$ with $U_p = O(\bar{N}_p^2)$. Likewise, $C^{(2)}_{M_\mathrm{t}, M_\mathrm{r}}$ will have a large central sparse URA segment $\{(M_\mathrm{t}k_\mathrm{t}, M_\mathrm{r}k_\mathrm{r}) \mid -(U_p - 1) \le k_p \le U_p - 1, p \in \{\mathrm{t}, \mathrm{r}\}\}$. This $M_\mathrm{t} \times M_\mathrm{r}$ times dilation of aperture can enhance the estimation performance, which we shall see in simulations.

From the estimates $R_{M_\mathrm{t}, M_\mathrm{r}}(k_\mathrm{t}, k_\mathrm{r})$, we can let

$$[\tilde{\mathbf{r}}_y]_{(k_\mathrm{t}+U_\mathrm{t}-1)(2U_\mathrm{r}-1)+k_\mathrm{r}+U_\mathrm{r}-1} = R_{M_\mathrm{t}, M_\mathrm{r}}(k_\mathrm{t}, k_\mathrm{r}) \tag{4.48}$$

for $-(U_p - 1) \le k_p \le U_p - 1$, $p \in \{\mathrm{t}, \mathrm{r}\}$ and construct

$$\tilde{\mathbf{r}}_y = (\widetilde{\mathbf{A}}_\mathrm{t} \odot \widetilde{\mathbf{A}}_\mathrm{r})\tilde{\mathbf{d}} + \mathbf{q} \in \mathbb{C}^{(2U_\mathrm{t}-1)(2U_\mathrm{r}-1)}, \tag{4.49}$$

where

$$\widetilde{\mathbf{A}}_p = [\tilde{\mathbf{a}}_p(M_p\omega_{p,1}) \ \cdots \ \tilde{\mathbf{a}}_p(M_p\omega_{p,D})] \tag{4.50}$$

with

$$\tilde{\mathbf{a}}_p(\omega) = [e^{-j\omega(U_p-1)} \ \cdots \ e^{-j\omega} \ 1 \ \ e^{j\omega} \ \cdots \ e^{j\omega(U_p-1)}]^T \tag{4.51}$$

for $p \in \{\mathrm{t}, \mathrm{r}\}$ and $\mathbf{q} = [\mathbf{0}_{1\times w} \ \sigma_\mathrm{e}^2 \ \mathbf{0}_{1\times w}]^T$ with $w = ((2U_\mathrm{t} - 1)(2U_\mathrm{r} - 1) - 1)/2$. By comparing (4.49) to (4.27), we see that $\tilde{\mathbf{r}}_y$ resembles a vectorized received block obtained from a pair of $(2U_\mathrm{t} - 1)$-antenna transmit ULA and $(2U_\mathrm{r} - 1)$-antenna receive ULA due to DODs $M_\mathrm{t}\omega_{\mathrm{t},i}$ and DOAs $M_\mathrm{r}\omega_{\mathrm{r},i}$. In the above derivations, we offer an intuitive way of understanding the construction of the 2-D coarray signal (4.49) via the 2-D correlation function (4.43). This viewpoint is different from [5], where a 2-D coarray signal is constructed only via applying selection matrices. Our derivation above provides a different insight about the construction and physical meaning of (4.49).

Since we only have a single time block (snapshot) $\tilde{\mathbf{r}}_y$, to estimate the DODs and DOAs using subspace methods, we apply the method in [5], which is the 2-D extension to the improved spatial smoothing method in [79]. We consider row selection matrices

$$\mathbf{J}_{p,i_p} = [\mathbf{0}_{U_p\times(U_p-i_p)} \ \ \mathbf{I}_{U_p} \ \ \mathbf{0}_{U_p\times(i_p-1)}] \tag{4.52}$$

and let

$$\mathbf{x}_{i_\mathrm{t},i_\mathrm{r}} = (\mathbf{J}_{\mathrm{t},i_\mathrm{t}} \otimes \mathbf{J}_{\mathrm{r},i_\mathrm{r}})\tilde{\mathbf{r}}_y \tag{4.53}$$

for $1 \le i_p \le U_p$, $p \in \{\mathrm{t}, \mathrm{r}\}$. Then we can show that the spatially smoothed matrix

$$\mathbf{Y}_{\mathrm{ss}} \triangleq [\mathbf{x}_{1,1}\ \mathbf{x}_{1,2}\ \cdots\ \mathbf{x}_{1,U_\mathrm{r}}\ \mathbf{x}_{2,1}\ \cdots\ \mathbf{x}_{U_\mathrm{t},U_\mathrm{r}}]^T \tag{4.54}$$

$$= (\mathbf{A}'_\mathrm{t} \odot \mathbf{A}'_\mathrm{r})\widetilde{\mathbf{R}}_d(\mathbf{A}'_\mathrm{t} \odot \mathbf{A}'_\mathrm{r})^H + \sigma_\mathrm{e}^2\mathbf{I}_{U_\mathrm{t}U_\mathrm{r}}, \tag{4.55}$$

where $\widetilde{\mathbf{R}}_d$ is diagonal with $[\widetilde{\mathbf{R}}_d]_{i,i} = [\tilde{\mathbf{d}}]_i$, and

$$\mathbf{A}'_p = [\mathbf{a}_{U_p}(M_p\omega_{p,1})\ \cdots\ \mathbf{a}_{U_p}(M_p\omega_{p,D})] \tag{4.56}$$

with $\mathbf{a}_N(\omega)$ as defined in (4.2). Again, if we assume that the first D_0 paths are in-band paths, we obtain

$$\mathbf{Y}_{\mathrm{ss}} \approx (\mathbf{A}'_{\mathrm{t,IB}} \odot \mathbf{A}'_{\mathrm{r,IB}})\widetilde{\mathbf{R}}_{d,\mathrm{IB}}(\mathbf{A}'_{\mathrm{t,IB}} \odot \mathbf{A}'_{\mathrm{r,IB}})^H + \sigma_\mathrm{e}^2\mathbf{I}_{U_\mathrm{t}U_\mathrm{r}},$$

where $\mathbf{A}'_{p,\mathrm{IB}}$ has the first D_0 columns of $\mathbf{A}'_p$, and $\widetilde{\mathbf{R}}_{d,\mathrm{IB}}$ is the top left $D_0 \times D_0$ subblock of $\widetilde{\mathbf{R}}_d$. We note that $\mathbf{Y}_{\mathrm{ss}}$ can be viewed as the covariance of vectorized received blocks obtained from a pair of U_t-antenna transmit ULA and U_r-antenna receive ULA due to DODs $M_\mathrm{t}\omega_{\mathrm{t},i}$ and DOAs $M_\mathrm{r}\omega_{\mathrm{r},i}$, and buried in white noise. (Again, ambiguity due to M_p can be resolved as explained in Sec. 4.3.) Equivalently, $\mathbf{Y}_{\mathrm{ss}}$ is like the covariance of vectorized received blocks obtained from a pair of U_t-antenna transmit sparse ULA (with antenna spacing M_t) and U_r-antenna receive sparse ULA (with antenna spacing M_r) due to DODs $\omega_{\mathrm{t},i}$ and DOAs $\omega_{\mathrm{r},i}$. Hence, the effective array apertures are enlarged by M_t and M_r times. Then, standard methods like 2-D unitary ESPRIT [51] can be readily used to estimate the DODs and DOAs. At most $\min(U_\mathrm{t}(U_\mathrm{r}-1), U_\mathrm{r}(U_\mathrm{t}-1))$ paths can be identified using 2-D unitary ESPRIT [51]. Thus, $O(U_\mathrm{t}U_\mathrm{r}) = O(\bar{N}_\mathrm{t}^2\bar{N}_\mathrm{r}^2)$ paths can be estimated theoretically. However, we are not aiming to estimate a very large number of DODs and DOAs here. Given a fixed number of RF chains and decimation ratio, the coarray aperture of nonuniform scheme can be much larger than the array aperture of uniform scheme (i.e., $U_p \gg \bar{N}_p$). Thus, the main benefit is that we can get higher DOD and DOA resolution and smaller estimation errors, as we shall see in simulations (Fig. 4.9). Moreover, this can be achieved for a significant number of paths with only a small number of RF chains.

Note that in (4.24), $\overline{\mathbf{D}}_p\mathbf{H}_p$, $p \in \{\mathrm{t}, \mathrm{r}\}$, are matrices obtained from nonuniformly decimating the rows of banded Toeplitz matrices. Such way of designing the precoder and combiner also appears in [5], but therein the coefficients $h_p(n)$ in $\mathbf{H}_p$ are not designed for filtering DODs and DOAs but for decorrelating path gains. Hence, they can only choose $n_{p,i} = \tilde{n}_{p,i}$ (i.e., $M_p = 1$) for the expanded and

decimated samples. Otherwise, there will be DOD and DOA ambiguities. We can therefore achieve M_p times larger effective aperture of the virtual sparse arrays than [5] given a fixed number of RF chains. This leads to smaller estimation errors, as shown in simulations (Sec. 4.8).

4.4.3 Channel Matrix Estimation

As described in the end of Sec. 4.3, to cover the full DOD and DOA ranges, we use a transmit filter bank with M_t filters and a receive filter bank with M_r filters, $p \in \{\mathrm{t}, \mathrm{r}\}$. For each pair of the transmit and receive filters, (4.27) can be approximated by

$$\mathbf{y}_k = (\overline{\mathbf{A}}_\mathrm{t}^* \odot \overline{\mathbf{A}}_\mathrm{r})\tilde{\mathbf{d}}_k + \mathbf{e}_k \tag{4.57}$$

$$\approx (\overline{\mathbf{A}}_{\mathrm{t,IB}}^* \odot \overline{\mathbf{A}}_{\mathrm{r,IB}})\tilde{\mathbf{d}}_{k,\mathrm{IB}} + \mathbf{e}_k \tag{4.58}$$

where $\overline{\mathbf{A}}_{p,\mathrm{IB}}$ has the columns of $\overline{\mathbf{A}}_p$ corresponding to the D_0 in-band DODs (or DOAs), and $\tilde{\mathbf{d}}_{k,\mathrm{IB}}$ has the corresponding entries of $\tilde{\mathbf{d}}_k$. After we obtain the in-band DOD and DOA estimates, we can construct estimated $\overline{\mathbf{A}}_{\mathrm{t,IB}}$ and $\overline{\mathbf{A}}_{\mathrm{r,IB}}$ and find the least-squares estimates [19] of the in-band path gains $\tilde{\mathbf{d}}_{k,\mathrm{IB}}$ based on (4.58) as long as $D_0 < \bar{N}_\mathrm{t}\bar{N}_\mathrm{r}$. After obtaining the DODs, DOAs, and path gains for in-band paths for all pairs of the transmit and receive filters, we can estimate the channel matrix $\mathbf{H}_{\alpha,k}$ via (4.1). Hence, if the DODs and DOAs are not known to be in some subbands a priori, the total number of training blocks required is $M_\mathrm{t}M_\mathrm{r}K$ to estimate the channel matrix, where K is the number of training blocks used for each pair of the transmit and receive filters.

4.5 Hybrid Analog and Digital Implementation of 2-D CBS

In the following, we describe the hybrid analog and digital implementation of CBS for mmWave MIMO channel estimation. We first consider the implementation of the receiver, while the transmitter for the mmWave MIMO system can be designed similarly since the precoder is the Hermitian transpose of a combiner. As shown in Fig. 4.1, the analog precoder $\mathbf{F}_{\mathrm{t,a}}^H$ and analog combiner $\mathbf{F}_{\mathrm{r,a}}$ are realized by phase shifter networks, so each entry of the matrices should have unit modulus. However, a digital filter that is properly designed does not have constant-modulus coefficients. To deal with this issue, we use the important result from [48, 96] as given in Lemma 3.1 in Sec. 3.3.

We first illustrate the implementation of CBS with nonuniform decimation and expansion. Consider CBS transmit and receive filters $h_p(n)$, $0 \le n \le L_p - 1$ and their vector forms $\mathbf{h}_p = [h_p(L_p - 1)\ h_p(L_p - 2)\ \cdots\ h_p(0)]$, $p \in \{\mathrm{t}, \mathrm{r}\}$. According

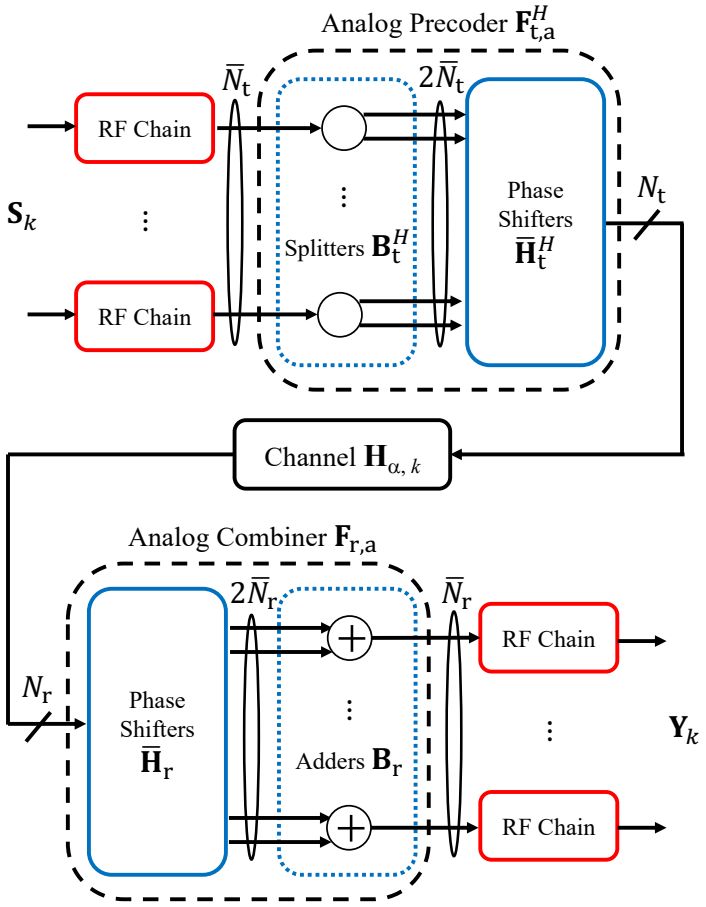


Figure 4.7: Detailed implementation of hybrid CBS for mmWave MIMO channel estimation. The digital precoder and digital combiner in Fig. 4.1 are not shown here as we design them to be identities. Typically, $N_\mathrm{t} \gg \bar{N}_\mathrm{t}$ and $N_\mathrm{r} \gg \bar{N}_\mathrm{r}$.

to Lemma 3.1, there exist $a_{p,1}, a_{p,2} \in \mathbb{C}$ and $\mathbf{h}_{p,1}, \mathbf{h}_{p,2} \in \mathbb{C}^{L_p}$ whose entries are unit-modulus such that

$$\mathbf{h}_p = a_{p,1}\mathbf{h}_{p,1} + a_{p,2}\mathbf{h}_{p,2}. \tag{4.59}$$

Hence, in (4.24), the overall precoder $\mathbf{F}_\mathrm{t}^H$ and overall combiner $\mathbf{F}_\mathrm{r}$ will satisfy

$$\mathbf{F}_p = \overline{\mathbf{D}}_p\mathbf{H}_p = \widetilde{\mathbf{H}}_p\overline{\mathbf{H}}_p \in \mathbb{C}^{\bar{N}_p \times N_p}, \tag{4.60}$$

where

$$\widetilde{\mathbf{H}}_p = \begin{bmatrix} a_{p,1} & a_{p,2} & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & a_{p,1} & a_{p,2} & \cdots & 0 & 0 \\ \vdots & \vdots & & & \ddots & & \\ 0 & 0 & 0 & 0 & \cdots & a_{p,1} & a_{p,2} \end{bmatrix} \tag{4.61}$$

are $\bar{N}_p \times 2\bar{N}_p$ matrices, and

$$\overline{\mathbf{H}}_p = [\overline{\mathbf{h}}_{p,1}^{(0)} \;\; \overline{\mathbf{h}}_{p,2}^{(0)} \;\; \overline{\mathbf{h}}_{p,1}^{(n_{p,1})} \; \overline{\mathbf{h}}_{p,2}^{(n_{p,1})} \cdots \overline{\mathbf{h}}_{p,1}^{(n_{p,\bar{N}_p-1})} \; \overline{\mathbf{h}}_{p,2}^{(n_{p,\bar{N}_p-1})}]^T$$

with $\overline{\mathbf{h}}_{p,m}^{(k)} = [\mathbf{0}_{1\times k} \;\; \mathbf{h}_{p,m}^T \;\; \mathbf{0}_{1\times(N_p-L_p-k)}]^T$ for $p \in \{\mathrm{t}, \mathrm{r}\}$, $m = 1, 2$, and nonnegative integers k. We can implement $\overline{\mathbf{H}}_p$ with $2\bar{N}_p L_p$ phase shifting operations because all the entries of $\mathbf{h}_{p,1}$ and $\mathbf{h}_{p,2}$ have unit modulus. For the general case where $|a_{p,1}| \neq |a_{p,2}|$, the decomposition $\mathbf{F}_p = \widetilde{\mathbf{H}}_p \overline{\mathbf{H}}_p$ implies that we require $2\bar{N}_\mathrm{t}$ and $2\bar{N}_\mathrm{r}$ RF chains at the transmitter and receiver. Alternatively, if we choose $a_{p,1} = a_{p,2}$, then

$$\widetilde{\mathbf{H}}_p = a_{p,1}\mathbf{B}_p, \tag{4.62}$$

where

$$\mathbf{B}_p = \mathrm{diag}([1 \;\; 1], [1 \;\; 1], \ldots, [1 \;\; 1]) \in \mathbb{C}^{\bar{N}_p \times 2\bar{N}_p} \tag{4.63}$$

for $p \in \{\mathrm{t}, \mathrm{r}\}$. We observe that $\mathbf{B}_\mathrm{t}^H$ and $\mathbf{B}_\mathrm{r}$ can be implemented with $\bar{N}_\mathrm{t}$ (1-to-2) splitters and with $\bar{N}_\mathrm{r}$ adders, respectively, in the analog domain (see Fig. 4.7). The constant scalars $a_{p,1}$ can be implemented in the digital precoder and combiner. Or, we may get rid of $a_{p,1}$ completely because the performance of standard methods like 2-D unitary ESPRIT [51] does not change when the received block is scaled by a constant. In summary, as shown in Fig. 4.7, the analog precoder is designed as

$$\mathbf{F}_{\mathrm{t,a}}^H = \overline{\mathbf{H}}_\mathrm{t}^H \mathbf{B}_\mathrm{t}^H \in \mathbb{C}^{N_\mathrm{t} \times \bar{N}_\mathrm{t}}, \tag{4.64}$$

realized with $2\bar{N}_\mathrm{t} L_\mathrm{t}$ phase shifting operations, and the analog combiner is designed as

$$\mathbf{F}_{\mathrm{r,a}} = \mathbf{B}_\mathrm{r} \overline{\mathbf{H}}_\mathrm{r} \in \mathbb{C}^{\bar{N}_\mathrm{r} \times N_\mathrm{r}}, \tag{4.65}$$

realized with $2\bar{N}_\mathrm{r} L_\mathrm{r}$ phase shifting operations, and we need only $\bar{N}_\mathrm{t}$ and $\bar{N}_\mathrm{r}$ RF chains at the transmitter and receiver, respectively. That is, the number of transmit RF chains equals the number of transmitted symbols in a training vector, and the number of receive RF chains equals the dimension of the decimated CBS output. This means that no additional cost in terms of RF chains is incurred despite the unit-modulus constraints (unlike in [48, 95], using twice the number of RF chains). The digital precoder is designed as $\mathbf{F}_{\mathrm{t,d}}^H = a_{\mathrm{t},1}^* \mathbf{I}$ or $\mathbf{F}_{\mathrm{t,d}}^H = \mathbf{I}$. The digital combiner is designed as $\mathbf{F}_{\mathrm{r,d}} = a_{\mathrm{r},1}\mathbf{I}$ or $\mathbf{F}_{\mathrm{r,d}} = \mathbf{I}$. After the digital combiner, the remaining steps for the 2-D coarray method described in Sec. 4.4.2 can then be realized in the digital domain.

For CBS with uniform decimation and expansion (4.9), we can implement the precoder (4.7) and combiner (4.8) in the same way as we do for the nonuniform case. This is because we can regard the uniform decimation matrices (4.6) as special cases of (4.17) by letting the virtual antenna locations $n_{p,i} = iM_p$, $p \in \{\mathrm{t}, \mathrm{r}\}$. Thus, we need $\bar{N}_\mathrm{t} = (N_\mathrm{t} - L_\mathrm{t} + 1)/M_\mathrm{t}$ RF chains at the transmitter and $\bar{N}_\mathrm{r} = (N_\mathrm{r} - L_\mathrm{r} + 1)/M_\mathrm{r}$ RF chains at the receiver.

4.6 Time Complexity Analysis

We now examine the time complexity of the proposed 2-D hybrid CBS. Here, time complexity translates into pilot overhead, which is the number of measurements required in training phase for channel estimation. Due to the large arrays used in mmWave MIMO transceivers, channel estimation can be a time-consuming task. Thus, it is interesting to compare our method with previous methods in terms of pilot overhaed. Suppose the DODs and DOAs are not known to be in some subbands a priori. Then, to cover the full DOD and DOA ranges, we use a transmit filter bank with M_t filters and a receive filter bank with M_r filters if the filters have passband widths $2\pi/M_p$, $p \in \{\mathrm{t}, \mathrm{r}\}$. For each pair of transmit and receive filters, we need K training blocks $\mathbf{S}_k$, $1 \le k \le K$ as shown in (4.4), where each block consists of $\bar{N}_\mathrm{t}$ training vectors. Thus, the total number of measurements required by hybrid CBS is

$$N_\mathrm{CBS} = M_\mathrm{t} M_\mathrm{r} \bar{N}_\mathrm{t} K. \tag{4.66}$$

This number is the same as the number of measurements required by the classical DFT beamspace MUSIC [19] or DFT beamspace ESPRIT [4]. This is because to cover the full DOD and DOA ranges, the two methods similarly require a set of M_t DFT beamformers at the transmitter and a set of M_r DFT beamformers at the receiver. That is,

$$N_\text{DFT-MUSIC} = N_\text{DFT-ESPRIT} = M_\mathrm{t} M_\mathrm{r} \bar{N}_\mathrm{t} K. \tag{4.67}$$

Thus, for a fair comparison between hybrid CBS and these previous methods, we can choose the same parameters so as to compare estimation performance while fixing pilot overhead.

Another type of channel estimation methods is based on beam search, such as the sequential beam search [6] and hierarchical beam search [6] methods. (We compare to the hierarchical beam search in [6] instead of the hierarchical beam search in [11]

since the former is shown in [6] to have smaller pilot overhead.) The number of measurements required by sequential beam search (i.e., exhaustive search) is

$$N_{\mathrm{SS}} = K_{\mathrm{os,t}} K_{\mathrm{os,r}} N_{\mathrm{t}} N_{\mathrm{r}}, \tag{4.68}$$

where $K_{\mathrm{os},p}$ are the oversampling factors[1] such that the transmit and receive codebooks consists of $K_{\mathrm{os},p} N_p$, $p \in \{\mathrm{t}, \mathrm{r}\}$, steering vectors with evenly sampled angles in the DOD and DOA ranges $\omega \in [-\pi, \pi)$. The number of measurements required by hierarchical beam search [6] is

$$N_{\mathrm{HS}} = D\Big[\big(\log_{M_{\mathrm{hf}}}(N_{\mathrm{t}}) + \log_{M_{\mathrm{hf}}}(N_{\mathrm{r}}) - 2i_{\mathrm{LY}}\big) M_{\mathrm{hf}} + M_{\mathrm{hf}}^{2i_{\mathrm{LY}}} + K_{\mathrm{os,t}} K_{\mathrm{os,r}}\Big], \tag{4.69}$$

where D is the number of paths, M_{hf} is the hierarchical factor, and i_{LY} is the initial layer index [6]. For a fair comparison between hybrid CBS and the beam search methods, we choose to design them to have similar estimation errors and then compare the pilot overhead. Since the beam search methods are grid-based methods, the estimation error comes from two sources, on-grid error due to incorrect beam found and off-grid error due to angles not on grid points. In this chapter, we design the system parameters so that assuming zero on-grid error, the off-grid error of the beam search methods is the same as the error of hybrid CBS. This guarantees that CBS has an error at least as small as the total error of the beam search methods. That is, the RMSE of CBS is a lower bound for those of beam search methods, as shown in Fig. 4.16(a). To evaluate the off-grid error, we assume the DODs and DOAs are uniformly distributed over the full range $\omega \in [-\pi, \pi)$. Then, the errors will be uniformly distributed in $(-\pi/(K_{\mathrm{os},p} N_p), \pi/(K_{\mathrm{os},p} N_p))$, and the error variances are $\pi^2/(3K_{\mathrm{os},p}^2 N_p^2)$, $p \in \{\mathrm{t}, \mathrm{r}\}$. Hence, if the RMSE of DOD and DOA estimates of hybrid CBS is ϵ, we shall design

$$K_{\mathrm{os},p} = \frac{\pi}{\sqrt{3}\epsilon N_p}. \tag{4.70}$$

In this way, the MSE of hybrid CBS is at least as small as the two beam search methods. As we will show in Sec. 4.8.2, with this setting, hybrid CBS requires smaller pilot overhead than the beam search methods.

Note that for each method mentioned in this section, since $\bar{N}_{\mathrm{r}}$ RF chains can be simultaneously used at the receiver to combine the received array data, the number of time instants required equals the number of measurements divided by $\bar{N}_{\mathrm{r}}$, as explained in [11].

[1] It is assumed that $K_{\mathrm{os,t}} = K_{\mathrm{os,r}}$ in [6].

4.7 Filters With Delays for Decorrelating Path Gains in 2-D CBS

We propose a new method for decorrelating path gains in the mmWave MIMO system. The new method, a 2-D extension to the decorrelation method in Sec. 3.4, is based on CBS filters with delays. In all the derivations in Secs. 4.3 and 4.4, the banded Toeplitz matrices $\mathbf{H}_{\mathrm{t}}$ and $\mathbf{H}_{\mathrm{r}}$ are assumed fixed for all received blocks $\mathbf{Y}_k$, e.g., in (4.24). However, we can also use different banded Toeplitz matrices for different time blocks. That is,

$$\mathbf{Y}_k = \overline{\mathbf{D}}_{\mathrm{r}}\mathbf{H}_{\mathrm{r},k}\mathbf{H}_{\alpha,k}\mathbf{H}_{\mathrm{t},k}^H\overline{\mathbf{D}}_{\mathrm{t}}^H\mathbf{S}_k + \overline{\mathbf{D}}_{\mathrm{r}}\mathbf{H}_{\mathrm{r},k}\mathbf{E}_k, \tag{4.71}$$

where $\mathbf{H}_{p,k}$ are $(N_p - L_p + 1) \times N_p$ banded Toeplitz matrices with entries

$$[\mathbf{H}_{p,k}]_{i,l} = h_{p,k}(L_p - 1 + i - l), \tag{4.72}$$

for $p \in \{\mathrm{t}, \mathrm{r}\}$ and $k = 1, \ldots, K$. In the following, we show that if we design $h_{p,k}(n)$ to be different delayed versions of some prototype filters $h_p(n)$, the path gains $\alpha_{k,i}$ can to some extent be decorrelated (if they were correlated in the beginning). The decorrelation method is applicable to both uniform decimation (Sec. 4.3) and nonuniform decimation (Sec. 4.4) schemes. However, it is more important to the latter scheme because we assume uncorrelated path gains when using coarray methods. Hence, the method will be illustrated through the nonuniform decimation scheme.

We design the filters $h_{p,k}(n)$ for the kth snapshot as delayed versions of some prototype filters $h_p(n)$ with length $L'_p < L_p$, $p \in \{\mathrm{t}, \mathrm{r}\}$. Let $q_{p,k} \in [0, L_p - L'_p]$ denote the delays. Then the filters $H_{p,k}(z) = \sum_{n=0}^{L_p-1} h_{p,k}(n)z^{-n}$ are related to the prototypes by

$$H_{p,k}(z) = H_p(z)z^{-q_{p,k}}, \tag{4.73}$$

where $H_p(z) = \sum_{n=0}^{L'_p-1} h_p(n)z^{-n}$. Note that each filter $h_{p,k}(n)$ only has at most L'_p nonzero coefficients. With such designed $\mathbf{H}_{p,k}$, we still have (4.25) but the diagonal matrix $\widetilde{\mathbf{D}}_k$ now has elements

$$[\widetilde{\mathbf{D}}_k]_{i,i} = \sigma_{\mathrm{s}}\alpha_{k,i}H_{\mathrm{r},k}(e^{j\omega_{\mathrm{r},i}})e^{j\omega_{\mathrm{r},i}(L_{\mathrm{r}}-1)}H^*_{\mathrm{t},k}(e^{j\omega_{\mathrm{t},i}})e^{-j\omega_{\mathrm{t},i}(L_{\mathrm{t}}-1)}. \tag{4.74}$$

Thus, we will also have (4.30), while $\widetilde{\mathbf{R}}_d = \mathrm{E}[\tilde{\mathbf{d}}_k\tilde{\mathbf{d}}_k^H]$ has entries

$$[\widetilde{\mathbf{R}}_d]_{m,l} = \sigma_{\mathrm{s}}^2\rho_{m,l}u_{m,l}e^{j(\omega_{\mathrm{t},l}-\omega_{\mathrm{t},m})(L_{\mathrm{t}}-1)}e^{j(\omega_{\mathrm{r},m}-\omega_{\mathrm{r},l})(L_{\mathrm{r}}-1)},$$

where

$$\begin{aligned} u_{m,l} &= \mathrm{E}[H_{\mathrm{t},k}^*(e^{j\omega_{\mathrm{t},m}})H_{\mathrm{t},k}(e^{j\omega_{\mathrm{t},l}})H_{\mathrm{r},k}(e^{j\omega_{\mathrm{r},m}})H_{\mathrm{r},k}^*(e^{j\omega_{\mathrm{r},l}})] \\ &= H_{\mathrm{t}}^*(e^{j\omega_{\mathrm{t},m}})H_{\mathrm{t}}(e^{j\omega_{\mathrm{t},l}})H_{\mathrm{r}}(e^{j\omega_{\mathrm{r},m}})H_{\mathrm{r}}^*(e^{j\omega_{\mathrm{r},l}}) \\ &\quad \cdot \mathrm{E}[e^{j(\omega_{\mathrm{t},m}-\omega_{\mathrm{t},l})q_{\mathrm{t},k}}e^{-j(\omega_{\mathrm{r},m}-\omega_{\mathrm{r},l})q_{\mathrm{r},k}}]. \end{aligned} \tag{4.75}$$

4.7.1 Case of Random Delays

Suppose the delays $(q_{\mathrm{t},k}, q_{\mathrm{r},k})$ are independent and identically distributed (i.i.d.) with joint probability mass function $f(q_{\mathrm{t}}, q_{\mathrm{r}})$. Since the 2-D discrete-time Fourier transform of $f(q_{\mathrm{t}}, q_{\mathrm{r}})$ is

$$\begin{aligned} F(e^{j\omega_{\mathrm{t}}}, e^{j\omega_{\mathrm{r}}}) &= \sum_{q_{\mathrm{t}}=0}^{L_{\mathrm{t}}-L_{\mathrm{t}}'} \sum_{q_{\mathrm{r}}=0}^{L_{\mathrm{r}}-L_{\mathrm{r}}'} f(q_{\mathrm{t}}, q_{\mathrm{r}})e^{-j\omega_{\mathrm{t}}q_{\mathrm{t}}}e^{-j\omega_{\mathrm{r}}q_{\mathrm{r}}} \\ &= \mathrm{E}[e^{-j\omega_{\mathrm{t}}q_{\mathrm{t},k}}e^{-j\omega_{\mathrm{r}}q_{\mathrm{r},k}}], \end{aligned} \tag{4.76}$$

we have that

$$\mathrm{E}[e^{j(\omega_{\mathrm{t},m}-\omega_{\mathrm{t},l})q_{\mathrm{t},k}}e^{-j(\omega_{\mathrm{r},m}-\omega_{\mathrm{r},l})q_{\mathrm{r},k}}] = F(e^{j(\omega_{\mathrm{t},l}-\omega_{\mathrm{t},m})}, e^{j(\omega_{\mathrm{r},m}-\omega_{\mathrm{r},l})}). \tag{4.77}$$

Now we design $F(e^{j\omega_{\mathrm{t}}}, e^{j\omega_{\mathrm{r}}})$ as a 2-D lowpass filter. Then assuming that there is not a pair of paths with both DODs close to each other *and* DOAs close to each other (so that $(\omega_{\mathrm{t},l} - \omega_{\mathrm{t},m}, \omega_{\mathrm{r},m} - \omega_{\mathrm{r},l})$ is in the passband), we can make $u_{m,l} \approx 0$ for all $m \neq l$. This, in turn, leads to an approximately diagonal $\widetilde{\mathbf{R}}_d$. In other words, the path gains are decorrelated. With this method, the 2-D coarray method described in Sec. 4.4.2 can even be applied when the path gains $\alpha_{k,i}$ are correlated, i.e., $\rho_{m,l} \neq 0$ for some $m \neq l$. For simplicity, we can design the transmit filter delays to be independent of the receive filter delays. In this case, the joint probability mass function $f(q_{\mathrm{t}}, q_{\mathrm{r}}) = f_{\mathrm{t}}(q_{\mathrm{t}})f_{\mathrm{r}}(q_{\mathrm{r}})$ and its discrete-time Fourier transform

$$F(e^{j\omega_{\mathrm{t}}}, e^{j\omega_{\mathrm{r}}}) = F_{\mathrm{t}}(e^{j\omega_{\mathrm{t}}})F_{\mathrm{r}}(e^{j\omega_{\mathrm{r}}}) \tag{4.78}$$

are both separable, where

$$F_p(e^{j\omega}) = \sum_{q=0}^{L_p-L_p'} f_p(q)e^{-j\omega q} = \mathrm{E}[e^{-j\omega q_{p,k}}] \tag{4.79}$$

for $p \in \{\mathrm{t}, \mathrm{r}\}$. Hence, we can use 1-D lowpass filters $F_{\mathrm{t}}(e^{j\omega})$ and $F_{\mathrm{r}}(e^{j\omega})$ to construct a 2-D lowpass filter. In some cases, introducing delays for either the transmit CBS filters or receive CBS filters is sufficient to decorrelate the path gains, but we can

also do it at both sides. Some simulation examples regarding such comparison will be shown in Sec. 4.8. We will also see (Example 7) that if we have rough initial estimates of the DODs and DOAs, we can design or redesign the probability mass function of the delays to get better decorrelating ability. The introduction of the filter delays does not ruin the traditional CBS filtering effect on DODs and DOAs. We can still realize it by designing $H_{\rm t}(e^{j\omega})$ and $H_{\rm r}(e^{j\omega})$ to be standard lowpass filters. However, there is a tradeoff between the CBS filtering effect and the decorrelating effect on the path gains when we choose L'_p given a fixed L_p because the effective filter lengths of $h_p(n)$ and $f_p(q)$ are L'_p and $L_p - L'_p + 1$, respectively. Still, for large arrays (large N_p), which are getting popular recently [9, 11], a large enough L_p can be used so that reasonably good filtering ability and decorrelating ability can be both obtained.

In [5], a method based on decimating rows to obtain submatrices of banded Toeplitz (SBT) random matrices is also proposed to decorrelate the path gains. However, instead of introducing random delays, the coefficients $h_{p,k}(n)$, $p \in \{\rm t, r\}$, are themselves chosen randomly with Rademacher distribution. Such design cannot result in a filtering effect for the DODs and DOAs, which is essential for our decimation scheme ($n_{p,i} = \tilde{n}_{p,i}M_p$, $M_p > 1$) to work without causing DOD and DOA ambiguities.

4.7.2 Case of Deterministic Delays

In the above derivations, we have assumed that the delays $(q_{{\rm t},k}, q_{{\rm r},k})$ are random and i.i.d. with joint probability mass function $f(q_{\rm t}, q_{\rm r})$. Instead of random delays, it is also possible to consider deterministic delays. In practice, we use K time blocks to obtain the estimated covariance

$$\widehat{\mathbf{R}}_y = \frac{1}{K}\sum_{k=1}^{K} \mathbf{y}_k \mathbf{y}_k^H, \tag{4.80}$$

where $\mathbf{y}_k$ is as in (4.27). We let $\hat{f}(q_{\rm t}, q_{\rm r})$ denote the number of time blocks with transmit and receive filter delays $q_{\rm t}$ and $q_{\rm r}$, divided by K, i.e.,

$$\hat{f}(q_{\rm t}, q_{\rm r}) = \frac{1}{K}\Big|\big\{1 \le k \le K \mid q_{{\rm t},k} = q_{\rm t}, q_{{\rm r},k} = q_{\rm r}\big\}\Big|. \tag{4.81}$$

The 2-D discrete-time Fourier transform of $\hat{f}(q_{\mathrm{t}}, q_{\mathrm{r}})$ is

$$\begin{aligned}\widehat{F}(e^{j\omega_{\mathrm{t}}}, e^{j\omega_{\mathrm{r}}}) &= \sum_{q_{\mathrm{t}}=0}^{L_{\mathrm{t}}-L'_{\mathrm{t}}} \sum_{q_{\mathrm{r}}=0}^{L_{\mathrm{r}}-L'_{\mathrm{r}}} \hat{f}(q_{\mathrm{t}}, q_{\mathrm{r}}) e^{-j\omega_{\mathrm{t}} q_{\mathrm{t}}} e^{-j\omega_{\mathrm{r}} q_{\mathrm{r}}} \\ &= \frac{1}{K} \sum_{k=1}^{K} e^{-j\omega_{\mathrm{t}} q_{\mathrm{t},k}} e^{-j\omega_{\mathrm{r}} q_{\mathrm{r},k}}, \end{aligned} \tag{4.82}$$

which represents an empirical counterpart of (4.76). Then similar to (4.30), we have

$$\mathrm{E}[\widehat{\mathbf{R}}_y] = (\overline{\mathbf{A}}_{\mathrm{t}}^{*} \odot \overline{\mathbf{A}}_{\mathrm{r}})\, \mathrm{E}\left[\widehat{\widetilde{\mathbf{R}}}_d\right] (\overline{\mathbf{A}}_{\mathrm{t}}^{*} \odot \overline{\mathbf{A}}_{\mathrm{r}})^H + \mathbf{R}_{\mathrm{e}}, \tag{4.83}$$

where

$$\mathrm{E}\left[\widehat{\widetilde{\mathbf{R}}}_d\right] = \frac{1}{K} \sum_{k=1}^{K} \mathrm{E}[\tilde{\mathbf{d}}_k \tilde{\mathbf{d}}_k^H] \tag{4.84}$$

with $[\tilde{\mathbf{d}}_k]_i = [\widetilde{\mathbf{D}}_k]_{i,i}$ defined in (4.74). Then with (4.82), we can derive that $\mathrm{E}\left[\widehat{\widetilde{\mathbf{R}}}_d\right]$ has entries

$$\left[\mathrm{E}\left[\widehat{\widetilde{\mathbf{R}}}_d\right]\right]_{i,l} = \sigma_{\mathrm{s}}^2 \rho_{i,l} \hat{u}_{i,l} e^{j[(\omega_{\mathrm{t},l}-\omega_{\mathrm{t},i})(L_{\mathrm{t}}-1)+(\omega_{\mathrm{r},i}-\omega_{\mathrm{r},l})(L_{\mathrm{r}}-1)]},$$

where

$$\hat{u}_{i,l} = H_{\mathrm{t}}^{*}(e^{j\omega_{\mathrm{t},i}}) H_{\mathrm{t}}(e^{j\omega_{\mathrm{t},l}}) H_{\mathrm{r}}(e^{j\omega_{\mathrm{r},i}}) H_{\mathrm{r}}^{*}(e^{j\omega_{\mathrm{r},l}}) \widehat{F}(e^{j(\omega_{\mathrm{t},l}-\omega_{\mathrm{t},i})}, e^{j(\omega_{\mathrm{r},i}-\omega_{\mathrm{r},l})}). \tag{4.85}$$

Hence, if we design $\hat{f}(q_{\mathrm{t}}, q_{\mathrm{r}})$ so that $\widehat{F}(e^{j\omega_{\mathrm{t}}}, e^{j\omega_{\mathrm{r}}})$ can approximate the desired low-pass filter $F(e^{j\omega_{\mathrm{t}}}, e^{j\omega_{\mathrm{r}}})$, it is again possible to achieve $\widehat{F}(e^{j(\omega_{\mathrm{t},l}-\omega_{\mathrm{t},i})}, e^{j(\omega_{\mathrm{r},i}-\omega_{\mathrm{r},l})}) \approx 0$ for all $i \neq l$, assuming $(\omega_{\mathrm{t},l} - \omega_{\mathrm{t},m}, \omega_{\mathrm{r},m} - \omega_{\mathrm{r},l})$ is not in the passband. An almost diagonal $\mathrm{E}\left[\widehat{\widetilde{\mathbf{R}}}_d\right]$ is then obtained, which means the path gains are decorrelated. We will see that using either random delays or deterministic delays give us similar estimation performance in simulations (Sec. 4.8). Thus, deterministic delays can be a better choice as they are easier to implement than random delays within the analog precoder and combiner.

4.8 Simulations

In this section, we study the performance of the proposed hybrid CBS for mmWave MIMO channel estimation via numerical examples. Estimation errors and time complexity are considered in Secs. 4.8.1 and 4.8.2, respectively. In the following examples, we assume the number of paths is known for element-space methods, while the number of in-band paths is known for CBS. To compare with CBS, for

element-space, we consider DOA and DOD estimates of only those paths that are in-band paths with respect to CBS. The root mean square errors (RMSEs) in detected in-band source angles shown in this section are obtained by averaging square errors measured in ω over all in-band paths and over those Monte Carlo runs which yield the correct number of in-band paths (since element-space may yield a wrong number of in-band paths). That is, the RMSE of DODs ($p = \mathrm{t}$) or DOAs ($p = \mathrm{r}$) is defined as

$$\epsilon_p = \left[\frac{1}{N_{\mathrm{mc}} D_0} \sum_{l=1}^{N_{\mathrm{mc}}} \sum_{i=1}^{D_0} (\hat{\omega}_{p,i}(l) - \omega_{p,i}(l))^2 \right]^{1/2}, \tag{4.86}$$

where $\omega_{p,i}(l)$ is the ith DOD or DOA in the lth Monte Carlo run, $\hat{\omega}_{p,i}(l)$ is the estimate of $\omega_{p,i}(l)$, D_0 is the number of in-band paths, and N_{mc} is the number of Monte Carlo runs yielding correct number of in-band paths. We design CBS transmit and receive filters as lowpass Parks-McClellan filters [23, 63] with passband edges $(1-\beta)\pi/M_p$ and stopband edges $(1+\beta)\pi/M_p$, where M_p are the expansion and decimation ratios, $p \in \{\mathrm{t}, \mathrm{r}\}$, and β is the roll-off factor which determines transition bandwidths. Covariance estimates are obtained by using $K = 100$ time blocks, and we average 500 Monte Carlo runs to get each plot. If not otherwise mentioned, all path gains $\alpha_{k,i}$ have the same power, and we define the SNR to be the ratio of the power of an in-band path to the noise power σ_{e}^2.

4.8.1 Estimation performance comparison

Example 1 (2-D Hybrid CBS with uniform decimation and expansion): We consider a pair of transmit and receive ULAs with $N_{\mathrm{t}} = N_{\mathrm{r}} = 96$ antennas. There are 2 in-band paths with DODs $\theta_{\mathrm{t}} = -4^\circ, 3^\circ$ and DOAs $\theta_{\mathrm{r}} = -3^\circ, 4^\circ$, and 2 out-of-band paths with DODs $\theta_{\mathrm{t}} = 40^\circ, 60^\circ$ and DOAs $\theta_{\mathrm{r}} = 45^\circ, 65^\circ$. The expansion and decimation ratios are $M_{\mathrm{t}} = M_{\mathrm{r}} = 8$. The CBS transmit and receive filters have length $L_{\mathrm{t}} = L_{\mathrm{r}} = 25$ and roll-off factor $\beta = 0.5$. Thus, the numbers of RF chains are $\bar{N}_p = (N_p - L_p + 1)/M_p = 9$, $p \in \{\mathrm{t}, \mathrm{r}\}$. The DODs and DOAs are estimated using 2-D unitary ESPRIT [51]. As shown in Fig. 4.8, the average RMSE of DOD and DOA estimates of hybrid CBS is compared to 2-D classical DFT beamspace (BS) MUSIC [19]. Classical BS is directly applicable to the analog domain because a submatrix of a DFT matrix with constant-modulus entries is used as its BS matrix. We use either 100×100 or 1000×1000 2-D MUSIC grid points for the passband angles. With the first grid, classical BS MUSIC has larger RMSE than hybrid CBS since the performance is limited by the density of grid points. With the denser grid, classical BS MUSIC achieves smaller RMSE than hybrid CBS at lower SNRs (but

vice versa at higher SNRs). However, the denser grid takes much higher computation and time complexity. The running time per Monte Carlo run of hybrid CBS is 0.058 seconds, which is smaller than and much smaller than those of classical BS MUSIC with 100×100 and 1000×1000 grid points, namely, 0.13 and 7.6 seconds. Just for reference, we also show the average RMSE when only the first $\bar{N}_p$ antennas are used, $p \in \{\mathrm{t}, \mathrm{r}\}$, and all other antennas are discarded (element-space truncation). In this case there are no analog precoder and combiner. Since not all antennas are effectively used, element-space truncation has the worst performance in this example.

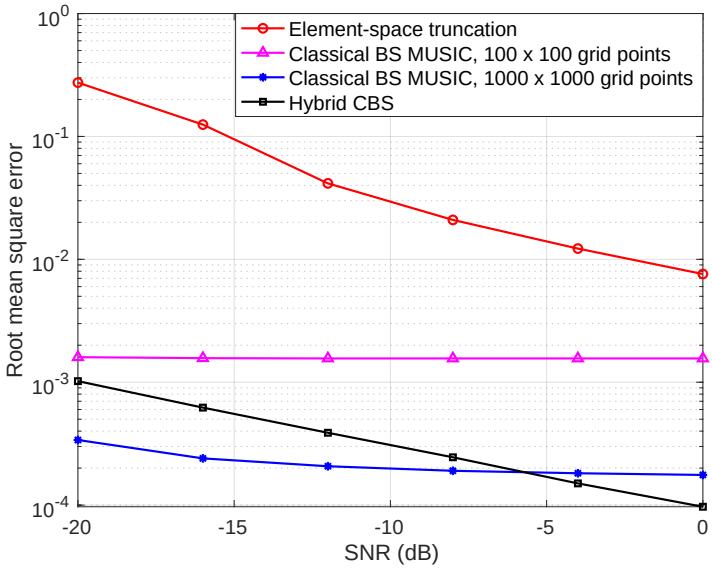


Figure 4.8: Average RMSE of DOD and DOA estimates of element-space truncation, classical beamspace (BS) MUSIC, and hybrid CBS with uniform decimation (and expansion). Except classical BS, 2-D unitary ESPRIT is used to estimate DODs and DOAs. Each method can be realized with 9 transmit and 9 receive RF chains in the hybrid model.

In the remaining examples, we study 2-D hybrid CBS with nonuniform decimation and expansion, applied with coarray method. The expansion and decimation ratios are $M_\mathrm{t} = M_\mathrm{r} = 4$. The nonuniform decimation matrices $\overline{\mathbf{D}}_p$ in (4.17) are chosen with $n_{p,i} = \tilde{n}_{p,i} M_p$, $0 \le i \le N_\mathrm{MRA} - 1$, where $\{\tilde{n}_{p,i}\}$ are the antenna locations of N_MRA-antenna restricted MRAs [3], $p \in \{\mathrm{t}, \mathrm{r}\}$. DODs and DOAs are estimated using 2-D unitary ESPRIT [51].

Example 2 (Uniform decimation vs. nonuniform decimation for hybrid CBS): We first compare nonuniform decimation (CBS coarray method) to uniform decimation

for hybrid CBS. We consider a pair of transmit and receive ULAs with $N_{\rm t} = N_{\rm r} = 52$ antennas. There are 7 in-band paths with DODs $\omega_{\rm t} = -0.125\pi + i\Delta$ and DOAs $\omega_{\rm r} = -0.125\pi + i\Delta$, $i = 0, 1, \ldots, 6$, where $\Delta = 0.25\pi/6$ and 2 out-of-band paths with DODs $\theta_{\rm t} = 40°, 60°$ and DOAs $\theta_{\rm r} = 45°, 65°$. (Recall that $\omega = \pi \sin\theta$.) The CBS transmit and receive filters have length $L_{\rm t} = L_{\rm r} = 16$ and roll-off factor $\beta = 0.5$. For CBS coarray method, the nonuniform expansion and decimation correspond to MRAs with $N_{\rm MRA} = 5$ antennas. For a fair comparison, we retain only 5 samples at $n_i = iM_p$, $0 \le i \le 4$ for uniform expansion and decimation. We also compare to the classical DFT beamspace ESPRIT [4], with 5 DFT beams used at each of the precoder and combiner. Hence, each method can be realized with $\bar{N}_{\rm t} = 5$ transmit and $\bar{N}_{\rm r} = 5$ receive RF chains. In Fig. 4.9, the RMSEs of DODs and DOAs for nonuniform CBS are the smallest, while uniform CBS and DFT beamspace ESPRIT have similar performance. Nonuniform CBS has smaller estimation errors since the coarray aperture of the MRA is larger than the ULA aperture of uniform CBS. The large errors of uniform CBS is because the DODs and DOAs are too closely spaced. Uniform CBS can still be a good method in other scenarios, as we saw in Example 1.

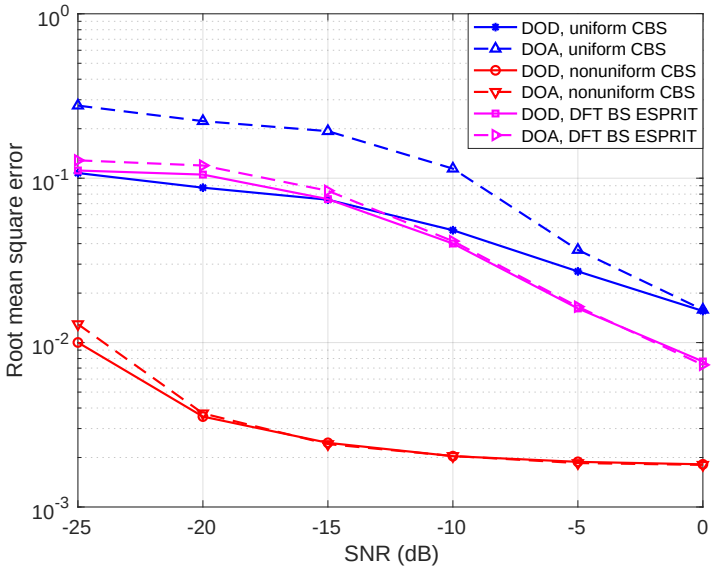


Figure 4.9: RMSE of DOD and DOA estimates of hybrid CBS with uniform or nonuniform decimation and classical DFT beamspace (BS) ESPRIT [4].

Example 3 (2-D Hybrid CBS coarray method vs. direct coarray method (both with nonuniform decimation and expansion)): We consider a pair of transmit and receive ULAs with $N_{\rm t} = N_{\rm r} = 108$ antennas. There are 3 in-band paths with DODs

$\theta_t = -7°, -2°, 3°$ and DOAs $\theta_r = -4°, 1°, 6°$, and 2 out-of-band paths with DODs $\theta_t = 40°, 60°$ and DOAs $\theta_r = 45°, 65°$. For CBS coarray method, the nonuniform expansion and decimation correspond to MRAs with $N_{MRA} = 8$ antennas. The CBS transmit and receive filters have length $L_t = L_r = 16$ and roll-off factor $\beta = 0.5$. For comparison, we show the performance of the direct coarray method in element-space. That is, we do nonuniform expansion and decimation in element-space at the transmitter and receiver, respectively, corresponding to samples at $\tilde{n}_{p,i}$, $0 \le i \le 7$, where $\{\tilde{n}_{p,i}\}$ are the antenna locations of an 8-antenna restricted MRA [3], $p \in \{t, r\}$. Thus, each method can be realized with 8 transmit and 8 receive RF chains in the hybrid model. As shown in Fig. 4.10, the RMSEs of DODs and DOAs of CBS coarray method are much smaller than those of the direct coarray method. This is because CBS has an effective 2-D coarray aperture that is 4×4 times larger (since CBS has dilation factors $M_t = M_r = 4$, but direct coarray method does not). Also, for each method the RMSE of DODs is similar to that of DOAs since similar parameters are used for the transmitter and receiver.

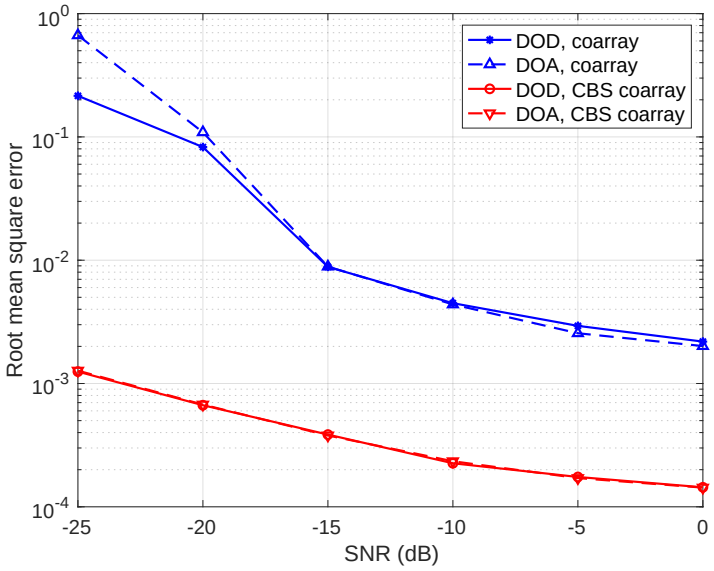


Figure 4.10: RMSE of DOD and DOA estimates of direct coarray method in element-space and hybrid CBS coarray method. For CBS coarray method, the nonuniform decimation matrices $\overline{\mathbf{D}}_p$ in (4.17) are chosen with $n_{p,i} = \tilde{n}_{p,i} M_p$, $0 \le i \le 7$, where $M_p = 4$ and $\{\tilde{n}_{p,i}\}$ are the antenna locations of an 8-antenna restricted MRA [3], $p \in \{t, r\}$. For direct coarray method, we do nonuniform expansion and decimation in element-space at the transmitter and receiver, respectively, corresponding to samples at $\tilde{n}_{p,i}$, $0 \le i \le 7$. Thus, each method can be realized with 8 transmit and 8 receive RF chains in the hybrid model.

Example 4 (2-D Hybrid CBS coarray method with nonuniform decimation, 12 in-band paths, 5 RF chains): Now we show the effectiveness of hybrid CBS coarray method by an extreme case with many paths. Suppose we have a pair of transmit and receive ULAs with $N_{\rm t} = N_{\rm r} = 61$ antennas. There are 12 in-band paths with DODs $\omega_{\rm t} = -0.2\pi + i\Delta$ and DOAs $\omega_{\rm r} = -0.2\pi + i\Delta$, $i = 0, 1, \ldots, 11$, where $\Delta = 0.4\pi/11$ and no out-of-band paths. For the decimation and expansion, MRAs with $N_{\rm MRA} = 5$ antennas are used. The CBS transmit and receive filters have length $L_{\rm t} = L_{\rm r} = 25$ and roll-off factor $\beta = 0.2$. As shown in Fig. 4.11, reasonably good RMSEs of DODs and DOAs can still be obtained. The RMSEs tend to saturate with SNR as coarray method is used [49, 50], but we can identify 12 in-band paths, more than the number $\bar{N}_p = 5$ of RF chains. Moreover, if we use two filter banks (each with $M_p = 4$ filters, $p \in \{\rm t, r\}$) as described in the end of Sec. 4.3, then we may identify the paths falling in the other $M_{\rm t}M_{\rm r} - 1 = 15$ combinations of transmit and receive filter passbands. Thus, much more paths can be identified. The direct coarray method as in Example 3 is not shown in Fig. 4.11 since it cannot identify those closely spaced DODs and DOAs.

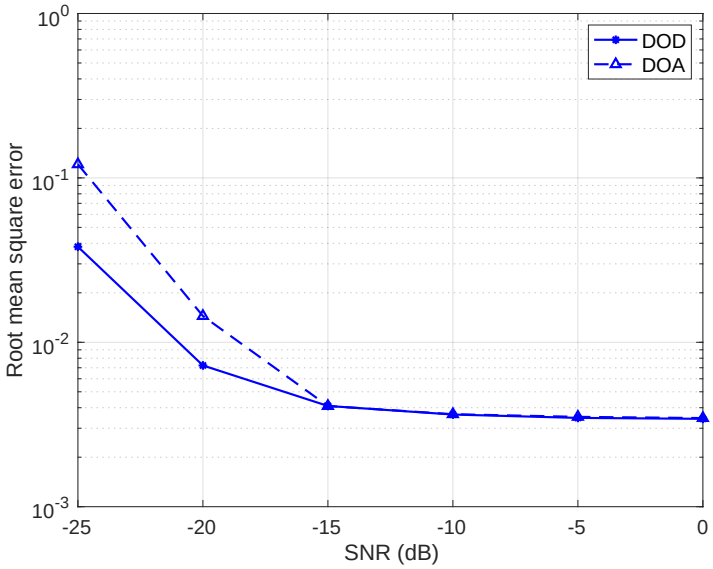


Figure 4.11: (12 in-band paths, 5 RF chains) RMSE of DOD and DOA estimates of hybrid CBS coarray method with nonuniform decimation.

Example 5 (Decorrelation method for 2-D hybrid CBS with nonuniform decimation and expansion): Next, we verify the effectiveness of the decorrelation method proposed in Sec. 4.7. We consider a pair of transmit and receive ULAs with $N_{\rm t} = N_{\rm r} = 71$ antennas. There are 2 in-band paths with DODs $\theta_{\rm t} = -7^\circ, 3^\circ$ and

DOAs $\theta_{\mathrm{r}} = -4^\circ, 6^\circ$, and 2 out-of-band paths with DODs $\theta_{\mathrm{t}} = 40^\circ, 60^\circ$ and DOAs $\theta_{\mathrm{r}} = 45^\circ, 65^\circ$. Each pair of the 4 path gains have a correlation coefficient $\rho = 0.4$. For CBS coarray method, the nonuniform expansion and decimation corresponds to MRAs with $N_{\mathrm{MRA}} = 5$ antennas dilated by $M_p = 4$ times, $p \in \{\mathrm{t}, \mathrm{r}\}$. The CBS transmit and receive prototype filters have length $L'_{\mathrm{t}} = L'_{\mathrm{r}} = 16$ and roll-off factor $\beta = 0.5$. To decorrelate the path gains, we introduce deterministic receive filter delays $q_{\mathrm{r},k}$ to approximate the uniform distribution over $[0, Q]$, where $Q = 19$. That is, the decorrelation filter $f_{\mathrm{r}}(q)$ in (4.78) is a rectangular window with length $L_{\mathrm{r}} - L'_{\mathrm{r}} + 1 = 20$. Since a 5-antenna MRA has aperture $A = 9$ (i.e., the last sensor location $\tilde{n}_{\mathrm{r},4} = 9$), the parameters are consistent as $AM_{\mathrm{r}}+Q+L'_{\mathrm{r}} = 9\times4+19+16 = N_{\mathrm{r}}$. For comparison, we consider the random SBT precoding method [5], which can also decorrelate path gains. Like hybrid CBS, the random SBT precoding method is designed to be realized with $\bar{N}_{\mathrm{t}} = 5$ transmit and $\bar{N}_{\mathrm{r}} = 5$ receive RF chains in the hybrid model. The submatrices of banded Toeplitz random matrices at the transmitter and receiver are obtained by keeping rows $\tilde{n}_{p,i}$, $1 \le i \le 5$, corresponding to the antenna locations of a 5-antenna MRA, $p \in \{\mathrm{t}, \mathrm{r}\}$. As shown in Fig. 4.12, the RMSEs of DODs and DOAs of CBS coarray method are much smaller than those of the random SBT precoding method. CBS has smaller estimation errors because the effective 2-D coarray aperture is $M_{\mathrm{t}}M_{\mathrm{r}} = 4 \times 4$ times larger. Also, for each method the RMSE of DODs is again similar to that of DOAs since similar parameters are used for the transmitter and receiver.

Example 6 (Random or deterministic filter delays at the transmitter, or receiver, or both, for the decorrelation method): Now we compare the performance of the proposed decorrelation method with several combinations of random or deterministic filter delays at the transmitter and receiver. All the simulation parameters are the same as Example 5, except that there are 6 in-band paths with DODs $\omega_{\mathrm{t}} = -0.1\pi+i\Delta$ and DOAs $\omega_{\mathrm{r}} = -0.12\pi+i\Delta$, $i = 0, 1, \ldots, 5$, where $\Delta = 0.22\pi/5$. The 2 out-of-band paths are the same as before, and each pair of the 8 path gains have a correlation coefficient $\rho = 0.4$. As shown in Fig. 4.13(a), for all SNRs, the average RMSEs of DODs and DOAs are almost the same for either random or deterministic delay at either the transmitter or receiver. Implementing the delays at the transmitter or receiver do not matter because of the similar parameters on both sides. When deterministic delays yield similar performance to random delays, we recommend deterministic ones as they are easier to implement in the analog precoder and combiner. Yet we also see that using independent random delays at both the transmitter and receiver results in slightly smaller average RMSE. This is due to the effect of

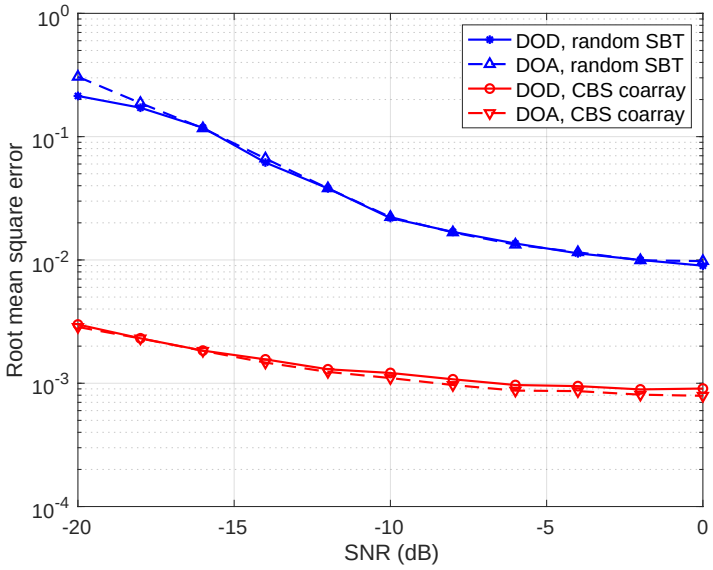
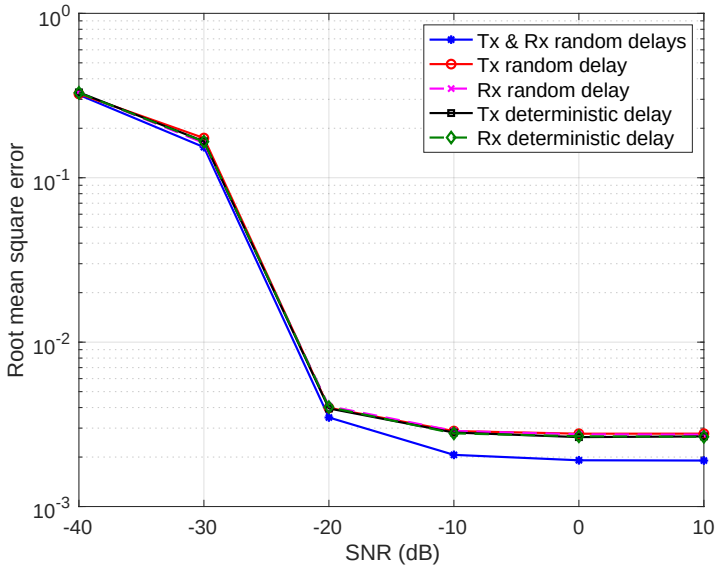


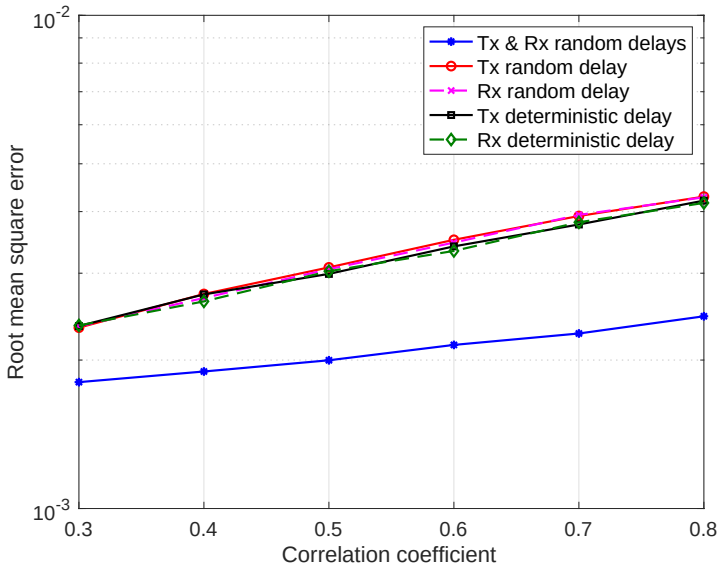
Figure 4.12: RMSE of DOD and DOA estimates of the random SBT precoding method [5] and hybrid CBS coarray method with nonuniform decimation and random filter delays. Each method can be realized with 5 transmit and 5 receive RF chains in the hybrid model.

the product $F_{\rm t}(e^{j\omega_{\rm t}})F_{\rm r}(e^{j\omega_{\rm r}})$ in (4.78). That is, the decorrelating ability is doubled. Similar results are obtained in Fig. 4.13(b) when we fix the SNR to be 0 dB and vary the correlation coefficient ρ. We note that in this example, considering either of the transmitter and receiver has only $\bar{N}_p = 5$ RF chains, we already achieve quite high DOA resolution when resolving those 6 in-band paths with closely spaced DOAs.

Example 7 (Filter delays with different probability mass functions for the decorrelation method): We compare the performance of the proposed decorrelation method with random receive filter delays with different probability mass functions: a normalized rectangular window, Hann window, or Hamming window [63]. All the simulation parameters are the same as Example 6, except that there are 2 in-band paths with DODs $\theta_{\rm t} = -7°, 3°$ and DOAs $\theta_{\rm r} = -4°, 6°$, and each pair of the 4 (in-band and out-of-band) path gains have a correlation coefficient $\rho = 0.9$. As shown in Fig. 4.14(a), at higher SNR regime, the average RMSE of DODs and DOAs of the Hamming window is slightly smaller than that of the Hann window, which is, in turn, slightly smaller than that of the rectangular window. This can be understood if we consider the magnitude responses of the used windows in Fig. 4.14(b). Here the difference of the 2 in-band DOAs measured in ω is also plotted. At this frequency, the attenuation of the Hamming window is larger than that of



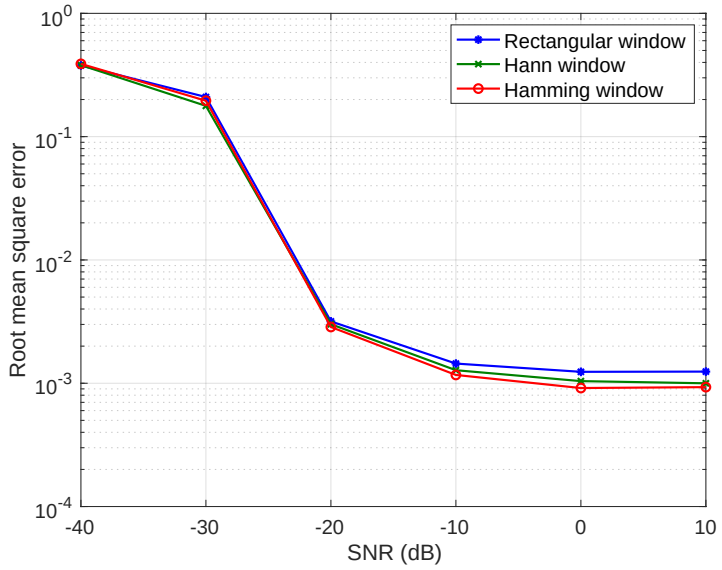
(a)



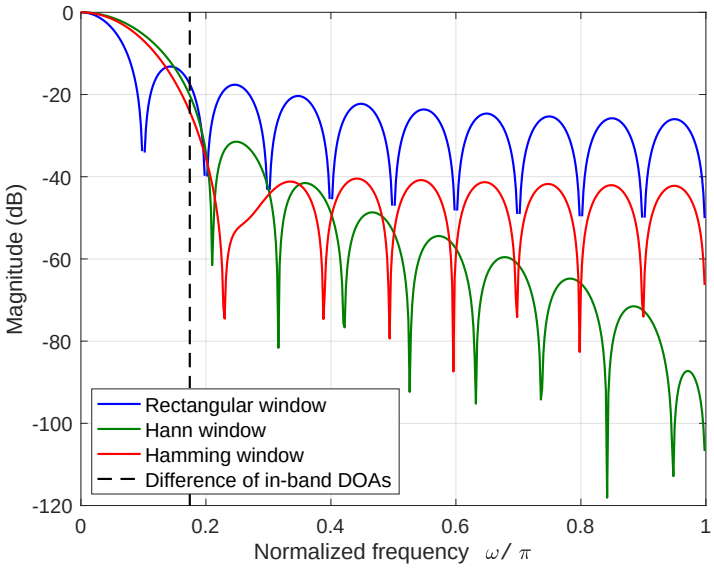
(b)

Figure 4.13: Average RMSE of DOD and DOA estimates of hybrid CBS coarray method with nonuniform decimation. To decorrelate the path gains, we introduce random or deterministic delays for the CBS filters at the transmitter (Tx), or receiver (Rx), or both. (a) RMSE as SNR varies. (b) RMSE as correlation coefficient varies.

the Hann window, which is, in turn, larger than that of the rectangular window. This determines the decorrelating ability due to the receive filter delays. Hence, in practice, if we have rough initial estimates of DODs and DOAs, we can design or redesign the probability mass function of the delays to get better decorrelating



(a)



(b)

Figure 4.14: Performance of hybrid CBS coarray method with nonuniform decimation and random filter delays at the receiver. The probability mass function of the delay is a normalized rectangular, Hann, or Hamming window. (a) Average RMSE of DOD and DOA estimates. (b) Magnitude responses of the used windows. The difference of the 2 in-band DOAs is indicated by the dashed vertical line.

ability.

Example 8 (Estimation errors of channel matrix): In all previous examples, we only show estimation errors of DODs and DOAs, but not of the channel matrix.

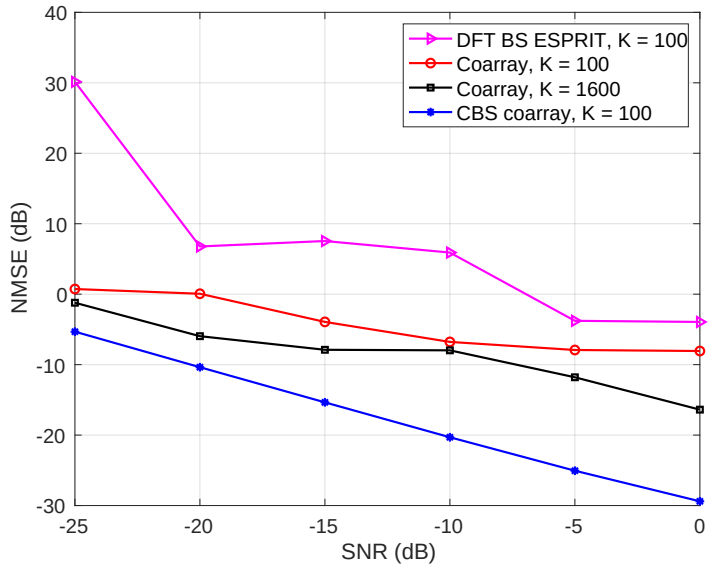
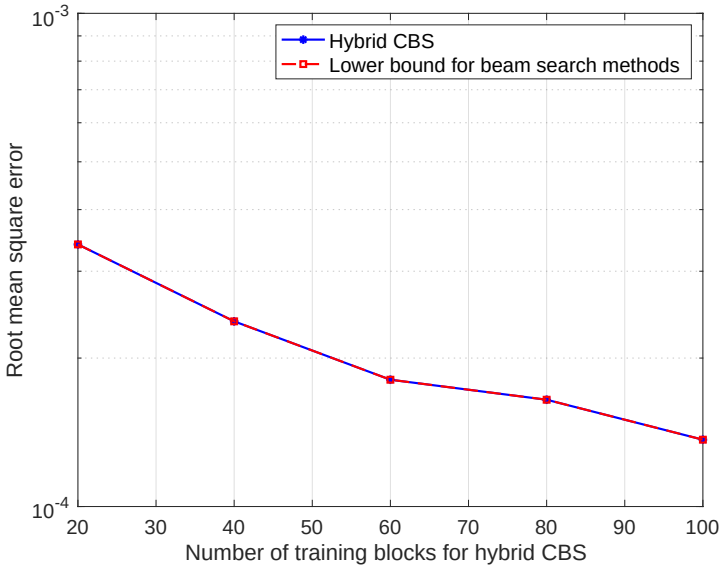


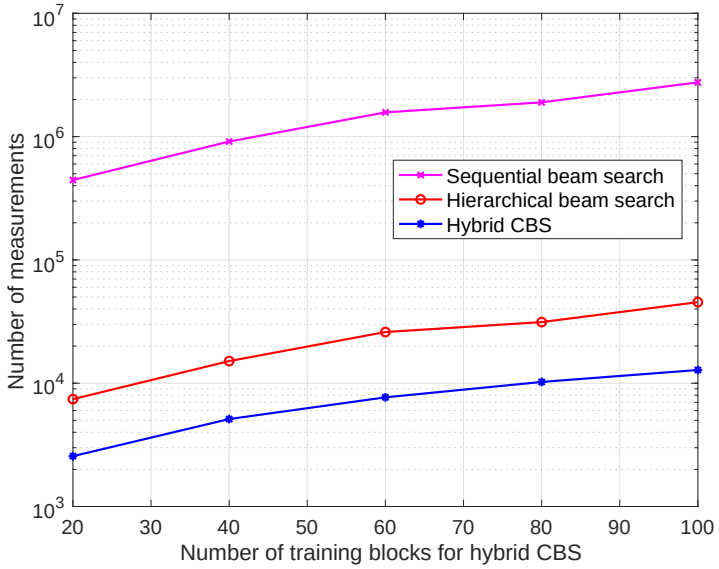
Figure 4.15: NMSE of channel matrix estimates of DFT beamspace (BS) ESPRIT [4], direct coarray method in element-space, and hybrid CBS coarray method. The expansion and decimation schemes are the same as in Fig. 4.10. For DFT BS ESPRIT and CBS coarray method, the total number of training blocks is $M_{\mathrm{t}}M_{\mathrm{r}}K = 4 \cdot 4 \cdot 100 = 1600$. For direct coarray method, the number of training blocks is $K = 100$ or $K = 1600$.

In general, smaller estimation errors of DODs and DOAs correspond to smaller estimation errors of the channel matrix. However, for completeness, we now show an example for estimation errors of channel matrix. We consider direct coarray method in element-space and hybrid CBS coarray method as in Example 3. We also compare to DFT beamspace ESPRIT [4]. There are 3 in-band paths with DODs $\theta_{\mathrm{t}} = -7^\circ, 2.5^\circ, 3^\circ$ and DOAs $\theta_{\mathrm{r}} = -4^\circ, 5.5^\circ, 6^\circ$, and 2 out-of-band paths with DODs $\theta_{\mathrm{t}} = 35^\circ, 75^\circ$ and DOAs $\theta_{\mathrm{r}} = 75^\circ, -30^\circ$. All the other simulation parameters are the same as in Example 3. As explained in Sec. 4.4.3, for CBS coarray method (and DFT beamspace ESPRIT), if the DODs and DOAs are not known to be in some subbands a priori, we need a total of $M_{\mathrm{t}}M_{\mathrm{r}}K = 4 \cdot 4 \cdot 100 = 1600$ training blocks to estimate the channel matrix, where K is the number of training blocks used for each pair of the transmit and receive filters. Thus, for direct coarray method, we consider both $K = 100$ and $K = 1600$ time blocks, where the latter case gives a fair comparison. As shown in Fig. 4.15, the normalized mean square errors (NMSEs) (as defined in [19]) of the channel matrix of CBS coarray method are smaller than those of the direct coarray method, even with $K = 1600$. DFT beamspace ESPRIT performs the worst as it fails to resolve the closely spaced DODs and DOAs.

4.8.2 Time Complexity Comparison



(a)



(b)

Figure 4.16: Time complexity comparison between hybrid CBS, sequential beam search [6], and hierarchical beam search [6]. (a) Average RMSE of DOD and DOA estimates. The lower bound for beam search methods is due to off-grid error as explained in Sec. 4.6. (b) Number of measurements.

We now compare the time complexity of hybrid CBS with other methods. As explained in Sec. 4.6, since the number of time blocks is chosen to be the same for non-beam-search methods like hybrid CBS, DFT beamspace MUSIC [19], and

DFT beamspace ESPRIT [4] in Sec. 4.8.1, they all have the same pilot overhead, as in (4.66) and (4.67). Thus, the results in Sec. 4.8.1 show that hybrid CBS has smaller estimation errors than the other methods when pilot overhead is fixed. To compare with beam search methods, as elaborated in Sec. 4.6, we design system parameters in a way such that RMSE of hybrid CBS is at least as small as beam search methods. To this end, we first do simulations for hybrid CBS and obtain the average RMSE of DOD and DOA estimates as in Fig. 4.16(a). In this example, we vary the number K of training blocks while fixing the SNR to be 0 dB. There are only 3 in-band paths with DODs $\theta_{\mathrm{t}} = -7°, -2°, 3°$ and DOAs $\theta_{\mathrm{r}} = -4°, 1°, 6°$. All the other simulation parameters are the same as in Example 3. Then we use (4.70) to compute the corresponding oversampling factors $K_{\mathrm{os},p}$, $p \in \{\mathrm{t}, \mathrm{r}\}$, for beam search methods. With such $K_{\mathrm{os},p}$, the RMSE of hybrid CBS equals the off-grid error of beam search methods, and is a lower bound for their RMSEs, as plotted in Fig. 4.16(a). Then, we can evaluate the pilot overhead of sequential beam search [6] and hierarchical beam search [6] using (4.68) and (4.69), respectively. The other parameters are chosen as $M_{\mathrm{hf}} = i_{\mathrm{LY}} = 2$ as in [6]. As shown in Fig. 4.16(b), hybrid CBS requires the smallest number of measurements.

4.9 Concluding Remarks

In this chapter, 2-D hybrid CBS is proposed for mmWave MIMO channel estimation. High-resolution gridless subspace methods like 2-D unitary ESPRIT can be readily applied to the combiner output without additional preparation since CBS preserves the Vandermonde structure of a ULA output. The transmitter precoder is designed as an expander followed by a CBS filter, whereas the receiver combiner is designed as a CBS filter followed by a decimator. We propose two decimation and expansion schemes: uniform and nonuniform. In the latter scheme, the expansion and decimation are nonuniform, corresponding to antenna locations of a virtual transmit dilated sparse array and receive dilated sparse array. Thus, we can use the 2-D coarray method to increase the number of identifiable paths. More interestingly, it is possible to get better DOD and DOA estimation performance due to the larger array aperture resulting from the dilation and sparse array structure. This can be done for quite a few paths with a limited number of RF chains. Besides, a method for decorrelating the path gains is proposed. The method is based on introducing CBS filter delays which vary with time blocks. With the method, we can apply 2-D coarray methods even for correlated path gains. One can get better decorrelating ability by designing the probability distribution of the delays. By sim-

ulations, we show that hybrid CBS yields smaller estimation errors than previous non-beam-search methods such as DFT beamspace MUSIC and DFT beamspace ESPRIT when having the same pilot overhead. Also, hybrid CBS requires smaller pilot overhead to achieve the same estimation error as beam search methods such as hierarchical beam search.

Recently, a new technology, reconfigurable intelligent surface (RIS) [86, 105], has been proposed by many to solve the issue of sensitivity to blockages for mmWave communications. With RIS, channel estimation is challenging because RIS is passive without active RF chains or computational power, and the number of RIS elements is often large. Hence, it is important to design low-complexity algorithms for RIS-aided mmWave MIMO channel estimation. It is particularly promising to adapt hybrid CBS to the RIS framework since hybrid CBS has low complexity and good estimation performance.

Chapter 5

THEORETICAL ANALYSIS OF 1-D CONVOLUTIONAL BEAMSPACE

5.1 Introduction

As proposed in Chapter 2, convolutional beamspace (CBS) is a beamspace method for direction-of-arrival (DOA) estimation. Given an N-sensor uniform linear array (ULA) output $\mathbf{x}$, traditionally one does *beamspace* processing by computing $\mathbf{y} = \mathbf{T}\mathbf{x}$ and estimating DOAs based on $\mathbf{y}$ instead of $\mathbf{x}$. Here $\mathbf{y}$ has length less than N, so there is reduction in computational complexity. Besides, beamspace methods usually enjoy higher DOA resolution and smaller bias compared to *element-space* methods (which estimate DOAs directly using $\mathbf{x}$) [2, 8, 16, 17]. For CBS, $\mathbf{T}$ is a banded Toeplitz matrix so that it is equivalent to filtering the array output. In Chapter 2, uniform decimation (downsampling) on $\mathbf{y}$ is further proposed to achieve significant complexity reduction. More precisely, we compute the average of the covariance matrices of all polyphase components [66] of $\mathbf{y}$, and DOAs can be estimated based on the eigenvalue decomposition of the average covariance. The idea of using FIR filtering in beamspace methods is also proposed in [106], but the detailed development of how we use the filter output to estimate DOAs only appears in this thesis and in our papers [23, 24].

This chapter aims to study MSE performance of the CBS method. Besides low computation and compatibility with root-MUSIC and ESPRIT, CBS also offers performance advantage over element-space in some scenarios. While CBS and element-space have similar DOA estimation errors for uncorrelated sources, the estimation error of CBS can be significantly smaller than that of element-space when there are correlated sources. This benefit of CBS is demonstrated in Chapter 2 mainly through numerical examples, while theoretical MSE analysis is given only for limited simple cases. Moreover, some details are bypassed in the analysis in Chapter 2, so the results are only approximations. One main goal of this chapter is thus to develop a rigorous and more accurate analysis for the MSE performance of CBS.

The foundations for the analysis of MSE performance for MUSIC and root-MUSIC were laid many decades ago in the classic papers [26, 56]. We extend the analysis

to CBS in this chapter. According to the results in [26, 56], MUSIC and root-MUSIC asymptotically have the same MSE in element-space, and we will show that it is also true for CBS. Compared to element-space, we have to tackle two additional complications in CBS. First, the different polyphase components of $\mathbf{y}$ are not independent, so it is more difficult to derive the asymptotic probability distribution of the eigenvectors of the average finite-snapshot covariance matrix. In the traditional asymptotic theory for principal component analysis [107], only independent observations are considered. This is directly applicable to element-space, but some modifications are required to adapt it to CBS. Second, the filter output $\mathbf{y}$ is represented only by the sources in the filter passband since we assume the attenuated stopband sources, if any, can be neglected, as shown in Chapter 2. This leads to an additional error term that should be analyzed. As we shall see, the effect of these filtered stopband sources is a bias, and it can be analyzed separately from the variance term. A first-order perturbation analysis is adopted. Furthermore, this bias term (squared) is often much smaller than the variance term if the CBS filter has reasonably good stopband attenuation and if the stopband source power is not very large.

In addition, the Cramér–Rao Bound (CRB) [43, 44, 108] for CBS is also derived in this chapter. Conventionally, the CRB offers a lower bound on the variances of unbiased estimates of parameters. As explained earlier, CBS yields biased DOA estimates if there are stopband sources. A modified form of CRB for a biased estimator of a scalar parameter is given in [109]. In our case, a biased estimator of a vector of parameters, i.e., DOAs, is considered, and we show that it can be viewed as an unbiased estimator for some transformation of the parameters. Hence, we can use the CRB for transformation of parameters [43] to get a modified form of lower bound on the variances of CBS DOA estimates. This bound depends on the Jacobian matrix of the transformation and can be numerically computed. Moreover, assuming the stopband sources are reasonably attenuated, the modified bound can be well approximated by the original CRB for unbiased estimates. We study the CRB under the stochastic model [44, 49], where the source amplitudes are assumed random. We derive two forms of CRB expressions. Form 1 is in the same style as the CRB in [49]. Although it is not the focus of this chapter, Form 1 can yield a CRB even when the noise is non-white with a singular covariance. On the other hand, Form 2 is in the same style as the CRB in [44] and offers some important insight. For example, the CRB for a DOA is approximately inversely proportional to its SNR and approximately independent of source powers of other DOAs. A necessary and

sufficient condition for the CBS CRB to exist is also given. Although the main goal of this chapter is to study the MSE and CRB for CBS, all the results also apply to element-space because element-space can be viewed as a special case of CBS.

Chapter outline: The basics of convolutional beamspace (CBS) are reviewed in Sec. 5.2, and notions for aiding later analysis are also presented. MSE for CBS is analyzed theoretically in Sec. 5.3. Cramér–Rao Bound (CRB) for CBS is then derived in Sec. 5.4. In particular, two forms of CRB expressions are proposed in Sec. 5.4.1 and Sec. 5.4.2. Numerical examples are shown in Sec. 5.5 to verify the theory. Finally, Sec. 5.6 concludes the chapter. Appendices 5.A, 5.B, and 5.C contain detailed proofs of the results.

5.2 Convolutional Beamspace

We consider an N-sensor ULA with sensor spacing $\lambda/2$, and assume D monochromatic plane waves of wavelength λ impinging on the array with DOAs $\theta_i \in [-\pi/2, \pi/2)$ measured from the normal to the line of array. The array output is thus

$$\mathbf{x} = \mathbf{A}\mathbf{c} + \mathbf{e}, \tag{5.1}$$

where $\mathbf{c}$ contains source amplitudes c_i, and $\mathbf{e}$ is additive noise. The array manifold $\mathbf{A} = [\mathbf{a}_N(\omega_1)\ \mathbf{a}_N(\omega_2)\ \cdots\ \mathbf{a}_N(\omega_D)]$, where $\omega_i = \pi \sin\theta_i$ and

$$\mathbf{a}_N(\omega) = [1\ e^{j\omega}\ e^{j2\omega}\ \cdots\ e^{j(N-1)\omega}]^T \tag{5.2}$$

for any positive integer N. The stochastic (also known as unconditional) model [44] is considered. That is, we assume $\mathbf{c}$ is a circularly-symmetric complex Gaussian random vector with covariance $\mathbf{P}$, which can be non-diagonal if sources are correlated. The noise $\mathbf{e}$ is assumed circularly-symmetric complex Gaussian with covariance $p_{\mathrm{e}}\mathbf{I}$ and uncorrelated with the sources. To apply subspace methods like MUSIC [52] or root-MUSIC [26], we compute the array output covariance

$$\mathbf{R}_{\mathbf{xx}} \triangleq \mathrm{E}[\mathbf{x}\mathbf{x}^H] = \mathbf{A}\mathbf{P}\mathbf{A}^H + p_{\mathrm{e}}\mathbf{I}. \tag{5.3}$$

Then the DOAs ω_i can be estimated by finding the signal and noise subspaces, which are spanned by appropriate subsets of eigenvectors of $\mathbf{R}_{\mathbf{xx}}$.

In Chapter 2, a new beamspace method, called convolutional beamspace (CBS) is proposed. In CBS, the ULA output $x(n), 0 \le n \le N-1$ is convolved with an FIR filter $H(z) = \sum_{n=0}^{L-1} h(n) z^{-n}$ to obtain the output $y(n)$, where $L < N$. Discarding the

transient samples $y(0), \ldots, y(L-2)$, we then collect the steady-state samples in a vector

$$\begin{aligned}\mathbf{y} &\triangleq [y(L-1)\ y(L)\ \cdots\ y(N-1)]^T = \mathbf{Hx} \\ &= \mathbf{A}_L\mathbf{d} + \mathbf{He},\end{aligned} \tag{5.4}$$

where

$$\mathbf{H} = \begin{bmatrix} h(L-1) & \cdots & h(0) & 0 & \cdots & 0 \\ 0 & h(L-1) & \cdots & h(0) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & h(L-1) & \cdots & h(0) \end{bmatrix}$$

is a $(N-L+1) \times N$ banded Toeplitz matrix,

$$\mathbf{A}_L = [\mathbf{a}_{N-L+1}(\omega_1)\ \cdots\ \mathbf{a}_{N-L+1}(\omega_D)], \tag{5.5}$$

and $\mathbf{d} = \mathbf{D}_H\mathbf{c}$ with

$$\mathbf{D}_H = \mathrm{diag}(H(e^{j\omega_1})e^{j\omega_1(L-1)}, \ldots, H(e^{j\omega_D})e^{j\omega_D(L-1)}). \tag{5.6}$$

Like the original array output $\mathbf{x}$, the CBS output $\mathbf{y}$ is represented in terms of a Vandermonde matrix, i.e., $\mathbf{A}_L$. Hence, we can compute the covariance

$$\mathbf{R}_{\mathbf{yy}} = \mathbf{A}_L\mathbf{R}_{\mathbf{dd}}\mathbf{A}_L^H + p_{\mathrm{e}}\mathbf{H}\mathbf{H}^H, \tag{5.7}$$

where

$$\mathbf{R}_{\mathbf{dd}} = \mathrm{E}[\mathbf{d}\mathbf{d}^H] = \mathbf{D}_H\mathbf{P}\mathbf{D}_H^H, \tag{5.8}$$

and estimate DOAs using root-MUSIC or ESPRIT without any further adjustment or processing to the data. Note from (5.6) that the source amplitudes c_i are filtered by the frequency response $H(e^{j\omega_i})$. We assume signals in the filter stopband are well attenuated, so $\mathbf{y}$ contains only those DOAs that fall in the passband of $H(e^{j\omega})$. Without loss of generality, assume $\omega_1, \ldots, \omega_{D_0}$ are in the passband. Then, $\mathbf{y} \approx \mathbf{A}_{L,0}\mathbf{d}_0 + \mathbf{He}$, where $\mathbf{A}_{L,0}$ has the first D_0 columns of $\mathbf{A}_L$, and $\mathbf{d}_0$ has the first D_0 entries of $\mathbf{d}$.

Since $\mathbf{y}$ contains only passband sources, we can decimate $\mathbf{y}$ without causing ambiguity. This gives us complexity reduction, which is an integral part of any beamspace

method. In particular, if $H(e^{j\omega})$ has passband width $2\pi/M$ for some integer M, we can decimate $\mathbf{y}$ by M and obtain

$$\mathbf{v}_l = \mathbf{D}_l\mathbf{y} = \mathbf{A}_{\text{dec}}\mathbf{d}_l + \mathbf{D}_l\mathbf{H}\mathbf{e} \approx \mathbf{A}_{\text{dec},0}\mathbf{d}_{l,0} + \mathbf{D}_l\mathbf{H}\mathbf{e}, \tag{5.9}$$

where

$$\mathbf{D}_l = \left[\boldsymbol{\delta}_l^{(N-L+1)} \;\; \boldsymbol{\delta}_{l+M}^{(N-L+1)} \;\; \cdots \;\; \boldsymbol{\delta}_{l+(J-1)M}^{(N-L+1)}\right]^T \tag{5.10}$$

is a decimation matrix, $\mathbf{A}_{\text{dec}} = [\mathbf{a}_J(M\omega_1) \;\cdots\; \mathbf{a}_J(M\omega_D)]$, $J = (N-L+1)/M$ (assumed an integer for simplicity), $\mathbf{d}_l$ has entries

$$[\mathbf{d}_l]_i = c_i e^{j(L-1+l)\omega_i} H(e^{j\omega_i}), \tag{5.11}$$

$\mathbf{A}_{\text{dec},0}$ has the first D_0 columns of $\mathbf{A}_{\text{dec}}$, and $\mathbf{d}_{l,0}$ has the first D_0 entries of $\mathbf{d}_l$. In (5.9), l can take values $0, \ldots, M-1$, corresponding to the M polyphase components [66] of $\mathbf{y}$. We will estimate only the D_0 passband DOAs based on $\mathbf{v}_l$. We assume $D_0 < J$ for MUSIC to identify DOAs without ambiguity [52]. To avoid wasting data, we compute the average covariance

$$\mathbf{R}_{\text{ave}} = \frac{1}{M}\sum_{l=0}^{M-1}\mathbf{R}_{v_l} = \mathbf{A}_{\text{dec}}\breve{\mathbf{R}}_d\mathbf{A}_{\text{dec}}^H + p_{\text{e}}\mathbf{G}_{\text{dec}}, \tag{5.12}$$

where $\breve{\mathbf{R}}_d$ is $\mathbf{R}_{d_l} = \mathrm{E}[\mathbf{d}_l\mathbf{d}_l^H]$ averaged over l (polyphase index),

$$\mathbf{R}_{v_l} = \mathrm{E}[\mathbf{v}_l\mathbf{v}_l^H] = \mathbf{A}_{\text{dec}}\mathbf{R}_{d_l}\mathbf{A}_{\text{dec}}^H + p_{\text{e}}\mathbf{G}_{\text{dec}}, \tag{5.13}$$

and $\mathbf{G}_{\text{dec}} \triangleq \mathbf{D}_l\mathbf{H}\mathbf{H}^H\mathbf{D}_l$ is independent of l. To aid our later analysis, we express

$$\mathbf{R}_{\text{ave}} = \mathbf{A}_{\text{dec},0}\breve{\mathbf{R}}_{d,0}\mathbf{A}_{\text{dec},0}^H + \delta\mathbf{R} + p_{\text{e}}\mathbf{G}_{\text{dec}}, \tag{5.14}$$

where the covariance perturbation

$$\delta\mathbf{R} = \mathbf{A}_{\text{dec}}\breve{\mathbf{R}}_d\mathbf{A}_{\text{dec}}^H - \mathbf{A}_{\text{dec},0}\breve{\mathbf{R}}_{d,0}\mathbf{A}_{\text{dec},0}^H \tag{5.15}$$

contains the auto-covariance of stopband sources and the cross-covariance between passband and stopband sources. This covariance perturbation is due to the filtered stopband sources, so we assume $\|\delta\mathbf{R}\|$ is small. In (5.15), $\breve{\mathbf{R}}_{d,0}$ is $\mathbf{R}_{d_{l,0}} = \mathrm{E}[\mathbf{d}_{l,0}\mathbf{d}_{l,0}^H]$ averaged over l. We assume that the CBS filter is a spectral factor of a Nyquist(M) filter so that $\mathbf{G}_{\text{dec}} = \mathbf{I}$ (see Sec. 2.2.4). That is, the noise after filtering and decimation remains white. Then we compute the eigenvalue decomposition

$$\mathbf{R}_{\text{ave}} = \mathbf{E}_{\text{s}}\boldsymbol{\Lambda}_{\text{s}}\mathbf{E}_{\text{s}}^H + \mathbf{E}_{\text{n}}\boldsymbol{\Lambda}_{\text{n}}\mathbf{E}_{\text{n}}^H, \tag{5.16}$$

where $\mathbf{E}_{\rm s} = [\mathbf{e}_1 \cdots \mathbf{e}_{D_0}]$ and $\mathbf{E}_{\rm n} = [\mathbf{e}_{D_0+1} \cdots \mathbf{e}_J]$ contain the signal and noise eigenvectors respectively, and $\mathbf{\Lambda}_{\rm s} = \text{diag}(\lambda_1, \ldots, \lambda_{D_0})$ and $\mathbf{\Lambda}_{\rm n} = \text{diag}(\lambda_{D_0+1}, \ldots, \lambda_J)$ contain the corresponding eigenvalues in descending order. Note that only the first D_0 eigenvalues are assumed dominant and correspond to signals. Then we can estimate the passband DOAs using MUSIC [52] or root-MUSIC [26]. Considering MUSIC as an example, we evaluate the MUSIC spectrum

$$P(\omega) = (\mathbf{a}_J^H(M\omega)\mathbf{E}_{\rm n}\mathbf{E}_{\rm n}^H\mathbf{a}_J(M\omega))^{-1} \tag{5.17}$$

on a dense grid of potential DOAs and identify local maxima as the estimates of $M\omega_i \bmod 2\pi$, or equivalently $\omega_i + 2\pi s_i/M$ for some integers s_i. Since ω_i are known to be in the passband of $H(e^{j\omega})$ which has width $2\pi/M$, the ambiguities s_i can be resolved. In practice, we use a finite number, say K, of independent snapshots to estimate the covariance matrix (5.12). That is, we compute noise subspace estimate $\widehat{\mathbf{E}}_{\rm n}$ based on

$$\widehat{\mathbf{R}}_{\rm ave} = \frac{1}{KM}\sum_{l=0}^{M-1}\sum_{k=1}^{K}\mathbf{v}_l[k]\mathbf{v}_l^H[k] \tag{5.18}$$

and then evaluate the MUSIC spectrum (5.17).

5.3 MSE Analysis for CBS

The goal of this section is to derive the MSE performance when we use MUSIC [52] or root-MUSIC [26] to estimate DOAs based on (5.18). In Chapter 2, comparison of MSE performance between CBS and element-space is given mainly based on numerical examples, and theoretical analysis of CBS MSE is given only for limited simple cases in an approximated way. In the following, we present a rigorous and more accurate MSE analysis for CBS. The results are derived based on the asymptotic eigenvector distribution of covariance. Also, we assume that the stopband attenuation of the CBS filter is large enough so that the filtered stopband sources can be dealt with first-order perturbation analysis.

MSE of DOA estimates for element-space is analyzed for MUSIC in [56] and for root-MUSIC in [26], and these papers have remained the foundation for such analysis for many years. Here we extend this analysis to CBS. In fact, the probability distributions of DOA estimation errors and thus MSEs in element-space are asymptotically the same for MUSIC and root-MUSIC. We will show that these are also true for CBS. Compared to element-space analysis, there are two complications we should deal with for CBS. First, $\mathbf{v}_l[k]$ is not independent of $\mathbf{v}_m[k]$ for $l \neq m$, which

makes the derivation of the distribution of the eigenvectors of (5.18) more difficult. Independent observations are assumed in the traditional asymptotic theory for principal component analysis [107], which is naturally applicable to element-space. We modify the method so that it can be used for CBS. Second, the presence of $\delta\mathbf{R}$ due to the filtered stopband sources is an additional source of estimation errors. We will show that the effect of the filtered stopband sources is a bias and can be analyzed separately from the variance term as we use a first-order perturbation analysis. To this end, we define $\mathbf{R}_{\mathbf{yy},0} = \mathbf{R}_{\mathbf{yy}}\big|_{\delta\mathbf{R}=\mathbf{O}}$ and

$$\mathbf{R}_{\text{ave},0} = \mathbf{R}_{\text{ave}}\big|_{\delta\mathbf{R}=\mathbf{O}} = \mathbf{E}_{\text{s},0}\mathbf{\Lambda}_{\text{s},0}\mathbf{E}_{\text{s},0}^H + \mathbf{E}_{\text{n},0}\mathbf{\Lambda}_{\text{n},0}\mathbf{E}_{\text{n},0}^H \tag{5.19}$$

to be the covariance matrices when $\delta\mathbf{R}$ is set to be zero (equivalently, when the stopband sources are nulled), where $\mathbf{E}_{\text{s},0} = [\mathbf{e}_{1,0} \ \cdots \ \mathbf{e}_{D_0,0}]$ and $\mathbf{E}_{\text{n},0} = [\mathbf{e}_{D_0+1,0} \ \cdots \ \mathbf{e}_{J,0}]$ contain the signal and noise eigenvectors respectively, and $\mathbf{\Lambda}_{\text{s},0} = \text{diag}(\lambda_{1,0}, \ldots, \lambda_{D_0,0})$ and $\mathbf{\Lambda}_{\text{n},0} = \text{diag}(\lambda_{D_0+1,0}, \ldots, \lambda_{J,0}) = p_{\text{e}}\mathbf{I}$ contain the corresponding eigenvalues in the diagonals. Here we only have D_0 signal eigenvectors because the $D - D_0$ stopband sources are null. According to the theory of perturbation of Hermitian matrices [110, 111], the signal eigenvectors of $\mathbf{R}_{\text{ave}}$ are

$$\mathbf{e}_l = \mathbf{e}_{l,0} + \sum_{\substack{r=1 \\ r\neq l}}^{J} \frac{\mathbf{e}_{r,0}^H \delta\mathbf{R}\mathbf{e}_{l,0}}{\lambda_{l,0} - \lambda_{r,0}}\mathbf{e}_{r,0}, \quad l = 1, \ldots, D_0 \tag{5.20}$$

if $\|\delta\mathbf{R}\|$ is small compared to the norm of the first term in (5.14). Here we assume the signal eigenvalues are distinct so that the denominators $\lambda_{l,0} - \lambda_{r,0} \neq 0$. This is true with probability one if the DOAs ω_i are randomly distributed with some continuous probability distribution.

In the following, we first present a lemma for the distribution of the signal eigenvectors $\hat{\mathbf{e}}_{l,0}$ of the K-snapshot estimate $\widehat{\mathbf{R}}_{\text{ave},0}$ when we null the stopband sources. This can be viewed as a generalized version of its element-space counterpart, Lemma 3.1 in [56].

Lemma 5.1 *The signal eigenvectors $\hat{\mathbf{e}}_{l,0}$ of $\widehat{\mathbf{R}}_{\text{ave},0}$ are asymptotically (for large K) jointly complex Gaussian with means $\mathbf{e}_{l,0}$, covariances*

$$\text{E}[(\hat{\mathbf{e}}_{l,0} - \mathbf{e}_{l,0})(\hat{\mathbf{e}}_{r,0} - \mathbf{e}_{r,0})^H] = \sum_{\substack{i=1 \\ i\neq l}}^{J}\sum_{\substack{k=1 \\ k\neq r}}^{J} \frac{\text{tr}\left(\widetilde{\mathbf{R}}_{\mathbf{yy}}^{(r,l)}(\widetilde{\mathbf{R}}_{\mathbf{yy}}^{(k,i)})^H\right)}{KM^2(\lambda_{l,0} - \lambda_{i,0})(\lambda_{r,0} - \lambda_{k,0})}\mathbf{e}_{i,0}\mathbf{e}_{k,0}^H \tag{5.21}$$

and relation matrices

$$\mathrm{E}[(\hat{\mathbf{e}}_{l,0}-\mathbf{e}_{l,0})(\hat{\mathbf{e}}_{r,0}-\mathbf{e}_{r,0})^T]=\sum_{\substack{i=1\\ i\neq l}}^{J}\sum_{\substack{k=1\\ k\neq r}}^{J}\frac{\mathrm{tr}\left(\widetilde{\mathbf{R}}_{\mathbf{yy}}^{(k,l)}(\widetilde{\mathbf{R}}_{\mathbf{yy}}^{(r,i)})^H\right)}{KM^2(\lambda_{l,0}-\lambda_{i,0})(\lambda_{r,0}-\lambda_{k,0})}\mathbf{e}_{i,0}\mathbf{e}_{k,0}^T \quad (5.22)$$

for $1\le l,r\le D_0$*, where* $\widetilde{\mathbf{R}}_{\mathbf{yy}}^{(p,q)}\in\mathbb{C}^{M\times M}$ *for* $1\le p,q\le J$ *are submatrices of*

$$\widetilde{\mathbf{R}}_{\mathbf{yy}}=(\mathbf{E}_0^H\otimes\mathbf{I}_M)\mathbf{R}_{\mathbf{yy},0}(\mathbf{E}_0\otimes\mathbf{I}_M) \quad (5.23)$$

$$\triangleq\begin{bmatrix}\widetilde{\mathbf{R}}_{\mathbf{yy}}^{(1,1)} & \cdots & \widetilde{\mathbf{R}}_{\mathbf{yy}}^{(1,J)}\\ \vdots & \ddots & \vdots\\ \widetilde{\mathbf{R}}_{\mathbf{yy}}^{(J,1)} & \cdots & \widetilde{\mathbf{R}}_{\mathbf{yy}}^{(J,J)}\end{bmatrix}. \quad (5.24)$$

Here $\mathbf{E}_0=[\mathbf{E}_{\mathrm{s},0}\ \mathbf{E}_{\mathrm{n},0}]$ *contains the eigenvectors in (5.19).*

Proof: See Appendix 5.A. ■

Note that although the signals and noise are circularly-symmetric complex Gaussian, the eigenvector estimates $\hat{\mathbf{e}}_{l,0}$ are not in general circularly symmetric since the relation matrices can be nonzero. Our expressions of covariance and relation matrices reduce to those in [56] if we set $L=1$, the CBS filter $H(z)=1$, the decimation ratio $M=1$, and $B_i=0$ (since all sources are in the "passband"). In this case, considering (5.19), we can simplify (5.23) as

$$\widetilde{\mathbf{R}}_{\mathbf{yy}}=\mathbf{E}_0^H\mathbf{R}_{\mathbf{yy},0}\mathbf{E}_0=\mathbf{E}_0^H\mathbf{R}_{\mathrm{ave},0}\mathbf{E}_0=\mathbf{\Lambda}_0, \quad (5.25)$$

where $\mathbf{\Lambda}_0=\mathrm{diag}(\mathbf{\Lambda}_{\mathrm{s},0},\mathbf{\Lambda}_{\mathrm{n},0})$. Hence, we obtain the covariances

$$\mathrm{E}[(\hat{\mathbf{e}}_{l,0}-\mathbf{e}_{l,0})(\hat{\mathbf{e}}_{r,0}-\mathbf{e}_{r,0})^H]=\sum_{\substack{i=1\\ i\neq l}}^{J}\frac{\lambda_{l,0}\lambda_{i,0}}{K(\lambda_{l,0}-\lambda_{i,0})^2}\mathbf{e}_{i,0}\mathbf{e}_{i,0}^H\cdot\delta_{lr} \quad (5.26)$$

and relation matrices

$$\mathrm{E}[(\hat{\mathbf{e}}_{l,0}-\mathbf{e}_{l,0})(\hat{\mathbf{e}}_{r,0}-\mathbf{e}_{r,0})^T]=\frac{-\lambda_{l,0}\lambda_{r,0}}{K(\lambda_{l,0}-\lambda_{r,0})^2}\mathbf{e}_{r,0}\mathbf{e}_{l,0}^T(1-\delta_{lr}). \quad (5.27)$$

These element-space expressions are much simpler than the CBS ones because $\widetilde{\mathbf{R}}_{\mathbf{yy}}$ is diagonal for element-space. Also, as mentioned earlier, CBS eigenvector estimates $\hat{\mathbf{e}}_{l,0}$ are generally not circularly-symmetric complex Gaussian. However, for element-space, they are circularly-symmetric complex Gaussian since $\mathrm{E}[(\hat{\mathbf{e}}_{l,0}-\mathbf{e}_{l,0})(\hat{\mathbf{e}}_{l,0}-\mathbf{e}_{l,0})^T]=0$.

Next, using Lemma 5.1 and (5.20), we can derive the distribution of the DOA estimation errors $\hat{\omega}_i-\omega_i$ as follows.

Theorem 5.1 *The CBS DOA estimation errors $\hat{\omega}_i - \omega_i$ for passband sources, when either MUSIC or root-MUSIC is used with (5.18), are asymptotically (for large K) jointly Gaussian distributed with means B_i and cross-correlations*

$$\begin{aligned}\mathrm{E}[(\hat{\omega}_i - \omega_i)(\hat{\omega}_k - \omega_k)] = B_i B_k + \frac{1}{2KM^2 g(\tilde{\omega}_i) g(\tilde{\omega}_k)} \mathrm{Re}\Bigg\{ & \\ \sum_{l=1}^{D_0} \sum_{r=1}^{D_0} \mathbf{e}_{l,0}^H \mathbf{a}_J(\tilde{\omega}_i) \mathbf{a}_J^H(\tilde{\omega}_k) \mathbf{e}_{r,0} \dot{\mathbf{a}}_J^H(\tilde{\omega}_i) \mathbf{E}_{\mathrm{n},0} \mathbf{B}_{lr} \mathbf{E}_{\mathrm{n},0}^H \dot{\mathbf{a}}_J(\tilde{\omega}_k) & \\ + \sum_{l=1}^{D_0} \sum_{r=1}^{D_0} \mathbf{e}_{l,0}^H \mathbf{a}_J(\tilde{\omega}_i) \mathbf{a}_J^T(\tilde{\omega}_k) \mathbf{e}_{r,0}^* \dot{\mathbf{a}}_J^H(\tilde{\omega}_i) \mathbf{E}_{\mathrm{n},0} \mathbf{C}_{lr} \mathbf{E}_{\mathrm{n},0}^T \dot{\mathbf{a}}_J^*(\tilde{\omega}_k) \Bigg\} &\end{aligned}$$

for $1 \le i, k \le D_0$, where $\tilde{\omega}_i = M\omega_i$,

$$B_i = \frac{\mathrm{Re}\{\mathbf{a}_J^H(\tilde{\omega}_i) \sum_{l=1}^{D_0} (\mathbf{e}_{l,0} \delta \mathbf{e}_l^H + \delta \mathbf{e}_l \mathbf{e}_{l,0}^H) \dot{\mathbf{a}}_J(\tilde{\omega}_i)\}}{M g(\tilde{\omega}_i)}, \tag{5.28}$$

$$\delta \mathbf{e}_l = \sum_{\substack{r=1 \\ r \ne l}}^{J} \frac{\mathbf{e}_{r,0}^H \delta \mathbf{R} \mathbf{e}_{l,0}}{\lambda_{l,0} - \lambda_{r,0}} \mathbf{e}_{r,0}, \quad l = 1, \ldots, D_0, \tag{5.29}$$

$\dot{\mathbf{a}}_J(\omega) = \frac{\mathrm{d}}{\mathrm{d}\omega} \mathbf{a}_J(\omega)$, $g(\omega) = \dot{\mathbf{a}}_J^H(\omega) \mathbf{E}_{\mathrm{n},0} \mathbf{E}_{\mathrm{n},0}^H \dot{\mathbf{a}}_J(\omega)$, $\mathbf{B}_{lr}$ and $\mathbf{C}_{lr}$ are $(J-D_0)\times(J-D_0)$ matrices with entries

$$[\mathbf{B}_{lr}]_{m,n} = \frac{\mathrm{tr}\left(\widetilde{\mathbf{R}}_{\mathbf{yy}}^{(r,l)} (\widetilde{\mathbf{R}}_{\mathbf{yy}}^{(D_0+n, D_0+m)})^H\right)}{M^2 (\lambda_{l,0} - p_{\mathrm{e}})(\lambda_{r,0} - p_{\mathrm{e}})} \tag{5.30}$$

and

$$[\mathbf{C}_{lr}]_{m,n} = \frac{\mathrm{tr}\left(\widetilde{\mathbf{R}}_{\mathbf{yy}}^{(D_0+n,l)} (\widetilde{\mathbf{R}}_{\mathbf{yy}}^{(r, D_0+m)})^H\right)}{M^2 (\lambda_{l,0} - p_{\mathrm{e}})(\lambda_{r,0} - p_{\mathrm{e}})} \tag{5.31}$$

for $1 \le m, n \le J - D_0$, $1 \le l, r \le D_0$, and $\widetilde{\mathbf{R}}_{\mathbf{yy}}^{(p,q)}$, $1 \le p, q \le J$ are as defined in (5.24).

Proof: See Appendix 5.B. ■

In particular, the MSEs $\mathrm{E}[(\hat{\omega}_i - \omega_i)^2]$ of the passband DOAs can be obtained by letting $k = i$ in Theorem 5.1:

$$\mathrm{E}[(\hat{\omega}_i - \omega_i)^2] = B_i^2 + V_i, \tag{5.32}$$

where

$$\begin{aligned}V_i = \frac{1}{2KM^2 g^2(\tilde{\omega}_i)} \mathrm{Re}\Bigg\{ & \sum_{l=1}^{D_0} \sum_{r=1}^{D_0} \mathbf{e}_{l,0}^H \mathbf{a}_J(\tilde{\omega}_i) \mathbf{a}_J^H(\tilde{\omega}_i) \mathbf{e}_{r,0} \dot{\mathbf{a}}_J^H(\tilde{\omega}_i) \mathbf{E}_{\mathrm{n},0} \mathbf{B}_{lr} \mathbf{E}_{\mathrm{n},0}^H \dot{\mathbf{a}}_J(\tilde{\omega}_i) \\ & + \sum_{l=1}^{D_0} \sum_{r=1}^{D_0} \mathbf{e}_{l,0}^H \mathbf{a}_J(\tilde{\omega}_i) \mathbf{a}_J^T(\tilde{\omega}_i) \mathbf{e}_{r,0}^* \dot{\mathbf{a}}_J^H(\tilde{\omega}_i) \mathbf{E}_{\mathrm{n},0} \mathbf{C}_{lr} \mathbf{E}_{\mathrm{n},0}^T \dot{\mathbf{a}}_J^*(\tilde{\omega}_i) \Bigg\}\end{aligned}$$

for $i = 1, \ldots, D_0$. As mentioned earlier, since $\mathbf{v}_l[k]$ is not independent of $\mathbf{v}_m[k]$ for $l \neq m$ in (5.18), it is more difficult to analyze CBS MSE. One can see such complications through the $M \times M$ matrices $\widetilde{\mathbf{R}}_{\mathbf{yy}}^{(p,q)}$ appearing in (5.30) and (5.31). On the other hand, the presence of the filtered stopband sources simply results in a bias B_i in the DOA estimates. Thus, the MSE is given in the form of a bias-variance decomposition. As we shall see in the simulations (Fig. 5.4), the bias term is typically much smaller than the variance term if the CBS filter is properly designed with good stopband attenuation and if the stopband sources are not too powerful. Hence, we may ignore the bias term in many practical cases. Due to the complicated MSE expression, it is not easy to get further insight. However, we will present more insight based on the CRB expressions in Sec. 5.4.

Our MSE expression can be viewed as a generalized version of the element-space MSE expression in [56]. Our expression reduces to that in [56] if we set $L = 1$, the CBS filter $H(z) = 1$, the decimation ratio $M = 1$, and $B_i = 0$. Then again we have $\widetilde{\mathbf{R}}_{\mathbf{yy}} = \mathbf{\Lambda}_0$ as in (5.25) and thus

$$\mathbf{B}_{lr} = \frac{\lambda_{l,0} p_{\mathrm{e}} \mathbf{I}}{(\lambda_{l,0} - p_{\mathrm{e}})^2} \delta_{lr} \tag{5.33}$$

and $\mathbf{C}_{lr} = \mathbf{O}$ for all l, r. This is why we can get the much more simplified expression for element-space MSE

$$\mathrm{E}[(\hat{\omega}_i - \omega_i)^2] = \frac{1}{2Kg(\omega_i)} \sum_{l=1}^{D} \frac{\lambda_{l,0} p_{\mathrm{e}}}{(\lambda_{l,0} - p_{\mathrm{e}})^2} |\mathbf{e}_{l,0}^H \mathbf{a}_J(\omega_i)|^2 \tag{5.34}$$

given in [56]. Although it is known that CBS CRB cannot be smaller than element-space CRB [23, 78], it is still unclear whether CBS MSE can be smaller than element-space MSE for uncorrelated sources because of the complicated MSE expression for CBS. For correlated sources, CBS MSE can be much smaller than element-space MSE (see Fig. 5.5(b)).

5.4 Cramér–Rao Bound for CBS

In this section, we derive the Cramér–Rao bound (CRB) [43] for the DOA estimates based on the CBS output $\mathbf{y}$ in (5.4), under the stochastic model [44, 49]. The CRB offers a lower bound on the variances of unbiased estimates of parameters. However, as shown in Sec. 5.3, CBS yields biased DOA estimates if there are stopband sources. Hence, some extra care will be taken in order to derive the CRB for biased estimators [43]. We shall derive two forms of CRB expressions. Form 1 is in the same style as the CRB in [49]. It is derived from inverting the Fisher

information matrix [43], and a necessary and sufficient condition for the existence of the CRB is naturally obtained in this process. Although it is not the focus of this chapter, Form 1 also yields a CRB even when the noise is non-white with a singular covariance. Meanwhile, Form 2 is in the same style as the CRB in [44] and offers some additional insight as shown later.

The probability model for stochastic CRB [44, 49] based on K snapshots of CBS outputs is

$$\begin{bmatrix} \mathbf{y}[1] \\ \mathbf{y}[2] \\ \vdots \\ \mathbf{y}[K] \end{bmatrix} \sim \mathcal{CN}\left(\mathbf{0}, \begin{bmatrix} \mathbf{R_{yy}} & \mathbf{O} & \cdots & \mathbf{O} \\ \mathbf{O} & \mathbf{R_{yy}} & \cdots & \mathbf{O} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{O} & \mathbf{O} & \cdots & \mathbf{R_{yy}} \end{bmatrix} \right), \tag{5.35}$$

where $\mathbf{R_{yy}}$ is as in (5.7). The parameter vector for this probability model is

$$\boldsymbol{\alpha} = \left[[\omega_i]_{i=1}^{D} \;\; [p_i]_{i=1}^{D} \;\; [P_{ik}^{(\mathrm{r})}]_{i>k} \;\; [P_{ik}^{(\mathrm{i})}]_{i>k} \;\; p_{\mathrm{e}} \right]^T, \tag{5.36}$$

where p_i is the ith diagonal entry of $\mathbf{P}$, and $P_{ik}^{(\mathrm{r})}$ and $P_{ik}^{(\mathrm{i})}$ are the real and imaginary parts of the (i,k)-entry of $\mathbf{P}$, respectively. Thus, there are $D+D^2+1$ real parameters, among which only the D parameters $\boldsymbol{\omega} = [\omega_1 \;\cdots\; \omega_D]^T$ are of interest.

For any unbiased estimator $\hat{\boldsymbol{\alpha}}$ with $\mathrm{E}[\hat{\boldsymbol{\alpha}}] = \boldsymbol{\alpha}$, the CRB is given by [43]

$$\mathrm{CRB}_{\mathrm{unb}}(\boldsymbol{\alpha}) = [\boldsymbol{\mathcal{I}}(\boldsymbol{\alpha})]^{-1}, \tag{5.37}$$

where $\boldsymbol{\mathcal{I}}(\boldsymbol{\alpha})$ is the Fisher information matrix for the model, such that the covariance of the estimator

$$\mathrm{cov}(\hat{\boldsymbol{\alpha}}) \succeq \mathrm{CRB}_{\mathrm{unb}}(\boldsymbol{\alpha}). \tag{5.38}$$

As shown in Sec. 5.3, however, CBS yields biased estimates $\hat{\boldsymbol{\omega}}_0$ for the in-band (passband) DOAs $\boldsymbol{\omega}_0 = [\omega_1 \;\cdots\; \omega_{D_0}]^T$ if there are out-of-band (stopband) sources. A modified form of CRB for a biased estimator of a scalar parameter is given in [109]. To derive a bound for the biased vector estimator $\hat{\boldsymbol{\omega}}_0$, we consider its expectation $\mathrm{E}[\hat{\boldsymbol{\omega}}_0]$. It must be some function of the model parameters $\boldsymbol{\alpha}$, say, $\mathrm{E}[\hat{\boldsymbol{\omega}}_0] = \boldsymbol{\psi}(\boldsymbol{\alpha})$. Then $\hat{\boldsymbol{\omega}}_0$ can be viewed as an unbiased estimator for the transformation $\boldsymbol{\psi}(\cdot)$ of the parameter vector $\boldsymbol{\alpha}$. Hence, assuming $\boldsymbol{\psi}(\boldsymbol{\alpha})$ is differentiable, we can use the CRB for transformations [43] to obtain a lower bound on the covariance of the biased estimate $\hat{\boldsymbol{\omega}}_0$:

$$\mathrm{cov}(\hat{\boldsymbol{\omega}}_0) \succeq \frac{\partial \boldsymbol{\psi}(\boldsymbol{\alpha})}{\partial \boldsymbol{\alpha}} \mathrm{CRB}_{\mathrm{unb}}(\boldsymbol{\alpha}) \left(\frac{\partial \boldsymbol{\psi}(\boldsymbol{\alpha})}{\partial \boldsymbol{\alpha}} \right)^T, \tag{5.39}$$

where $\mathrm{CRB}_{\mathrm{unb}}(\boldsymbol{\alpha})$ is defined in (5.37), and $\frac{\partial \boldsymbol{\psi}}{\partial \boldsymbol{\alpha}}$ is the Jacobian matrix with entries

$$\left[\frac{\partial \boldsymbol{\psi}}{\partial \boldsymbol{\alpha}}\right]_{i,k} = \frac{\partial [\boldsymbol{\psi}]_i}{\partial [\boldsymbol{\alpha}]_k} = \begin{cases} 1 + \frac{\partial B_i}{\partial [\boldsymbol{\alpha}]_i}, & i = k \\ \frac{\partial B_i}{\partial [\boldsymbol{\alpha}]_k}, & i \neq k \end{cases}. \tag{5.40}$$

Here B_i are the biases defined in (5.28). Although the derivatives $\frac{\partial B_i}{\partial [\boldsymbol{\alpha}]_k}$ can be analytically computed from (5.28), the results will be lengthy due to the complicated dependence of $\mathbf{e}_{l,0}$ and $\lambda_{l,0}$ on $[\boldsymbol{\alpha}]_k$. Thus, we omit the tedious derivations, which may not give much insight. In this chapter, we only compute the derivatives numerically in the example of Fig. 5.4. Moreover, these derivatives $\frac{\partial B_i}{\partial [\boldsymbol{\alpha}]_k}$ are typically small since the covariance perturbation $\delta\mathbf{R}$ due to the filtered stopband sources is small. Thus,

$$\frac{\partial \boldsymbol{\psi}(\boldsymbol{\alpha})}{\partial \boldsymbol{\alpha}} \approx \begin{bmatrix} \mathbf{I} & \mathbf{O}_{D_0, D-D_0+D^2+1} \end{bmatrix} \tag{5.41}$$

so that we have

$$\mathrm{cov}(\hat{\boldsymbol{\omega}}_0) \geq \frac{\partial \boldsymbol{\psi}(\boldsymbol{\alpha})}{\partial \boldsymbol{\alpha}} \mathrm{CRB}_{\mathrm{unb}}(\boldsymbol{\alpha}) \left(\frac{\partial \boldsymbol{\psi}(\boldsymbol{\alpha})}{\partial \boldsymbol{\alpha}}\right)^T \tag{5.42}$$

$$\approx \mathrm{CRB}_{\mathrm{unb}}(\boldsymbol{\omega}_0), \tag{5.43}$$

where $\mathrm{CRB}_{\mathrm{unb}}(\boldsymbol{\omega}_0)$ is the top left $D_0 \times D_0$ block of $\mathrm{CRB}_{\mathrm{unb}}(\boldsymbol{\alpha})$. Note that (5.43) is precisely the CRB for CBS if there are no stopband sources so that $\hat{\boldsymbol{\omega}}_0$ is unbiased. If stopband sources exist, as we shall see in simulations (Fig. 5.4), usually (5.43) also gives a good lower bound. It does not make a big difference to consider (5.42) unless the out-of-band source power is extremely large. In the following, we focus on the $\boldsymbol{\omega}$-block $\mathrm{CRB}_{\mathrm{unb}}(\boldsymbol{\omega})$, i.e., the top left $D \times D$ block of $\mathrm{CRB}_{\mathrm{unb}}(\boldsymbol{\alpha})$ because it leads to more elegant and insightful expressions. Once we have $\mathrm{CRB}_{\mathrm{unb}}(\boldsymbol{\omega})$, we can immediately obtain $\mathrm{CRB}_{\mathrm{unb}}(\boldsymbol{\omega}_0)$ as a submatrix. Then in particular, we have that the variance

$$\mathrm{var}(\hat{\omega}_i) \geq [\mathrm{CRB}_{\mathrm{unb}}(\boldsymbol{\omega}_0)]_{i,i} \tag{5.44}$$

for each in-band DOA estimate $\hat{\omega}_i$. Now we are ready to derive Form 1 of the CRB by investigating the Fisher information matrix $\boldsymbol{\mathcal{I}}(\boldsymbol{\alpha})$.

5.4.1 Form 1

The (i,k)-entry of the Fisher information matrix $\boldsymbol{\mathcal{I}}(\boldsymbol{\alpha})$ for the model (5.35) can be derived as [49]

$$[\boldsymbol{\mathcal{I}}(\boldsymbol{\alpha})]_{i,k} = K \frac{\partial \mathbf{r}_{\mathbf{yy}}^H}{\partial [\boldsymbol{\alpha}]_i} \left(\mathbf{R}_{\mathbf{yy}}^T \otimes \mathbf{R}_{\mathbf{yy}}\right)^{-1} \frac{\partial \mathbf{r}_{\mathbf{yy}}}{\partial [\boldsymbol{\alpha}]_k}, \tag{5.45}$$

where $\mathbf{r}_{\mathbf{yy}} = \mathrm{vec}(\mathbf{R}_{\mathbf{yy}})$. Separating the parameters of interest from the other parameters, we have

$$\mathcal{I}(\boldsymbol{\alpha}) = K \begin{bmatrix} \mathbf{G}^H\mathbf{G} & \mathbf{G}^H\boldsymbol{\Delta}_1 \\ \boldsymbol{\Delta}_1^H\mathbf{G} & \boldsymbol{\Delta}_1^H\boldsymbol{\Delta}_1 \end{bmatrix}, \tag{5.46}$$

where

$$\mathbf{G} = \left(\mathbf{R}_{\mathbf{yy}}^T \otimes \mathbf{R}_{\mathbf{yy}}\right)^{-\frac{1}{2}} \begin{bmatrix} \dfrac{\partial \mathbf{r}_{\mathbf{yy}}}{\partial [\boldsymbol{\alpha}]_1} & \cdots & \dfrac{\partial \mathbf{r}_{\mathbf{yy}}}{\partial [\boldsymbol{\alpha}]_D} \end{bmatrix}, \tag{5.47}$$

$$\boldsymbol{\Delta}_1 = \left(\mathbf{R}_{\mathbf{yy}}^T \otimes \mathbf{R}_{\mathbf{yy}}\right)^{-\frac{1}{2}} \begin{bmatrix} \dfrac{\partial \mathbf{r}_{\mathbf{yy}}}{\partial [\boldsymbol{\alpha}]_{D+1}} & \cdots & \dfrac{\partial \mathbf{r}_{\mathbf{yy}}}{\partial [\boldsymbol{\alpha}]_{D+D^2+1}} \end{bmatrix}. \tag{5.48}$$

Here we assume $\mathbf{R}_{\mathbf{yy}}$ is positive definite so that $(\mathbf{R}_{\mathbf{yy}}^T \otimes \mathbf{R}_{\mathbf{yy}})$ is also positive definite. Then $(\mathbf{R}_{\mathbf{yy}}^T \otimes \mathbf{R}_{\mathbf{yy}})^{-1/2}$ denotes the inverse of its positive definite square root. For the CRB (5.37) to exist, we require $\mathcal{I}(\boldsymbol{\alpha})$ to be invertible, so we obtain the following theorem.

Theorem 5.2 *The CRB (5.37) exists if and only if*

$$\mathbf{M} = \begin{bmatrix} \mathbf{M}_1 & (\mathbf{A}_L\mathbf{D}_H)^* \otimes (\mathbf{A}_L\mathbf{D}_H) & \mathrm{vec}(\mathbf{H}\mathbf{H}^H) \end{bmatrix} \tag{5.49}$$

has full column rank, where

$$\mathbf{M}_1 = \dot{\mathbf{A}}_L^* \odot (\mathbf{A}_L\mathbf{D}_H\mathbf{P}\mathbf{D}_H^*) + (\mathbf{A}_L\mathbf{D}_H\mathbf{P}\mathbf{D}_H^*)^* \odot \dot{\mathbf{A}}_L, \tag{5.50}$$

and

$$\dot{\mathbf{A}}_L = \begin{bmatrix} \dot{\mathbf{a}}_{N-L+1}(\omega_1) & \cdots & \dot{\mathbf{a}}_{N-L+1}(\omega_D) \end{bmatrix} \tag{5.51}$$

with $\dot{\mathbf{a}}_{N-L+1}(\omega) = \frac{\mathrm{d}}{\mathrm{d}\omega}\mathbf{a}_{N-L+1}(\omega)$.

Proof: See Appendix 5.C. ■

Theorem 5.2 offers a necessary and sufficient condition for the CRB to exist. Since $\mathbf{M}$ has $(N-L+1)^2$ rows and $D+D^2+1$ columns, it can have full column rank only if $D < N-L+1$. In [49], a similar condition for the existence of the element-space CRB is given assuming sources are known to be uncorrelated. In the following, we assume that $\mathbf{M}$ indeed has full column rank. Now using the ideas of block matrix inversion and Schur complements [49], we can show that (5.37) and (5.46) imply that

$$\mathrm{CRB}_{\mathrm{unb}}(\boldsymbol{\omega}) = \frac{1}{K}\left(\mathbf{G}^H\boldsymbol{\Pi}_{\boldsymbol{\Delta}_1}^{\perp}\mathbf{G}\right)^{-1}, \tag{5.52}$$

where $\mathbf{\Pi}_{\mathbf{\Delta}_1}^{\perp} = \mathbf{I} - \mathbf{\Delta}_1(\mathbf{\Delta}_1^H\mathbf{\Delta}_1)^{-1}\mathbf{\Delta}_1^H$ denotes the orthogonal projection onto the null space of $\mathbf{\Delta}_1^H$. Computing the above derivatives and simplifying the results, we can obtain the CRB as follows.

Theorem 5.3 *If there are no stopband sources, the CRB for the DOAs $\boldsymbol{\omega}$ for CBS is*

$$\mathrm{CRB}_{\mathrm{unb}}(\boldsymbol{\omega}) = \frac{1}{K}\left(\mathbf{G}^H\mathbf{\Pi}_{\mathbf{\Delta}}^{\perp}\mathbf{G}\right)^{-1}, \tag{5.53}$$

where

$$\mathbf{G} = \left(\mathbf{R}_{\mathbf{yy}}^T \otimes \mathbf{R}_{\mathbf{yy}}\right)^{-\frac{1}{2}}\left(\dot{\mathbf{A}}_L^* \odot (\mathbf{A}_L\mathbf{D}_H\mathbf{P}\mathbf{D}_H^*) + (\mathbf{A}_L\mathbf{D}_H\mathbf{P}\mathbf{D}_H^*)^* \odot \dot{\mathbf{A}}_L\right), \tag{5.54}$$

$$\mathbf{\Delta} = \left(\mathbf{R}_{\mathbf{yy}}^T \otimes \mathbf{R}_{\mathbf{yy}}\right)^{-\frac{1}{2}}\left[(\mathbf{A}_L\mathbf{D}_H)^* \otimes (\mathbf{A}_L\mathbf{D}_H) \quad \mathrm{vec}(\mathbf{H}\mathbf{H}^H)\right]. \tag{5.55}$$

Here $\mathbf{A}_L$, $\mathbf{D}_H$, and $\dot{\mathbf{A}}_L$ are as defined in (5.5), (5.6), and (5.51). If stopband sources exist, (5.53) is an approximate bound.

Remark 1) Suppose there are stopband sources so that the CBS DOA estimator is biased. Then the exact CRB (5.42) can be computed from (5.40), (5.37), and (5.46) with $\mathbf{G}$ as in (5.54) and $\mathbf{\Delta}_1$ as in (5.141). In this chapter, we only numerically compute the derivatives in the Jacobian matrix in the example of Fig. 5.4.

Remark 2) By slightly modifying the proof of Theorem 5.3, one can verify that if the noise covariance is $p_{\mathrm{e}}\mathbf{R}_{\mathrm{e}}$ instead of $p_{\mathrm{e}}\mathbf{I}$ for any $\mathbf{R}_{\mathrm{e}}$ known a priori, then the CRB is as in (5.53)–(5.55) except that $\mathbf{H}\mathbf{H}^H$ is replaced by $\mathbf{H}\mathbf{R}_{\mathrm{e}}\mathbf{H}^H$ in (5.55). It is valid even when $\mathbf{R}_{\mathrm{e}}$ is singular. This result for a singular noise covariance cannot be obtained using Form 2 in Sec. 5.4.2.

Proof of Theorem 5.3: See Appendix 5.C. ■

The CRB in Theorem 5.3 serves as a good reference for determining how well CBS performs in practice. In Sec. 5.5, we will show that the MSE performance of CBS is close to the CRB in many cases. Besides, since element-space can be viewed as a special case of CBS, we can obtain the following corollary.

Corollary 5.1 *The CRB for the DOAs $\boldsymbol{\omega}$ for element-space is*

$$\mathrm{CRB}_{\mathrm{elm}}(\boldsymbol{\omega}) = \frac{1}{K}\left(\mathbf{G}^H\mathbf{\Pi}_{\mathbf{\Delta}}^{\perp}\mathbf{G}\right)^{-1}, \tag{5.56}$$

where

$$\mathbf{G} = \left(\mathbf{R}_{\mathbf{xx}}^T \otimes \mathbf{R}_{\mathbf{xx}}\right)^{-\frac{1}{2}} \left(\dot{\mathbf{A}}^* \odot (\mathbf{AP}) + (\mathbf{AP})^* \odot \dot{\mathbf{A}}\right), \tag{5.57}$$

$$\mathbf{\Delta} = \left(\mathbf{R}_{\mathbf{xx}}^T \otimes \mathbf{R}_{\mathbf{xx}}\right)^{-\frac{1}{2}} \left[\mathbf{A}^* \otimes \mathbf{A} \quad \mathrm{vec}(\mathbf{I})\right]. \tag{5.58}$$

Here

$$\dot{\mathbf{A}} = \left[\dot{\mathbf{a}}_N(\omega_1) \ \cdots \ \dot{\mathbf{a}}_N(\omega_D)\right] \tag{5.59}$$

with $\dot{\mathbf{a}}_N(\omega) = \frac{\mathrm{d}}{\mathrm{d}\omega}\mathbf{a}_N(\omega)$.

Remark: Similar to Remark 2 for Theorem 5.3, if the noise covariance is $p_{\mathrm{e}}\mathbf{R}_{\mathrm{e}}$ instead of $p_{\mathrm{e}}\mathbf{I}$ for any $\mathbf{R}_{\mathrm{e}}$ known a priori, then the CRB is as in (5.56)–(5.58) except that $\mathbf{I}$ is replaced by $\mathbf{R}_{\mathrm{e}}$ in (5.58). It is valid even when $\mathbf{R}_{\mathrm{e}}$ is singular.

Proof of Corollary 5.1: This corollary is obtained by setting the filter length $L = 1$ and CBS filter $H(z) = 1$ in Theorem 5.3. ■

Note that although Theorem 5.3 for CBS applies only to ULAs, Corollary 5.1 applies to any linear arrays. One can verify that our proof under the special case of element-space is valid for sparse arrays. It is valid even when the sources are correlated.

In [49, 50], different expressions of element-space CRBs are also derived, but uncorrelated sources are assumed therein. It is important to consider correlated sources because CBS is especially advantageous over element-space in this case. Yet, to compare with previous works, we also consider uncorrelated sources in the following. When the sources are known to be uncorrelated a priori, i.e., $\mathbf{P} = \mathrm{diag}(p_1, \ldots, p_D)$, the parameter vector becomes

$$\boldsymbol{\alpha} = \left[[\omega_i]_{i=1}^D \ \ [p_i]_{i=1}^D \ \ p_{\mathrm{e}}\right]^T \tag{5.60}$$

instead of (5.36), and the CRB for CBS can be derived as follows.

Theorem 5.4 *Suppose the sources are known to be uncorrelated a priori. Then if there are no stopband sources, the CRB for the DOAs* $\boldsymbol{\omega}$ *for CBS is*

$$\mathrm{CRB}_{\mathrm{unb}}^{\mathrm{unc}}(\boldsymbol{\omega}) = \frac{1}{K}\left(\mathbf{G}^H \mathbf{\Pi}_{\mathbf{\Delta}_{\mathrm{unc}}}^{\perp} \mathbf{G}\right)^{-1}, \tag{5.61}$$

where $\mathbf{G}$ *is as in (5.54) and*

$$\mathbf{\Delta}_{\mathrm{unc}} = \left(\mathbf{R}_{\mathbf{yy}}^T \otimes \mathbf{R}_{\mathbf{yy}}\right)^{-\frac{1}{2}} \left[(\mathbf{A}_L \mathbf{D}_H)^* \odot (\mathbf{A}_L \mathbf{D}_H) \quad \mathrm{vec}(\mathbf{H}\mathbf{H}^H)\right]. \tag{5.62}$$

The CRB (5.61) exists if and only if

$$\mathbf{M}_{\text{unc}} = \begin{bmatrix} \mathbf{M}_1 & (\mathbf{A}_L\mathbf{D}_H)^* \odot (\mathbf{A}_L\mathbf{D}_H) & \text{vec}(\mathbf{H}\mathbf{H}^H) \end{bmatrix} \tag{5.63}$$

has full column rank, where $\mathbf{M}_1$ *is as in (5.50). If stopband sources exist, (5.61) is an approximate bound.*

Proof: The proof is similar to that of Theorem 5.2 and Theorem 5.3. ■

Again, element-space can be viewed as a special case of CBS, so we obtain the following corollary.

Corollary 5.2 *When the sources are known to be uncorrelated a priori, the CRB for the DOAs* $\boldsymbol{\omega}$ *for element-space is*

$$\text{CRB}_{\text{elm}}^{\text{unc}}(\boldsymbol{\omega}) = \frac{1}{K}\left(\mathbf{G}^H\boldsymbol{\Pi}_{\boldsymbol{\Delta}_{\text{unc}}}^{\perp}\mathbf{G}\right)^{-1}, \tag{5.64}$$

where $\mathbf{G}$ *is as in (5.57) and*

$$\boldsymbol{\Delta}_{\text{unc}} = \left(\mathbf{R}_{\mathbf{xx}}^T \otimes \mathbf{R}_{\mathbf{xx}}\right)^{-\frac{1}{2}} \begin{bmatrix} \mathbf{A}^* \odot \mathbf{A} & \text{vec}(\mathbf{I}) \end{bmatrix}. \tag{5.65}$$

Proof: This corollary is obtained by setting the filter length $L = 1$ and CBS filter $H(z) = 1$ in Theorem 5.4. ■

One can check that (5.64) is equivalent to the CRB expressions in [49, 50]. Besides, comparing Theorem 5.4 to Theorem 5.3 and Corollary 5.2 to Corollary 5.1, we observe that the only difference in the CRB expressions is the substitution of a Khatri–Rao product for a Kronecker product in the expression for $\boldsymbol{\Delta}$. Therefore, $\boldsymbol{\Delta}_{\text{unc}}$ is a submatrix of $\boldsymbol{\Delta}$, obtained by selecting proper columns of $\boldsymbol{\Delta}$. This allows us to formally establish the following result. This essentially follows the intuition that additional prior knowledge can only decrease the CRB.

Fact 5.1 *Suppose the sources are uncorrelated. Then the CRB for the DOAs* $\boldsymbol{\omega}$ *for CBS when the information of the uncorrelatedness of the sources is unknown a priori is not smaller than that when this information is known a priori:*

$$\text{CRB}_{\text{unb}}(\boldsymbol{\omega})\big|_{\mathbf{P}=\text{diag}(p_1,\ldots,p_D)} \geq \text{CRB}_{\text{unb}}^{\text{unc}}(\boldsymbol{\omega}). \tag{5.66}$$

Similarly, it is true for element-space:

$$\text{CRB}_{\text{elm}}(\boldsymbol{\omega})\big|_{\mathbf{P}=\text{diag}(p_1,\ldots,p_D)} \geq \text{CRB}_{\text{elm}}^{\text{unc}}(\boldsymbol{\omega}). \tag{5.67}$$

Proof: Since $\mathbf{\Delta}_{\text{unc}}$ is a submatrix of $\mathbf{\Delta}$, obtained by selecting proper columns of $\mathbf{\Delta}$ (for both CBS and element-space), we have $\mathbf{\Pi}_{\mathbf{\Delta}}^{\perp} \preceq \mathbf{\Pi}_{\mathbf{\Delta}_{\text{unc}}}^{\perp}$. Hence, $(\mathbf{G}^H \mathbf{\Pi}_{\mathbf{\Delta}}^{\perp} \mathbf{G})^{-1} \succeq (\mathbf{G}^H \mathbf{\Pi}_{\mathbf{\Delta}_{\text{unc}}}^{\perp} \mathbf{G})^{-1}$, which completes the proof. ■

Fact 5.1 shows that the CRB cannot be larger if more prior information is given, which is not surprising. We will verify this theorem by simulations (Fig. 5.6).

5.4.2 Form 2

Now we get back to our correlated model with the parameter vector (5.36). In [44], an alternative form of element-space stochastic CRB different from Corollary 5.1 is derived for any linear array under two added assumptions. First, assume the manifold matrix $\mathbf{A}$ has full column rank. Second, assume the number of sources $D < N$, the number of sensors. In the case of a ULA, since $\mathbf{A}$ is Vandermonde, the second assumption implies the first one. Hence, for CBS CRB based on the filter output $\mathbf{y}$, if we assume $D < N - L + 1$, then we can derive a second form of CRB using results in [44]. Note that this same assumption is required for Form 1 to be valid. In the following, we assume this inequality is satisfied. Since the model in [44] only applies to white noise, a noise-whitening transformation is used to obtain the following theorem.

Theorem 5.5 *If there are no stopband sources, the CRB for the DOAs $\boldsymbol{\omega}$ for CBS is*

$$\text{CRB}_{\text{unb}}(\boldsymbol{\omega}) = \frac{p_{\text{e}}}{2K} \left[\text{Re}\{\mathbf{S}_1 \circ \mathbf{S}_2^*\} \right]^{-1}, \tag{5.68}$$

where

$$\mathbf{S}_1 = \mathbf{P}\mathbf{A}^H \mathbf{H}^H \mathbf{R}_{\mathbf{yy}}^{-1} \mathbf{H}\mathbf{A}\mathbf{P}, \tag{5.69}$$

$$\mathbf{S}_2 = \dot{\mathbf{A}}^H \mathbf{W}^H \mathbf{\Pi}_{\mathbf{WA}}^{\perp} \mathbf{W}\dot{\mathbf{A}} \tag{5.70}$$

with $\dot{\mathbf{A}}$ as defined in (5.59) and

$$\mathbf{W} = (\mathbf{H}\mathbf{H}^H)^{-1/2}\mathbf{H}. \tag{5.71}$$

Like in Theorem 5.3, (5.68) is precisely the CRB for CBS if there are no stopband sources. If stopband sources exist, it is an approximate bound. An exact bound can be computed as in Remark 1 of Theorem 5.3.

Remark: This noise-whitening method (5.72) requires a nonsingular noise covariance, i.e., $\mathbf{H}\mathbf{H}^H$ in the case of CBS. By definition we assume $h(0) \neq 0$ and

$h(L-1) \neq 0$ so that L is the filter length. This implies that the banded Toeplitz matrix $\mathbf{H}$ has full row rank, so $\mathbf{H}\mathbf{H}^H$ is positive definite, and $\mathbf{W}$ is well defined. This is also why we cannot obtain a Form-2 CRB if the noise covariance is $p_{\mathrm{e}}\mathbf{R}_{\mathrm{e}}$ instead of $p_{\mathrm{e}}\mathbf{I}$ for any singular $\mathbf{R}_{\mathrm{e}}$ known a priori (but we can obtain a Form-1 CRB as in Remark 2 for Theorem 5.3).

Proof of Theorem 5.5: Consider the noise-whitening transformation

$$\mathbf{z} = (\mathbf{H}\mathbf{H}^H)^{-1/2}\mathbf{y}. \tag{5.72}$$

Since this transformation is invertible, the CRB based on $\mathbf{z}$ is the same as the CRB based on $\mathbf{y}$. We can show that the covariance of $\mathbf{z}$ is given by

$$\mathbf{R}_{\mathbf{z}\mathbf{z}} = \mathbf{A}_{\mathbf{z}}\mathbf{P}\mathbf{A}_{\mathbf{z}}^H + p_{\mathrm{e}}\mathbf{I}, \tag{5.73}$$

where $\mathbf{A}_{\mathbf{z}} = \mathbf{W}\mathbf{A}$ is the equivalent manifold for $\mathbf{z}$, and $\mathbf{W}$ is defined in (5.71). Since (5.73) has the same form as the element-space model in [44], we can apply the CRB expression therein and obtain (5.68), with

$$\mathbf{S}_1 = \mathbf{P}\mathbf{A}_{\mathbf{z}}^H\mathbf{R}_{\mathbf{z}\mathbf{z}}^{-1}\mathbf{A}_{\mathbf{z}}\mathbf{P} \tag{5.74}$$

and $\mathbf{S}_2$ as in (5.70). Using (5.71) and (5.72), we can show that (5.74) is equivalent to (5.69). ■

Note that if $D < N - L + 1$, one can verify that the RHS of (5.53) indeed equals the RHS of (5.68). The detailed derivations are lengthy and omitted here, but the main idea is to start from computing the block matrix inversion $(\mathbf{\Delta}^H\mathbf{\Delta})^{-1}$ in (5.53), to express quantities in terms of signal and noise subspaces, and to simplify things using properties of the Kronecker product and Khatri–Rao product. Then, one can finally obtain (5.68). In this process, we need $D < N - L + 1$ to guarantee the noise subspace to have a nonzero dimension.

Theorem 5.5 offers a second form of CRB for CBS. Some additional insight can be obtained from Form 2. To this end, we first show an approximation as follows.

Fact 5.2 *Let* $\mathbf{\Pi}_{\mathbf{H}^H} = \mathbf{H}^H(\mathbf{H}\mathbf{H}^H)^{-1}\mathbf{H}$ *denote the orthogonal projection onto the column space of* $\mathbf{H}^H$*. Assume*

$$\left\| p_{\mathrm{e}}(\mathbf{A}^H\mathbf{\Pi}_{\mathbf{H}^H}\mathbf{A})^{-1/2}\mathbf{P}^{-1}(\mathbf{A}^H\mathbf{\Pi}_{\mathbf{H}^H}\mathbf{A})^{-1/2} \right\| \ll 1. \tag{5.75}$$

Then the CRB (5.68) can be approximated by

$$\mathrm{CRB}_{\mathrm{unb}}(\boldsymbol{\omega}) \approx \frac{p_{\mathrm{e}}}{2K}\left[\mathrm{Re}\{\mathbf{P} \circ \mathbf{S}_2^*\}\right]^{-1}, \tag{5.76}$$

where $\mathbf{S}_2$ *is as in (5.70).*

Remark: The insight to be gained from this fact will be clear from Corollaries 5.3 and 5.4 below.

Proof of Fact 5.2: To prove this fact, we only have to show that $\mathbf{S}_1 \approx \mathbf{P}$ under the assumption (5.75). Consider (5.73) and (5.74). Applying the matrix inversion lemma [112], we have

$$\begin{aligned}\mathbf{R}_{\mathbf{zz}}^{-1} &= p_{\mathrm{e}}^{-1}\mathbf{I} - p_{\mathrm{e}}^{-2}\mathbf{A}_{\mathbf{z}}(\mathbf{P}^{-1} + p_{\mathrm{e}}^{-1}\mathbf{A}_{\mathbf{z}}^H\mathbf{A}_{\mathbf{z}})^{-1}\mathbf{A}_{\mathbf{z}}^H \\ &= p_{\mathrm{e}}^{-1}\mathbf{I} - p_{\mathrm{e}}^{-1}\mathbf{A}_{\mathbf{z}}(\mathbf{A}_{\mathbf{z}}^H\mathbf{A}_{\mathbf{z}})^{-\frac{1}{2}}(\mathbf{S}+\mathbf{I})^{-1}(\mathbf{A}_{\mathbf{z}}^H\mathbf{A}_{\mathbf{z}})^{-\frac{1}{2}}\mathbf{A}_{\mathbf{z}}^H,\end{aligned}$$

where

$$\mathbf{S} = p_{\mathrm{e}}(\mathbf{A}_{\mathbf{z}}^H\mathbf{A}_{\mathbf{z}})^{-\frac{1}{2}}\mathbf{P}^{-1}(\mathbf{A}_{\mathbf{z}}^H\mathbf{A}_{\mathbf{z}})^{-\frac{1}{2}} \tag{5.77}$$

$$= p_{\mathrm{e}}(\mathbf{A}^H\mathbf{\Pi}_{\mathbf{H}^H}\mathbf{A})^{-1/2}\mathbf{P}^{-1}(\mathbf{A}^H\mathbf{\Pi}_{\mathbf{H}^H}\mathbf{A})^{-1/2}. \tag{5.78}$$

Now since $\|\mathbf{S}\| \ll 1$ due to (5.75), we have $(\mathbf{S}+\mathbf{I})^{-1} \approx \mathbf{I} - \mathbf{S}$. Thus,

$$\mathbf{S}_1 = \mathbf{P}\mathbf{A}_{\mathbf{z}}^H\mathbf{R}_{\mathbf{zz}}^{-1}\mathbf{A}_{\mathbf{z}}\mathbf{P} \tag{5.79}$$

$$\begin{aligned}&\approx p_{\mathrm{e}}^{-1}\mathbf{P}\mathbf{A}_{\mathbf{z}}^H\mathbf{A}_{\mathbf{z}}\mathbf{P} - p_{\mathrm{e}}^{-1}\mathbf{P}\mathbf{A}_{\mathbf{z}}^H\mathbf{A}_{\mathbf{z}}(\mathbf{A}_{\mathbf{z}}^H\mathbf{A}_{\mathbf{z}})^{-\frac{1}{2}}(\mathbf{A}_{\mathbf{z}}^H\mathbf{A}_{\mathbf{z}})^{-\frac{1}{2}}\mathbf{A}_{\mathbf{z}}^H\mathbf{A}_{\mathbf{z}}\mathbf{P} \\ &\quad + p_{\mathrm{e}}^{-1}\mathbf{P}\mathbf{A}_{\mathbf{z}}^H\mathbf{A}_{\mathbf{z}}(\mathbf{A}_{\mathbf{z}}^H\mathbf{A}_{\mathbf{z}})^{-\frac{1}{2}}\mathbf{S}(\mathbf{A}_{\mathbf{z}}^H\mathbf{A}_{\mathbf{z}})^{-\frac{1}{2}}\mathbf{A}_{\mathbf{z}}^H\mathbf{A}_{\mathbf{z}}\mathbf{P}\end{aligned} \tag{5.80}$$

$$= p_{\mathrm{e}}^{-1}\mathbf{P}\mathbf{A}_{\mathbf{z}}^H\mathbf{A}_{\mathbf{z}}\mathbf{P} - p_{\mathrm{e}}^{-1}\mathbf{P}\mathbf{A}_{\mathbf{z}}^H\mathbf{A}_{\mathbf{z}}\mathbf{P} + \mathbf{P} \tag{5.81}$$

$$= \mathbf{P}, \tag{5.82}$$

where we have used (5.77) to obtain (5.81). This completes the proof. ■

The assumption (5.75) is satisfied in practical CBS and element-space systems with large arrays. To understand why it is the case, note that the LHS of (5.75) is less than or equal to $\|p_{\mathrm{e}}\mathbf{P}^{-1}\| \cdot \|(\mathbf{A}^H\mathbf{\Pi}_{\mathbf{H}^H}\mathbf{A})^{-1}\|$. If the number of sensors N is large and if the DOAs are not very close to one another, then typically $\mathbf{A}^H\mathbf{A} \approx N\mathbf{I}$. That is, the columns of $\mathbf{A}$ are approximately orthogonal. Meanwhile, according to our numerical experiments, $\mathbf{\Pi}_{\mathbf{H}^H}\mathbf{A} \approx \mathbf{A}$ for properly designed CBS filter. Together, we have $\|(\mathbf{A}^H\mathbf{\Pi}_{\mathbf{H}^H}\mathbf{A})^{-1}\| \approx N^{-1}$. Moreover, we assume that the SNR is not very small and that the correlation coefficients between the sources are not close to 1, so that $\|p_{\mathrm{e}}\mathbf{P}^{-1}\| \ll N$. Hence, we finally obtain (5.75). Numerical values for the LHS of (5.75) will be given in the example of Fig. 5.3.

The fact that $\mathbf{\Pi}_{\mathbf{H}^H}\mathbf{A} \approx \mathbf{A}$ for properly designed CBS filter is expectable to some extent. In [78], it is shown that the CRB based on $\mathbf{y} = \mathbf{T}\mathbf{x}$ for any beamspace matrix $\mathbf{T}$ is larger than or equal to the element-space CRB, and that $\mathbf{\Pi}_{\mathbf{T}^H}\mathbf{A} = \mathbf{A}$

is a necessary condition for the beamspace CRB to equal the element-space CRB. Thus, it is not surprising that the well-performing method CBS leads to $\mathbf{\Pi}_{\mathbf{H}^H}\mathbf{A} \approx \mathbf{A}$. This means that all steering vectors roughly lie in the column space of $\mathbf{H}^H$, so no significant signal information is lost.

Having proved Fact 5.2 and discussed the practical side of the assumption (5.75), we now derive the following corollary relating CRB to source powers.

Corollary 5.3 *Suppose the assumption in Fact 5.2 holds so that we have (5.76). Also assume the correlation coefficient of each pair of sources is fixed. Then the CRB for a DOA is approximately inversely proportional to its own source power and approximately independent of the source powers of the other DOAs. This is true for both CBS and element-space.*

Proof: We can express the source covariance $\mathbf{P}$ as

$$\mathbf{P} = \mathbf{D}_p \mathbf{R}_\rho \mathbf{D}_p, \tag{5.83}$$

where $\mathbf{D}_p = \mathrm{diag}(\sqrt{p_1}, \ldots, \sqrt{p_D})$, and the correlation coefficient matrix $\mathbf{R}_\rho$ has entries $[\mathbf{R}_\rho]_{i,k} = \mathrm{E}[c_i c_k^*]/\sqrt{p_i p_k}$. Hence, (5.76) implies

$$\mathrm{CRB}_{\mathrm{unb}}(\boldsymbol{\omega}) \approx \frac{p_\mathrm{e}}{2K}\left[\mathrm{Re}\{\mathbf{D}_p \mathbf{R}_\rho \mathbf{D}_p \circ \mathbf{S}_2^*\}\right]^{-1}, \tag{5.84}$$

$$= \frac{p_\mathrm{e}}{2K}\mathbf{D}_p^{-1}\left[\mathrm{Re}\{\mathbf{R}_\rho \circ \mathbf{S}_2^*\}\right]^{-1}\mathbf{D}_p^{-1}. \tag{5.85}$$

Thus, the CRB for the ith DOA is

$$\mathrm{CRB}_{\mathrm{unb}}(\omega_i) \approx \frac{p_\mathrm{e}}{2K p_i}\left[\left[\mathrm{Re}\{\mathbf{R}_\rho \circ \mathbf{S}_2^*\}\right]^{-1}\right]_{i,i}, \tag{5.86}$$

which is inversely proportional to p_i and independent of p_k for all $k \neq i$. This is also true for element-space since CBS reduces to element-space if we set the filter length $L = 1$ and CBS filter $H(z) = 1$. ■

Corollary 5.3 shows that the CRB for a DOA almost does not depend on the power of another DOA. In particular, the CBS CRB for in-band DOAs is almost independent of out-of-band source powers. That is, a more powerful out-of-band jammer does not impose a larger lower bound on MSE of in-band DOA estimates. As we shall see in simulations (Fig. 5.3), CBS can yield an in-band MSE almost independent of the out-of-band source powers as long as the filter stopband attenuation is large enough to sufficiently attenuate the stopband sources.

Another corollary relating CRB to noise power can be obtained from Fact 5.2 as stated below.

Corollary 5.4 *Suppose the assumption in Fact 5.2 holds so that we have (5.76). Then the CRB for a DOA is approximately proportional to the noise power p_e. This is true for both CBS and element-space.*

Proof: It is immediately proved by (5.76). ■

Corollary 5.3 and Corollary 5.4 together imply that the CRB of a DOA ω_i is approximately inversely proportional to its own SNR p_i/p_e. This will be verified by simulations (Fig. 5.5). Note that the two corollaries may not be easily proved by directly using the CRB expression in Theorem 5.3 or Theorem 5.5. Even Theorem 5.5 does not immediately lead to Corollary 5.4 because $\mathbf{S}_1$ in (5.68) depends on p_e. In this sense, Fact 5.2 is an important result which gives much more insight.

5.5 Simulations

In the following numerical examples, we assume the number of DOAs is known. To compare with CBS using a filter $H(z)$, for element-space, we just consider DOA estimates in the passband of $H(z)$ and ignore those in the stopband. Whenever we mention mean square errors (MSEs) or root mean square errors (RMSEs) in detected in-band source angles, we refer to averaging square errors measured in ω over all in-band DOAs. Similarly, since the stochastic CRBs [44] differ for different DOAs, we average the variance bounds over all in-band DOAs. The theoretical MSEs of CBS and element-space are computed from (5.32) and (5.34), respectively. Unless otherwise stated, the CBS and element-space CRBs are computed from (5.53) and (5.56), respectively. Note that (5.53) is precisely the CRB for CBS only if there are no stopband sources so that the in-band DOA estimates are unbiased. If stopband sources exist, (5.53) is approximately a lower bound on the variance of DOA estimates of CBS due to (5.43). The exact lower bound (5.42) for the biased CBS estimator will be compared to the approximate bound in Fig. 5.4. The CBS filter $H(z)$ is designed to be a spectral factor of a lowpass Nyquist-equiripple filter [71], with passband edge $\pi/2M$ and stopband edge $3\pi/2M$, where M is the decimation ratio.

In Fig. 5.1, we compare RMSEs of DOA estimates and CRBs for CBS with various filter length L, and for element-space. The RMSEs based on Monte Carlo simulation and based on our theoretical analysis in Sec. 5.3 are both shown for comparison. We consider a ULA with $N = 99$ sensors receiving 6 sources at angles $\theta = -5°, 0°, 5°, 40°, 60°$, and $80°$. All sources have power 1. Sources n and $n + 3$ have a correlation coefficient $\rho = 0.9$ for $n = 1, 2, 3$. For CBS, the decimation

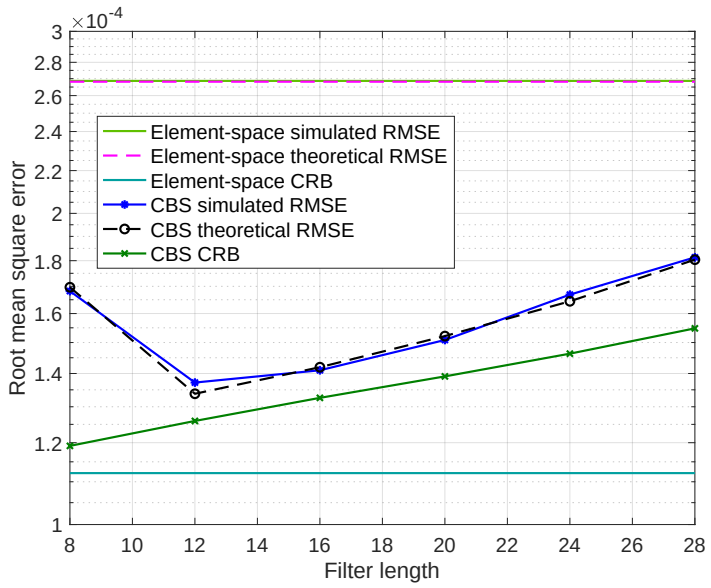


Figure 5.1: Simulated RMSE, theoretical RMSE, and CRB for CBS with various filter length L, and for element-space.

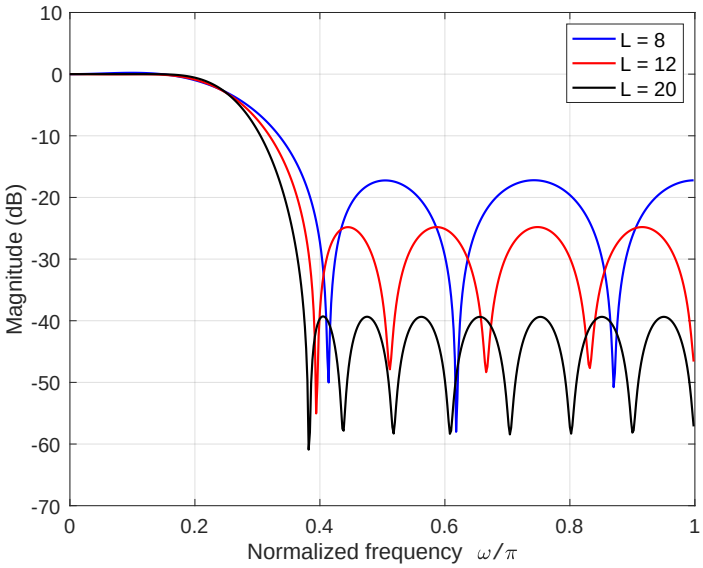


Figure 5.2: Magnitude responses of the Nyquist-equiripple filters used for CBS with several typical values of filter length L.

ratio is $M = 4$. Hence, the three sources at $-5°, 0°$ and $5°$ are in the passband, and the others are in the stopband. Noise variance is $p_e = 1$. Root-MUSIC is used to estimate DOAs. Covariance estimates are obtained by using 500 snapshots, and 500 Monte Carlo runs are used. Several observations can be made from Fig. 5.1.

First, the theoretical RMSE curves almost coincide with simulated RMSE curves for both CBS and element-space. This verifies our MSE analysis in Sec. 5.3. Second, the CBS CRB is always larger than the element-space CRB. This is consistent with the known fact that beamspace CRB cannot be smaller than element-space CRB [78]. However, the RMSE of CBS is uniformly smaller than that of element-space in this example. That is, CBS offers a practical algorithm with RMSE approaching the CRB when there are correlated sources, while there is a large gap between RMSE and CRB for element-space. Finally, there is an optimal filter length such that the RMSE of CBS is minimized. If the filter length is too small, the stopband attenuation is not good enough (see Fig. 5.2), and the filtered stopband sources contribute to a large bias B_i as defined in (5.28). If the filter length is too large, we need to discard many transient samples in the filter output (since we retain only steady-state samples). This leads to a larger RMSE. By plotting a RMSE curve as a function of the filter length using our analysis in Sec. 5.3, we can determine the optimal filter length. However, we also note that the dependence of CBS RMSE on the filter length L is not very significant. For a wide range of values of L, CBS is better than element-space, so it is not a difficult task to choose L in practice. Magnitude responses of the Nyquist-equiripple filters used for CBS with several typical values of filter length L are shown in Fig. 5.2. These filters indeed have equiripple stopband attenuation. Besides, we obtain better stopband attenuation as the filter length increases.

In Fig. 5.3, we compare RMSEs of DOA estimates and CRBs for CBS and element-space as we vary the out-of-band source power. The filter length is now fixed at $L = 16$, and all the other simulation parameters are the same as in the example of Fig. 5.1. We do not use the optimal value $L = 12$ as in Fig. 5.1 because it was obtained when the out-of-band source power is 0 dB. We need a larger filter length to get better stopband attenuation to deal with larger out-of-band source power. According to Fig. 5.3, both the CBS CRB and element-space CRB almost do not depend on the out-of-band source power, as implied by Corollary 5.3. To obtain this corollary, we need the assumption (5.75) in Fact 5.2 to be valid. This is indeed the case since the LHS of (5.75) gradually decreases from 0.112 to 0.0627 as the out-of-band source power increases from 0 dB to 30 dB. Besides, just like in Fig. 5.1, the CBS CRB is larger than the element-space CRB. However, the RMSE of CBS is smaller than that of element-space if the out-of-band source power is not too large. To better understand how the MSE and CRB of CBS change with the out-of-band source power, in Fig. 5.4, we show the simulated MSE, theoretical MSE $B_i^2 + V_i$

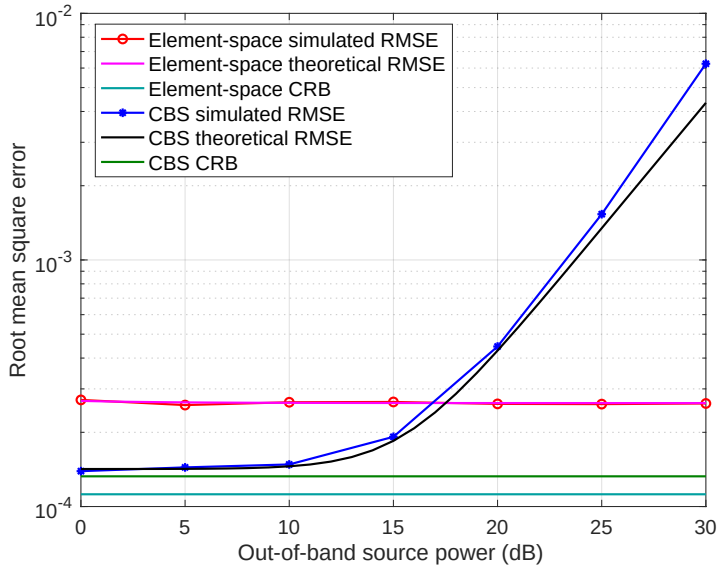


Figure 5.3: Simulated RMSE, theoretical RMSE, and CRB for element-space and CBS as the out-of-band source power varies.

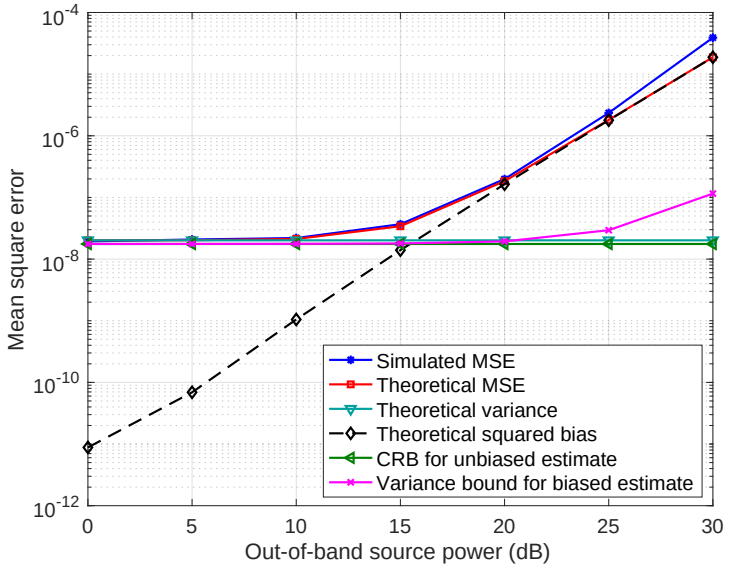
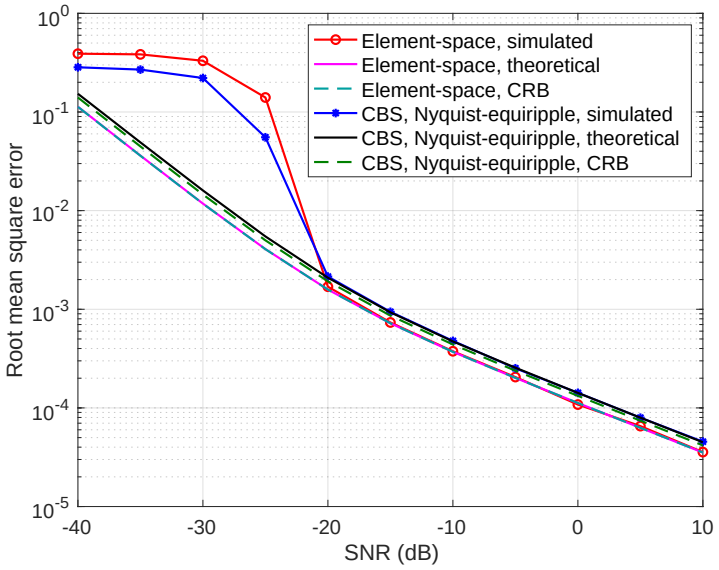


Figure 5.4: Simulated MSE, theoretical MSE, theoretical variance, theoretical squared bias, CRB for unbiased estimate (5.43), and the exact variance bound (5.42) for biased estimate for CBS as the out-of-band source power varies.

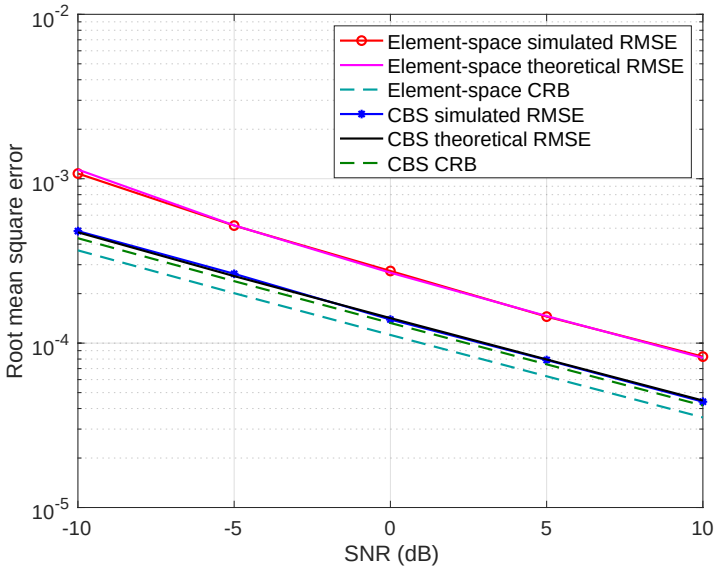
(as defined in (5.32)), theoretical variance V_i, theoretical squared bias B_i^2, CRB for unbiased estimate (5.43), and the exact variance bound (5.42) for biased estimate for CBS. As long as the out-of-band source power is not extremely large, say, not

greater than 20 dB in this example, our theoretical expression gives a good estimate of the true MSE. If the out-of-band source power is extremely large, our assumption that the covariance perturbation $\delta\mathbf{R}$ in (5.15) due to the filtered stopband sources is small will be invalid. Thus, there is a larger difference between theoretical MSE and simulated MSE. Another observation is that the theoretical squared bias is much smaller than the theoretical variance for out-of-band source power not greater than 10 dB. Hence, we may ignore the bias term in the MSE analysis if the passband and stopband sources have similar powers. Finally, as mentioned in the beginning of this section, the CBS CRB in each previous plot is approximately a lower bound on the variance of DOA estimate due to (5.43). Here we compare the exact lower bound (5.42) for the biased CBS estimator with the approximate bound. The exact CRB (5.42) is computed from (5.40), (5.37), and (5.46) with $\mathbf{G}$ as in (5.54) and $\mathbf{\Delta}_1$ as in (5.141), and the derivatives in the Jacobian matrix are computed numerically. We see that the approximate bound is similar to the exact bound if the out-of-band source power is not greater than 20 dB. Hence, (5.43) gives a good lower bound as long as the out-of-band source is not too strong compared to the stopband attenuation of the CBS filter.

In Fig. 5.5, we compare RMSEs of DOA estimates and CRBs for CBS and element-space as we vary the SNR. We consider the same three passband sources and three stopband sources as in the example of Fig. 5.1. All sources have power 1. The SNR is thus $1/p_e$, where the noise power p_e is varied. Sources n and $n+3$ have a correlation coefficient ρ for $n = 1, 2, 3$. All the other simulation parameters are the same as before. In Fig. 5.5(a), we consider $\rho = 0$, i.e., the uncorrelated case. When the SNR is greater than -20 dB, both CBS and element-space RMSEs approach the CRBs, and the two systems have similar performance. For the extremely low SNR regime, the simulated RMSEs deviate from the theoretical RMSEs. This can be expected because our analysis is asymptotic. It is not a serious issue since the analysis already works well for an SNR as low as -20 dB. In Fig. 5.5(b), we consider $\rho = 0.9$. In this correlated case, CBS RMSE still approaches its CRB, but element-space RMSE does not. CBS RMSE is significantly smaller than element-space RMSE. In these two subfigures, the theoretical RMSE curves again almost coincide with simulated RMSE curves for both CBS and element-space. Moreover, both CBS and element-space CRBs look linear in the log-log plots, and the CRBs decrease by a factor of 10 as SNR increases by 20 dB. That is, the CRBs for MSE is inversely proportional to the SNR. This verifies the result of Corollary 5.4. We note that in the example of Fig. 5.5(b), the in-band sources are correlated only with



(a)



(b)

Figure 5.5: Simulated RMSE, theoretical RMSE, and CRB for element-space and CBS as the SNR varies. Sources n and $n+3$ have a correlation coefficient ρ for $n = 1, 2, 3$. (a) $\rho = 0$. (b) $\rho = 0.9$.

out-of-band sources. CBS is especially advantageous over element-space in this setting. If in-band sources are correlated with one another, then element-space can be better than CBS, as shown in Fig. 2.6.

As mentioned in the beginning of this section, in all the previous examples, the CBS

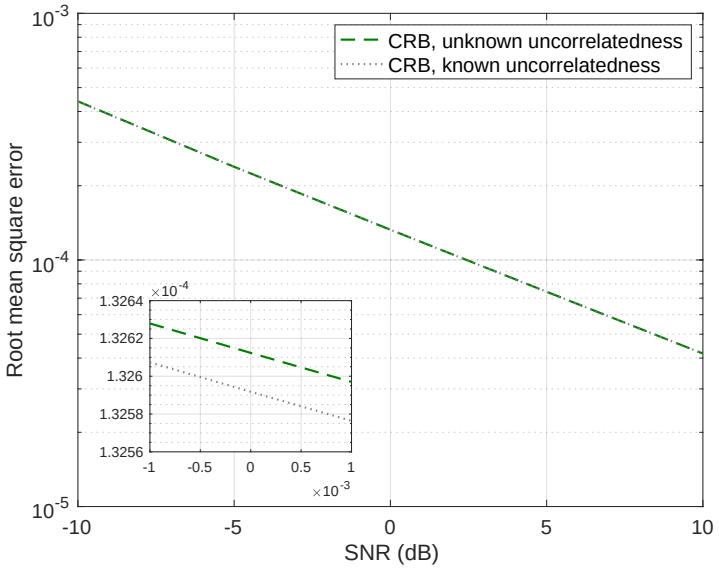


Figure 5.6: CRB for CBS as the SNR varies. The uncorrelatedness of the sources are assumed either unknown or known a priori.

and element-space CRBs are computed from (5.53) and (5.56), respectively. That is, the correlations P_{ik} between sources are assumed unknown as in (5.36). When all the sources are uncorrelated, we can also assume that the uncorrelatedness is known a priori. Under this assumption, we have obtained the CBS and element-space CRBs in (5.61) and (5.64), respectively. To compare the two cases, we consider again the example of Fig. 5.5(a) and compute the CBS CRB assuming either unknown or known uncorrelatedness. The results are shown in Fig. 5.6. The CRB assuming known uncorrelatedness is smaller than the CRB assuming unknown uncorrelatedness, which is consistent with Fact 5.1. Interestingly, the difference between them is very small, so this prior information is not so influential.

5.6 Concluding Remarks

The MSE performance and Cramér–Rao bound (CRB) for convolutional beamspace (CBS) are analyzed in this chapter. Theoretical expressions of MSE are derived assuming that MUSIC or root-MUSIC is used to estimate DOAs. (The performance is the same for both.) The bias of the CBS estimator, though negligible in some cases, is given via a first-order perturbation analysis. To obtain the variance of the CBS estimator, we develop an approach to derive the asymptotic probability distribution of the eigenvectors of the average finite-snapshot covariance matrices of dependent random vectors. This approach can be useful to other applications because previous

results are only for independent random vectors. As for CRB, we offer two forms of expressions. Form 1 is derived directly from the Fisher information matrix. Form 2 is derived via a noise-whitening approach and offers further insight. In particular, the CRB for a DOA is approximately inversely proportional to its own source power and approximately independent of the source powers of the other DOAs. Also, the CRB for a DOA is approximately proportional to the noise power. These results are also true for element-space and to the best of our knowledge, have not been theoretically established in previous works. Extensive numerical examples are also given, which verify the theoretical results.

Appendices

5.A Proof of Lemma 5.1

Consider the K-snapshot estimate $\widehat{\mathbf{R}}_{\text{ave},0} = \widehat{\mathbf{R}}_{\text{ave}}\big|_{\delta\mathbf{R}=\mathbf{O}}$ with stopband sources nulled, where $\widehat{\mathbf{R}}_{\text{ave}}$ is defined in (5.18). We have

$$\mathrm{E}[\widehat{\mathbf{R}}_{\text{ave},0}] = \mathbf{R}_{\text{ave},0} = \mathbf{A}_{\text{dec},0}\breve{\mathbf{R}}_{d,0}\mathbf{A}_{\text{dec},0}^{H} + p_{\text{e}}\mathbf{G}_{\text{dec}} \tag{5.87}$$

and

$$\begin{aligned}
&K^2M^2\,\mathrm{E}\left[[\widehat{\mathbf{R}}_{\text{ave},0}]_{i,p}[\widehat{\mathbf{R}}_{\text{ave},0}]_{g,h}^{*}\right] \\
&\quad= \mathrm{E}\left[\sum_{k=1}^{K}\sum_{l=0}^{M-1}[\mathbf{v}_l[k]]_i[\mathbf{v}_l[k]]_p^{*}\sum_{n=1}^{K}\sum_{m=0}^{M-1}[\mathbf{v}_m[n]]_g^{*}[\mathbf{v}_m[n]]_h\right] \\
&\quad= \mathrm{E}\left[\sum_{k=1}^{K}\sum_{l=0}^{M-1}\sum_{m=0}^{M-1}[\mathbf{v}_l[k]]_i[\mathbf{v}_l[k]]_p^{*}[\mathbf{v}_m[k]]_g^{*}[\mathbf{v}_m[k]]_h\right. \\
&\qquad\left.+\sum_{k=1}^{K}\sum_{n\neq k}\sum_{l=0}^{M-1}\sum_{m=0}^{M-1}[\mathbf{v}_l[k]]_i[\mathbf{v}_l[k]]_p^{*}[\mathbf{v}_m[n]]_g^{*}[\mathbf{v}_m[n]]_h\right] \\
&\quad= \mathrm{E}\left[K\sum_{l=0}^{M-1}\sum_{m=0}^{M-1}[\mathbf{v}_l]_i[\mathbf{v}_l]_p^{*}[\mathbf{v}_m]_g^{*}[\mathbf{v}_m]_h\right. \\
&\qquad\left.+K(K-1)M^2[\mathbf{R}_{\text{ave},0}]_{i,p}[\mathbf{R}_{\text{ave},0}]_{g,h}^{*}\right]
\end{aligned} \tag{5.88}$$

for $0 \le i, p, g, h \le J-1$. Since the entries of $\mathbf{v}_l$ for all l are circularly-symmetric complex Gaussian, using Wick's theorem (or Isserlis' theorem) [113, 114], we have

$$\begin{aligned}
\mathrm{E}\left[[\mathbf{v}_l]_i[\mathbf{v}_l]_p^{*}[\mathbf{v}_m]_g^{*}[\mathbf{v}_m]_h\right] &= \mathrm{E}\left[[\mathbf{v}_l]_i[\mathbf{v}_l]_p^{*}\right]\mathrm{E}\left[[\mathbf{v}_m]_g^{*}[\mathbf{v}_m]_h\right] \\
&\quad+\mathrm{E}\left[[\mathbf{v}_l]_i[\mathbf{v}_m]_g^{*}\right]\mathrm{E}\left[[\mathbf{v}_l]_p^{*}[\mathbf{v}_m]_h\right].
\end{aligned}$$

Hence, we can obtain the covariance

$$\begin{aligned}
&\mathrm{E}\left[([\widehat{\mathbf{R}}_{\mathrm{ave},0}]_{i,p} - [\mathbf{R}_{\mathrm{ave},0}]_{i,p})([\widehat{\mathbf{R}}_{\mathrm{ave},0}]_{g,h} - [\mathbf{R}_{\mathrm{ave},0}]_{g,h})^*\right] \\
&\qquad = \frac{1}{KM^2}\sum_{l=0}^{M-1}\sum_{m=0}^{M-1}\mathrm{E}\left[[\mathbf{v}_l]_i[\mathbf{v}_m]_g^*\right]\mathrm{E}\left[[\mathbf{v}_l]_p^*[\mathbf{v}_m]_h\right] \qquad (5.89)\\
&\qquad = \frac{1}{KM^2}\mathrm{tr}\left(\mathbf{R}_{\mathbf{yy}}^{(i,g)}(\mathbf{R}_{\mathbf{yy}}^{(p,h)})^H\right), \qquad (5.90)
\end{aligned}$$

where $\mathbf{R}_{\mathbf{yy}}^{(i,g)} \in \mathbb{C}^{M\times M}$ are submatrices of

$$\mathbf{R}_{\mathbf{yy},0} = \mathbf{R}_{\mathbf{yy}}\big|_{\delta\mathbf{R}=\mathbf{O}} \triangleq \begin{bmatrix} \mathbf{R}_{\mathbf{yy}}^{(1,1)} & \cdots & \mathbf{R}_{\mathbf{yy}}^{(1,J)} \\ \vdots & \ddots & \vdots \\ \mathbf{R}_{\mathbf{yy}}^{(J,1)} & \cdots & \mathbf{R}_{\mathbf{yy}}^{(J,J)} \end{bmatrix}. \qquad (5.91)$$

Throughout the above derivations, we implicitly assume the stopband sources have been nulled.

Then consider (5.19) and let $\mathbf{E}_0 = [\mathbf{E}_{\mathrm{s},0}\ \mathbf{E}_{\mathrm{n},0}]$ and $\mathbf{\Lambda}_0 = \mathrm{diag}(\mathbf{\Lambda}_{\mathrm{s},0}, \mathbf{\Lambda}_{\mathrm{n},0})$. Define

$$\mathbf{U} = K^{1/2}(\mathbf{E}_0^H\widehat{\mathbf{R}}_{\mathrm{ave},0}\mathbf{E}_0 - \mathbf{\Lambda}_0). \qquad (5.92)$$

Based on the multivariate central limit theorem and (5.90), we can derive that $\mathbf{U}$ is asymptotically (for large K) complex Gaussian with mean zero and covariance

$$\mathrm{E}\left[[\mathbf{U}]_{i,p}[\mathbf{U}]_{g,h}^*\right] = \frac{1}{M^2}\mathrm{tr}\left(\widetilde{\mathbf{R}}_{\mathbf{yy}}^{(i,g)}(\widetilde{\mathbf{R}}_{\mathbf{yy}}^{(p,h)})^H\right), \qquad (5.93)$$

where $\widetilde{\mathbf{R}}_{\mathbf{yy}}^{(i,g)}$ are as defined in (5.24). Let

$$\mathbf{V} = K^{-1/2}\mathbf{U} + \mathbf{\Lambda}_0 = \mathbf{E}_0^H\widehat{\mathbf{R}}_{\mathrm{ave},0}\mathbf{E}_0 \qquad (5.94)$$

and suppose that

$$\mathbf{V} = \mathbf{Q}\mathbf{D}_{\mathbf{V}}\mathbf{Q}^H \qquad (5.95)$$

is the eigenvalue decomposition of $\mathbf{V}$ with eigenvalues in descending order. We

partition $\mathbf{U}$ and $\mathbf{Q}$ into

$$\mathbf{U} = \begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1D_0} & \mathbf{u}_{1\mathrm{e}}^T \\ u_{21} & u_{22} & \cdots & u_{2D_0} & \mathbf{u}_{2\mathrm{e}}^T \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ u_{D_0 1} & u_{D_0 2} & \cdots & u_{D_0 D_0} & \mathbf{u}_{D_0\mathrm{e}}^T \\ \mathbf{u}_{\mathrm{e}1} & \mathbf{u}_{\mathrm{e}2} & \cdots & \mathbf{u}_{\mathrm{e}D_0} & \mathbf{U}_{\mathrm{ee}} \end{bmatrix}, \tag{5.96}$$

$$\mathbf{Q} = \begin{bmatrix} q_{11} & q_{12} & \cdots & q_{1D_0} & \mathbf{q}_{1\mathrm{e}}^T \\ q_{21} & q_{22} & \cdots & q_{2D_0} & \mathbf{q}_{2\mathrm{e}}^T \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ q_{D_0 1} & q_{D_0 2} & \cdots & q_{D_0 D_0} & \mathbf{q}_{D_0\mathrm{e}}^T \\ \mathbf{q}_{\mathrm{e}1} & \mathbf{q}_{\mathrm{e}2} & \cdots & \mathbf{q}_{\mathrm{e}D_0} & \mathbf{Q}_{\mathrm{ee}} \end{bmatrix}, \tag{5.97}$$

where $\mathbf{U}_{\mathrm{ee}}, \mathbf{Q}_{\mathrm{ee}} \in \mathbb{C}^{(J-D_0)\times(J-D_0)}$ and $\mathbf{u}_{i\mathrm{e}}, \mathbf{u}_{\mathrm{e}i}, \mathbf{q}_{i\mathrm{e}}, \mathbf{q}_{\mathrm{e}i} \in \mathbb{C}^{J-D_0}$, $i = 1, \ldots, D_0$. Following the derivations in [107], we can show that asymptotically (for large K), $K^{1/2}(q_{ll} - 1)$ converges stochastically to 0, and the limiting distributions of $K^{1/2}q_{il}$ and $K^{1/2}\mathbf{q}_{\mathrm{e}l}$ are the same as the limiting distributions of $u_{li}^*/(\lambda_{l,0} - \lambda_{i,0})$ and $\mathbf{u}_{l\mathrm{e}}^*/(\lambda_{l,0} - p_{\mathrm{e}})$, $1 \le i, l \le D_0$, $i \ne l$. Using (5.94) and (5.95), we obtain

$$\widehat{\mathbf{R}}_{\mathrm{ave},0} = (\mathbf{E}_0\mathbf{Q})\mathbf{D}_{\mathbf{V}}(\mathbf{E}_0\mathbf{Q})^H, \tag{5.98}$$

which is the eigendecomposition of $\widehat{\mathbf{R}}_{\mathrm{ave},0}$. Hence, asymptotically, the signal eigenvector estimates are

$$\hat{\mathbf{e}}_{l,0} = \mathbf{E}_0[\mathbf{Q}]_{:,l} \tag{5.99}$$

$$= \mathbf{e}_{l,0} + K^{-1/2}\left[\sum_{\substack{i=1 \\ i\ne l}}^{D_0} \frac{[\mathbf{U}]_{l,i}^*}{\lambda_{l,0} - \lambda_{i,0}}\mathbf{e}_{i,0} + \frac{\mathbf{E}_{\mathrm{n},0}\mathbf{u}_{l\mathrm{e}}^*}{\lambda_{l,0} - p_{\mathrm{e}}}\right] \tag{5.100}$$

$$= \mathbf{e}_{l,0} + K^{-1/2}\sum_{\substack{i=1 \\ i\ne l}}^{J} \frac{[\mathbf{U}]_{l,i}^*}{\lambda_{l,0} - \lambda_{i,0}}\mathbf{e}_{i,0}, \tag{5.101}$$

$l = 1, \ldots, D_0$. Using (5.93) and (5.101), we obtain the statement of the lemma. To obtain the relation matrices, we have also used the fact that $\mathbf{U}$ is Hermitian symmetric.

5.B Proof of Theorem 5.1

Following the idea (Eq. (B.2a)) of [56], asymptotically (for large K), the DOA estimates $\hat{\omega}_i$ of MUSIC based on (5.18) satisfy

$$\hat{\omega}_i - \omega_i \approx \frac{-\mathrm{Re}\{\mathbf{a}_J^H(\tilde{\omega}_i)\widehat{\mathbf{E}}_{\mathrm{n}}\widehat{\mathbf{E}}_{\mathrm{n}}^H\dot{\mathbf{a}}_J(\tilde{\omega}_i)\}}{M\dot{\mathbf{a}}_J^H(\tilde{\omega}_i)\mathbf{E}_{\mathrm{n}}\mathbf{E}_{\mathrm{n}}^H\dot{\mathbf{a}}_J(\tilde{\omega}_i)}, \tag{5.102}$$

where $\tilde{\omega}_i = M\omega_i$, $\dot{\mathbf{a}}_J(\omega) = \frac{\mathrm{d}}{\mathrm{d}\omega}\mathbf{a}_J(\omega)$, and the terms neglected in the approximation are $O(1/K)$. Besides, according to [26], root-MUSIC has the same asymptotic error expression (5.102). Then in view of (5.20), for small $\|\delta\mathbf{R}\|$,

$$\dot{\mathbf{a}}_J^H(\tilde{\omega}_i)\mathbf{E}_{\mathrm{n}}\mathbf{E}_{\mathrm{n}}^H\dot{\mathbf{a}}_J(\tilde{\omega}_i) = \dot{\mathbf{a}}_J^H(\tilde{\omega}_i)(\mathbf{I} - \mathbf{E}_{\mathrm{s}}\mathbf{E}_{\mathrm{s}}^H)\dot{\mathbf{a}}_J(\tilde{\omega}_i) \tag{5.103}$$

$$\approx \dot{\mathbf{a}}_J^H(\tilde{\omega}_i)(\mathbf{I} - \mathbf{E}_{\mathrm{s},0}\mathbf{E}_{\mathrm{s},0}^H)\dot{\mathbf{a}}_J(\tilde{\omega}_i) \tag{5.104}$$

$$= \dot{\mathbf{a}}_J^H(\tilde{\omega}_i)\mathbf{E}_{\mathrm{n},0}\mathbf{E}_{\mathrm{n},0}^H\dot{\mathbf{a}}_J(\tilde{\omega}_i), \tag{5.105}$$

where the terms neglected in the approximation are $O(\|\delta\mathbf{R}\|)$. Thus,

$$\hat{\omega}_i - \omega_i \approx -\mathrm{Re}\{\mathbf{a}_J^H(\tilde{\omega}_i)\widehat{\mathbf{E}}_{\mathrm{n}}\widehat{\mathbf{E}}_{\mathrm{n}}^H\dot{\mathbf{a}}_J(\tilde{\omega}_i)\}/(Mg(\tilde{\omega}_i)), \tag{5.106}$$

where $g(\omega) = \dot{\mathbf{a}}_J^H(\omega)\mathbf{E}_{\mathrm{n},0}\mathbf{E}_{\mathrm{n},0}^H\dot{\mathbf{a}}_J(\omega)$. Besides,

$$\widehat{\mathbf{E}}_{\mathrm{n}}\widehat{\mathbf{E}}_{\mathrm{n}}^H = \mathbf{I} - \widehat{\mathbf{E}}_{\mathrm{s}}\widehat{\mathbf{E}}_{\mathrm{s}}^H \tag{5.107}$$

$$\approx \mathbf{I} - \widehat{\mathbf{E}}_{\mathrm{s},0}\widehat{\mathbf{E}}_{\mathrm{s},0}^H - \sum_{l=1}^{D_0}\left(\mathbf{e}_{l,0}\delta\mathbf{e}_l^H + \delta\mathbf{e}_l\mathbf{e}_{l,0}^H\right) \tag{5.108}$$

$$= \widehat{\mathbf{E}}_{\mathrm{n},0}\widehat{\mathbf{E}}_{\mathrm{n},0}^H - \sum_{l=1}^{D_0}\left(\mathbf{e}_{l,0}\delta\mathbf{e}_l^H + \delta\mathbf{e}_l\mathbf{e}_{l,0}^H\right), \tag{5.109}$$

where the terms neglected in the approximation are $O(\|\delta\mathbf{R}\|^2)$, and $\delta\mathbf{e}_l$ is defined in (5.29). Hence, for large K and small $\|\delta\mathbf{R}\|$,

$$\hat{\omega}_i - \omega_i = B_i - \frac{\mathrm{Re}\{\mathbf{a}_J^H(\tilde{\omega}_i)\widehat{\mathbf{E}}_{\mathrm{n},0}\widehat{\mathbf{E}}_{\mathrm{n},0}^H\dot{\mathbf{a}}_J(\tilde{\omega}_i)\}}{Mg(\tilde{\omega}_i)}, \tag{5.110}$$

where B_i is defined in (5.28). Next, in view of (5.101), we can write $\widehat{\mathbf{E}}_{\mathrm{s},0} = \mathbf{E}_{\mathrm{s},0} + \widehat{\boldsymbol{\Xi}}$, where $\|\widehat{\boldsymbol{\Xi}}\| = O(K^{-1/2})$. Therefore, since $\mathbf{a}_J(\tilde{\omega}_i)$ lies in the signal subspace, we have

$$\mathbf{a}_J^H(\tilde{\omega}_i)\widehat{\mathbf{E}}_{\mathrm{n},0}\widehat{\mathbf{E}}_{\mathrm{n},0}^H = \mathbf{a}_J^H(\tilde{\omega}_i)\mathbf{E}_{\mathrm{s},0}\mathbf{E}_{\mathrm{s},0}^H\widehat{\mathbf{E}}_{\mathrm{n},0}\widehat{\mathbf{E}}_{\mathrm{n},0}^H \tag{5.111}$$

$$= \mathbf{a}_J^H(\tilde{\omega}_i)\mathbf{E}_{\mathrm{s},0}(\mathbf{E}_{\mathrm{s},0} - \widehat{\mathbf{E}}_{\mathrm{s},0})^H\widehat{\mathbf{E}}_{\mathrm{n},0}\widehat{\mathbf{E}}_{\mathrm{n},0}^H \tag{5.112}$$

$$= -\mathbf{a}_J^H(\tilde{\omega}_i)\mathbf{E}_{\mathrm{s},0}\widehat{\boldsymbol{\Xi}}^H(\mathbf{I} - \widehat{\mathbf{E}}_{\mathrm{s},0}\widehat{\mathbf{E}}_{\mathrm{s},0}^H), \tag{5.113}$$

where we used the fact that $\widehat{\mathbf{E}}_{\mathrm{s},0}^H\widehat{\mathbf{E}}_{\mathrm{n},0} = \mathbf{O}$ to obtain (5.112). Replacing $\widehat{\mathbf{E}}_{\mathrm{s},0}$ by $\mathbf{E}_{\mathrm{s},0} + \widehat{\boldsymbol{\Xi}}$ and neglecting $O(1/K)$ terms, we obtain

$$\mathbf{a}_J^H(\tilde{\omega}_i)\widehat{\mathbf{E}}_{\mathrm{n},0}\widehat{\mathbf{E}}_{\mathrm{n},0}^H \approx -\mathbf{a}_J^H(\tilde{\omega}_i)\mathbf{E}_{\mathrm{s},0}\widehat{\boldsymbol{\Xi}}^H(\mathbf{I} - \mathbf{E}_{\mathrm{s},0}\mathbf{E}_{\mathrm{s},0}^H) \tag{5.114}$$

$$= -\mathbf{a}_J^H(\tilde{\omega}_i)\mathbf{E}_{\mathrm{s},0}\widehat{\boldsymbol{\Xi}}^H\mathbf{E}_{\mathrm{n},0}\mathbf{E}_{\mathrm{n},0}^H \tag{5.115}$$

$$= -\mathbf{a}_J^H(\tilde{\omega}_i)\mathbf{E}_{\mathrm{s},0}\widehat{\mathbf{E}}_{\mathrm{s},0}^H\mathbf{E}_{\mathrm{n},0}\mathbf{E}_{\mathrm{n},0}^H, \tag{5.116}$$

where we used the fact that $\mathbf{E}_{\mathrm{s},0}^H \mathbf{E}_{\mathrm{n},0} = \mathbf{O}$ to obtain the last equality. Substituting this into (5.110), we have

$$\hat{\omega}_i - \omega_i = B_i + \widehat{R}_i, \tag{5.117}$$

where

$$\widehat{R}_i = \frac{\mathrm{Re}\{\mathbf{a}_J^H(\tilde{\omega}_i)\mathbf{E}_{\mathrm{s},0}\widehat{\mathbf{E}}_{\mathrm{s},0}^H\mathbf{E}_{\mathrm{n},0}\mathbf{E}_{\mathrm{n},0}^H\dot{\mathbf{a}}_J(\tilde{\omega}_i)\}}{Mg(\tilde{\omega}_i)}. \tag{5.118}$$

Here we see that the DOA estimation errors $\hat{\omega}_i - \omega_i$ are asymptotically (for large K) jointly Gaussian since the signal eigenvector estimates in $\widehat{\mathbf{E}}_{\mathrm{s},0}$ are jointly complex Gaussian due to Lemma 5.1. Note that $\mathrm{E}\left[\widehat{R}_i\right] = 0$ because $\mathrm{E}\left[\widehat{\mathbf{E}}_{\mathrm{s},0}^H\mathbf{E}_{\mathrm{n},0}\right] = \mathbf{E}_{\mathrm{s},0}^H\mathbf{E}_{\mathrm{n},0} = \mathbf{O}$, so $\mathrm{E}[\hat{\omega}_i - \omega_i] = B_i$. Moreover,

$$\mathrm{E}[(\hat{\omega}_i - \omega_i)(\hat{\omega}_k - \omega_k)] = B_iB_k + B_i\,\mathrm{E}\left[\widehat{R}_k\right] + \mathrm{E}\left[\widehat{R}_i\right]B_k + \mathrm{E}\left[\widehat{R}_i\widehat{R}_k\right] \tag{5.119}$$

$$= B_iB_k + \mathrm{E}\left[\widehat{R}_i\widehat{R}_k\right], \tag{5.120}$$

where

$$\mathrm{E}\left[\widehat{R}_i\widehat{R}_k\right] = \mathrm{E}\left[\frac{\mathrm{Re}\{\mathbf{a}_J^H(\tilde{\omega}_i)\sum_{l=1}^{D_0}\mathbf{e}_{l,0}\hat{\mathbf{e}}_{l,0}^H\mathbf{E}_{\mathrm{n},0}\mathbf{E}_{\mathrm{n},0}^H\dot{\mathbf{a}}_J(\tilde{\omega}_i)\}}{Mg(\tilde{\omega}_i)}\right.$$
$$\left.\cdot\,\frac{\mathrm{Re}\{\mathbf{a}_J^H(\tilde{\omega}_k)\sum_{r=1}^{D_0}\mathbf{e}_{r,0}\hat{\mathbf{e}}_{r,0}^H\mathbf{E}_{\mathrm{n},0}\mathbf{E}_{\mathrm{n},0}^H\dot{\mathbf{a}}_J(\tilde{\omega}_k)\}}{Mg(\tilde{\omega}_k)}\right].$$

Using the fact that $\mathrm{Re}\{u\}\mathrm{Re}\{v\} = \mathrm{Re}\{uv + uv^*\}/2$ for any $u, v \in \mathbb{C}$, we obtain

$$\mathrm{E}\left[\widehat{R}_i\widehat{R}_k\right] = \frac{\mathrm{Re}\{\gamma_1 + \gamma_2\}}{2M^2g(\tilde{\omega}_i)g(\tilde{\omega}_k)}, \tag{5.121}$$

where

$$\gamma_1 = \sum_{l=1}^{D_0}\sum_{r=1}^{D_0}\left[\mathbf{e}_{l,0}^H\mathbf{a}_J(\tilde{\omega}_i)\mathbf{a}_J^H(\tilde{\omega}_k)\mathbf{e}_{r,0}\dot{\mathbf{a}}_J^H(\tilde{\omega}_i)\right.$$
$$\left.\cdot\,\mathbf{E}_{\mathrm{n},0}\mathbf{E}_{\mathrm{n},0}^H\,\mathrm{E}\left[\hat{\mathbf{e}}_{l,0}\hat{\mathbf{e}}_{r,0}^H\right]\mathbf{E}_{\mathrm{n},0}\mathbf{E}_{\mathrm{n},0}^H\dot{\mathbf{a}}_J(\tilde{\omega}_k)\right] \tag{5.122}$$

and

$$\gamma_2 = \sum_{l=1}^{D_0}\sum_{r=1}^{D_0}\left[\mathbf{e}_{l,0}^H\mathbf{a}_J(\tilde{\omega}_i)\mathbf{a}_J^T(\tilde{\omega}_k)\mathbf{e}_{r,0}^*\dot{\mathbf{a}}_J^H(\tilde{\omega}_i)\right.$$
$$\left.\cdot\,\mathbf{E}_{\mathrm{n},0}\mathbf{E}_{\mathrm{n},0}^H\,\mathrm{E}\left[\hat{\mathbf{e}}_{l,0}\hat{\mathbf{e}}_{r,0}^T\right]\mathbf{E}_{\mathrm{n},0}^*\mathbf{E}_{\mathrm{n},0}^T\dot{\mathbf{a}}_J^*(\tilde{\omega}_k)\right]. \tag{5.123}$$

Using (5.21) and noting that $\mathbf{E}_{\mathrm{n},0}^H\mathbf{e}_{i,0} = 0$ for $i = 1, \ldots, D_0$ and $\mathbf{E}_{\mathrm{n},0}\mathbf{E}_{\mathrm{n},0}^H\mathbf{e}_{i,0} = \mathbf{e}_{i,0}$ for $i = D_0 + 1, \ldots, J$, we obtain

$$\mathbf{E}_{\mathrm{n},0}\mathbf{E}_{\mathrm{n},0}^H\,\mathrm{E}\left[\hat{\mathbf{e}}_{l,0}\hat{\mathbf{e}}_{r,0}^H\right]\mathbf{E}_{\mathrm{n},0}\mathbf{E}_{\mathrm{n},0}^H = \mathbf{E}_{\mathrm{n},0}\mathbf{B}_{lr}\mathbf{E}_{\mathrm{n},0}^H/K, \tag{5.124}$$

where $\mathbf{B}_{lr}$ are as defined in (5.30). Similarly, using (5.22), we obtain

$$\mathbf{E}_{\mathrm{n},0}\mathbf{E}_{\mathrm{n},0}^{H}\,\mathrm{E}\left[\hat{\mathbf{e}}_{l,0}\hat{\mathbf{e}}_{r,0}^{T}\right]\mathbf{E}_{\mathrm{n},0}^{*}\mathbf{E}_{\mathrm{n},0}^{T} = \mathbf{E}_{\mathrm{n},0}\mathbf{C}_{lr}\mathbf{E}_{\mathrm{n},0}^{T}/K, \tag{5.125}$$

where $\mathbf{C}_{lr}$ are as defined in (5.31). Using (5.120)–(5.125), we finally derive the expressions for covariance in the theorem statement.

5.C Proof of Theorems 5.2 and 5.3

In view of (5.47), (5.48), and (5.52), we need to compute the derivative of $\mathbf{r}_{\mathbf{yy}}$ with respect to each parameter in $\boldsymbol{\alpha}$. According to (5.6)–(5.8), we have that

$$\begin{aligned}
\mathbf{r}_{\mathbf{yy}} &= \mathrm{vec}(\mathbf{R}_{\mathbf{yy}}) && (5.126)\\
&= \mathrm{vec}\left(\sum_{i=1}^{D}\sum_{k=1}^{D} P_{i,k}\mathbf{b}(\omega_i)\mathbf{b}^{H}(\omega_k) + p_{\mathrm{e}}\mathbf{H}\mathbf{H}^{H}\right) && (5.127)\\
&= \sum_{i=1}^{D}\sum_{k=1}^{D} P_{i,k}\mathbf{b}^{*}(\omega_k)\otimes\mathbf{b}(\omega_i) + p_{\mathrm{e}}\mathrm{vec}(\mathbf{H}\mathbf{H}^{H}) && (5.128)\\
&= \sum_{i=1}^{D} p_i\mathbf{b}^{*}(\omega_i)\otimes\mathbf{b}(\omega_i) + p_{\mathrm{e}}\mathrm{vec}(\mathbf{H}\mathbf{H}^{H})\\
&\quad + \sum_{i>k} P_{ik}^{(\mathrm{r})}\left[\mathbf{b}^{*}(\omega_k)\otimes\mathbf{b}(\omega_i) + \mathbf{b}^{*}(\omega_i)\otimes\mathbf{b}(\omega_k)\right]\\
&\quad + \sum_{i>k} jP_{ik}^{(\mathrm{i})}\left[\mathbf{b}^{*}(\omega_k)\otimes\mathbf{b}(\omega_i) - \mathbf{b}^{*}(\omega_i)\otimes\mathbf{b}(\omega_k)\right], && (5.129)
\end{aligned}$$

where $\mathbf{b}(\omega_i) = H(e^{j\omega_i})e^{j\omega_i(L-1)}\mathbf{a}_{N-L+1}(\omega_i)$ for each i. We used the fact that $\mathbf{P}$ is Hermitian symmetric to obtain (5.129). Using (5.128), we can derive that

$$\left[\frac{\partial\mathbf{r}_{\mathbf{yy}}}{\partial\omega_1}\;\cdots\;\frac{\partial\mathbf{r}_{\mathbf{yy}}}{\partial\omega_D}\right] = \dot{\mathbf{A}}_L^{*}\odot(\mathbf{A}_L\mathbf{D}_H\mathbf{P}\mathbf{D}_H^{*}) + (\mathbf{A}_L\mathbf{D}_H\mathbf{P}\mathbf{D}_H^{*})^{*}\odot\dot{\mathbf{A}}_L, \tag{5.130}$$

where $\dot{\mathbf{A}}_L$ is as in (5.51), and that

$$\frac{\partial\mathbf{r}_{\mathbf{yy}}}{\partial p_{\mathrm{e}}} = \mathrm{vec}(\mathbf{H}\mathbf{H}^{H}). \tag{5.131}$$

Using (5.129), we can derive that

$$\begin{aligned}
\frac{\partial\mathbf{r}_{\mathbf{yy}}}{\partial p_i} &= \mathbf{b}^{*}(\omega_i)\otimes\mathbf{b}(\omega_i) && (5.132)\\
&= \left[(\mathbf{A}_L\mathbf{D}_H)^{*}\otimes(\mathbf{A}_L\mathbf{D}_H)\right]_{:,(i-1)D+i} && (5.133)
\end{aligned}$$

for $i = 1, \ldots, D$ and

$$\frac{\partial \mathbf{r}_{\mathbf{yy}}}{\partial P_{ik}^{(\mathrm{r})}} = \mathbf{b}^*(\omega_k) \otimes \mathbf{b}(\omega_i) + \mathbf{b}^*(\omega_i) \otimes \mathbf{b}(\omega_k) \tag{5.134}$$

$$= [(\mathbf{A}_L\mathbf{D}_H)^* \otimes (\mathbf{A}_L\mathbf{D}_H)]_{:,(k-1)D+i} + [(\mathbf{A}_L\mathbf{D}_H)^* \otimes (\mathbf{A}_L\mathbf{D}_H)]_{:,(i-1)D+k} \tag{5.135}$$

$$\frac{\partial \mathbf{r}_{\mathbf{yy}}}{\partial P_{ik}^{(\mathrm{i})}} = j\mathbf{b}^*(\omega_k) \otimes \mathbf{b}(\omega_i) - j\mathbf{b}^*(\omega_i) \otimes \mathbf{b}(\omega_k) \tag{5.136}$$

$$= j[(\mathbf{A}_L\mathbf{D}_H)^* \otimes (\mathbf{A}_L\mathbf{D}_H)]_{:,(k-1)D+i} - j[(\mathbf{A}_L\mathbf{D}_H)^* \otimes (\mathbf{A}_L\mathbf{D}_H)]_{:,(i-1)D+k} \tag{5.137}$$

for $i > k$. Hence, there exists a matrix $\mathbf{T} \in \mathbb{C}^{D^2 \times D^2}$ such that

$$\left[\left[\frac{\partial \mathbf{r}_{\mathbf{yy}}}{\partial p_i}\right]_{i=1}^{D} \left[\frac{\partial \mathbf{r}_{\mathbf{yy}}}{\partial P_{ik}^{(\mathrm{r})}}\right]_{i>k} \left[\frac{\partial \mathbf{r}_{\mathbf{yy}}}{\partial P_{ik}^{(\mathrm{i})}}\right]_{i>k}\right] = [(\mathbf{A}_L\mathbf{D}_H)^* \otimes (\mathbf{A}_L\mathbf{D}_H)]\mathbf{T} \tag{5.138}$$

holds. More precisely, $[\mathbf{T}]_{:,i} = \boldsymbol{\delta}^{(D^2)}_{(i-1)D+i}$ for $i = 1, \ldots, D$, and the column of $\mathbf{T}$ corresponding to $\frac{\partial \mathbf{r}_{\mathbf{yy}}}{\partial P_{ik}^{(\mathrm{r})}}$, $\frac{\partial \mathbf{r}_{\mathbf{yy}}}{\partial P_{ik}^{(\mathrm{i})}}$ are

$$\boldsymbol{\delta}^{(D^2)}_{(k-1)D+i} + \boldsymbol{\delta}^{(D^2)}_{(i-1)D+k}, j\boldsymbol{\delta}^{(D^2)}_{(k-1)D+i} - j\boldsymbol{\delta}^{(D^2)}_{(i-1)D+k}, \tag{5.139}$$

respectively, for $i > k$. One can check that $\mathbf{T}$ is invertible. In view of (5.46), the CRB (5.37) exists if and only if $[\mathbf{G}\ \boldsymbol{\Delta}_1]$ has full column rank. This proves Theorem 5.2 when we substitute (5.130), (5.131), and (5.138) into (5.47) and (5.48) and note that $(\mathbf{R}_{\mathbf{yy}}^T \otimes \mathbf{R}_{\mathbf{yy}})^{-1/2}$ and $\mathbf{T}$ are invertible. Besides, (5.52) becomes

$$\mathrm{CRB}_{\mathrm{unb}}(\boldsymbol{\omega}) = \frac{1}{K}\left(\mathbf{G}^H\boldsymbol{\Pi}^{\perp}_{\boldsymbol{\Delta}_1}\mathbf{G}\right)^{-1}, \tag{5.140}$$

where $\mathbf{G}$ is as in (5.54), and

$$\boldsymbol{\Delta}_1 = \left(\mathbf{R}_{\mathbf{yy}}^T \otimes \mathbf{R}_{\mathbf{yy}}\right)^{-\frac{1}{2}} \left[[(\mathbf{A}_L\mathbf{D}_H)^* \otimes (\mathbf{A}_L\mathbf{D}_H)]\mathbf{T} \quad \mathrm{vec}(\mathbf{H}\mathbf{H}^H)\right]. \tag{5.141}$$

We finally obtain (5.53) by noting that $\boldsymbol{\Pi}^{\perp}_{\boldsymbol{\Delta}_1} = \boldsymbol{\Pi}^{\perp}_{\boldsymbol{\Delta}}$ since $\mathbf{T}$ is invertible and does not alter the column space.

Chapter 6

DISTRIBUTED ALGORITHMS FOR ARRAY SIGNAL PROCESSING

6.1 Introduction

Beamforming and direction-of-arrival (DOA) estimation are two important areas of sensor array processing [8, 26, 27, 36, 52]. Popular algorithms include the Capon method for beamforming [36] and MUSIC [52], root-MUSIC [26], ESPRIT [27], and FOCUSS [37, 38] for DOA estimation. Traditionally, these algorithms require data collection and centralized computation at a fusion center. However, following the pioneering work of Scaglione, et al. [7], distributed algorithms for DOA estimation and beamforming have gained more research interest. In these algorithms, the sensor array is partitioned into subarrays. The data in subarray i is available only to the processor in that subarray, and between the processors there is some minimal exchange of intermediate results, in order to implement an average consensus. Based on such local computations and limited data exchange between processors, the goal is to perform usual array processing tasks such as DOA estimation and beamforming. Thus, such arrays work without the help of a central processor or fusion center. The communication bottleneck that is present in the case of large arrays with a central processor is thus mitigated in these decentralized (or distributed) systems. A detailed discussion of the relevance and advantages of such distributed processors can be found in [42] and references therein.

An excellent overview of distributed implementations of principal component methods is presented in [42], with a mention of applications in array processing as well. In the above papers for distributed array processing, a network gossiping protocol called average consensus (AC) [20–22] is extensively used as a backbone algorithm to avoid the need of fusion centers. AC is a method for computing the average of some values stored at all the subarrays. Subspace-based methods for DOA estimation, including MUSIC [52], root-MUSIC [26], and ESPRIT [27], require the computation of the eigenvalue decomposition (EVD) of the covariance matrix of the array output. To compute the EVD in a fully distributed manner, a distributed power method is proposed in [7]. In [35], Suleiman, et al. propose a distributed algorithm for ESPRIT based on least-squares estimates (LS-ESPRIT), although the

total least squares (TLS) ESPRIT is not considered. All the above are AC-based methods and apply to any network structure. Bertrand, et al. [115] propose a non-AC-based method that uses an alternating optimization procedure to estimate covariance matrix eigenvectors. The communication cost is reduced by sending array data projected onto lower dimensional subspaces. However, this method applies only to fully connected networks or networks with a tree topology. Using a similar idea, a distributed DOA estimation method which applies only to fully connected networks is presented in [116]. Partially distributed algorithms for MUSIC and root-MUSIC are introduced in earlier literature [117] and [118], but fusion centers are still required therein. We focus on AC-based methods in this chapter as it applies to any network structure. The main goal of this chapter is to show how to use the distributed power method and AC to implement several important array processing algorithms, but not to compare different kinds of distributed eigenvector estimation methods or AC methods.

Importantly, there are two families of AC methods, *asymptotic convergence* [20] and the recently introduced *finite-time convergence* methods [21, 22]. Asymptotic AC is used for the distributed DOA estimation algorithms in previous works [7, 35, 42, 53]. In these methods, one uses only finite but sufficiently many iterations to approximate asymptotic behaviors, so additional estimation errors arise from the use of asymptotic AC. This kind of error is analyzed in [53] for distributed LS-ESPRIT. By contrast, finite-time AC offers exact convergence in a finite number of iterations, so no additional estimation errors are introduced. Hence, we choose to use finite-time AC in this chapter, and this is why our distributed algorithms can achieve the same performance as the centralized counterparts. Finite-time AC can be applied without any limitations, so they are readily applicable to the previous works [7, 35, 42, 53] as well. Exact convergence is guaranteed in finite iterations as long as the subarray network is connected [21]. The finite-time AC methods [21, 22] are based on the idea of linear graph filters. When we are allowed to use a large enough filter order, i.e., a sufficient number of AC iterations, these finite-time AC methods already yield zero MSE due to exact convergence. We consider this scenario in this chapter. If one wants to use fewer iterations while sacrificing some MSE, a new finite-time AC method [119] may be considered. The new method extends the idea to nonlinear graph filters and designs filter coefficients using a learning framework of graph convolutional neural networks.

Although there are several inspiring papers on distributed algorithms for array

processing, distributed algorithms have not yet been reported for a number of well-known methods for DOA estimation and beamforming. In this chapter, we develop distributed algorithms for DOA estimation methods such as root-MUSIC, and total least squares (TLS) ESPRIT which is known to be more accurate than LS-ESPRIT [35]. We also derive distributed versions of the Capon beamformer and the well-known FOCUSS method for sparse-solver based DOA estimation. The above DOA estimation methods are classical methods, and we only consider these. More recent methods like grid-based Lasso [59] and grid-less atomic norm minimization [120] are left as future work. Besides, *beamspace* processing is a well-known technique in array processing [2, 8, 12, 14–19, 23, 24, 55], and in Sec. 6.4, we propose distributed algorithms for the convolutional beamspace (CBS) method introduced in Chapter 2. We will show that distributed DOA estimation algorithms, including root-MUSIC, TLS-ESPRIT, and FOCUSS, can be applied either directly to the original array domain, called *element-space*, or in series with a beamspace method like CBS. We also propose a distributed algorithm for spatial smoothing [39], which is a technique used for DOA estimation when there are coherent or correlated sources.

All the proposed algorithms are *fully distributed* in that a fusion center is not required. The novelties of the proposed algorithms mainly lie in finding a way to implementing the algorithm so that all the data exchange among subarrays can be realized using AC. This is achieved by transforming problems at hand into steps where computing the average of some values across the network is the only step that involves data exchange among subarrays. In particular, we often transform the problems into a series of linear operations on the involved distributed variables, such as array outputs and signal eigenvectors. For instance, in Capon beamforming, it would not be easy if one tries to compute the inverse of the covariance matrix directly when the array output is distributed. The novelty of our distributed Capon beamforming is to propose to use the conjugate gradient method [121] to replace the inversion of the covariance matrix with linear operations on array outputs. Likewise, only linear operations on array outputs are needed in distributed FOCUSS. As another example, in root-MUSIC, it would not be easy if one tries to explicitly compute $\hat{\mathbf{E}}_{\mathrm{s}}\hat{\mathbf{E}}_{\mathrm{s}}^{H}$ when the eigenvectors are distributed, where the columns of $\hat{\mathbf{E}}_{\mathrm{s}}$ are the estimates of the signal eigenvectors. The novelty of our distributed root-MUSIC is to propose to avoid explicit computation of $\hat{\mathbf{E}}_{\mathrm{s}}\hat{\mathbf{E}}_{\mathrm{s}}^{H}$ and use the Aberth method [122] for rooting polynomials in root-MUSIC to ensure that only linear operations on signal eigenvectors are involved. That is, many of the algorithms in their original form require operations other than weighted averages of the involved data, and we

show how to transform those operations into a series of weighted averages. In order to put the new methods in the right context, this chapter includes short reviews of all important techniques which form the backbone of the new methods. Moreover, the existing and new, centralized and distributed methods are presented in a unified framework. In this sense the chapter is self contained and has some tutorial value as well.

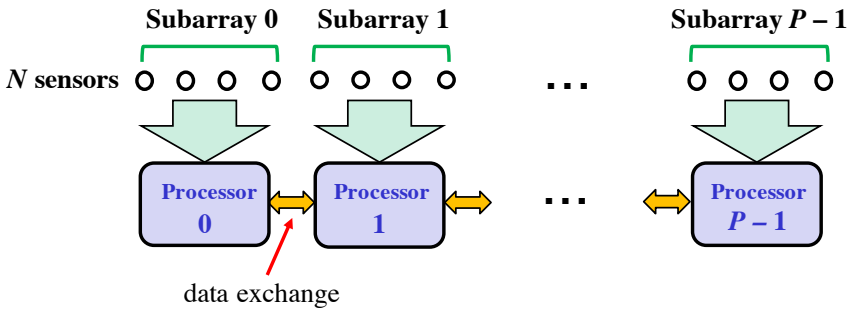


Figure 6.1: Schematic of distributed array processing. The N-sensor linear array is divided into P subarrays. The sensor data from subarray p is directly available only to processor p, corresponding to node p in a communication network modeled by an undirected graph $\mathcal{G}$. Between the processors there is some minimal exchange of intermediate results, in order to implement an average consensus.

The system model used throughout this chapter is shown in Fig. 6.1. This is a network composed of P nodes, each of which is a Q-sensor linear subarray [7, 35]. For ease of presentation, we assume that each subarray has the same number of sensors, but many algorithms in this chapter can be readily extended to subarrays with different numbers of sensors. The sensor data from subarray p is directly available only to a local processor at node p. Between the processors there is some minimal exchange of intermediate results, in order to implement an average consensus. The communication network is modeled by an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of the P nodes, and $\mathcal{E}$ is the set of edges. Each node represents a set of sensors and the edges represent the communication links. If there is an edge between two nodes, then two way communication is allowed between these nodes (for average consensus and so forth). The P subarrays, which do not have overlapping sensors, collectively form a linear array with $N = PQ$ sensors. That is, all the subarrays are on the same vertical positions. As in Sec. 1.1, the unit sensor spacing is $\lambda/2$, and monochromatic plane waves of wavelength λ arrive from D directions. Assume all the subarrays are located not too far away, and they receive the same set of source amplitudes. Let $\mathbf{x}_p \in \mathbb{C}^Q$ be the output of the pth

subarray. Then the array output is

$$\mathbf{x} = [\mathbf{x}_0^T \ \mathbf{x}_1^T \ \cdots \ \mathbf{x}_{P-1}^T]^T = \mathbf{A}\mathbf{c} + \mathbf{e}, \tag{6.1}$$

where $\mathbf{c}$ contains source amplitudes c_i, $\mathbf{e}$ contains additive noise terms, and $\mathbf{A} = [\mathbf{a}_N(\omega_1) \ \mathbf{a}_N(\omega_2) \ \cdots \ \mathbf{a}_N(\omega_D)]$ with

$$\mathbf{a}_N(\omega) = [e^{j\omega z_0} \ e^{j\omega z_1} \ \cdots \ e^{j\omega z_{N-1}}]^T. \tag{6.2}$$

Here $z_i \in \mathbb{Z}$ is the ith sensor location. Without loss of generality, assume $z_0 = 0$. In (6.2), $\omega = \pi \sin\theta$, with DOA $\theta \in [-\pi/2, \pi/2)$ measured from the normal to the line of array. We assume $\mathrm{E}[\mathbf{c}] = \mathbf{0}$, $\mathrm{E}[|c_i|^2] = p_i$, $\mathrm{E}[\mathbf{e}] = \mathbf{0}$, $\mathrm{E}[\mathbf{e}\mathbf{e}^H] = \sigma_e^2\mathbf{I}$, and $\mathrm{E}[\mathbf{c}\mathbf{e}^H] = \mathbf{0}$. In this chapter, we consider in general non-uniform linear arrays, but some of the proposed algorithms only apply to uniform linear arrays (ULAs), and we will mention it whenever it is the case.

Chapter outline: The distributed power method and average consensus are reviewed in Sec. 6.2. The proposed distributed algorithms for Capon beamforming, root-MUSIC, TLS-ESPRIT, and FOCUSS are introduced in Sec. 6.3. Then distributed convolutional beamspace methods, along with a variation called the "robust Capon beamspace filter" are introduced in Sec. 6.4. Distributed spatial smoothing is described in Sec. 6.5. Finally, Sec. 6.6 concludes the chapter.

6.2 Review of Distributed Power Method and Average Consensus

In this section, the distributed power method [7] and average consensus (AC) [20–22] are reviewed. The distributed power method is the first step for subspace-based DOA estimation algorithms as it estimates the eigenvectors of the covariance matrix of the array output. AC is a backbone subroutine for the distributed power method and other algorithms proposed in this chapter.

Subspace-based methods for DOA estimation, including MUSIC [52], root-MUSIC [26], and ESPRIT [27], require the computation of the eigenvalue decomposition (EVD) of the covariance matrix $\mathbf{R}_{\mathbf{xx}}$ of the array output $\mathbf{x}$. In practice, we use K snapshots to get the covariance estimate

$$\widehat{\mathbf{R}}_{\mathbf{xx}} = \frac{1}{K}\sum_{k=1}^{K} \mathbf{x}[k]\mathbf{x}^H[k]. \tag{6.3}$$

To compute the EVD of $\widehat{\mathbf{R}}_{\mathbf{xx}}$ in a fully distributed manner, a distributed power method was proposed in [7]. The method uses average consensus (AC) [20–22] as

a subroutine to compute the average of some values stored at all the subarrays. For average consensus, one can use methods which have either *asymptotic convergence* [20] or *finite-time convergence* [21, 22]. We propose to use finite-time AC in this chapter since it offers exact convergence in a finite number of iterations. A numerical example in Sec. 6.3.1 also shows that finite-time AC performs better than asymptotic AC. As shown below, as long as $\mathcal{G}$ is a connected graph, exact convergence can be achieved in at most $P-1$ iterations [21].

6.2.1 Average Consensus

The aim of AC is to compute the average of some initial scalar values $u_p(0)$ stored at nodes $p = 0, \ldots, P-1$. Nodes communicate with their neighbors and update their values through distributed linear iterations of the form

$$u_p(t+1) = W_{pp}(t)u_p(t) + \sum_{i \in \mathcal{N}_p} W_{pi}(t)u_i(t), \tag{6.4}$$

where $W_{pi}(t)$ is the weight on $u_i(t)$ at node p for iteration t, and $\mathcal{N}_p = \{i \mid \{p,i\} \in \mathcal{E}\}$ is the set of neighbors of node p. The weights $W_{pi}(t)$ are in general complex-valued and time-varying, though they are restricted to be time independent in some papers [7, 20]. We can write (6.4) in the vector form

$$\mathbf{u}(t+1) = \mathbf{W}(t)\mathbf{u}(t). \tag{6.5}$$

The goal of finite-time AC is to achieve

$$\mathbf{u}(I_{\text{ac}}) = \mathbf{1}\mathbf{1}^T\mathbf{u}(0)/P \tag{6.6}$$

after a finite number of iterations I_{ac}, while the goal of asymptotic AC is to achieve $\lim_{t\to\infty} \mathbf{u}(t) = \mathbf{1}\mathbf{1}^T\mathbf{u}(0)/P$. We propose to use finite-time AC in this chapter. In [21], it was shown that if $\mathcal{G}$ is a connected graph, then there exist weight matrices $\mathbf{W}(0), \ldots, \mathbf{W}(I_{\text{ac}}-1)$ with $I_{\text{ac}} \leq P-1$ such that (6.6) holds. Such weights are not unique, but one way to get a feasible solution is to start from the Laplacian $\mathbf{L}$ of $\mathcal{G}$. Suppose the distinct eigenvalues of $\mathbf{L}$ are $\lambda_1, \ldots, \lambda_R$, where $R \leq P$. Since $\mathcal{G}$ is connected, $\mathbf{L}$ must have the simple eigenvalue 0 [123]. Without loss of generality, assume $\lambda_R = 0$. It is shown in [21] that if we take

$$\mathbf{W}(0) = \frac{(-1)^{R-1}}{\lambda_1\lambda_2\cdots\lambda_{R-1}}(\mathbf{L} - \lambda_1\mathbf{I}) \tag{6.7}$$

and

$$\mathbf{W}(t) = \mathbf{L} - \lambda_{t+1}\mathbf{I} \tag{6.8}$$

for $t = 1, \ldots, R-2$, the iteration (6.5) converges after $I_{\text{ac}} = R-1 \leq P-1$ steps.

6.2.2 Distributed Power Method

Now we summarize the distributed power method [7]. We first consider the derivation of the first eigenvector of $\widehat{\mathbf{R}}_{\mathbf{xx}}$. Let $\mathbf{e}_1(0) \in \mathbb{C}^N$ be an initial random vector. Suppose all eigenvalues are distinct. Then the power iterations [124]

$$\mathbf{e}_1(n+1) = \widehat{\mathbf{R}}_{\mathbf{xx}}\mathbf{e}_1(n) \tag{6.9}$$

will converge to the first eigenvector as $n \to \infty$. Let

$$\mathbf{e}_1(n) = [\mathbf{e}_{1,0}^T(n)\ \mathbf{e}_{1,1}^T(n)\ \cdots\ \mathbf{e}_{1,P-1}^T(n)]^T \tag{6.10}$$

with each $\mathbf{e}_{1,p}(n) \in \mathbb{C}^Q$ stored locally at node p. Using (6.1) and (6.3), we can rewrite (6.9) as

$$\mathbf{e}_1(n+1) = \frac{1}{K}\sum_{k=1}^{K}\mathbf{x}[k]\sum_{p=0}^{P-1}\mathbf{x}_p^H[k]\mathbf{e}_{1,p}(n) \tag{6.11}$$

$$= \frac{P}{K}\sum_{k=1}^{K}\mathbf{x}[k]\,\mathrm{AC}_p(\mathbf{x}_p^H[k]\mathbf{e}_{1,p}(n)), \tag{6.12}$$

where $\mathrm{AC}_p(u_p)$ denotes the average consensus of the scalar values u_p with enough iterations so that the output is the exact average of u_p's. Hence, by first computing $b[k] = \mathrm{AC}_p(\mathbf{x}_p^H[k]\mathbf{e}_{1,p}(n))$, node p can locally compute

$$\mathbf{e}_{1,p}(n+1) = \frac{P}{K}\sum_{k=1}^{K}\mathbf{x}_p[k]b[k]. \tag{6.13}$$

Note that here we use the notation $b[k]$ instead of $b_p[k]$ because each node obtains a copy of the exact average, but one should understand that node p is using its own copy of $b[k]$ to compute (6.13). Suppose we run (6.9) for I_{pm} iterations. Then we do normalization $\hat{\mathbf{e}}_1 = \mathbf{e}_1(I_{\mathrm{pm}})/\|\mathbf{e}_1(I_{\mathrm{pm}})\|$ to obtain our final estimate $\hat{\mathbf{e}}_1$ of the normalized first eigenvector. The norm $\|\mathbf{e}_1(I_{\mathrm{pm}})\|$ can also be computed via AC since

$$\|\mathbf{e}_1(I_{\mathrm{pm}})\|^2 = P \cdot \mathrm{AC}_p(\mathbf{e}_{1,p}^H(I_{\mathrm{pm}})\mathbf{e}_{1,p}(I_{\mathrm{pm}})). \tag{6.14}$$

To derive the qth eigenvector $\hat{\mathbf{e}}_q$ of $\widehat{\mathbf{R}}_{\mathbf{xx}}$, we note that $\hat{\mathbf{e}}_q$ is the eigenvector corresponding to the largest eigenvalue of $(\mathbf{I} - \sum_{i=1}^{q-1}\hat{\mathbf{e}}_i\hat{\mathbf{e}}_i^H)\widehat{\mathbf{R}}_{\mathbf{xx}}$, where $\hat{\mathbf{e}}_i$ is the ith eigenvector of $\widehat{\mathbf{R}}_{\mathbf{xx}}$. Thus, it can be obtained by running the power iterations

$$\mathbf{e}_q(n+1) = \left(\mathbf{I} - \sum_{i=1}^{q-1}\hat{\mathbf{e}}_i\hat{\mathbf{e}}_i^H\right)\widehat{\mathbf{R}}_{\mathbf{xx}}\mathbf{e}_q(n). \tag{6.15}$$

Note that $\bar{\mathbf{e}}_q(n) \triangleq \widehat{\mathbf{R}}_{\mathbf{xx}}\mathbf{e}_q(n)$ can be computed in a manner similar to (6.12). Then,

$$\mathbf{e}_q(n+1) = \bar{\mathbf{e}}_q(n) - P\sum_{i=1}^{q-1}\hat{\mathbf{e}}_i\,\mathrm{AC}_p(\hat{\mathbf{e}}_{i,p}^H\bar{\mathbf{e}}_{q,p}(n)), \tag{6.16}$$

where $\hat{\mathbf{e}}_{i,p}, \bar{\mathbf{e}}_{q,p}(n) \in \mathbb{C}^Q$ are subvectors of $\hat{\mathbf{e}}_i$ and $\bar{\mathbf{e}}_q(n)$, respectively, corresponding to node p. Hence, by first computing $f_{i,q} = \mathrm{AC}_p(\hat{\mathbf{e}}_{i,p}^H\bar{\mathbf{e}}_{q,p}(n))$, node p can locally compute

$$\mathbf{e}_{q,p}(n+1) = \bar{\mathbf{e}}_{q,p}(n) - P\sum_{i=1}^{q-1}\hat{\mathbf{e}}_{i,p}f_{i,q}, \tag{6.17}$$

which is the subvector of $\mathbf{e}_q(n+1)$ corresponding to node p. Finally, after I_{pm} iterations, we do normalization $\hat{\mathbf{e}}_q = \mathbf{e}_q(I_{\mathrm{pm}})/\|\mathbf{e}_q(I_{\mathrm{pm}})\|$ to obtain our final estimate $\hat{\mathbf{e}}_q$ of the normalized qth eigenvector. The norm $\|\mathbf{e}_q(I_{\mathrm{pm}})\|$ can be computed in a manner similar to (6.14). The total communication cost per edge for estimating D eigenvectors is $O(D(D+K)I_{\mathrm{ac}}I_{\mathrm{pm}}) \approx O(DKI_{\mathrm{ac}}I_{\mathrm{pm}})$ if $D \ll K$. The cost is mainly dominated by computing $\bar{\mathbf{e}}_q(n)$ and (6.16). For comparison, the communication cost of the method in [115] applied to a fully connected network is $O(DKI_{\mathrm{ao}})$, where I_{ao} is the number of iterations for alternating optimization. When the distributed power method is applied to a fully connected network, $I_{\mathrm{ac}} = 1$, so the communication cost is $O(DKI_{\mathrm{pm}})$. The communication costs of the two methods have the same dependence on D and K. We use distributed power method in this chapter as it applies to any network structure.

The existence of $\hat{\mathbf{e}}_i$, $i < q$ in (6.15) implies that we can start the power iterations for $\hat{\mathbf{e}}_q$ only after the first $q-1$ eigenvectors are obtained. That is, the eigenvectors are updated sequentially. Alternatively, we can replace each $\hat{\mathbf{e}}_i$ by $\mathbf{e}_i(n+1)$, so the power iterations for all eigenvectors can be done in parallel. This can reduce time complexity. However, the modified method typically requires more iterations to compute each eigenvector, as shown later in Fig. 6.5. The total computational and communication costs are thus increased. Hence, we use only the original method in all other examples in this chapter.

6.3 Distributed DOA Estimation and Beamforming

In this section, we propose distributed algorithms for Capon beamforming [36], root-MUSIC [26], ESPRIT [27] based on total least-squares estimates (TLS-ESPRIT), and FOCUSS [37, 38]. (Note that LS-ESPRIT has been reported [35] and analyzed [53].) The distributed DOA estimation algorithms, including root-MUSIC, TLS-ESPRIT, and FOCUSS, can be applied either directly to element-space or in series

with a beamspace method, such as CBS presented in Sec. 6.4. In the following, we first show how to do distributed Capon beamforming in Sec. 6.3.1. Distributed root-MUSIC, ESPRIT, and FOCUSS are presented in Secs. 6.3.2, 6.3.3, and 6.3.4, respectively. FOCUSS is a method to obtain sparse solutions to underdetermined equations, so distributed FOCUSS has applications much broader than distributed DOA estimation. The communication, computation, and storage costs of the proposed distributed methods are compared to centralized methods in Sec. 6.3.5.

6.3.1 Distributed Capon Beamforming

In traditional beamforming, we consider a linear array output

$$\mathbf{x} = c_0 \mathbf{a}_N(\omega_0) + \mathbf{u}, \tag{6.18}$$

where $\mathbf{u} = \mathbf{A}\mathbf{c} + \mathbf{e}$ is the interference plus noise. The quantity ω_0 represents the "look direction," i.e., the direction in which we want to point the beam, c_0 is the amplitude of the signal coming from that direction, and $\mathbf{A}$, $\mathbf{c}$, $\mathbf{e}$ are as defined in (6.1), with $\omega_1, \ldots, \omega_D$ representing D interferer directions. The output of the beamformer can be expressed as

$$z_{\mathrm{BF}} = \mathbf{h}^H \mathbf{x}, \tag{6.19}$$

where $\mathbf{h} = [h(0) \cdots h(N-1)]^T$ is a complex weighting vector. The output signal-to-interference-plus-noise ratio (SINR) of the beamformer is defined as

$$\mathrm{SINR} = \frac{\mathrm{E}[|c_0 \mathbf{h}^H \mathbf{a}_N(\omega_0)|^2]}{\mathrm{E}[|\mathbf{h}^H \mathbf{u}|^2]} = \frac{p_0 |\mathbf{h}^H \mathbf{a}_N(\omega_0)|^2}{\mathbf{h}^H \mathbf{R}_{\mathbf{uu}} \mathbf{h}}, \tag{6.20}$$

where $p_0 = \mathrm{E}[|c_0|^2]$ and $\mathbf{R}_{\mathbf{uu}} = \mathrm{E}[\mathbf{u}\mathbf{u}^H]$. When the signal is uncorrelated with the interference, the Capon beamformer [8, 36], which is the solution to the optimization problem

$$\begin{aligned} \min_{\mathbf{h}} \quad & \mathbf{h}^H \mathbf{R}_{\mathbf{xx}} \mathbf{h} \\ \text{subject to} \quad & \mathbf{h}^H \mathbf{a}_N(\omega_0) = 1, \end{aligned} \tag{6.21}$$

maximizes the SINR, where $\mathbf{R}_{\mathbf{xx}} = \mathrm{E}[\mathbf{x}\mathbf{x}^H]$. The solution to this problem is given by

$$\mathbf{h} = c \cdot \mathbf{R}_{\mathbf{xx}}^{-1} \mathbf{a}_N(\omega_0), \tag{6.22}$$

where $c = 1/(\mathbf{a}_N^H(\omega_0) \mathbf{R}_{\mathbf{xx}}^{-1} \mathbf{a}_N(\omega_0))$. In practice, $\mathbf{R}_{\mathbf{xx}}$ is replaced by its estimate $\widehat{\mathbf{R}}_{\mathbf{xx}}$ as in (6.3).

Algorithm 1 Conjugate gradient method

1: $\mathbf{r}_0 = \mathbf{a}_N(\omega_0)$
2: $\mathbf{p}_0 = \mathbf{r}_0$
3: $\mathbf{w}_0 = \mathbf{0}$
4: **for** $i = 0$ to $N - 1$ **do**
5: $\quad \alpha_i = (\mathbf{r}_i^H \mathbf{r}_i)/(\mathbf{p}_i^H \widehat{\mathbf{R}}_{\mathbf{xx}} \mathbf{p}_i)$
6: $\quad \mathbf{w}_{i+1} = \mathbf{w}_i + \alpha_i \mathbf{p}_i$
7: $\quad \mathbf{r}_{i+1} = \mathbf{r}_i - \alpha_i \widehat{\mathbf{R}}_{\mathbf{xx}} \mathbf{p}_i$
8: $\quad$ **if** $\|\mathbf{r}_{i+1}\|^2 < \epsilon$ **then**
9: $\quad\quad$ **break** ▷ End iteration if residual approaches zero
10: $\quad$ **end if**
11: $\quad \beta_i = (\mathbf{r}_{i+1}^H \mathbf{r}_{i+1})/(\mathbf{r}_i^H \mathbf{r}_i)$
12: $\quad \mathbf{p}_{i+1} = \mathbf{r}_{i+1} + \beta_i \mathbf{p}_i$
13: **end for**
14: **return** $\mathbf{w}_{i+1}$

Now we show how to compute the Capon beamformer in our distributed setting, where the array output $\mathbf{x} = [\mathbf{x}_0^T \ \mathbf{x}_1^T \cdots \mathbf{x}_{P-1}^T]^T$ with $\mathbf{x}_p \in \mathbb{C}^Q$ stored locally at node p. The crucial step is to compute the inverse of the covariance estimate (6.3) appearing in (6.22). Instead of directly computing this inverse, we propose to use the conjugate gradient (CG) method [121] to compute

$$\mathbf{w} = \widehat{\mathbf{R}}_{\mathbf{xx}}^{-1} \mathbf{a}_N(\omega_0). \tag{6.23}$$

Equivalently, we want to solve the system of linear equations

$$\widehat{\mathbf{R}}_{\mathbf{xx}} \mathbf{w} = \mathbf{a}_N(\omega_0), \tag{6.24}$$

where $\widehat{\mathbf{R}}_{\mathbf{xx}} = \widehat{\mathbf{R}}_{\mathbf{xx}}^H$ is assumed to be positive definite (it is generally true if $K \geq N$). The CG method is a method for solving a set of positive definite linear equations [125]. Hence, (6.24) can be solved by the CG method, which is summarized as a pseudocode in Algorithm 1 shown in the table. In the CG method, we construct a set of vectors $\mathbf{p}_0, \ldots, \mathbf{p}_{N-1}$ that form a basis for $\mathbb{C}^N$ and are mutually conjugate with respect to $\widehat{\mathbf{R}}_{\mathbf{xx}}$, i.e., $\mathbf{p}_i^H \widehat{\mathbf{R}}_{\mathbf{xx}} \mathbf{p}_k = 0$ for all $i \neq k$. Then the final solution $\mathbf{w}$ can be expressed as

$$\mathbf{w} = \sum_{i=0}^{N-1} \alpha_i \mathbf{p}_i \tag{6.25}$$

for some α_i. The basis vectors $\mathbf{p}_i$ and coefficients α_i are obtained iteratively as in Algorithm 1. The quantity $\mathbf{w}_{i+1} = \sum_{k=0}^{i} \alpha_k \mathbf{p}_k$ is the partial solution in iteration $i + 1$, and $\mathbf{r}_{i+1} = \mathbf{a}_N(\omega_0) - \widehat{\mathbf{R}}_{\mathbf{xx}} \mathbf{w}_{i+1}$ is the corresponding residual. Line 9 serves as a

checkpoint for ending the CG iterations. If the residual $\mathbf{r}_{i+1} = \mathbf{0}$, the CG iterations should be terminated. In practice, we check if $\|\mathbf{r}_{i+1}\|^2 < \epsilon$ for some small positive ϵ. For more details of the CG method, one may refer to [121]. The key point of using the CG method here is that $\widehat{\mathbf{R}}_{\mathbf{xx}}$ appears only in the linear operation $\mathbf{q}_i = \widehat{\mathbf{R}}_{\mathbf{xx}}\mathbf{p}_i$ in Lines 5 and 7. This *can be computed using AC* in a way similar to (6.12).

Algorithm 2 Distributed design of Capon beamformer

1: $\mathbf{r}_0 = \mathbf{a}_N(\omega_0)$
2: $a_0 = \mathbf{r}_0^H \mathbf{r}_0$
3: $\mathbf{p}_0 = \mathbf{r}_0$
4: $\mathbf{w}_0 = \mathbf{0}$
5: **for** $i = 0$ to $N-1$ **do**
6: **for** $k = 1$ to K **do**
7: $t[k] = P \cdot \mathrm{AC}_p(\mathbf{x}_p^H[k]\mathbf{p}_{i,p})$
8: **end for**
9: **for** $p = 0$ to $P-1$ **do**
10: $\mathbf{q}_{i,p} = \frac{1}{K}\sum_{k=1}^{K} \mathbf{x}_p[k]t[k]$ ▷ Compute $\mathbf{q}_i = \widehat{\mathbf{R}}_{\mathbf{xx}}\mathbf{p}_i$
11: **end for**
12: $b_i = P \cdot \mathrm{AC}_p(\mathbf{p}_{i,p}^H \mathbf{q}_{i,p})$
13: $\alpha_i = a_i / b_i$
14: **for** $p = 0$ to $P-1$ **do**
15: $\mathbf{w}_{i+1,p} = \mathbf{w}_{i,p} + \alpha_i \mathbf{p}_{i,p}$
16: $\mathbf{r}_{i+1,p} = \mathbf{r}_{i,p} - \alpha_i \mathbf{q}_{i,p}$
17: **end for**
18: $a_{i+1} = P \cdot \mathrm{AC}_p(\mathbf{r}_{i+1,p}^H \mathbf{r}_{i+1,p})$
19: **if** $a_{i+1} < \epsilon$ **then**
20: **break** ▷ End iteration if residual approaches zero
21: **end if**
22: $\beta_i = a_{i+1} / a_i$
23: **for** $p = 0$ to $P-1$ **do**
24: $\mathbf{p}_{i+1,p} = \mathbf{r}_{i+1,p} + \beta_i \mathbf{p}_{i,p}$
25: **end for**
26: **end for**
27: **return** $\mathbf{w}_{i+1}$

The proposed distributed Capon beamforming based on CG method is presented in Algorithm 2 shown in the table. It is essentially using the CG method to solve (6.24) and making the algorithm distributed. Some new variables are defined in Algorithm 2 so that some partial results can be reused. Throughout the algorithm, the vectors $\mathbf{p}_{i,p}, \mathbf{q}_{i,p}, \mathbf{r}_{i,p}, \mathbf{w}_{i,p} \in \mathbb{C}^Q$ are subvectors of $\mathbf{p}_i, \mathbf{q}_i, \mathbf{r}_i, \mathbf{w}_i \in \mathbb{C}^N$, respectively, corresponding to node p, and are accessible only to node p. Moreover, the computations in Lines 10, 15, 16, and 24 are done locally at each node *in parallel*. AC appears

in the algorithm in three places, Lines 7, 12, and 18. The inner products in the AC arguments are also computed locally at node p in parallel. Line 7 is the dominant operation for communication, leading to $O(KNI_{\text{ac}})$ communication cost per edge, where I_{ac} is the number of AC iterations. After computing the Capon beamformer (6.22) using Algorithm 2, we simply have to apply the beamformer as in (6.19). This can be also done using AC since

$$z_{\text{BF}} = P \cdot \text{AC}_p(\mathbf{h}_p^H \mathbf{x}_p). \tag{6.26}$$

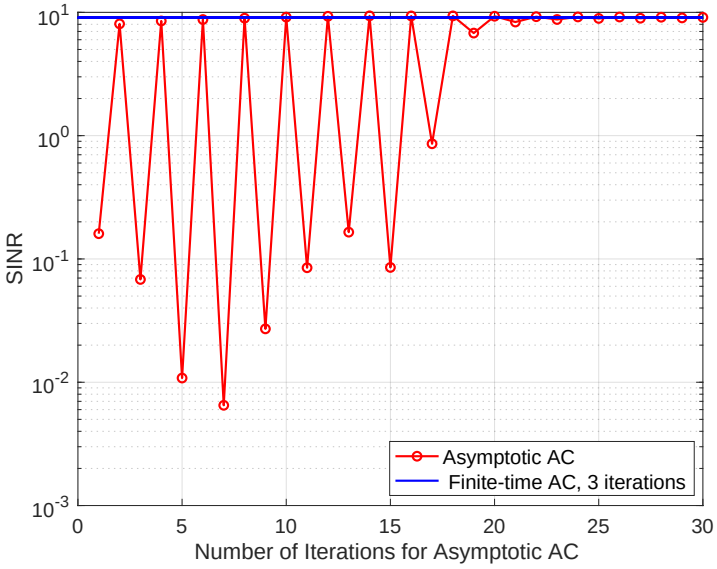
Here $\mathbf{h}_p \in \mathbb{C}^Q$ is the subvector of $\mathbf{h}$ corresponding to node p.

Our idea of using the CG method to compute the inverse of the covariance matrix can be applied to situations other than Capon beamforming. It can be used for any problem that requires the inversion of an array output covariance. For example, we can derive a distributed algorithm for the method of space-time adaptive processing (STAP) for MIMO radar systems proposed in [126].

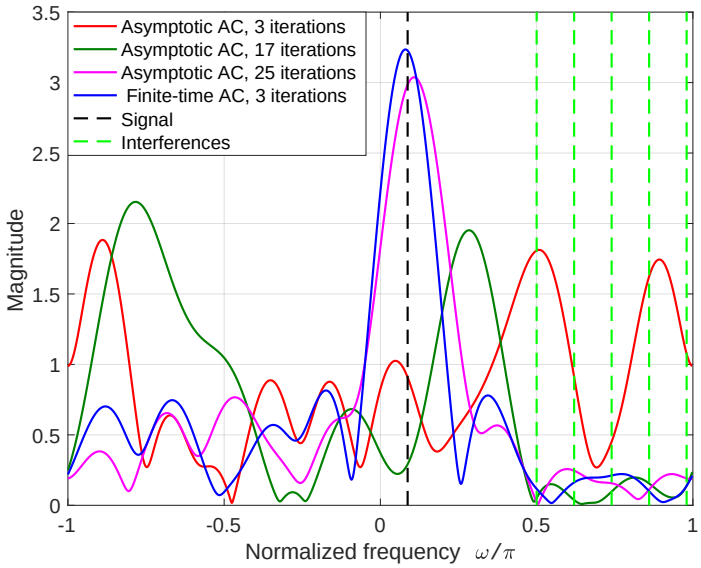
A numerical example is next given to demonstrate the effectiveness of the proposed distributed Capon beamforming. We consider the network as in [35] composed of $P = 6$ nodes, and the neighboring sets are

$$\begin{aligned} &\mathcal{N}_0 = \{1, 2\}, \mathcal{N}_1 = \{0, 2\}, \mathcal{N}_2 = \{0, 1, 3\}, \\ &\mathcal{N}_3 = \{2, 4, 5\}, \mathcal{N}_4 = \{3, 5\}, \mathcal{N}_5 = \{3, 4\}. \end{aligned} \tag{6.27}$$

For this network, using the finite-time AC method in [21], we can achieve exact convergence in 3 iterations. This is because, in this case, the Laplacian $\mathbf{L}$ has $R = 4$ distinct values. The weight matrix $\mathbf{W}(t)$ of each iteration can be found using (6.7) and (6.8). We compare this finite-time AC method with the asymptotic AC method using the optimal symmetric weight matrix proposed in [20]. Each node is a ULA with $Q = 2$ sensors and they together form a 12-sensor ULA. The signal of interest is at angle $\theta = 5°$, which is assumed known. There are 5 interfering sources with DOAs $\omega = 0.5\pi, 0.62\pi, 0.74\pi, 0.86\pi$, and 0.98π. (Recall $\omega = \pi \sin\theta$.) All sources are uncorrelated with power 1. The noise variance is $\sigma_e^2 = 1$. Output SINRs for various number of iterations for asymptotic AC are shown in Fig. 6.2(a). It does not make sense to consider the SINRs of finite-time AC before it reaches exact convergence with a sufficient finite number of iterations, so we just fix it to be that number, 3 in this example. We use 500 snapshots and average 2000 Monte Carlo runs to get the plot. Indeed, finite-time AC converges much faster than asymptotic AC. Although asymptotic AC gets SINRs similar to finite-time AC also in some of



(a)



(b)

Figure 6.2: Distributed Capon beamforming with asymptotic AC and finite-time AC methods. The number of iterations for finite-time AC is fixed at 3. (a) Output SINR. (b) Typical beamformer responses.

the small numbers of iterations, we cannot know which iterations yield larger SINRs if we do not know the ground truth, so in practice we still have to use a large enough number of iterations for asymptotic AC, like 20 in this example. Hence, we propose to use finite-time AC in our distributed algorithms. Note that the centralized Capon

beamformer yields exactly the same SINR as the finite-time AC method, so it is not plotted. Typical beamformer responses are also shown in Fig. 6.2(b). The beam pattern of asymptotic AC gradually converges to that of finite-time AC as iterations progress.

6.3.2 Distributed MUSIC and Root-MUSIC

In algorithms such as MUSIC [52] and root-MUSIC [26], we first compute the EVD of the covariance matrix $\mathbf{R_{xx}}$ of the array output $\mathbf{x}$ and obtain

$$\mathbf{R_{xx}} = \mathbf{E}_\mathrm{s}\mathbf{\Lambda}_\mathrm{s}\mathbf{E}_\mathrm{s}^H + \mathbf{E}_\mathrm{n}\mathbf{\Lambda}_\mathrm{n}\mathbf{E}_\mathrm{n}^H, \tag{6.28}$$

where $\mathbf{E}_\mathrm{s} = [\mathbf{e}_1 \cdots \mathbf{e}_D]$ and $\mathbf{E}_\mathrm{n} = [\mathbf{e}_{D+1} \cdots \mathbf{e}_N]$ contain the signal and noise eigenvectors respectively with $\mathbf{e}_k^H\mathbf{e}_m = \delta[k-m]$, and $\mathbf{\Lambda}_\mathrm{s}$ and $\mathbf{\Lambda}_\mathrm{n}$ are diagonal matrices containing the corresponding eigenvalues. In the MUSIC algorithm, we evaluate the MUSIC spectrum

$$P(\omega) = (\mathbf{a}_N^H(\omega)\mathbf{E}_\mathrm{n}\mathbf{E}_\mathrm{n}^H\mathbf{a}_N(\omega))^{-1} \tag{6.29}$$

on a dense grid of potential DOAs and identify local maxima as the DOA estimates. This is because theoretically, if ω is a true DOA, then the steering vector $\mathbf{a}_N(\omega)$ is orthogonal to noise subspace $\mathbf{E}_\mathrm{n}$, i.e., $\mathbf{a}_N^H(\omega)\mathbf{E}_\mathrm{n}\mathbf{E}_\mathrm{n}^H\mathbf{a}_N(\omega) = 0$. The converse (i.e., if $\mathbf{a}_N^H(\omega)\mathbf{E}_\mathrm{n}\mathbf{E}_\mathrm{n}^H\mathbf{a}_N(\omega) = 0$, then ω is a true DOA) is also true for a ULA (given the number of sources $D < N$), but not true in general for a non-ULA [56, 127, 128]. Our focus here is the development of distributed algorithms. We will see that these algorithms are insensitive to whether the array is uniform or not, and we do not focus on well-known identifiability issues here. In practice, $\mathbf{R_{xx}}$ and $\mathbf{e}_i$ are replaced by their finite-snapshot estimates $\widehat{\mathbf{R}}_{\mathbf{xx}}$ and $\hat{\mathbf{e}}_i$ respectively.

Root-MUSIC is a variation of the MUSIC algorithm. It avoids the evaluation of MUSIC spectrum on a dense grid by computing the roots of a polynomial, thereby improving accuracy, and reducing complexity. Thus, instead of finding local maxima of $P(\omega)$, we find roots ρ_i of the polynomial

$$f(z) = \underline{\mathbf{a}}_N^T(z)\hat{\mathbf{E}}_\mathrm{n}\hat{\mathbf{E}}_\mathrm{n}^H\overline{\mathbf{a}}_N(z), \tag{6.30}$$

where $\underline{\mathbf{a}}_N(z) = z^{z_N-1}[1 \;\; z^{-z_1} \cdots z^{-z_{N-1}}]^T$ and $\overline{\mathbf{a}}_N(z) = [1 \;\; z^{z_1} \cdots z^{z_{N-1}}]^T$ (with z_i denoting sensor locations as before). This polynomial is obtained by replacing $e^{j\omega}$ with z in $P^{-1}(\omega)$ and then multiplying it with z^{z_N-1}. The DOAs ω_i are obtained from the arguments of ρ_i.

Now we show how to do MUSIC and root-MUSIC in our distributed setting. In this case, the eigenvector estimates are obtained using the distributed power method in Sec. 6.2, and the results are $\hat{\mathbf{e}}_i = [\hat{\mathbf{e}}_{i,0}^T \cdots \hat{\mathbf{e}}_{i,P-1}^T]^T$ with $\hat{\mathbf{e}}_{i,p}$ only known to node p. Then the MUSIC spectrum can be evaluated as

$$P(\omega) = \left(\sum_{i=D+1}^{N} \left| P \cdot \mathrm{AC}_p(\hat{\mathbf{e}}_{i,p}^H \mathbf{a}_{N,p}(\omega)) \right|^2 \right)^{-1}, \tag{6.31}$$

where $\mathbf{a}_{N,p}(\omega) \in \mathbb{C}^Q$ is the subvector of $\mathbf{a}_N(\omega)$ corresponding to node p. Alternatively, since $\hat{\mathbf{E}} \triangleq [\hat{\mathbf{E}}_\mathrm{s}\ \hat{\mathbf{E}}_\mathrm{n}]$ is a unitary matrix so that $\hat{\mathbf{E}}_\mathrm{s}\hat{\mathbf{E}}_\mathrm{s}^H + \hat{\mathbf{E}}_\mathrm{n}\hat{\mathbf{E}}_\mathrm{n}^H = \mathbf{I}$, we also have

$$P(\omega) = (\mathbf{a}_N^H(\omega)(\mathbf{I} - \hat{\mathbf{E}}_\mathrm{s}\hat{\mathbf{E}}_\mathrm{s}^H)\mathbf{a}_N(\omega))^{-1} \tag{6.32}$$

$$= (N - \mathbf{a}_N^H(\omega)\hat{\mathbf{E}}_\mathrm{s}\hat{\mathbf{E}}_\mathrm{s}^H\mathbf{a}_N(\omega))^{-1} \tag{6.33}$$

$$= \left(N - \sum_{i=1}^{D} \left| P \cdot \mathrm{AC}_p(\hat{\mathbf{e}}_{i,p}^H \mathbf{a}_{N,p}(\omega)) \right|^2 \right)^{-1}. \tag{6.34}$$

Typically, (6.34) is advantageous over (6.31) in terms of computation and communication among nodes because dominant eigenvectors $\hat{\mathbf{e}}_1, \ldots, \hat{\mathbf{e}}_D$ are obtained first by the distributed power method, unless D is almost as large as N. The communication cost per edge for evaluating $P(\omega)$ using (6.34) is $O(DGI_\mathrm{ac})$, where G is the number of grid points used for ω.

For distributed root-MUSIC, we can similarly consider either (6.30) or

$$f(z) = Nz^{z_{N-1}} - \underline{\mathbf{a}}_N^T(z)\hat{\mathbf{E}}_\mathrm{s}\hat{\mathbf{E}}_\mathrm{s}^H\overline{\mathbf{a}}_N(z). \tag{6.35}$$

In the following, we use the form in (6.35) to derive our distributed algorithm because it is easier to obtain $\hat{\mathbf{E}}_\mathrm{s}$ than $\hat{\mathbf{E}}_\mathrm{n}$. A similar algorithm can be derived for (6.30). To understand the crucial step that is required here, consider $\hat{\mathbf{E}}_\mathrm{s} = [\hat{\mathbf{E}}_{\mathrm{s},0}^T \cdots \hat{\mathbf{E}}_{\mathrm{s},P-1}^T]^T$, where $\hat{\mathbf{E}}_{\mathrm{s},p} \in \mathbb{C}^{Q \times D}$ contains the local subvectors of eigenvectors for node p. To explicitly compute all entries of $\hat{\mathbf{E}}_\mathrm{s}\hat{\mathbf{E}}_\mathrm{s}^H$, we need to compute $\hat{\mathbf{E}}_{\mathrm{s},p}\hat{\mathbf{E}}_{\mathrm{s},q}^H$ for each pair of p and q. This cannot be done without a fusion center or sharing data among all nodes. Instead of explicitly computing $\hat{\mathbf{E}}_\mathrm{s}\hat{\mathbf{E}}_\mathrm{s}^H$, we propose to use the *Aberth method* [122] as our polynomial-rooting algorithm and show that *the method can be done in a distributed way* based on AC. Given a polynomial $f(z)$ of degree n, the Aberth method is an algorithm that finds all roots simultaneously by the iteration

$$z_{i+1,k} = z_{i,k} - \left[\frac{f'(z_{i,k})}{f(z_{i,k})} - \sum_{\substack{l=1 \\ l \neq k}}^{n} \frac{1}{z_{i,k} - z_{i,l}} \right]^{-1} \tag{6.36}$$

Algorithm 3 Distributed root-MUSIC

Randomly initialize $z_{0,1}, \ldots, z_{0,2z_{N-1}}$
for $i = 0$ to $I_{\max}$ **do**
  for $k = 1$ to $2z_{N-1}$ **do**
    $\overline{\mathbf{u}} = P \cdot \mathrm{AC}_p(\hat{\mathbf{E}}_{\mathrm{s},p}^H \overline{\mathbf{a}}_{N,p}(z_{i,k}))$
    $\underline{\mathbf{u}} = P \cdot \mathrm{AC}_p(\hat{\mathbf{E}}_{\mathrm{s},p}^T \underline{\mathbf{a}}_{N,p}(z_{i,k}))$
    $\overline{\mathbf{v}} = P \cdot \mathrm{AC}_p(\hat{\mathbf{E}}_{\mathrm{s},p}^H \overline{\mathbf{b}}_{N,p}(z_{i,k}))$
    $\underline{\mathbf{v}} = P \cdot \mathrm{AC}_p(\hat{\mathbf{E}}_{\mathrm{s},p}^T \underline{\mathbf{b}}_{N,p}(z_{i,k}))$
    $s = N z_{i,k}^{z_{N-1}} - \underline{\mathbf{u}}^T \overline{\mathbf{u}}$
    $t = N z_{N-1} z_{i,k}^{z_{N-1}-1} - \underline{\mathbf{v}}^T \overline{\mathbf{u}} - \underline{\mathbf{u}}^T \overline{\mathbf{v}}$
    $z_{i+1,k} = z_{i,k} - [t/s - \sum_{l \neq k}(1/(z_{i,k} - z_{i,l}))]^{-1}$
  end for
  if $\sum_k |z_{i+1,k} - z_{i,k}|^2 < \epsilon$ **then**
    break ▷ End iteration if all roots converge
  end if
end for
return $z_{i+1,1}, \ldots, z_{i+1,2z_{N-1}}$

for $k = 1, \ldots, n$. Here $z_{i,k}$ stands for the kth root in the ith iteration, and $f'(z_{i,k})$ is the first derivative of $f(z)$ evaluated at $z = z_{i,k}$. The key point of using the Aberth method is that $\hat{\mathbf{E}}_{\mathrm{s}}$ appears only in $f'(z_{i,k})$ and $f(z_{i,k})$, and we can show that distributed computation of these can be done using AC. After plugging (6.35) into (6.36) and some derivations, we can realize distributed root-MUSIC as summarized in Algorithm 3 shown in the table. In this algorithm, $\overline{\mathbf{a}}_{N,p}(z), \underline{\mathbf{a}}_{N,p}(z), \overline{\mathbf{b}}_{N,p}(z), \underline{\mathbf{b}}_{N,p}(z) \in \mathbb{C}^Q$ are subvectors of $\overline{\mathbf{a}}_N(z), \underline{\mathbf{a}}_N(z), \frac{\mathrm{d}}{\mathrm{d}z}\overline{\mathbf{a}}_N(z)$, and $\frac{\mathrm{d}}{\mathrm{d}z}\underline{\mathbf{a}}_N(z)$, respectively, corresponding to node p. The vectors $\overline{\mathbf{a}}_{N,p}(z_{i,k})$, $\underline{\mathbf{a}}_{N,p}(z_{i,k})$, $\overline{\mathbf{b}}_{N,p}(z_{i,k})$, $\underline{\mathbf{b}}_{N,p}(z_{i,k})$ can be evaluated at node p because each $z_{i,k}$ is known to all nodes. The matrix multiplications in the four AC arguments are computed locally at node p in parallel. The iterative algorithm stops either when all roots converge, i.e.,

$$\sum_k |z_{i+1,k} - z_{i,k}|^2 < \epsilon \tag{6.37}$$

for some small positive ϵ, or when a predefined maximum number of iterations $I_{\max}$ is reached. The total communication cost per edge is $O(z_{N-1} D I_{\mathrm{ac}} I_{\mathrm{ab}})$, where I_{ab} is the number of Aberth iterations. A numerical example is given in Sec. 6.4.4.

The Aberth method works even for non-distinct roots [122]. One scenario that hinders convergence is when the roots are symmetrically positioned in the complex plane with respect to some line, and the initial guess of the roots makes them also symmetrically placed with respect to the line. However, this happens with

probability zero in our case due to two reasons. First, the actual roots resulting from a finite number of snapshots of the array output are not symmetrical with probability one. Second, we initialize the roots randomly with some continuous probability distribution, so they are not symmetrical with probability one. Hence, essentially it will always converge in our case.

6.3.3 Distributed ESPRIT

ESPRIT [27] is another commonly used subspace-based DOA estimation algorithm. For ESPRIT, we assume that each node is a Q-sensor ULA with the same sensor spacing $\lambda/2$, but the displacements between the subarrays are unknown, so the entire array can be non-uniform. This array setting follows [35]. In [35], ESPRIT based on least-squares estimates (LS-ESPRIT) was shown to be realizable in this distributed setting. It is well-known that total least-squares (TLS) ESPRIT produces more accurate estimates than LS-ESPRIT [129]. In this subsection, we will show that distributed TLS-ESPRIT can also be done, and we will present distributed LS-ESPRIT and TLS-ESPRIT in a unified framework.

In ESPRIT, we require two groups of sensors with a shift invariance between them. Here we define the first group to be the first $Q-1$ sensors of each node and the second group to be the last $Q-1$ sensors of each node. Hence, the shift invariance is the unit sensor spacing $\lambda/2$. Equivalently, consider the selection matrices

$$\bar{\mathbf{J}}_1 = [\mathbf{I}_{Q-1}\ \mathbf{0}_{Q-1}] \tag{6.38}$$

and

$$\bar{\mathbf{J}}_2 = [\mathbf{0}_{Q-1}\ \mathbf{I}_{Q-1}], \tag{6.39}$$

where $\mathbf{0}_{Q-1}$ is the $(Q-1)\times 1$ zero vector. Moreover, let

$$\mathbf{J}_l = \mathbf{I}_P \otimes \bar{\mathbf{J}}_l \tag{6.40}$$

for $l = 1, 2$, where $\otimes$ denotes the Kronecker product. Then in ESPRIT, we have to select the rows of signal eigenvectors $\hat{\mathbf{E}}_{\mathrm{s}}$ corresponding to each group, i.e., $\hat{\mathbf{E}}_l = \mathbf{J}_l\hat{\mathbf{E}}_{\mathrm{s}}$ for $l = 1, 2$. Ideally, $\hat{\mathbf{E}}_1$ and $\hat{\mathbf{E}}_2$ have the same column space if we have infinite snapshots. In practice, we find the LS or TLS solution [27] to $\hat{\mathbf{E}}_1\boldsymbol{\Psi} = \hat{\mathbf{E}}_2$. The LS solution is

$$\boldsymbol{\Psi}_{\mathrm{LS}} = (\hat{\mathbf{E}}_1^H\hat{\mathbf{E}}_1)^{-1}(\hat{\mathbf{E}}_1^H\hat{\mathbf{E}}_2). \tag{6.41}$$

Then the DOA estimates of LS-ESPRIT are [27]

$$\hat{\omega}_{\mathrm{LS},k} = \arg(\lambda_k(\mathbf{\Psi}_{\mathrm{LS}})), \tag{6.42}$$

the argument of the kth eigenvalue of $\mathbf{\Psi}_{\mathrm{LS}}$. To compute the TLS solution, we consider

$$\mathbf{F} = \begin{bmatrix} \mathbf{F}_{1,1} & \mathbf{F}_{1,2} \\ \mathbf{F}_{2,1} & \mathbf{F}_{2,2} \end{bmatrix}, \tag{6.43}$$

where $\mathbf{F}_{l,m} = \hat{\mathbf{E}}_l^H \hat{\mathbf{E}}_m$ for $l = 1, 2$ and $m = 1, 2$, and compute its EVD

$$\mathbf{F} = \bar{\mathbf{E}} \bar{\Lambda} \bar{\mathbf{E}}^H, \tag{6.44}$$

with the eigenvalues in descending order. Then we decompose the matrix $\bar{\mathbf{E}} \in \mathbb{C}^{2D \times 2D}$ into $D \times D$ submatrices, i.e.,

$$\bar{\mathbf{E}} = \begin{bmatrix} \bar{\mathbf{E}}_{1,1} & \bar{\mathbf{E}}_{1,2} \\ \bar{\mathbf{E}}_{2,1} & \bar{\mathbf{E}}_{2,2} \end{bmatrix}. \tag{6.45}$$

The TLS solution is then

$$\mathbf{\Psi}_{\mathrm{TLS}} = -\bar{\mathbf{E}}_{1,2} \bar{\mathbf{E}}_{2,2}^{-1}, \tag{6.46}$$

and the DOA estimates of TLS-ESPRIT are

$$\hat{\omega}_{\mathrm{TLS},k} = \arg(\lambda_k(\mathbf{\Psi}_{\mathrm{TLS}})). \tag{6.47}$$

Now we show how to do ESPRIT in our distributed setting. In [35], the authors showed how to do this for LS-ESPRIT. We now give a unified derivation for distributed ESPRIT, which works whether it is LS- or TLS-ESPRIT. For $l = 1, 2$ and $m = 1, 2$, we have

$$[\hat{\mathbf{E}}_l^H \hat{\mathbf{E}}_m]_{i,k} = \sum_{p=0}^{P-1} (\bar{\mathbf{J}}_l \hat{\mathbf{e}}_{i,p})^H (\bar{\mathbf{J}}_m \hat{\mathbf{e}}_{k,p}) \tag{6.48}$$

$$= P \cdot \mathrm{AC}_p((\bar{\mathbf{J}}_l \hat{\mathbf{e}}_{i,p})^H (\bar{\mathbf{J}}_m \hat{\mathbf{e}}_{k,p})) \tag{6.49}$$

for each (i, k)-entry. Again, $\hat{\mathbf{e}}_i = [\hat{\mathbf{e}}_{i,0}^T \cdots \hat{\mathbf{e}}_{i,P-1}^T]^T$ is the ith eigenvector with $\hat{\mathbf{e}}_{i,p} \in \mathbb{C}^Q$ being the subvector corresponding to node p. Thus, $(\bar{\mathbf{J}}_l \hat{\mathbf{e}}_{i,p})^H (\bar{\mathbf{J}}_m \hat{\mathbf{e}}_{k,p})$ can be computed locally at node p. Then with AC, each node can obtain the entire matrix $\mathbf{F}$. Hence for LS-ESPRIT, we can compute

$$\mathbf{\Psi}_{\mathrm{LS}} = \mathbf{F}_{1,1}^{-1} \mathbf{F}_{1,2} \tag{6.50}$$

and then (6.42) locally at each node. Likewise, for TLS-ESPRIT, (6.44) through (6.47) can also be realized locally at each node. This leads to an increase in the total computational complexity since all the nodes are performing the same operations. An alternative would be for one node to do the computations and then send the final DOA estimates to the other nodes. This is a tradeoff between computation and communication. The communication cost per edge for both LS-ESPRIT and TLS-ESPRIT is $O(D^2 I_{\text{ac}})$, though TLS-ESPRIT has twice communication cost of LS-ESPRIT because TLS-ESPRIT requires the entire matrix $\mathbf{F}$ while LS-ESPRIT requires only $\mathbf{F}_{1,1}$ and $\mathbf{F}_{1,2}$. This is a tradeoff for getting better DOA estimates with TLS-ESPRIT.

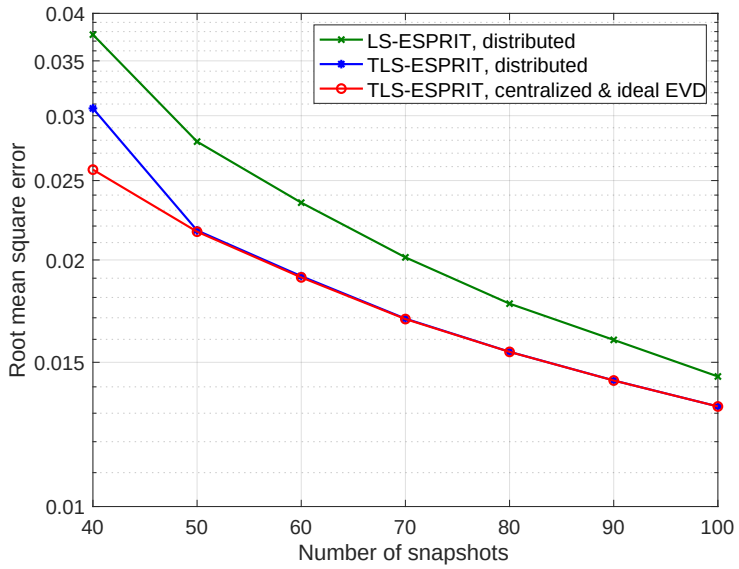


Figure 6.3: RMSE of DOA estimates using distributed LS-ESPRIT, distributed TLS-ESPRIT, and centralized TLS-ESPRIT.

A numerical example is given next to demonstrate the effectiveness of the proposed distributed TLS-ESPRIT. We consider the same network (6.27) and use the finite-time AC method with 3 iterations as explained in Sec. 6.3.1. We compare distributed TLS-ESPRIT with distributed LS-ESPRIT [35] and centralized TLS-ESPRIT [27]. Here the distributed power method shown in Sec. 6.2 is used to estimate eigenvectors of the array output covariance for distributed TLS-ESPRIT and distributed LS-ESPRIT, whereas ideal EVD of the covariance is assumed for centralized TLS-ESPRIT. The sensor locations of the pth node are $15p, 15p+1, \ldots, 15p+9$, so each node is a ULA with $Q = 10$ sensors, and there are 60 sensors in total. Note that the entire array is not a ULA. The displacement 15 between the subarrays is arbitrary

and assumed unknown. There are 6 DOAs at angles $\theta = -5^\circ, -3^\circ - 1^\circ, \ldots, 5^\circ$. The sources are uncorrelated with power -5 dB. The noise variance is $\sigma_e^2 = 1$. We assume the number of DOAs is known. The number of power iterations is $I_{\text{pm}} = 5$. Root mean square errors (RMSEs) of DOA estimates for various number of snapshots are shown in Fig. 6.3. In this chapter, whenever we mention RMSEs in detected source angles, we refer to averaging square errors measured in ω over all involved DOAs. We average 5000 Monte Carlo runs to get the plot. As expected, the RMSE of distributed TLS-ESPRIT is smaller than that of distributed LS-ESPRIT. The RMSE of centralized TLS-ESPRIT is even smaller at 40 snapshots because ideal EVD is used for it. However for 50 or more snapshots, distributed TLS-ESPRIT using only 5 power iterations performs almost the same as centralized TLS-ESPRIT.

6.3.4 Distributed FOCUSS

The FOCUSS algorithm [37, 38] for finding sparse solutions to an underdetermined system can also be used to estimate DOAs. We first briefly review the centralized FOCUSS for the case of multiple measurement vectors [38], or multiple snapshots in our language. To use FOCUSS for DOA estimation, a dictionary $\mathbf{D}$ of steering vectors $\mathbf{a}_N(\bar{\omega}_i)$ on a dense grid of potential DOAs $\{\bar{\omega}_i\}_{i=1}^d$ is considered, and the goal is to find sparse signals $\mathbf{Q} = [\mathbf{q}(1)\ \mathbf{q}(2) \cdots \mathbf{q}(K)]$ that well represent the K snapshots of array output

$$\mathbf{X} = \underbrace{[\mathbf{a}_N(\bar{\omega}_1)\ \mathbf{a}_N(\bar{\omega}_2)\ \cdots\ \mathbf{a}_N(\bar{\omega}_d)]}_{\text{dictionary } \mathbf{D}} \mathbf{Q} + \mathbf{E}. \tag{6.51}$$

Here $\mathbf{X} = [\mathbf{x}(1)\ \mathbf{x}(2) \cdots \mathbf{x}(K)]$ and $\mathbf{E} = [\mathbf{e}(1)\ \mathbf{e}(2) \cdots\ \mathbf{e}(K)]$ are the K snapshots of array outputs and noise, respectively. The number of grid points d is typically much larger than D, the number of sources. If D is not too large [38] (the exact bound depending on N and K), there exists a solution to (6.51) such that $\mathbf{q}(k)$ has a common sparse pattern across all snapshots. The locations of nonzero entries of $\mathbf{q}(k)$ reveal the DOAs, and the values represent the source amplitudes. FOCUSS is an iterative algorithm for solving such a problem. It is initialized with

$$\mathbf{Q}_0 = \mathbf{D}^+\mathbf{X}. \tag{6.52}$$

Then we iterate

$$\mathbf{W}_{n+1} = \text{diag}(\|\mathbf{q}_{n,1}\|^{1-p/2}, \ldots, \|\mathbf{q}_{n,d}\|^{1-p/2}) \tag{6.53}$$

$$\mathbf{Q}_{n+1} = \mathbf{W}_{n+1}(\mathbf{D}\mathbf{W}_{n+1})^+\mathbf{X} \tag{6.54}$$

for $n = 0, 1, \ldots,$ until convergence. Here $\mathbf{q}_{n,i}^T$ is the ith row of $\mathbf{Q}_n$, and $p \in [0, 1]$ is a parameter to be chosen so that the solution aims to minimize the ℓ_p diversity measure [38], i.e., the ℓ_p pseudo-norm of the vector whose elements are the ℓ_2 norms of the rows of $\mathbf{Q}$.

Now we show how to do FOCUSS in our distributed setting, where

$$\mathbf{X} = [\mathbf{X}_0^T \; \mathbf{X}_1^T \cdots \mathbf{X}_{P-1}^T]^T \tag{6.55}$$

with $\mathbf{X}_p \in \mathbb{C}^{Q \times K}$ stored locally at node p. Let $\mathbf{B}_0 = \mathbf{D}^+ = [\mathbf{B}_{0,0} \cdots \mathbf{B}_{0,P-1}]$ with each $\mathbf{B}_{0,p} \in \mathbb{C}^{d \times Q}$. Since the dictionary $\mathbf{D}$ is known to all nodes, $\mathbf{B}_{0,p}$ can also be obtained at node p. Hence, (6.52) can be computed from

$$\mathbf{Q}_0 = P \cdot \mathrm{AC}_p(\mathbf{B}_{0,p}\mathbf{X}_p). \tag{6.56}$$

Let $\mathbf{B}_{n+1} = \mathbf{W}_{n+1}(\mathbf{D}\mathbf{W}_{n+1})^+ = [\mathbf{B}_{n+1,0} \cdots \mathbf{B}_{n+1,P-1}]$ with each $\mathbf{B}_{n+1,p} \in \mathbb{C}^{d \times Q}$. Then (6.54) can be computed from

$$\mathbf{Q}_{n+1} = P \cdot \mathrm{AC}_p(\mathbf{B}_{n+1,p}\mathbf{X}_p). \tag{6.57}$$

Since each $\mathbf{Q}_{n+1}$ is the result of AC, it is known to all nodes, so (6.53) can also be formed at each node. Thus, each $\mathbf{B}_{n+1,p}$ can indeed be obtained at node p. The fact that only linear operations on array outputs are involved in the iterations makes FOCUSS readily realizable using AC. As in distributed ESPRIT, computing $(\mathbf{D}\mathbf{W}_{n+1})^+$ at all the nodes leads to an increase in the total computational complexity since they are performing the same operations. An alternative would be for one node to do the computations and then send the results to the other nodes. This is a tradeoff between computation and communication. The communication cost per edge is $O(dKI_{\mathrm{ac}}I_{\mathrm{fo}})$, where I_{fo} is the number of FOCUSS iterations. A numerical example is given in Sec. 6.4.4.

Distributed FOCUSS evidently has broader applications than DOA estimation. It can be used for any problem that requires a sparse solution to an underdetermined system [38].

6.3.5 Communication, Computation, and Storage

We now compare the communication costs of the proposed distributed algorithms and centralized algorithms. Using the results in Sec. 6.2.2 for distributed power method and in this section for the DOA estimation methods, the communication costs per edge for the various distributed algorithms are summarized in Table 6.1.

Table 6.1: Communication costs of distributed and centralized DOA estimation methods

Algorithm	Communication cost	Typical numbers
Distributed MUSIC	$O(DKI_{\text{ac}}I_{\text{pm}} + DGI_{\text{ac}})$	3000
Distributed root-MUSIC	$O(DKI_{\text{ac}}I_{\text{pm}} + z_{N-1}DI_{\text{ac}}I_{\text{ab}})$	8940
Distributed TLS-ESPRIT	$O(DKI_{\text{ac}}I_{\text{pm}} + D^2I_{\text{ac}})$	1812
Distributed FOCUSS	$O(dKI_{\text{ac}}I_{\text{fo}})$	9×10^5
All centralized methods	$O(KN)$	12000

For a ULA, $z_{N-1} = N - 1$. Typically, the number of DOAs D is much smaller than either of the number of grid points G used for ω in MUSIC, number of sensors N, number of snapshots K, and number of grid points d used for ω in FOCUSS. How we choose the numbers of iterations I_{pm}, I_{ab}, and I_{fo} depends on the particular DOA problem settings, but empirically we need a relatively small I_{pm}. Also, according to Sec. 6.2.1, $I_{\text{ac}} < P$, and the exact number of I_{ac} depends on the network structure. Hence, the communication cost of TLS-ESPRIT is typically the smallest. That is, TLS-ESPRIT is the DOA estimation method particularly suitable for distributed implementations. On the other hand, to implement any centralized DOA estimation method, each node always sends the raw data to a fusion center to do centralized computation. Thus, the communication cost of any centralized DOA estimation method is the same. Given K snapshots of the array output $\mathbf{x}$, the total communication cost across the sensor network is $O(KNT)$, where T is the average distance between each node and the fusion center. The distance between two nodes is the number of edges in a shortest path connecting them. In other words, we have assumed that an efficient routing protocol is used so that each node can send its raw data to the fusion center via a shortest path. In this case, the average communication cost per edge for any centralized algorithm is $O(KNT/|\mathcal{E}|)$. Since we assume the network is a connected graph, $O(1) \leq T \leq O(P)$ and $O(P) \leq |\mathcal{E}| \leq O(P^2)$. The possible range of the average communication cost per edge is thus $O(KNP^r)$, where $-2 \leq r \leq 0$. Hence, whether the communication cost of a distributed algorithm is smaller than the average communication cost of a centralized one depends on the values of the parameters and on the connectivity of the network. Distributed algorithms are typically more advantageous given larger arrays or more snapshots, especially for distributed TLS-ESPRIT.

Besides the average communication cost per edge, another important metric is the

maximum communication cost among all the edges because it can determine the existence of a communication bottleneck. Unless the network is almost fully connected, the maximum communication cost for any centralized algorithm is typically close to $O(KN)$, due to an edge nearby the fusion center. By contrast, the maximum communication costs of the distributed algorithms follow the same expressions in Table 6.1 because each edge has the same communication cost. Therefore, distributed algorithms are more likely to be better than centralized ones in terms of maximum communication costs, which can determine a communication bottleneck. For comparison, the maximum communication cost of centralized methods is also listed in Table 6.1. Also, we give an example of typical numbers by setting $P = 6, Q = 20, D = 2, K = 100, d = G = 200, I_{\text{ac}} = I_{\text{pm}} = 3, I_{\text{ab}} = 10, I_{\text{fo}} = 15$, and $z_{N-1} = 119$. The numbers of iterations are set according to empirical results. We see that distributed TLS-ESPRIT has the smallest communication cost. All the distributed algorithms have smaller costs than centralized ones, except distributed FOCUSS. The large communication cost of FOCUSS is mainly due to the factor dK in the expression. Unless there is some other prior information (e.g., DOAs known to be in some range so that a smaller d can be used), distributed FOCUSS would not be advantageous in terms of communication cost. The numbers in Table 6.1 are typical when we assume the number of sources $D \ll N$. The small-D assumption is also made in the literature [115] to show the advantage in communication costs for distributed methods. However, even when this assumption does not hold so that distributed methods do not lower communication costs, other issues can still prevent the use of centralized methods in the first place, as explained in the following.

In the proposed distributed algorithms, the local computation cost at a node depends on the amount of local array data and the number of its neighbors, but not directly on the size of the network. Take TLS-ESPRIT as an example. The computational complexity of (6.49) locally at node i is $O(D^2Q + D^2 I_{\text{ac}}|\mathcal{N}_i|)$, where $|\mathcal{N}_i|$ is the number of its neighbors. Besides, each node only has to store its local array data and AC weights for each edge between each neighbor and itself. It is unlike centralized systems where the fusion center has to store the array data collected from all the nodes to do centralized computation. Therefore, the proposed distributed systems should scale better than centralized systems with fusion centers, as the sensor network expands. The requirement of local storage space and computation power that grow with the network size makes it very challenging to implement centralized algorithms for large networks.

In summary, we have the following key observations.

1. TLS-ESPRIT typically has the smallest communication cost among the proposed distributed algorithms.

2. There are two ways to measure communication cost: average communication cost per edge and maximum communication cost among all the edges. The maximum communication cost can determine a communication bottleneck, so it is more important for the system to be scalable. The two measures do not make a difference for distributed algorithms, but centralized algorithms have larger maximum communication cost than average communication cost. Hence, distributed algorithms are more likely to be better than centralized ones in terms of maximum communication costs.

3. In the proposed distributed algorithms, the local computation and storage at a node depend on the amount of local array data and the number of its neighbors, but not directly on the network size. Thus, these distributed systems should scale better than centralized systems.

6.4 Distributed Convolutional Beamspace

In this section, we propose distributed algorithms for the *convolutional beamspace (CBS)* method introduced in Sec. 2.2, and for its variant *Capon-CBS* presented in Sec. 2.5. Given an N-sensor array with output $\mathbf{x} \in \mathbb{C}^N$, the idea of beamspace is to compute a transformation $\mathbf{y} = \mathbf{T}\mathbf{x} \in \mathbb{C}^B$, where $B < N$, and estimate DOA using $\mathbf{y}$. (For clarity, DOA estimation using $\mathbf{x}$ directly is called *element-space* method.) With a properly designed $\mathbf{T}$, the DOAs falling outside a chosen subband are attenuated, and we focus on estimating only the DOAs in the subband. A bank of transformations $\{\mathbf{T}_i\}$ can be operated in parallel to cover all subbands. One major advantage of beamspace processing is the lowered complexity due to dimensionality reduction ($B < N$). Beamspace methods also tend to have smaller SNR threshold for resolution of closely spaced sources [8, 16, 17] and smaller bias [2, 23].

Classical beamspace methods compromise the Vandermonde structure in the output of a uniform linear array (ULA), so elaborate steps have to be taken to apply root-MUSIC [2] or ESPRIT [18]. By contrast, CBS methods allow root-MUSIC and ESPRIT to be performed directly for ULAs without additional preparation since the Vandermonde structure is preserved. This is achieved by convolving the ULA output with an FIR filter $H(z)$. In traditional CBS (Sec. 2.2), the filter $H(z)$ was

predefined as a standard lowpass filter, such as the Parks-McClellan filter [63]. But the filter was not designed by taking input statistics into account. In Capon-CBS (Sec. 2.5), the filter is designed based on data using the idea of Capon beamforming, so it should do a better job of suppressing the sources falling in the stopband, as they are treated as interference in the Capon method.

The CBS methods have two stages. Stage 1 is the convolutional beamspace (CBS) stage, where the ULA output is convolved with an FIR filter $H(z)$. Stage 2 is the DOA estimation stage. The filter in Stage 1 needs to have a flat passband, and it is a standard lowpass filter in traditional CBS or the *robust Capon beamspace filter* in Capon-CBS. The robust Capon beamspace filter is based on the robust version [83] of Capon beamformer, as proposed in Sec. 2.5 to get a flat passband. Distributed implementation of the CBS filter, i.e., distributed convolution, is introduced in Sec. 6.4.1. Distributed design of the robust Capon beamspace filter is presented in Sec. 6.4.2. The DOA estimation algorithms proposed in Sec. 6.3 can be used in Stage 2 to form an overall distributed system together with the Stage 1 methods in Secs. 6.4.1 and 6.4.2. This distributed DOA estimation in Stage 2 is presented in Sec. 6.4.3. Numerical examples are given in Sec. 6.4.4.

6.4.1 Distributed Implementation of the CBS Filter

Convolutional beamspace (CBS) methods as in Chapter 2 are used for ULAs. That is, we assume the sensor locations $z_i = i$ in (6.2). We convolve the N-sensor ULA output sequence $x(n), 0 \le n \le N-1$ with an FIR filter

$$H(z) = \sum_{n=0}^{L-1} h(n) z^{-n} \tag{6.58}$$

with length $L < N$ and retain *steady state* output samples:

$$\begin{aligned} \mathbf{y} &\triangleq [y(L-1)\ y(L) \cdots y(N-1)]^T \\ &= \mathbf{H}[x(0)\ x(1) \cdots x(N-1)]^T = \mathbf{H}\mathbf{x}, \end{aligned} \tag{6.59}$$

where $\mathbf{H}$ is a $(N-L+1) \times N$ banded Toeplitz matrix with first row $[h(L-1)\ h(L-2) \cdots h(0)\ 0\ \cdots 0]$ and first column $[h(L-1)\ 0\ \cdots 0]^T$. As in Sec. 2.2, one can show that

$$\mathbf{y} = \mathbf{A}_L \mathbf{d} + \mathbf{H}\mathbf{e}, \tag{6.60}$$

where $\mathbf{A}_L$ is a Vandermonde matrix obtained from $\mathbf{A}$ (defined in (6.1), (6.2)), by keeping the first $N-L+1$ rows, and $\mathbf{d}$ has elements $d_k = c_k e^{j(L-1)\omega_k} H(e^{j\omega_k})$. The

arriving signals with DOAs ω_k are thus filtered by the response $H(e^{j\omega})$. The filter is designed as a standard lowpass filter such as the Parks-McClellan filter [63] in traditional CBS (Sec. 2.2), and as a Capon beamspace filter in Capon-CBS (Sec. 2.5). The design of the Capon beamspace filter will be discussed in Sec. 6.4.2. Assuming signals in the stopband are well attenuated so that $\mathbf{y}$ contains only those DOAs that fall in the passband of $H(e^{j\omega})$, we have $\mathbf{y} \approx \mathbf{A}_{L,0}\bar{\mathbf{d}}_0 + \mathbf{He}$. Here $\mathbf{A}_{L,0}$ has D_0 columns of $\mathbf{A}_L$ corresponding to the D_0 sources that fall in the passband, and $\bar{\mathbf{d}}_0$ has the corresponding D_0 rows of $\mathbf{d}$.

To achieve complexity reduction in CBS, we *decimate* $y(n)$ with a uniform downsampler. If the passband of $H(z)$ has width $\approx 2\pi/M$, we can decimate $y(n)$ by the integer M. Let $v(n) = y(n + L - 1)$ so that $\mathbf{y} = [v(0)\ v(1)\ \cdots\ v(N - L)]^T$. Then we take the *polyphase components* of $v(n)$ [66]. That is, we define

$$\mathbf{v}_l = [v(l)\ v(l + M)\ \cdots\ v(l + (J - 1)M)\ ]^T, \tag{6.61}$$

for $l = 0, \ldots, M - 1$, where $J = \lfloor (N - L + 1)/M \rfloor$. As in Sec. 2.2, it can then be verified that [23]

$$\mathbf{v}_l = \mathbf{A}_{\text{dec}}\mathbf{d}_l + \mathbf{D}_l\mathbf{He}, \tag{6.62}$$

where

$$\mathbf{A}_{\text{dec}} = [\mathbf{a}_J(M\omega_1)\ \mathbf{a}_J(M\omega_2)\ \cdots\ \mathbf{a}_J(M\omega_D)]\,, \tag{6.63}$$

$\mathbf{d}_l$ has elements $d_{l,k} = c_k e^{j(L-1+l)\omega_k} H(e^{j\omega_k})$ and $\mathbf{D}_l = [\boldsymbol{\delta}_l\ \boldsymbol{\delta}_{l+M}\ \cdots\ \boldsymbol{\delta}_{l+(J-1)M}]^T$ is a decimation matrix, with $\boldsymbol{\delta}_l$ being the lth standard basis vector for the $(N - L + 1)$-dimensional space. The noise term $\mathbf{D}_l\mathbf{He}$ can be whitened by making $\mathbf{D}_l\mathbf{H}\mathbf{H}^H\mathbf{D}_l^H = \mathbf{I}$. This is achieved by choosing $H(z)$ as a spectral factor of a Nyquist(M) filter as shown in Sec. 2.2.4. Since $\mathbf{v}_l$ is represented in terms of the Vandermonde matrix $\mathbf{A}_{\text{dec}}$ just like the original ULA output $\mathbf{x}$, standard DOA estimation methods such as root-MUSIC or ESPRIT can be directly applied. Details will be shown in Sec. 6.4.3. The columns of $\mathbf{A}_{\text{dec}}$ are $\mathbf{a}_J(M\omega_k)$ rather than $\mathbf{a}_J(\omega_k)$, so ω_k can be determined only up to $M\omega_k \bmod 2\pi$, creating *ambiguity*. But since ω_k are known to be in the passband of $H(e^{j\omega})$ which has width $2\pi/M$, the ambiguity can be resolved.

Now we show how to implement the convolution (6.59) in the CBS stage in our distributed setting. Once we can do this, there is no difficulty in doing decimation. We make the following assumptions to simplify our distributed algorithm.

- *Assumption 1*: The CBS filter has order $L-1 \leq Q$. Since CBS is typically used for large arrays (large N), this assumption is reasonable.
- *Assumption 2*: There is an edge between each pair of adjacent nodes $p-1$ and p.

The method described in the following applies to CBS using any kinds of filters, including Capon-CBS. With $\mathbf{y}$ and $\mathbf{H}$ defined as in (6.59), let $\mathbf{y}_0 \in \mathbb{C}^{Q-L+1}$ and $\mathbf{y}_p \in \mathbb{C}^Q$, $p = 1 \ldots, P-1$ so that $\mathbf{y} = [\mathbf{y}_0^T\ \mathbf{y}_1^T \cdots \mathbf{y}_{P-1}^T]^T$. Since $L-1 \leq Q$, we have

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_{0,0} & \mathbf{0} & & \cdots & \mathbf{0} \\ \mathbf{H}_{1,0} & \mathbf{H}_{1,1} & \mathbf{0} & \cdots & \vdots \\ \mathbf{0} & \mathbf{H}_{2,1} & \mathbf{H}_{2,2} & \ddots & \\ \vdots & \ddots & \ddots & \ddots & \\ & & & & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{H}_{P-1,P-2} & \mathbf{H}_{P-1,P-1} \end{bmatrix}. \tag{6.64}$$

Here $\mathbf{H}_{0,0}$ is a $(Q-L+1) \times Q$ Toeplitz matrix with first row $[h(L-1)\ h(L-2) \cdots h(0)\ 0 \cdots 0]$ and first column $[h(L-1)\ 0 \cdots 0]^T$. Besides,

$$\mathbf{H}_{p,p-1} = \left[\begin{array}{c|cccc} & h(L-1) & h(L-2) & \cdots & h(1) \\ & 0 & h(L-1) & \cdots & h(2) \\ \mathbf{0} & \vdots & \ddots & \ddots & \vdots \\ & 0 & \cdots & 0 & h(L-1) \\ \hline \mathbf{0} & \multicolumn{4}{c}{\mathbf{0}} \end{array}\right]$$

are $Q \times Q$ Toeplitz matrices, and $\mathbf{H}_{p,p}$ are $Q \times Q$ Toeplitz matrices with first row $[h(0)\ 0 \cdots 0]$ and first column $[h(0)\ h(1) \cdots h(L-1)\ 0 \cdots 0]^T$, for $p = 1, \ldots, P-1$. Thus, we have $\mathbf{y}_0 = \mathbf{H}_{0,0}\mathbf{x}_0$, which can be computed locally at node 0. Besides, since $L-1 \leq Q$, we have

$$\mathbf{y}_p = \mathbf{H}_{p,p-1}\mathbf{x}_{p-1} + \mathbf{H}_{p,p}\mathbf{x}_p, \tag{6.65}$$

which can be computed locally at node p once node $p-1$ sends the last $L-1$ entries of $\mathbf{x}_{p-1}$, i.e., $x(pQ-L+1), \ldots, x(pQ-1)$, to node p, for $p = 1, \ldots, P-1$. This can be done without using AC because we assume that there is an edge between nodes $p-1$ and p. The communication cost per edge is $O(KL)$, where K is the number of snapshots.

If *no assumption* is made on the filter length L, then in the most general case, we can compute

$$y(i+L-1) = \mathbf{h}_i^T \mathbf{x} = \mathrm{AC}_p(\mathbf{h}_{i,p}^T \mathbf{x}_p), \tag{6.66}$$

where $\mathbf{h}_i^T$ is the ith row of $\mathbf{H}$, and $\mathbf{h}_{i,p} \in \mathbb{C}^Q$ is the subvector of $\mathbf{h}_i$ corresponding to node p, for $i = 0, \ldots, N-L$. The communication cost per edge is $O(K(N-L)I_{\mathrm{ac}})$.

6.4.2 Distributed Design of the Capon-CBS Filter

In Sec. 2.2, the CBS filter is a standard lowpass filter, such as the Parks-McClellan filter. In this case we simply assume all nodes have the filter coefficients available. It is beneficial to replace such a standard filter with a data-dependent filter based on the idea of Capon beamforming, as explained in Sec. 2.5. Among all the unknown DOAs, those falling in the stopband are treated as interferers (or jammers). We call this the Capon-CBS filter. This is nothing but a *sliding Capon beamspace generator*. This will provide optimal attenuation of the stopband signals.

Note that the output of this Capon-CBS filter will be used to estimate the DOAs in its passband. Thus, unlike in traditional beamforming applications, this Capon-CBS filter should be designed to have a *flat passband*. Since the traditional Capon filter does not yield a flat passband around ω_0 although $H(e^{j\omega_0}) = 1$, we will use the robust Capon beamforming method reported in [83]. This is a method to ensure that $H(e^{j\omega}) \approx 1$ in a specified passband range, while at the same time optimally rejecting the interferers in the stopband. We call this CBS filter the *robust Capon beamspace filter*. For purposes of computation [83], this filter will be expressed explicitly in terms of real and imaginary components: $[\mathrm{Re}\{\mathbf{h}^T\} \;\; \mathrm{Im}\{\mathbf{h}^T\}]^T \triangleq \widetilde{\mathbf{h}}$ where $\mathbf{h} = [h^*(L-1) \;\; h^*(L-2) \;\; \cdots \;\; h^*(0)]^T$. The CBS filter coefficients $\widetilde{\mathbf{h}}$ are designed as a robust Capon beamspace generator, which is the solution to

$$\begin{aligned} \min_{\widetilde{\mathbf{h}}} \quad & \widetilde{\mathbf{h}}^T \widetilde{\mathbf{R}}_L \widetilde{\mathbf{h}} \\ \text{subject to} \quad & \widetilde{\mathbf{h}}^T \mathbf{a} \geq 1 \;\; \forall \mathbf{a} \in \mathcal{E}, \end{aligned} \tag{6.67}$$

where

$$\widetilde{\mathbf{R}}_L = \frac{1}{N-L+1} \sum_{i=0}^{N-L} \mathrm{E}[\widetilde{\mathbf{x}}_{L,i} \widetilde{\mathbf{x}}_{L,i}^T] \tag{6.68}$$

with $\widetilde{\mathbf{x}}_{L,i} = [\mathrm{Re}\{\mathbf{x}_{L,i}^T\} \; \mathrm{Im}\{\mathbf{x}_{L,i}^T\}]^T$ and

$$\mathbf{x}_{L,i} = [x(i) \; x(i+1) \; \cdots x(i+L-1)]^T. \tag{6.69}$$

Here, $\mathcal{E}$ is an $2L$-dimensional ellipsoid that covers the range of values of $\widetilde{\mathbf{a}}_L(\omega) = [\mathrm{Re}\{\mathbf{a}_L(\omega)^T\} \;\; \mathrm{Im}\{\mathbf{a}_L(\omega)^T\}]^T$ in the passband. The low-complexity algorithm given in [83] can be used to solve the problem.

Now we show how to compute $\widetilde{\mathbf{h}}$ in our distributed setting. We make the same *Assumptions 1* and *2* as in Sec. 6.4.1 to simplify the algorithm. Then, with K snapshots, we estimate

$$\widehat{\widetilde{\mathbf{R}}}_L = \frac{1}{K(N-L+1)} \sum_{k=1}^{K} \sum_{i=0}^{N-L} \widetilde{\mathbf{x}}_{L,i}[k] \widetilde{\mathbf{x}}_{L,i}^T[k] \tag{6.70}$$

$$= \frac{P}{N-L+1} \mathrm{AC}_p(\widetilde{\mathbf{R}}_{L,p}), \tag{6.71}$$

where

$$\widetilde{\mathbf{R}}_{L,0} = \frac{1}{K} \sum_{k=1}^{K} \sum_{i=0}^{Q-L} \widetilde{\mathbf{x}}_{L,i}[k] \widetilde{\mathbf{x}}_{L,i}^T[k] \tag{6.72}$$

and

$$\widetilde{\mathbf{R}}_{L,p} = \frac{1}{K} \sum_{k=1}^{K} \sum_{i=pQ-L+1}^{(p+1)Q-L} \widetilde{\mathbf{x}}_{L,i}[k] \widetilde{\mathbf{x}}_{L,i}^T[k] \tag{6.73}$$

for $p = 1, \ldots, P-1$. Considering (6.69), we see that (6.72) can be computed locally at node 0 since it involves only $x(0), \ldots, x(Q-1)$. To compute (6.73), since it involves $x(pQ-L+1), \ldots, x((p+1)Q-1)$, node $p-1$ has to send the $L-1$ samples $x(pQ-L+1), \ldots, x(pQ-1)$ to node p. Then (6.73) can be computed at node p. Note that these samples to be sent are exactly the same as those we need to send for the distributed implementation of the CBS filter. In other words, there is no additional communication cost for the "convolution" part of CBS if we already did the "Capon design" of Capon-CBS. After each node obtains $\widehat{\widetilde{\mathbf{R}}}_L$ via AC in (6.71), the problem (6.67) can be solved locally at each node using the method in [83]. The total communication cost per edge for the design of the Capon-CBS filter is $O(KL + L^2 I_{\mathrm{ac}})$, where I_{ac} is the number of AC iterations.

6.4.3 Distributed DOA Estimation in Stage 2

For subspace-based methods such as MUSIC, root-MUSIC, and ESPRIT, we estimate the average of the $J \times J$ covariances of all the polyphase components

$$\widehat{\mathbf{R}}_{\mathrm{ave}} = \frac{1}{KM} \sum_{k=1}^{K} \sum_{l=0}^{M-1} \mathbf{v}_l[k] \mathbf{v}_l^H[k], \tag{6.74}$$

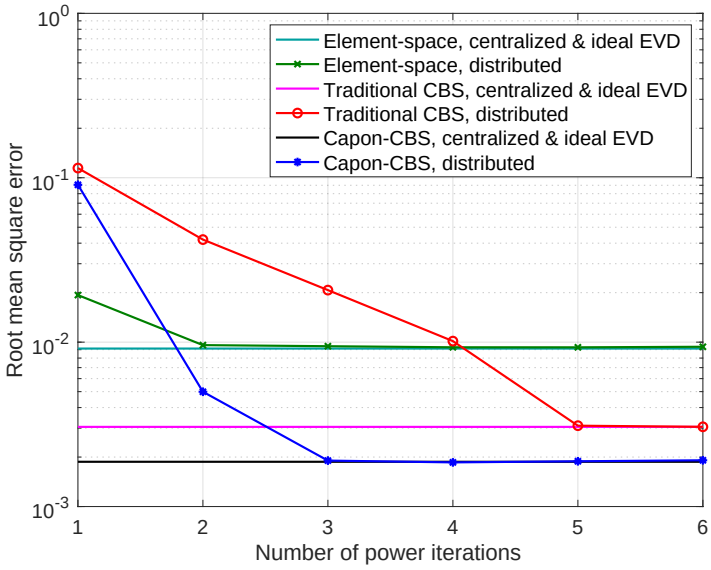
where the polyphase components $\mathbf{v}_l$ are defined in (6.61). Then we only have to compute the signal and noise eigenvectors of (6.74), and the remaining steps just follow Secs. 6.3.2 and 6.3.3. Note that (6.74) is similar to (6.3). The elements of each vector $\mathbf{v}_l$ are distributed among the P nodes in exactly the same way that the elements of $\mathbf{x}$ were distributed in (6.3). The eigenvectors of $\widehat{\mathbf{R}}_{\text{ave}}$ can therefore be computed exactly as we computed eigenvectors of $\widehat{\mathbf{R}}_{\mathbf{xx}}$ in Sec. 6.2.2. From these the DOAs can be estimated unambiguously as described in Sec. 2.2. The summation over l in (6.74) leads to an increase in communication cost compared to element-space, but CBS offers a smaller computational complexity. It is a tradeoff between computation and communication.

FOCUSS introduced in Sec. 6.3.4 can also be used to estimate the DOAs in Stage 2. How we use FOCUSS with CBS is similar to the formulation of the Lasso problem with CBS, described in Sec. 2.4. We just replace $\mathbf{D}$ and $\mathbf{X}$ in (6.52) and (6.54) by $\mathbf{D}_{L,0} = [\mathbf{a}_J(M\bar{\omega}_1)\ \mathbf{a}_J(M\bar{\omega}_2)\cdots\mathbf{a}_J(M\bar{\omega}_{d_0})]$ and $\mathbf{V}_0 = \mathbf{D}_0\mathbf{H}\mathbf{X}$, respectively. Here $\mathbf{D}_0$ is the decimation matrix as in (6.62), and $\{\bar{\omega}_1, \ldots, \bar{\omega}_{d_0}\}$ is the grid of frequencies within the passband of $H(e^{j\omega})$. If the grid is uniform, $d_0 \approx d/M$. Note that since the passband width is $2\pi/M$, the arguments $M\bar{\omega}_k$ span a range of 2π. The rows of $\mathbf{V}_0$ are distributed among the P nodes in the same way that the rows of $\mathbf{X}$ were distributed in (6.55). Hence, the FOCUSS iterations can be done using AC in a similar way. Significant reduction in computational complexity is obtained due to the smaller size of the dictionary $\mathbf{D}_{L,0}$. Moreover, the communication cost per edge is $O(d_0 K I_{\text{ac}} I_{\text{fo}})$, smaller than $O(dKI_{\text{ac}}I_{\text{fo}})$ for the element-space FOCUSS. Thus, among the distributed DOA estimation methods, FOCUSS gets more benefits when used with CBS.

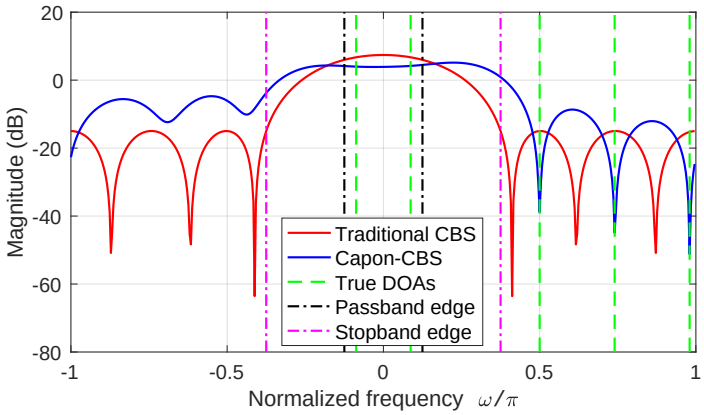
6.4.4 Simulations

We now present numerical examples to demonstrate the effectiveness of the proposed distributed CBS methods together with distributed DOA estimation algorithms in Sec. 6.3. We again consider the same network (6.27) with $P = 6$, and use the finite-time average consensus (AC) method with 3 iterations as explained in Sec. 6.3.1. The noise variance is $\sigma_e^2 = 1$.

Example 1: Distributed root-MUSIC with distributed CBS. We show the DOA estimation performance of distributed root-MUSIC when it is used together with distributed CBS (both traditional and Capon). The results are also compared to element-space and the centralized counterparts. Again, the distributed power method is used



(a)



(b)

Figure 6.4: Distributed root-MUSIC and centralized root-MUSIC for element-space, traditional CBS, and Capon-CBS. (a) RMSE of in-band DOA estimates. (b) Typical filter responses. The distributed and centralized algorithms result in the same filter for each system, so only one curve is plotted for each system.

to estimate eigenvectors of the array output covariance for distributed algorithms, whereas ideal EVD is assumed for centralized algorithms. Each node is a ULA with $Q = 8$ sensors and they together form a 48-sensor ULA. For CBS methods, the filter length is $L = 9$, and the decimation ratio is $M = 4$. The traditional CBS filter is designed to be a lowpass Parks-McClellan filter [63], with passband edge $\pi/2M$ and stopband edge $3\pi/2M$. The Capon-CBS filter is the solution to Problem (6.67) with $\mathcal{E}$ designed as in Sec. 2.5 using $r = 10$ equally spaced samples of the

array response in the passband. Note that the two assumptions $L-1 \leq Q$ and that there is an edge between each pair of adjacent nodes $p-1$ and p are satisfied, so the distributed CBS methods with lower communication costs can be used for both traditional and Capon CBS. There are 2 in-band sources (sources in the passband) with power -5 dB and DOAs $\theta = -5°, 5°$. There are 3 out-of-band sources (sources in the stopband) with power 15 dB and DOAs $\omega = 0.5\pi, 0.74\pi, 0.98\pi$. Each pair of the 5 sources has correlation coefficient 0.6, except that the two in-band sources are uncorrelated. We assume the number of DOAs is known. RMSEs of in-band DOA estimates for various number of power iterations are shown in Fig. 6.4(a). We use 100 snapshots and average 2000 Monte Carlo runs to get the plot. With only a few power iterations, all distributed algorithms converge to the centralized counterparts. Given enough number of power iterations, distributed traditional CBS and Capon-CBS perform significantly better than element-space. As shown in Sec. 2.5, in this harsh environment with powerful out-of-band sources correlated with in-band ones, centralized Capon-CBS has even smaller RMSE than centralized traditional CBS. This behavior is also true for distributed counterparts here. Fig. 6.4(b) also shows that the Capon-CBS filter indeed has a flat passband while suppressing the out-of-band sources. Capon-CBS filters keep a balance between attenuating interference and noise, which is the benefit of design based on data. The distributed and centralized algorithms result in the same filter for each of the two systems, traditional CBS and Capon-CBS, so only one curve is plotted for each system.

The results in Fig. 6.4 are obtained when we estimate eigenvectors of the array output covariance using the traditional distributed power method [7], where eigenvectors are updated sequentially (method 1). As mentioned in the end of Sec. 6.2.2, we can also update the eigenvectors in parallel (method 2). For the same example, we compare the two distributed power methods in Fig. 6.5. We see that method 2 requires more iterations than method 1 to compute each eigenvector. This can be expected because we do not use the finally converged results of the first $q-1$ eigenvectors to update the qth eigenvector in method 2. Thus, we choose to use method 1 in all other examples in this chapter.

Example 2: Distributed FOCUSS with distributed CBS. Next we show the DOA estimation performance of distributed FOCUSS when it is used together with distributed CBS. The results are also compared to element-space and the centralized counterparts. For FOCUSS, the number of DOAs and their locations are estimated together. Specifically, after getting the estimate $\hat{\mathbf{Q}}$ in (6.51), we plot the *FOCUSS*

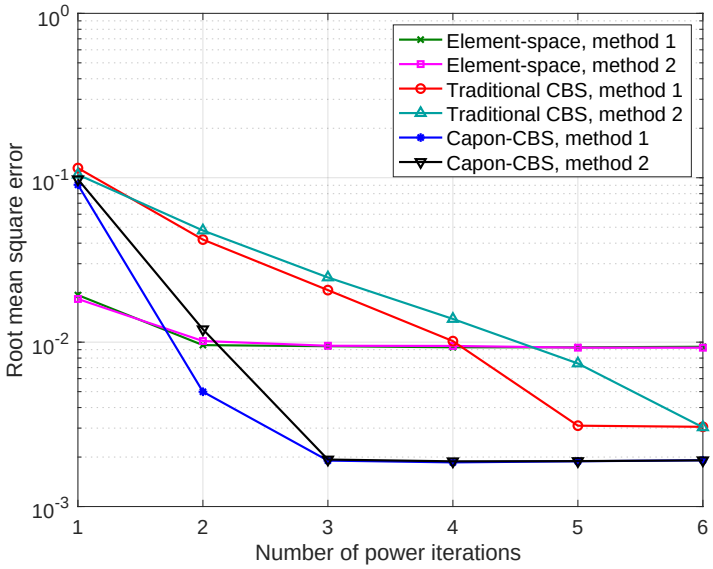
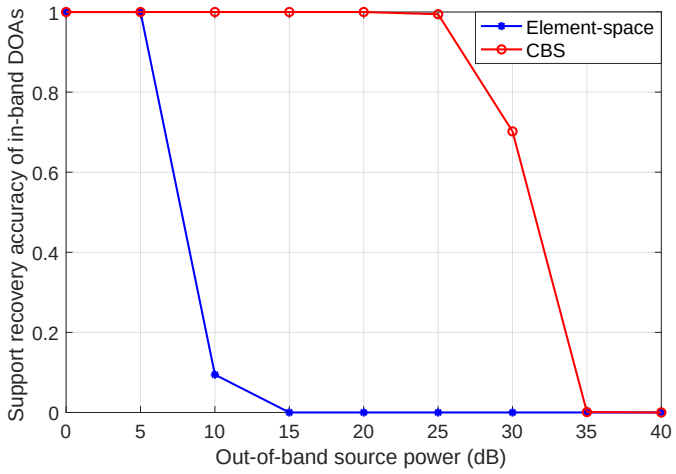
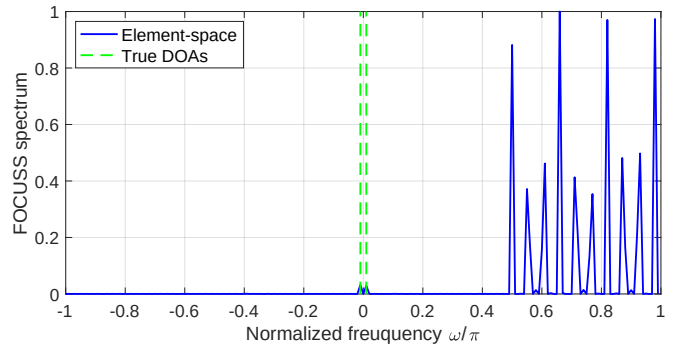


Figure 6.5: RMSE of in-band DOA estimates for distributed root-MUSIC using the two distributed power methods. Method 1 is the traditional power method [7], where eigenvectors are updated sequentially. Method 2 is a modified version where all eigenvectors are updated in parallel.

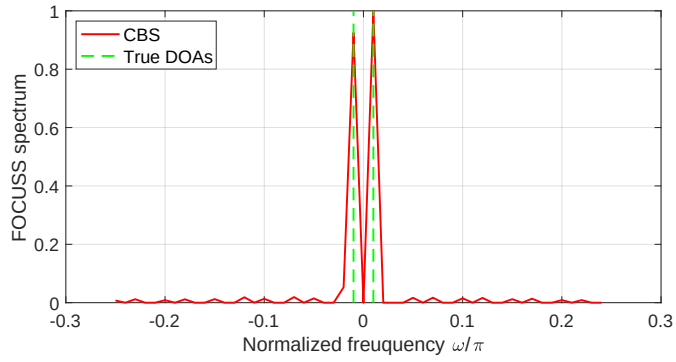
spectrum $P(\bar{\omega}_i) = \sum_n |\hat{\mathbf{Q}}_{in}|^2$ for $1 \le i \le d$. Then, we declare that there is a source at $\bar{\omega}_i$ if there is a peak (local maximum) that is larger than a particular threshold: $P(\bar{\omega}_i) \ge \epsilon_{\text{fo}}$. Here we use $\epsilon_{\text{fo}} = 0.1$ (with the spectrum normalized to have a maximum value 1). Each node is a ULA with $Q = 16$ sensors and they together form a 96-sensor ULA. For CBS, the decimation ratio is $M = 4$, and the filter is designed to be a lowpass Parks-McClellan filter [63] of length $L = 17$, with passband edge $\pi/2M$ and stopband edge $3\pi/2M$. A grid of $d = 200$ points uniform in ω is used for the potential DOAs. There are 2 in-band sources with power 1 and DOAs $\theta = -0.573°, 0.573°$. There are 10 out-of-band sources with identical power (which is varied in the experiment), with DOAs $\omega = 0.5\pi, 0.5\pi + \delta, 0.5\pi + 2\delta, \ldots, 0.98\pi$ with $\delta = 0.48\pi/9$. The in-band DOAs are exactly on the grid for simplicity. All sources are uncorrelated. Support recovery accuracy (SRC) of in-band DOAs for various out-of-band source powers are shown in Fig. 6.6(a). The SRC is defined as the probability of recovering the two and only two DOAs in the passband and without errors among all Monte Carlo runs. We use 100 snapshots and 2000 Monte Carlo runs to get the plot. CBS can tolerate more powerful out-of-band sources than element-space. More importantly, since distributed and centralized algorithms perform exactly the same for each of the two systems (i.e., element space and CBS),



(a)



(b)



(c)

Figure 6.6: Distributed FOCUSS and centralized FOCUSS for element-space and CBS. (a) Support recovery accuracy (SRC) of in-band DOAs. Both distributed and centralized algorithms have the same SRC for each system, so only one curve is plotted for each system. (b)-(c) Typical FOCUSS spectra of element-space and CBS when the out-of-band source power is 15 dB.

only one curve is plotted for each system. That is, the proposed distributed FOCUSS yields exactly the same solution as centralized FOCUSS. Typical FOCUSS spectra

of element-space and CBS for a Monte Carlo run are shown in Fig. 6.6(b)-(c). For CBS, only the passband part is plotted. The two in-band sources can clearly be resolved with CBS FOCUSS, but not with element-space FOCUSS. In the latter the out-of-band sources are too powerful to allow the in-band sources to be resolved.

6.5 Distributed Spatial Smoothing

In this section, we propose distributed algorithms for spatial smoothing [39]. Spatial smoothing is a technique used for DOA estimation when there are coherent or correlated sources. Consider a ULA output $\mathbf{x}$ as in (6.1), which has covariance

$$\mathbf{R_{xx}} = \mathbf{A}\mathbf{R_{cc}}\mathbf{A}^H + \sigma_e^2\mathbf{I}, \tag{6.75}$$

where $\mathbf{R_{cc}} = \mathrm{E}[\mathbf{c}\mathbf{c}^H]$. When there are coherent sources, the rank of $\mathbf{R_{cc}}$ and hence the rank of $\mathbf{A}\mathbf{R_{cc}}\mathbf{A}^H$ will be less than the number of sources D. Therefore, the signal subspace, i.e., the column space of $\mathbf{A}$, cannot be fully identified from the EVD of $\mathbf{R_{xx}}$. To overcome this, one computes the spatially smoothed covariance

$$\mathbf{R}_{\mathrm{ss}} = \frac{1}{L_{\mathrm{ss}}} \sum_{i=0}^{L_{\mathrm{ss}}-1} \mathrm{E}[\mathbf{x}_{\mathrm{ss},i}\mathbf{x}_{\mathrm{ss},i}^H], \tag{6.76}$$

where L_{ss} is a parameter and

$$\mathbf{x}_{\mathrm{ss},i} = [x(i)\ x(i+1)\ \cdots x(i+N-L_{\mathrm{ss}})]^T. \tag{6.77}$$

It can be shown [39] that the reduced rank due to coherent sources can be fully restored back to D based on this spatial smoothing if $L_{\mathrm{ss}} \geq D$. Thus, more accurate DOA estimates can be obtained by applying subspace-based methods to $\mathbf{R}_{\mathrm{ss}}$, for at most $N - L_{ss}$ sources [39]. With this method we can therefore identify $D \leq N/2$ sources. Empirically, spatial smoothing also helps when sources are correlated but not necessarily coherent [130].

We now show how to do spatial smoothing in our distributed setting. The task here is to estimate the eigenvectors of $\widehat{\mathbf{R}}_{\mathrm{ss}}$ instead of $\widehat{\mathbf{R}}_{\mathbf{xx}}$ obtained from K snapshots and then estimate DOAs using, e.g., root-MUSIC. One may notice that both distributed design of the Capon-CBS filter and spatial smoothing involve the average of subarray covariance matrices. More precisely, $\widetilde{\mathbf{R}}_L$ in (6.68) and $\mathbf{R}_{\mathrm{ss}}$ in (6.76) are exactly the same if $L = N - L_{\mathrm{ss}} + 1$ and $\widetilde{\mathbf{x}}_{L,i}$ is replaced by $\mathbf{x}_{L,i}$. So actually, the method for dealing with $\widetilde{\mathbf{R}}_L$ can be used to deal with $\mathbf{R}_{\mathrm{ss}}$ if the assumptions (a) $(N - L_{\mathrm{ss}} + 1) - 1 \leq Q$ (b) that there is an edge between each pair of adjacent nodes $p-1$ and p, are satisfied.

Algorithm 4 Distributed power iteration of spatial smoothing

1: **for** $i = 0$ to $L_{\text{ss}} - 1$ **do**
2: **for** $k = 1$ to K **do**
3: $t_i[k] = P \cdot \text{AC}_p(\mathbf{x}_{\text{ss},i,p}^H[k]\mathbf{e}_{1,i,p}(n))$
4: **end for**
5: **for** $p = 0$ to $P - 1$ **do**
6: $\mathbf{q}_{i,p} = \frac{1}{K}\sum_{k=1}^{K} \mathbf{x}_{\text{ss},i,p}[k]t_i[k]$
7: **end for**
8: **end for**
9: **for** $m = 0$ to $L - 1$ **do**
10: **for** $p = 0$ to $P - 1$ **do**
11: $s_{m,p} = \frac{1}{L_{\text{ss}}}\sum_{i:pQ\le i+m<(p+1)Q}[\mathbf{q}_i]_m$
12: **end for**
13: $[\mathbf{e}_1(n+1)]_m = P \cdot \text{AC}_p(s_{m,p})$
14: **end for**
15: **return** $\mathbf{e}_1(n+1)$

In this case, $\widehat{\mathbf{R}}_{\text{ss}}$ can be computed via average consensus in a manner similar to (6.71). Then the eigenvectors of $\widehat{\mathbf{R}}_{\text{ss}}$ can be computed locally at each node.

Case of unrestricted L_{ss}. If no assumption is made on L_{ss}, then we can modify the distributed power method in Sec. 6.2.2 to realize spatial smoothing. If we can show how to compute the power iterations

$$\mathbf{e}_1(n+1) = \widehat{\mathbf{R}}_{\text{ss}}\mathbf{e}_1(n) \tag{6.78}$$

for the first eigenvector $\mathbf{e}_1$, then the remaining steps follow the same way as in Sec. 6.2.2. The main idea is to clarify what computations can be done locally at each node p. The detailed procedure is presented in Algorithm 4 shown in the table. The vector $\mathbf{x}_{\text{ss},i,p}$ is the subvector of $\mathbf{x}_{\text{ss},i} \in \mathbb{C}^{N-L_{\text{ss}}+1}$ stored at node p, and its length can vary from 0 to $N - L_{\text{ss}} + 1$ for different i or p. Then, the vector $\mathbf{e}_{1,i,p}(n)$ is the subvector of $\mathbf{e}_1(n)$ with entries corresponding to $\mathbf{x}_{\text{ss},i,p}$. The inner products $\mathbf{x}_{\text{ss},i,p}^H[k]\mathbf{e}_{1,i,p}$ are computed locally at node p in parallel. The computations in Lines 6 and 11 are also done locally at each node in parallel. Finally, $\mathbf{q}_i = [\mathbf{q}_{i,0}^T \cdots \mathbf{q}_{i,P-1}^T]^T$, and the notations $[\mathbf{q}_i]_m$ and $[\mathbf{e}_1(n)]_m$ mean the mth entry of $\mathbf{q}_i$ and $\mathbf{e}_1(n)$, respectively. AC appears in Algorithm 4 in two places, Lines 3 and 13. Line 3 is the dominant operation for communication, so the total communication cost per edge for estimating D eigenvectors is $O(DKL_{\text{ss}}I_{\text{ac}}I_{\text{pm}})$. This is L_{ss} times the cost of the basic distributed power method in Sec. 6.2.2. This is a tradeoff for getting better DOA estimates when there are coherent or correlated sources and when $(N - L_{\text{ss}} + 1) - 1 > Q$.

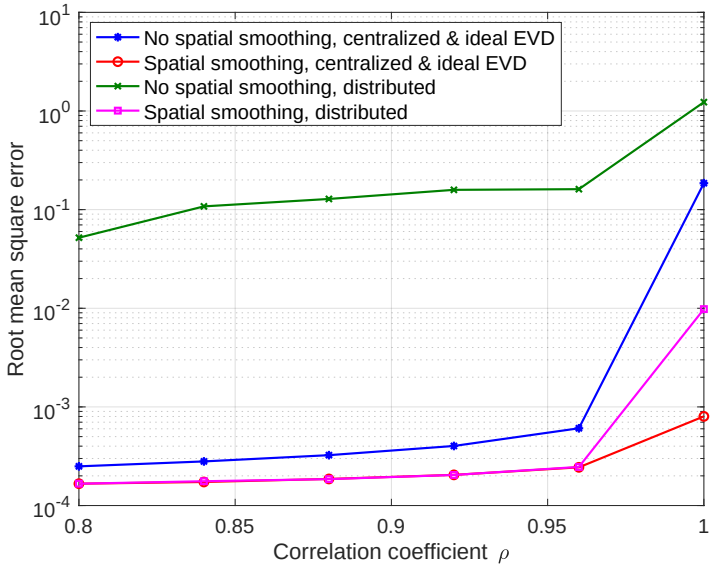


Figure 6.7: RMSE of DOA estimates with and without spatial smoothing using distributed root-MUSIC and centralized root-MUSIC.

A numerical example is given to demonstrate the effectiveness of the proposed distributed spatial smoothing. We consider the same network (6.27) and use the finite-time AC method with 3 iterations as explained in Sec. 6.3.1. We show the DOA estimation performance of distributed spatial smoothing used with root-MUSIC. Again, the distributed power method is used to estimate eigenvectors of the covariance for distributed algorithms, whereas ideal EVD is assumed for centralized algorithms. Each node is a ULA with $Q = 16$ sensors and they together form a 96-sensor ULA. There are 6 DOAs at angles $\theta = -10°, -6°, -2°, \ldots, 10°$. All sources have power 1, and each pair of sources have the same correlation coefficient ρ. The noise variance is $\sigma_e^2 = 1$. The parameter $L_{\text{ss}} = 17$ is used for spatial smoothing. The assumption $(N - L_{\text{ss}} + 1) - 1 \leq Q$ is not satisfied, so Algorithm 4 is used. We assume the number of DOAs is known. The number of power iterations is $I_{\text{pm}} = 5$. RMSEs of DOA estimates for various ρ are shown in Fig. 6.7. We use 500 snapshots and average 2000 Monte Carlo runs to get the plot. Spatial smoothing improves performance significantly, especially for the distributed case. Except for the coherent case ($\rho = 1$), RMSE of distributed spatial smoothing is almost the same as that of centralized spatial smoothing, though we use a small $I_{\text{pm}} = 5$. When there is no spatial smoothing, or when $\rho = 1$, the distributed algorithm has larger RMSE than the centralized one because the covariance is closer to being rank-deficient, so errors due to finite I_{pm} are magnified.

6.6 Concluding Remarks

In this chapter, we propose distributed algorithms for several DOA estimation and beamforming methods, including spatial smoothing methods and the convolutional beamspace methods introduced in previous chapters. These algorithms are truly distributed in that a fusion center is not required. The novelties of the proposed algorithms lie in transforming problems at hand into steps where computing the average of some values across the network is the only step that involves data exchange among subarrays. Using average consensus with finite-time exact convergence as a subroutine, these distributed algorithms can achieve the same performance as centralized algorithms in just a few iterations. The effectiveness of the distributed algorithms is verified through simulations.

In the future, one can consider developing distributed algorithms for the difference coarray method [32] for sparse arrays. It is possible that some sparse arrays can facilitate the design of distributed coarray method, whereas others can cause difficulties. Hence, the sensor locations of the sparse arrays may need to be jointly designed.

Chapter 7

CONCLUSIONS AND FUTURE DIRECTIONS

In this thesis, we proposed a new beamspace method, convolutional beamspace (CBS). CBS is especially suitable for large arrays since it achieves DOA and channel estimation with low complexity. We presented the basic digital CBS for DOA estimation based on passive arrays in Chapter 2. We further developed the hybrid (analog and digital) CBS for DOA estimation based on passive arrays in Chapter 3. Then we proposed hybrid CBS for mmWave MIMO channel estimation in Chapter 4. We analyzed the MSE performance and derived the Cramér–Rao bound of CBS for DOA estimation in Chapter 5. As another line of research on array processing algorithms with low computation and communication complexity, we developed distributed algorithms for several well-known DOA estimation and beamforming methods in Chapter 6.

In addition to the results presented in this thesis, there are several related array processing topics worth studying in the future:

1. *CBS with gridless sparse recovery methods:* In the discussions about CBS in this thesis, we focus on beamspace processing followed by subspace DOA estimation methods. However, the DOAs can also be estimated from the CBS output using sparse recovery methods, such as dictionary methods as in Sec. 2.4. Dictionary methods are grid-based methods and suffer from off-grid errors, so one may want to consider CBS with gridless sparse recovery methods such as atomic norm minimization method [92]. Atomic norm minimization method suffers from high computational complexity, so the use of CBS should be beneficial.

2. *CBS for beamforming:* In the CBS method for passive arrays, there are two stages. In stage 1, we do filtering and decimation. In stage 2, we only do DOA estimation in this thesis. We have considered designing the filter based on the idea of Capon beamforming [36, 83] in stage 1, leading to the Capon-CBS method. In the future, we can also consider beamforming in stage 2. Besides low complexity, there can be other advantages. When we decimate the CBS filter output by M, we can get M polyphase components [66]. Hence, higher

signal to interference-plus-noise ratio (SINR) can be potentially obtained by taking a weighted average of the beamformer outputs resulting from the M polyphase components. The design of the weights is thus an interesting optimization problem.

3. *Reconfigurable intelligent surfaces for mmWave MIMO channels:* Besides severe path loss, mmWave communications are also faced with the challenge of sensitivity to blockages. Many propose to use a new technology, reconfigurable intelligent surface (RIS) [86, 105], to solve the issue. An RIS can be installed on large flat surfaces like buildings to reflect RF energy around obstacles. It consists of an array of passive reflecting elements. Each reflecting element can independently impose the required phase shift on the incoming signal so that the overall reflected signal can be reconfigured to propagate toward the desired directions. To make good use of such RIS-aided communication systems, many methods require channel state information. Thus, channel estimation is a crucial step. This channel estimation is challenging because the RIS is passive without active RF chains or computational power, and the number of RIS elements is often large. Hence, it is important to design practical algorithms with low hardware and computational complexity for RIS-aided mmWave MIMO channel estimation. In particular, it is promising to adapt hybrid CBS to the RIS framework since hybrid CBS has low complexity and good estimation performance.

4. *Machine learning for array processing and communications:* The use of machine learning for classical array processing tasks has gained more attention recently [131, 132]. Classical methods based on theoretical modeling and derivations typically perform well or even the best if the modeling is perfect. In contrast, when there is model mismatch such as array imperfections, machine learning-based methods can perform better since they are data-driven. To combine the benefits of both methods, one may consider using general system architectures as in classical designs while tuning relevant parameters with machine learning. For instance, machine learning, particularly deep neural networks, can be used to optimize the analog and digital precoder and combiner coefficients in mmWave MIMO communication systems, or the filter coefficients of CBS. These problems are nonconvex and difficult to solve using classical techniques. Machine learning opens up a new possibility of solving them.

5. *Applications in autonomous vehicles:* As an emerging technology, autonomous vehicles have attracted much research attention recently. To achieve the self-driving ability, a huge amount of data should be communicated among vehicles and between vehicles and the infrastructure. Thus, mmWave communications, which can offer large bandwidths, will be likely considered [133]. Again, to overcome the severe path loss of mmWaves, large antenna arrays will be used to get large beamforming gain. To steer the beams towards the target directions, the DOAs of the targets have to be estimated first. Besides DOA estimation, the estimation of other parameters such as range and Doppler frequency can also be relevant [134]. Hence, it is interesting to explore how to design low-complexity and high-resolution beamspace methods like hybrid CBS in this high-dimensional estimation framework.

Bibliography

[1] M. D. Zoltowski, S. D. Silverstein, and C. P. Mathews, "Beamspace root-MUSIC for minimum redundancy linear arrays," *IEEE Trans. Signal Process.*, vol. 41, no. 7, pp. 2502–2507, Jul. 1993.

[2] M. D. Zoltowski, G. M. Kautz, and S. D. Silverstein, "Beamspace root-MUSIC," *IEEE Trans. Signal Process.*, vol. 41, no. 1, pp. 344–364, Jan. 1993.

[3] A. T. Moffet, "Minimum-redundancy linear arrays," *IEEE Trans. Antennas Propag.*, vol. 16, no. 2, pp. 172–175, Mar. 1968.

[4] J. Zhang and M. Haardt, "Channel estimation and training design for hybrid multi-carrier mmWave massive MIMO systems: The beamspace ESPRIT approach," in *25th European Signal Process. Conf. (EUSIPCO)*, 2017, pp. 385–389.

[5] Y.-P. Lin and T.-M. Yang, "Random SBT precoding for angle estimation of mmWave massive MIMO systems using sparse arrays spacing," *IEEE Access*, vol. 8, pp. 163 380–163 393, 2020.

[6] Z. Xiao, P. Xia, and X.-G. Xia, "Channel estimation and hybrid precoding for millimeter-wave MIMO systems: A low-complexity overall solution," *IEEE Access*, vol. 5, pp. 16 100–16 110, 2017.

[7] A. Scaglione, R. Pagliari, and H. Krim, "The decentralized estimation of the sample covariance," in *Proc. Asilomar Conf. on Signal, Syst., Comput.*, 2008, pp. 1722–1726.

[8] H. L. Van Trees, *Optimum array processing*. John Wiley & Sons, Inc., New York, 2002.

[9] A. L. Swindlehurst, E. Ayanoglu, P. Heydari, and F. Capolino, "Millimeter-wave massive MIMO: the next wireless revolution?" *IEEE Commun. Mag.*, vol. 52, no. 9, pp. 56–62, Sep. 2014.

[10] J. G. Andrews, S. Buzzi, W. Choi, S. V. Hanly, A. Lozano, A. C. Soong, and J. C. Zhang, "What will 5G be?" *IEEE J. on Select. Areas in Commun.*, vol. 32, no. 6, pp. 1065–1082, 2014.

[11] A. Alkhateeb, O. El Ayach, G. Leus, and R. W. Heath, "Channel estimation and hybrid precoding for millimeter wave cellular systems," *IEEE J. of Select. Topics in Signal Process.*, vol. 8, no. 5, pp. 831–846, 2014.

[12] G. Bienvenu and L. Kopp, "Decreasing high resolution method sensitivity by conventional beamformer preprocessing," in *Proc. IEEE Intl. Conf. Acoust., Speech, and Signal Process.*, Mar. 1984, pp. 714–717.

[13] B. Van Veen and B. Williams, "Structured covariance matrices and dimensionality reduction in array processing," in *Fourth Annual ASSP Workshop on Spec. Est. and Modeling*, 1988, pp. 168–171.

[14] K. M. Buckley and X. L. Xu, "Reduced-dimension beam-space broad-band source localization: preprocessor design," in *Proc. SPIE, Adv. Algorithms and Architectures for Signal Process. III*, Feb. 1988, pp. 368–376.

[15] H. B. Lee and M. S. Wengrovitz, "Improved high-resolution direction-finding through use of homogeneous constraints," in *Fourth Annual ASSP Workshop on Spec. Est. and Modeling*, Aug. 1988, pp. 152–157.

[16] ——, "Resolution threshold of beamspace MUSIC for two closely spaced emitters," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 38, no. 9, pp. 1545–1559, Sep. 1990.

[17] X. L. Xu and K. M. Buckley, "Statistical performance comparison of MUSIC in element-space and beam-space," in *Proc. IEEE Intl. Conf. Acoust., Speech, and Signal Process.*, 1989, pp. 2124–2127.

[18] G. Xu, S. D. Silverstein, R. H. Roy, and T. Kailath, "Beamspace ESPRIT," *IEEE Trans. Signal Process.*, vol. 42, no. 2, pp. 349–356, Feb. 1994.

[19] Z. Guo, X. Wang, and W. Heng, "Millimeter-wave channel estimation based on 2-D beamspace MUSIC method," *IEEE Trans. Wireless Commun.*, vol. 16, no. 8, pp. 5384–5394, Aug. 2017.

[20] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Systems & Control Letters*, vol. 53, no. 1, pp. 65–78, 2004.

[21] A. Sandryhaila, S. Kar, and J. M. F. Moura, "Finite-time distributed consensus through graph filters," in *Proc. IEEE Intl. Conf. Acoust., Speech, and Signal Process.*, 2014, pp. 1080–1084.

[22] S. Safavi and U. A. Khan, "Revisiting finite-time distributed algorithms via successive nulling of eigenvalues," *IEEE Signal Process. Lett.*, vol. 22, no. 1, pp. 54–57, 2015.

[23] P.-C. Chen and P. P. Vaidyanathan, "Convolutional beamspace for linear arrays," *IEEE Trans. Signal Process.*, vol. 68, pp. 5395–5410, 2020.

[24] P. P. Vaidyanathan and P.-C. Chen, "Convolutional beamspace for array signal processing," in *IEEE Intl. Conf. Acoust., Speech, and Signal Process.*, 2020.

[25] A. Barabell, "Improving the resolution performance of eigenstructure-based direction-finding algorithms," in *IEEE Intl. Conf. Acoust., Speech, and Signal Process.*, vol. 8, 1983, pp. 336–339.

[26] B. D. Rao and K. V. S. Hari, "Performance analysis of Root-Music," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 37, no. 12, pp. 1939–1949, 1989.

[27] R. Roy and T. Kailath, "ESPRIT – estimation of signal parameters via rotational invariance techniques," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 37, no. 7, pp. 984–995, Jul. 1989.

[28] P.-C. Chen and P. P. Vaidyanathan, "Channel estimation for mmWave using the convolutional beamspace approach," *IEEE Trans. Signal Process.*, 2024 (to appear).

[29] P.-C. Chen and P. P. Vaidyanathan, "Hybrid convolutional beamspace method for mmWave MIMO channel estimation," in *2023 Asilomar Conf. on Signal, Syst., Comput.*, 2023.

[30] P.-C. Chen and P. P. Vaidyanathan, "DOA estimation for mmWave passive arrays with hybrid beamspace processing," *IEEE Access*, under review.

[31] P.-C. Chen and P. P. Vaidyanathan, "Hybrid convolutional beamspace for DOA estimation of millimeter wave sources," in *2022 Asilomar Conf. on Signal, Syst., Comput.*, 2022, pp. 86–90.

[32] P. Pal and P. P. Vaidyanathan, "Nested arrays: A novel approach to array processing with enhanced degrees of freedom," *IEEE Trans. Signal Process.*, vol. 58, no. 8, pp. 4167–4181, Aug. 2010.

[33] P.-C. Chen and P. P. Vaidyanathan, "Distributed algorithms for array signal processing," *IEEE Trans. Signal Process.*, vol. 69, pp. 4607–4622, 2021.

[34] ——, "Distributed root-MUSIC using finite-time average consensus," in *2021 Asilomar Conf. on Signal, Syst., Comput.*, 2021, pp. 539–543.

[35] W. Suleiman, M. Pesavento, and A. Zoubir, "Decentralized direction finding using partly calibrated arrays," in *European Signal Process. Conf. (EUSIPCO)*, 2013, pp. 1–5.

[36] J. Capon, "High-resolution frequency-wavenumber spectrum analysis," *Proc. IEEE*, vol. 57, no. 8, pp. 1408–1418, 1969.

[37] I. F. Gorodnitsky and B. D. Rao, "Sparse signal reconstruction from limited data using FOCUSS: a re-weighted minimum norm algorithm," *IEEE Trans. Signal Process.*, vol. 45, no. 3, pp. 600–616, 1997.

[38] S. F. Cotter, B. D. Rao, Kjersti Engan, and K. Kreutz-Delgado, "Sparse solutions to linear inverse problems with multiple measurement vectors," *IEEE Trans. Signal Process.*, vol. 53, no. 7, pp. 2477–2488, 2005.

[39] Tie-Jun Shan, M. Wax, and T. Kailath, "On spatial smoothing for direction-of-arrival estimation of coherent signals," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 33, no. 4, pp. 806–811, 1985.

[40] P. P. Vaidyanathan and P. Pal, "Sparse sensing with co-prime samplers and arrays," *IEEE Trans. Signal Process.*, vol. 59, no. 2, pp. 573–586, Feb. 2011.

[41] O. El Ayach, S. Rajagopal, S. Abu-Surra, Z. Pi, and R. W. Heath, "Spatially sparse precoding in millimeter wave MIMO systems," *IEEE Trans. Wireless Commun.*, vol. 13, no. 3, pp. 1499–1513, 2014.

[42] S. X. Wu, H. Wai, L. Li, and A. Scaglione, "A review of distributed algorithms for principal component analysis," *Proc. of the IEEE*, vol. 106, no. 8, pp. 1321–1340, 2018.

[43] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Prentice-Hall, Inc., 1993.

[44] P. Stoica and A. Nehorai, "Performance study of conditional and unconditional direction-of-arrival estimation," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 38, no. 10, pp. 1783–1795, Oct. 1990.

[45] X. Gao, L. Dai, and A. M. Sayeed, "Low RF-complexity technologies to enable millimeter-wave MIMO with large antenna array for 5G wireless communications," *IEEE Commun. Mag.*, vol. 56, no. 4, pp. 211–217, Apr. 2018.

[46] P.-C. Chen and P. P. Vaidyanathan, "Convolutional beamspace using IIR filters," in *IEEE Intl. Conf. Acoust., Speech, and Signal Process.*, 2022, pp. 5003–5007.

[47] ——, "Sliding-Capon based convolutional beamspace for linear arrays," in *IEEE Intl. Conf. Acoust., Speech, and Signal Process.*, 2021, pp. 4565–4569.

[48] X. Zhang, A. F. Molisch, and S.-Y. Kung, "Variable-phase-shift-based RF-baseband codesign for MIMO antenna selection," *IEEE Trans. Signal Process.*, vol. 53, no. 11, pp. 4091–4103, 2005.

[49] C.-L. Liu and P. P. Vaidyanathan, "Cramér–Rao bounds for coprime and other sparse arrays, which find more sources than sensors," *Digital Signal Processing*, vol. 61, pp. 43–61, 2017.

[50] M. Wang and A. Nehorai, "Coarrays, MUSIC, and the Cramér–Rao bound," *IEEE Trans. Signal Process.*, vol. 65, no. 4, pp. 933–946, 2017.

[51] M. D. Zoltowski, M. Haardt, and C. P. Mathews, "Closed-form 2-D angle estimation with rectangular arrays in element space or beamspace via unitary ESPRIT," *IEEE Trans. Signal Process.*, vol. 44, no. 2, pp. 316–328, 1996.

[52] R. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Trans. Antennas Propag.*, vol. 34, no. 3, pp. 276–280, Mar. 1986.

[53] W. Suleiman, M. Pesavento, and A. M. Zoubir, "Performance analysis of the decentralized eigendecomposition and ESPRIT algorithm," *IEEE Trans. Signal Process.*, vol. 64, no. 9, pp. 2375–2386, 2016.

[54] R. Yang, D. Gray, and W. Al-Ashwal, "Estimation of the DOAs of coherent signals in beam space processing for phased arrays," in *IEEE Intl. Conf. on Radar*, Aug. 2018, pp. 1–5.

[55] H. Zhao, N. Zhang, and Y. Shen, "Robust beamspace design for direct localization," in *IEEE Intl. Conf. Acoust., Speech and Signal Process.*, May 2019, pp. 4360–4364.

[56] P. Stoica and A. Nehorai, "MUSIC, maximum likelihood, and Cramer-Rao bound," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 37, no. 5, pp. 720–741, May 1989.

[57] C.-L. Liu and P. P. Vaidyanathan, "Super nested arrays: Linear sparse arrays with reduced mutual coupling–Part I: Fundamentals," *IEEE Trans. Signal Process.*, vol. 64, no. 15, pp. 3997–4012, Aug. 2016.

[58] S. Qin, Y. D. Zhang, and M. G. Amin, "Generalized coprime array configurations for direction-of-arrival estimation," *IEEE Trans. Signal Process.*, vol. 63, no. 6, pp. 1377–1390, Mar. 2015.

[59] D. Malioutov, M. Cetin, and A. S. Willsky, "A sparse signal reconstruction perspective for source localization with sensor arrays," *IEEE Trans. Signal Process.*, vol. 53, no. 8, pp. 3010–3022, Aug. 2005.

[60] S. D. Silverstein, W. E. Engeler, and J. A. Tardif, "Parallel architectures for multirate superresolution spectrum analyzers," *IEEE Trans. Circuits Syst.*, vol. 38, no. 4, pp. 449–453, Apr. 1991.

[61] A. Tkacenko and P. P. Vaidyanathan, "The role of filter banks in sinusoidal frequency estimation," *Journal of the Franklin Institute*, vol. 338, no. 5, pp. 517–547, 2001.

[62] L. Qiu, Y. Cai, R. C. de Lamare, and M. Zhao, "Reduced-rank DOA estimation algorithms based on alternating low-rank decomposition," *IEEE Signal Process. Lett.*, vol. 23, no. 5, pp. 565–569, 2016.

[63] A. V. Oppenheim and R. W. Schafer, *Discrete-time signal processing*. Prentice Hall, 2010.

[64] L. Wang, R. C. de Lamare, and M. Haardt, "Direction finding algorithms based on joint iterative subspace optimization," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 50, no. 4, pp. 2541–2553, 2014.

[65] R. C. de Lamare and R. Sampaio-Neto, "Adaptive reduced-rank processing based on joint and iterative interpolation, decimation, and filtering," *IEEE Trans. Signal Process.*, vol. 57, no. 7, pp. 2503–2514, 2009.

[66] P. P. Vaidyanathan, *Multirate systems and filter banks*. Prentice Hall, Englewood Cliffs, N.J., 1993.

[67] ——, "Multirate digital filters, filter banks, polyphase networks, and applications: a tutorial," *Proc. IEEE*, vol. 78, no. 1, pp. 56–93, 1990.

[68] J. G. Proakis, *Digital communications*. Mc-Graw Hill, New York, 2008.

[69] H. Samueli, "On the design of optimal equiripple FIR digital filters for data transmission applications," *IEEE Trans. Circuits Syst.*, vol. 35, no. 12, pp. 1542–1546, Dec. 1988.

[70] P. P. Vaidyanathan, T. Q. Nguyen, Z. Doganata, and T. Saramaki, "Improved technique for design of perfect reconstruction FIR QMF banks with lossless polyphase matrices," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 37, no. 7, pp. 1042–1056, Jul. 1989.

[71] P. P. Vaidyanathan and T. Q. Nguyen, "Eigenfilters: a new approach to least-squares FIR filter design and applications including Nyquist filters," *IEEE Trans. Circuits Syst.*, vol. 34, no. 1, pp. 11–23, Jan. 1987.

[72] H. Samueli, "On the design of FIR digital data transmission filters with arbitrary magnitude specifications," *IEEE Trans. Circuits Syst.*, vol. 38, no. 12, pp. 1563–1567, Dec. 1991.

[73] R. D. Koilpillai and P. P. Vaidyanathan, "Cosine-modulated FIR filter banks satisfying perfect reconstruction," *IEEE Trans. Signal Process.*, vol. 40, no. 4, pp. 770–783, Apr. 1992.

[74] Y.-P. Lin and P. P. Vaidyanathan, "A Kaiser window approach for the design of prototype filters of cosine modulated filterbanks," *IEEE Signal Process. Lett.*, vol. 5, no. 6, pp. 132–134, Jun. 1998.

[75] P. P. Vaidyanathan and P.-Q. Hoang, "Lattice structures for optimal design and robust implementation of two-channel perfect-reconstruction QMF banks," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 36, no. 1, pp. 81–94, Jan. 1988.

[76] M. Vetterli and J. Kovacevic, *Wavelets and subband coding*. Prentice Hall, Englewood Cliffs, N.J., 1995.

[77] P. Stoica and A. Nehorai, "Comparative performance study of element-space and beam-space MUSIC estimators," *Circuits, Systems and Signal Process.*, vol. 10, no. 3, pp. 285–292, Sep. 1991.

[78] A. J. Weiss and B. Friedlander, "Preprocessing for direction finding with minimal variance degradation," *IEEE Trans. Signal Process.*, vol. 42, no. 6, pp. 1478–1485, 1994.

[79] C.-L. Liu and P. P. Vaidyanathan, "Remarks on the spatial smoothing step in coarray MUSIC," *IEEE Signal Process. Lett.*, vol. 22, no. 9, pp. 1438–1442, Sep. 2015.

[80] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996.

[81] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 2.1," http://cvxr.com/cvx, Mar. 2014.

[82] Z. Yang and L. Xie, "A weighted atomic norm approach to spectral super-resolution with probabilistic priors," in *IEEE Intl. Conf. Acoust., Speech, and Signal Process.*, 2016, pp. 4598–4602.

[83] R. G. Lorenz and S. P. Boyd, "Robust minimum variance beamforming," *IEEE Trans. Signal Process.*, vol. 53, no. 5, pp. 1684–1696, 2005.

[84] P. Stoica, Z. Wang, and J. Li, "Robust Capon beamforming," *IEEE Signal Process. Lett.*, vol. 10, no. 6, pp. 172–175, 2003.

[85] C.-Y. Chen and P. P. Vaidyanathan, "Quadratically constrained beamforming robust against direction-of-arrival mismatch," *IEEE Trans. Signal Process.*, vol. 55, no. 8, pp. 4139–4150, 2007.

[86] C. Pan, H. Ren, K. Wang, J. F. Kolb, M. Elkashlan, M. Chen, M. Di Renzo, Y. Hao, J. Wang, A. L. Swindlehurst *et al.*, "Reconfigurable intelligent surfaces for 6G systems: Principles, applications, and research directions," *IEEE Commun. Mag.*, vol. 59, no. 6, pp. 14–20, 2021.

[87] P. P. Vaidyanathan, S. Mitra, and Y. Neuvo, "A new approach to the realization of low-sensitivity IIR digital filters," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 34, no. 2, pp. 350–361, 1986.

[88] P. P. Vaidyanathan, P. Regalia, and S. Mitra, "Design of doubly-complementary IIR digital filters using a single complex allpass filter, with multirate applications," *IEEE Trans. Circuits Syst.*, vol. 34, no. 4, pp. 378–389, 1987.

[89] M. Renfors and T. Saramaki, "Recursive Nth-band digital filters–Part I: Design and properties," *IEEE Trans. Circuits Syst.*, vol. 34, no. 1, pp. 24–39, 1987.

[90] T. Q. Nguyen, T. I. Laakso, and T. E. Tuncer, "On perfect-reconstruction allpass-based cosine-modulated IIR filter banks," in *Proc. of IEEE Int. Symp. on Circuits and Syst.*, vol. 2, 1994, pp. 33–36.

[91] H. W. Lollmann and P. Vary, "Design of critically subsampled DFT filter-banks with allpass polyphase filters and near-perfect reconstruction," in *IEEE Intl. Conf. Acoust., Speech and Signal Process.*, 2009, pp. 3185–3188.

[92] Y. Li and Y. Chi, "Off-the-grid line spectrum denoising and estimation with multiple measurement vectors," *IEEE Trans. Signal Process.*, vol. 64, no. 5, pp. 1257–1269, 2015.

[93] S. H. Dokhanchi, B. S. Mysore, K. V. Mishra, and B. Ottersten, "A mmWave automotive joint radar-communications system," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 55, no. 3, pp. 1241–1260, 2019.

[94] J. Mo, P. Schniter, and R. W. Heath, "Channel estimation in broadband millimeter wave MIMO systems with few-bit ADCs," *IEEE Trans. Signal Process.*, vol. 66, no. 5, pp. 1141–1154, 2018.

[95] F. Sohrabi and W. Yu, "Hybrid digital and analog beamforming design for large-scale antenna arrays," *IEEE J. Sel. Topics Signal Process.*, vol. 10, no. 3, pp. 501–513, 2016.

[96] R. Rajamäki, S. P. Chepuri, and V. Koivunen, "Hybrid beamforming for active sensing using sparse arrays," *IEEE Trans. Signal Process.*, vol. 68, pp. 6402–6417, 2020.

[97] A. Liao, Z. Gao, Y. Wu, H. Wang, and M.-S. Alouini, "2D unitary ESPRIT based super-resolution channel estimation for millimeter-wave massive MIMO with hybrid precoding," *IEEE Access*, vol. 5, pp. 24 747–24 757, 2017.

[98] M. F. Keskin, A. Fascista, F. Jiang, A. Coluccia, G. Seco-Granados, and H. Wymeersch, "ESPRIT-oriented precoder design for mmWave channel estimation," *arXiv preprint arXiv:2301.01585*, 2023.

[99] C. Lin and G. Y. L. Li, "Terahertz communications: An array-of-subarrays solution," *IEEE Commun. Mag.*, vol. 54, no. 12, pp. 124–131, 2016.

[100] V. Va, H. Vikalo, and R. W. Heath, "Beam tracking for mobile millimeter wave communication systems," in *IEEE Global Conf. on Signal and Inf. Process.*, 2016, pp. 743–747.

[101] Z. Xiao, T. He, P. Xia, and X.-G. Xia, "Hierarchical codebook design for beamforming training in millimeter-wave communication," *IEEE Trans. Wireless Commun.*, vol. 15, no. 5, pp. 3380–3392, 2016.

[102] A. Liu, L. Lian, V. K. Lau, and X. Yuan, "Downlink channel estimation in multiuser massive MIMO with hidden Markovian sparsity," *IEEE Trans. Signal Process.*, vol. 66, no. 18, pp. 4796–4810, 2018.

[103] W. Xu, A. Liu, B. Zhou, and M. Zhao, "Successive linear approximation VBI for joint sparse signal recovery and dynamic grid parameters estimation," *arXiv preprint arXiv:2307.09149*, 2023.

[104] S. Qin, Y. D. Zhang, M. G. Amin, and F. Gini, "Frequency diverse coprime arrays with coprime frequency offsets for multitarget localization," *IEEE J. Sel. Topics Signal Process.*, vol. 11, no. 2, pp. 321–335, 2017.

[105] A. L. Swindlehurst, G. Zhou, R. Liu, C. Pan, and M. Li, "Channel estimation with reconfigurable intelligent surfaces–a general framework," *Proc. IEEE*, vol. 110, no. 9, pp. 1312–1338, 2022.

[106] R. R. Pote and B. D. Rao, "Reduced dimension beamspace design incorporating nested array for mmWave channel estimation," in *2019 Asilomar Conf. on Signal, Syst., Comput.*, 2019, pp. 1212–1216.

[107] T. W. Anderson, "Asymptotic theory for principal component analysis," *The Annals of Mathematical Statistics*, vol. 34, no. 1, pp. 122–148, 1963.

[108] Y. C. Eldar, "Uniformly improving the Cramér-Rao bound and maximum-likelihood estimation," *IEEE Trans. Signal Process.*, vol. 54, no. 8, pp. 2943–2956, 2006.

[109] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. Wiley, 2012.

[110] F. Rellich and J. Berkowitz, *Perturbation theory of eigenvalue problems*. CRC Press, 1969.

[111] B. Champagne, "Adaptive eigendecomposition of data covariance matrices based on first-order perturbations," *IEEE Trans. Signal Process.*, vol. 42, no. 10, pp. 2758–2770, 1994.

[112] R. A. Horn and C. R. Johnson, *Matrix analysis*. Cambridge university press, 2012.

[113] C. Fassino, G. Pistone, and M. P. Rogantin, "Computing the moments of the complex gaussian: Full and sparse covariance matrix," *Mathematics*, vol. 7, no. 3, p. 263, 2019.

[114] S. Janson, *Gaussian Hilbert Spaces*. Cambridge university press, 1997, no. 129.

[115] A. Bertrand and M. Moonen, "Distributed adaptive estimation of covariance matrix eigenvectors in wireless sensor networks with application to distributed PCA," *Signal Process.*, vol. 104, pp. 120–135, 2014.

[116] A. Hassani, A. Bertrand, and M. Moonen, "Cooperative integrated noise reduction and node-specific direction-of-arrival estimation in a fully connected wireless acoustic sensor network," *Signal Process.*, vol. 107, pp. 68–81, 2015.

[117] M. Wax and T. Kailath, "Decentralized processing in sensor arrays," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 33, no. 5, pp. 1123–1129, 1985.

[118] W. Suleiman and P. Parvazi, "Search-free decentralized direction-of-arrival estimation using common roots for non-coherent partly calibrated arrays," in *IEEE Intl. Conf. Acoust., Speech, and Signal Process.*, 2014, pp. 2292–2296.

[119] B. Iancu and E. Isufi, "Towards finite-time consensus with graph convolutional neural networks," in *28th Eur. Signal Process. Conf. (EUSIPCO)*, 2021, pp. 2145–2149.

[120] B. N. Bhaskar, G. Tang, and B. Recht, "Atomic norm denoising with applications to line spectral estimation," *IEEE Trans. Signal Process.*, vol. 61, no. 23, pp. 5987–5999, 2013.

[121] P. E. Gill, W. Murray, and M. H. Wright, *Practical optimization*. Academic Press, New York, 1981.

[122] O. Aberth, "Iteration methods for finding all zeros of a polynomial simultaneously," *Mathematics of computation*, vol. 27, no. 122, pp. 339–344, 1973.

[123] R. Merris, "Laplacian matrices of graphs: a survey," *Linear algebra and its applications*, vol. 197, pp. 143–176, 1994.

[124] G. W. Stewart, *Matrix Algorithms: Vol. II: Eigensystems*. SIAM, 2001.

[125] M. R. Hestenes and E. Stiefel, "Methods of conjugate gradients for solving linear systems," *Journal of Research of the National Bureau of Standards*, vol. 49, pp. 409–435, 1952.

[126] C.-Y. Chen and P. P. Vaidyanathan, "MIMO radar space-time adaptive processing using prolate spheroidal wave functions," *IEEE Trans. Signal Process.*, vol. 56, no. 2, pp. 623–635, 2008.

[127] P. P. Vaidyanathan and P. Pal, "Direct-MUSIC on sparse arrays," in *2012 Intl. Conf. on Signal Process. and Commun. (SPCOM)*, 2012, pp. 1–5.

[128] ——, "Why does direct-MUSIC on sparse-arrays work?" in *2013 Asilomar Conf. on Signal, Syst., Comput.*, 2013, pp. 2007–2011.

[129] B. Ottersten, M. Viberg, and T. Kailath, "Performance analysis of the total least squares ESPRIT algorithm," *IEEE Trans. Signal Process.*, vol. 39, no. 5, pp. 1122–1135, 1991.

[130] J. S. Thompson, P. M. Grant, and B. Mulgrew, "Performance of spatial smoothing algorithms for correlated sources," *IEEE Trans. Signal Process.*, vol. 44, no. 4, pp. 1040–1046, 1996.

[131] Z.-M. Liu, C. Zhang, and S. Y. Philip, "Direction-of-arrival estimation based on deep neural networks with robustness to array imperfections," *IEEE Trans. Antennas Propag.*, vol. 66, no. 12, pp. 7315–7327, 2018.

[132] P. Kulkarni and P. P. Vaidyanathan, "Feature engineering for DOA estimation using a convolutional neural network, for sparse arrays," in *2021 Asilomar Conf. on Signal, Syst., Comput.*, 2021, pp. 246–250.

[133] L. Kong, M. K. Khan, F. Wu, G. Chen, and P. Zeng, "Millimeter-wave wireless communications for IoT-cloud supported autonomous vehicles: Overview, design, and challenges," *IEEE Commun. Mag.*, vol. 55, no. 1, pp. 62–68, 2017.

[134] S. Sun and Y. D. Zhang, "4D automotive radar sensing for autonomous vehicles: A sparsity-oriented approach," *IEEE J. of Select. Topics in Signal Process.*, vol. 15, no. 4, pp. 879–891, 2021.