Scalable Sparse Optimization in Dense Cloud-RAN

by

Yuanming SHI

A Thesis Submitted to
The Hong Kong University of Science and Technology
in Partial Fulfillment of the Requirements for
the Degree of Doctor of Philosophy
in the Department of Electronic and Computer Engineering

August 2015, Hong Kong

Authorization

I hereby declare that I am the sole author of the thesis.

I authorize the Hong Kong University of Science and Technology to lend this thesis to other institutions or individuals for the purpose of scholarly research.

I further authorize the Hong Kong University of Science and Technology to reproduce the thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

Yuanming SHI

August 2015

Scalable Sparse Optimization in Dense Cloud-RAN

by

Yuanming SHI

This is to certify that I have examined the above PhD thesis
and have found that it is complete and satisfactory in all respects,
and that any and all revisions required by
the thesis examination committee have been made.

Prof. Khaled B. LETAIEF, ECE (Thesis Supervisor)

Prof. Jidong ZHAO, CIVL (Committee Chairperson)

Prof. Vincent LAU, ECE (Committee Member)

Prof. Danny H. TSANG, ECE (Committee Member)

Prof. Dit-Yan YEUNG, CSE (Committee Member)

Prof. Bertram SHI, ECE (Department Head)

The Department of Electronic and Computer Engineering

August 2015

To My Family

Acknowledgements

I am deeply indebted to my supervisor Prof. Khaled B. Letaief for his endless support and continual encouragement throughout my PhD studies. His brilliant insights, energetic enthusiasm and deep knowledge have greatly shaped my research. This thesis would not have been possible without his aesthetic taste on the topic of cloud radio access networks. I want to express my deepest gratitude to him for providing me with invaluable guidance on approaching research, making presentations, writing papers, as well as positive encouragement despite his busy schedules. In addition, I appreciate his endless patience and the freedom he gave, which allow me to purse my research interests among many subjects as wireless communication networks, optimization, cloud computing, machine learning, and differential geometry.

I am very grateful to Prof. Jun Zhang for being my co-advisor during my PhD studies. Without his constant guidance and support at every stage of my PhD studies, my life at HKUST would not have been comfortable and fruitful. I very enjoyed the discussions with him on research during the weekly individual meetings. I learned so much from him on how to find out a meaningful topic, formulate the problem in the cleanest way, solve the problem in a quite smart way and present the results in an elegant way. His insights, inspiration, well-known wisdom, the way of conducting research have always been inspirational to me. My collaboration experience with Prof. Jun Zhang has mostly shaped the way that I view research and the field of wireless networks. I would also like thank Prof. Shenghui Song for his inspiration discussions and insightful comments during the group meetings.

I am very fortunate to have the opportunity to collaborate with Prof. Wei Chen and Prof. Bo Bai at Tsinghua since my undergraduate studies. Their enthusiasm and insights on research have led me into the wireless communications realm. I would like to express my deepest gratitude to them for their endless help, generosity of heart, insightful discussions, invaluable guidance and positive encouragement over the past five years. Without their help, my life would not have been so colorful and meaningful.

I thank Prof. Vincent Lau, Prof. Danny H. Tsang and Prof. Dit-Yan Yeung for serving my thesis defense committee. I thank Prof. Ling Shi for being my thesis proposal committee. I am also grateful to Prof. Weihua Zhuang for serving my thesis external examiner despite her busy schedules and the long trip for coming to Hong Kong. I also would like to thank the faculty of the Department of Electronic and Computer Engineering and the Department of Mathematics for providing great curricula and inspirational environment for graduate studies. In particular, I thank Prof. Vincent Lau, Prof. Daniel P. Palomar, Prof. Matthew McKay, Prof. Ross D. Murch, Prof. Bertram Shi, Prof. Bingyi Jing, Prof. Kani Chen for their teaching. I owe my particular gratitude to Prof. L. Jeff Hong for his invaluable insights and discussions on the chance-constrained optimization problem.

I have been fortunate to have many friends and colleagues who made my four years at HKUST a memorable journey. In particular, I thank my groupmates for their rewarding discussions and endless wisdom: Xiang Chen, Fangyong Li, Yaming Luo, Chang Li, Lu Yang, Xi Peng, Rui Wang, Yuyi Mao, Xianghao Yu, Changming Li, Xiaoyu Chen, Juei-Chin Shen and Juan Liu. I would like to thank Dr. Brendan O'Donoghue at Stanford for his enormous insightful discussions and endless support for developing the large-scale convex optimization framework.

Finally and most importantly, it is my greatest honor to thank my family: my mother, my father and my sisters, for their unconditional love and unwavering support throughout all my endeavors. No words can fully express my deepest gratitude to them. As a futile attempt, I dedicate this thesis to them.

Table of Contents

List of Tables

List of Figures

Abstract

The cloud radio access network (Cloud-RAN) provides a revolutionary way to densify radio access networks, thereby addressing the challenges in the era of mobile data deluge. In this architecture, all the baseband signal processing is shifted to a single cloud data center, which enables centralized resource coordination and signal processing for efficient interference management and flexible network adaptation. Thus, it can resolve the main challenges for next-generation wireless networks, including higher energy and spectral efficiency, higher cost efficiency, lower latency, as well as massive connectivity. However, with multi-entity collaboration and enlarged network sizes in dense Cloud-RANs, unique issues arise in green networking and large-scale computing. Fundamental methodologies and algorithms need to be developed to address these challenges by exploiting the problem structures, e.g., group sparsity and low-rankness.

In dense Cloud-RANs, due to the multi-entity network collaboration, network power consumption originated from the radio access network and the fronthaul network becomes huge. To design a green Cloud-RAN, a holistic approach is required for network power minimization. By exploiting the spatial and temporal mobile data traffic variation, we develop a group sparse beamforming framework to minimize the network power consumption by enabling network adaptation. This is achieved by adaptively selecting active remote radio heads (RRHs) and the corresponding fronthaul links via controlling the group sparsity structure of the aggregative beamforming vector at all the RRHs, thereby adapting to the spatial and temporal mobile data traffic fluctuations. In particular, the group sparsity structure is induced by minimizing the mixed ℓ_1/ℓ_2-norm of the aggregative beamforming vector.

To further demonstrate the power of the group sparse beamforming framework, a more challenging scenario with multicast transmission and imperfect channel state information (CSI) is also investigated. In particular, we present the PhaseLift and semidefinite relaxation techniques to convexify the robust non-convex quadratic quality-of-service (QoS) constraints. A smoothed ℓ_p-minimization approach is further proposed to induce the group-sparsity structure in the aggregative multicast beamforming vector, which indicates those RRHs that can be switched off. To solve the resultant non-convex group-sparsity inducing optimization problem, an iterative reweighted-ℓ_2 algorithm is then proposed based on the principle of the majorization-minimization (MM) algorithm.

As the design problem sizes scale up with the network size in dense Cloud-RANs, we demonstrate that it is critical to take the inherent characteristics of wireless channels into consideration to reduce the CSI acquisition overhead, while new optimization methods will be needed. We first present a low rank matrix completion approach via Riemannian pursuit to maximize the achievable degrees of freedom (DoF) only based on the network topology information without the knowledge of CSI at transmitters. Furthermore, to deal with the uncertainty in the available CSI, a chance-constrained programming based stochastic coordinated beamforming framework is proposed. In particular, a novel stochastic difference-of-convex (DC) programming algorithm for the resultant highly intractable chance constrained programming problem is developed with optimality guarantee, while all the previous algorithms can only find feasible but sub-optimal solutions.

The last part of the thesis is devoted to dealing with the computing issues in dense Cloud-RANs. This is motivated by the fact that the design problems in dense Cloud-RANs are entering a new era characterized by a high dimension and/or a large number of constraints as well as complicated structures. It is thus critical to exploit unique structures of the design problems, while convex optimization will serve as a powerful tool for such purposes. In particular, to enable scalable network densification and cooperation, we develop a two-stage framework to solve the general large-scale convex optimization problems. This is achieved by equivalently transforming the original convex problem into the standard conic optimization problem by the matrix stuffing technique. The operator splitting method, namely, the alternating direction method of multipliers (ADMM), is further adopted to solve the resultant large-scale self-dual embedding of the transformed cone programming problem. This enables the capability of parallel computing and infeasibility detection.

In summary, the central theme of this thesis is developing scalable sparse optimization methodologies and algorithms to address the networking and computing issues for network densification in Cloud-RANs.

Chapter 1

Introduction

In this chapter we introduce the network architecture of Cloud-RAN to densify the radio access networks via radio resource and computation resource coordination, thereby boosting the network capacity. The main advantages, the design and operating challenges, as well as the main contributions of this thesis will be presented.

1.1 Motivations

With the dramatic increase of smart mobile devices, and diversified wireless applications propelled by the advent of mobile social networks and Internet of Things (IoT), we are in an era of mobile data deluge. In particular, mobile data traffic has recently been doubling every year, which implies an astounding 1000 times increase for 5G networks by 2020 [1]. Furthermore, new wireless applications bring new service requirements. For instance, intensive data services will be needed in crowed places as stadiums and in densely populated metropolitan areas, while IoT applications call for scalable connectivity with diversified quality-of-service (QoS) requirements. With extremely low latency, accompanied by high availability, reliability and security, Tactile Internet [2] will enable numerous new services and allow for new experiences.

To meet these key requirements, network densification becomes a dominated theme by deploying more antennas or radio access units. In particular, massive multiple-input and multiple-output (MIMO) wireless system [3] is regarded as an effective and promising approach for multi-cell multi-user MIMO implementation by equipping base stations (BSs) with excessive antennas (up to a few hundred antennas), thereby offering unprecedented spectral efficiency and energy efficiency. With the additional antennas, this can be achieved even with low-complexity beamforming techniques (e.g., zero-forcing beamforming), while inter-cell interference can be effectively eliminated, assuming that the full channel state information (CSI) is available at BSs.

However, the performance of massive MIMO is limited by correlated scattering with the antenna spacing constraints, which also brings high deployment cost to maintain the minimum spacing [4]. Furthermore, the prerequisite of the massive CSI raises the concern about the training overhead, which is proportional to the BS antenna size in frequency-division duplexing (FDD) massive MIMO. In the alternative time-division duplexing (TDD) mode, the inevitable reuse of the same pilot in neighboring cells causes pilot contamination, which will seriously degrade the CSI accuracy and has been regarded as a main bottleneck of TDD massive MIMO [5]. Therefore, the capacity gain of the massive MIMO will be fundamentally limited. Other approach needs to be further exploited to meet the ever increasing mobile data traffic demands.

Heterogeneous and small cell networks (HetSNets) are regarded as the other promising way to address the challenges for mobile data traffic explosion. This is achieved by deploying more radio access points to exploit the spatial reuse, thereby bringing the radio access network closer to the mobile users. Meanwhile, as stated in [6], placing radio access points based on the traffic demand is an effective way for compensating path-loss, resulting in energy efficient cellular networks. However, efficient interference management is challenging for dense small-cell networks. Moreover, deploying more and more small-cells will cause significant cost and operating challenges for operators.

Network cooperation [7] is an enabling technique to manage the interference in the wireless networks, thereby fully exploiting the benefits of the network densification. The cooperative gain is achieved by sharing the user data and CSI among multiple BSs to mitigate the interference efficiently. However, the full cooperation gain is limited by the overhead of the orthogonal pilot-assisted channel estimation for uplink transmission in large-scale cooperative cellular networks [8]. In particular, Huh *et al.* [9] quantified the downlink training overhead for large-scale network MIMO, which is regarded as the system overhead bottleneck even if the uplink feedback overhead is ignored. Furthermore, with the additional limited computing capability at BSs, the cooperative gain is fundamentally limited by the overhead of computing and CSI acquisition in dense networks.

To address the above limitations of the existing solutions, a paradigm shift is thus needed in radio access networks to achieve the benefit of dense networks, including the magnified interference issue, the high capital expenditure (CAPEX) and operating expenditure (OPEX), and mobility management, etc. In particular, a holistic approach to deploy and manage dense networks is required, and efficient multi-tier collaboration should be supported.

The cloud radio access network (Cloud-RAN) [10, 11] is a promising centralized radio access technology to address the key challenges towards network densification by leveraging recent advances in cloud-computing technology [12]. With the baseband unit (BBU) implemented in the centralized computation resources located at the cloud data center (i.e., BBU pool), it can significantly reduce the complexity and cost of the radio access units, i.e., the CAPEX will be reduced. Moreover, this architecture makes it easy to upgrade, as only software upgrade at the cloud center is needed. It is also easily scalable, as more low-cost remote radio heads (RRHs) can be handily deployed to increase the network capacity. Meanwhile, intra-tier and inter-tier interference can be effectively mitigated by centralized signal processing and coordination at the BBU pool. Furthermore, with elastic network reconfiguration and adaptation, the operation efficiency of the Cloud-RAN can be significantly improved. For example, by adaptively switching on/off RRHs, and adjusting computing resources at the BBU pool, the network can be well adapted to the spatial and temporal traffic fluctuations.

In summary, network densification and cooperation in Cloud-RANs by deploying more RRHs and collaboration among RRHs at the radio access network and computation units at the BBU pool, provide a disruptive way to achieve the higher energy efficiency and spectral efficiency, higher cost efficiency, and scalable connectivity.

1.2 Network Architecture and Research Challenges

In this section, we will introduce the main entities of Cloud-RAN, including the cloud data center, the radio access network, and the mobile hauling network, followed by an overview of new research challenges.

1.2.1 Network Architecture

Cloud-RAN is a disruptive technology that takes advantage of recent advances in cloud-computing to revolutionize next-generation wireless networks. Its architecture is shown in

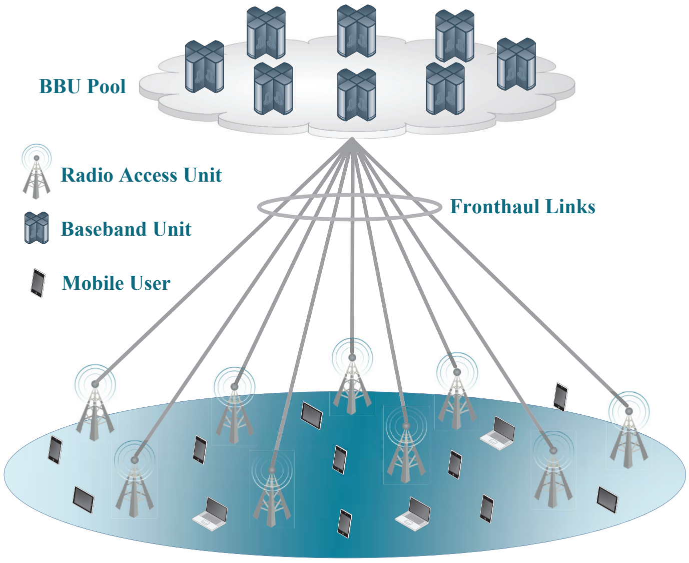

Figure 1.1: The architecture of Cloud-RAN, in which, all the RRHs are connected to a BBU pool through high-capacity and low-latency optical fronthaul links.

Fig. 1.1. Specifically, the BBU pool serves as a central cloud infrastructure for the dense radio access network consisting of large number of RRHs. The key advantage of the Cloud-RAN lies in the centralized coordination at the BBU pool, supported by the fronthaul network to transfer the information to and from different RRHs. In the following, we will introduce the main functionality of each entity in Cloud-RAN, as well as some deployment issues.

1.2.1.1 Cloud Data Center

The cloud data center (i.e., the BBU pool) consists of shared and reconfigurable computation and storage resources. Thanks to such a shared hardware platform, both the CAPEX (e.g. via low-cost site construction) and OPEX (e.g. via centralized cooling), as well as the management effort, can be significantly reduced. Besides performing basic baseband digital signal processing for transmission and reception, the cloud data center can also provide cloud-computing functionalities, such as on-demand services via virtualization with multiple virtual machines and resource pooling, and parallel computing for scalable algorithm implementation.

The centralized signal processing enabled by the cloud data center is essential for the performance gains of Cloud-RAN. Specifically, with densely deployed RRHs, by applying advanced signal processing algorithms in the computationally powerful cloud data center, large-scale cooperation can be achieved, thereby improving both spectral efficiency and energy efficiency. Moreover, with centralized coordination, effective dynamic resource allocation can be provided to smooth out spatial and temporal traffic fluctuations.

1.2.1.2 Radio Access Network

In Cloud-RANs, each RRH only consists of a passband signal processor, an amplifier and an A/D converter to support basic transmission and reception functionality, while the baseband signal processing is carried in the cloud data center. The data transmitted between the cloud data center and RRHs are typically oversampled realtime I/Q digitalized baseband streams in the order of Gbps [10]. For such access nodes, the mobile hauling network that provides high-capacity and low-latency connection to the cloud data center is usually called the mobile fronthaul network. The typical requirements of fronthaul links are: link capacity (110 Gb/s), latency (0.1 ms), and distance (110 km) [11].

1.2.1.3 Mobile Fronthaul Network

A key success of Cloud-RANs is to transfer the data traffic between the cloud data center and the radio access network by the mobile fronthaul network. As the capacity of each fronthaul link will affect both the network performance and deployment cost, it should be carefully picked.

Specifically, for the mobile fronthaul network connecting RRHs and the cloud data center, there is stringent requirements on latency and synchronization, as well as low jitter and error tolerance [10]. Both the low-cost wavelength-division multiplexing passive optical network (WDM-PON) and orthogonal frequency-division multiple access passive optical network (OFDMA-PON) are promising candidates to provide high-capacity and low-latency fronthaul solutions [13]. Furthermore, to alleviate the performance bottleneck due to the limited fronthaul capacity, a major effort is being made to develop advanced fronthaul compression schemes [14].

1.2.2 Research Challenges

The new architecture of Cloud-RAN will bring new opportunities as well as new design challenges, mainly due to the enlarged problem size as the design parameters and the required side information grow substantially. In this thesis, we will provide a holistic viewpoint for green Cloud- RAN design via convex optimization. It has been well recognized that convex optimization provides an indispensable set of tools for designing wireless communication systems [15–17], e.g. coordinated beamforming [18], power control [19], user admission control [20], as well as data routing and flow control [21]. The main reason for the success of convex optimization lies in its capability of flexible formulations, efficient globally optimal algorithms, e.g. the interior-point method [22], and the ability to leverage convex analysis to explore the solution structure, e.g. the uplink-downlink duality in the multiuser beamforming problem [18]. However, in dense Cloud-RANs, with its complex architecture, as well high-dimensional optimization variables and a large number of parameters, new challenges arise.

In this thesis, we will present new convex optimization methods based on sparse optimization to address the main design challenges for dense green Cloud-RAN. We will demonstrate the strength of convex optimization by developing new methodologies for the key design problems in green Cloud-RAN, i.e., network power minimization, diversified services, CSI acquisition and uncertainty, and large-scale convex optimization.

1.2.2.1 Green Cloud-RAN Design

In dense Cloud-RAN with a large number of RRHs, it is critical to select RRHs to adapt to the temporal and spatial data dynamics, thereby improving the energy efficiency and operating efficiency. In this way, we can reduce the power consumption of both the radio access points and fronthaul links, as well as reduce the signaling overhead. However, to enable such adaptation, the new challenge comes from the composite design variables, which consist of both discrete and continuous variables for RRH selection and coordinated beamforming design, respectively. This often yields a mixed integer nonlinear programming problem and is NP-hard.

To efficiently solve such problems, in Chapter 2, we will develop a group sparse beamforming framework to minimize the network power consumption by adaptively selecting active RRHs via controlling the group-sparsity structure of the beamforming vectors.

1.2.2.2 Diversified Services

Although network adaption by selecting the active RRHs provides a promising way to minimize the network power consumption in green Cloud-RAN design, exploiting the benefits of integrating diversified unicast and multicast services [23] has been well recognized as a promising way to further improve the energy efficiency. Therefore, multicast service should be incorporated in green Cloud-RAN design. Moreover, in practice, the available CSI may be imperfect due to channel estimation errors.

To design a green Cloud-RAN with multicast transmission and imperfect CSI, in Chapter 3, we will present a holistic approach to enable network adaptation for network power minimization.

1.2.2.3 CSI Acquisition and Uncertainty

CSI is essential for various cooperation strategies design in the dense Cloud-RAN, but its acquisition becomes challenging as a large number of RRHs are involved in cooperation. To address the channel estimation challenge with limited radio resources, a major effort is being made to design efficient CSI acquisition strategies by exploiting unique structures of wireless channels, e.g., partial connectivity in wireless networks [24]. To resolve the CSI challenge for Cloud-RAN, we propose a novel CSI acquisition method, called compressive CSI acquisition. This new method can effectively reduce the CSI signaling overhead by obtaining instantaneous coefficients of only a subset of all the channel links. As a result, the BBU pool will obtain mixed CSI consisting of instantaneous values of some links and statistical CSI for the others.

To effectively exploit the available mixed CSI (i.e., partial and imperfect), in Chapter 4, we shall present a new beamforming approach based on chance-constrained programming to alleviate the performance degradation due to the CSI uncertainty. We also present a low-rank matrix completion approach for the topological interference management only based on the network partial connectivity pattern [25] with the minimal CSI requirements.

1.2.2.4 Network Densification

In dense Cloud-RAN, the cloud data center will typically support hundreds of RRHs [10], and thus all the optimization algorithms need to scale to large problem sizes. However, solving convex quadratic programs has cubic complexity using the interior-point method. Moreover, a sequence of convex feasibility problems need to be solved for lots of design problems, e.g., for the RRH selection in the network power minimization problem. However, most existing custom algorithms, e.g., the ADMM based algorithms [26] and the uplink-downlink duality approach [27], cannot provide the certificates of infeasibility. Thus, effective feasibility detection should be embedded in the developed algorithms.

To resolve these challenges, in Chapter 5, we will present a two-stage approach framework to solve general large-scale convex optimization problems by leveraging the cloud-computing environment in the cloud data center.

1.3 Thesis Goals and Contributions

This focus of this thesis is to develop algorithmic approaches to deal with networking and computing issues in dense Cloud-RANs. Using unicast service with perfect CSI as our staring point, we develop a group sparse beamforming framework for green Cloud-RAN design, thereby minimizing the network power consumption. This is achieved by adaptively selecting active RRHs via controlling the group-sparsity structure of the beamforming vectors. Building on this insight, we develop algorithms based on convex optimization for more general scenarios with CSI uncertainty and multicast services, yielding a unified and flexible sparse optimization framework for green Cloud-RAN design. To address the massive CSI acquisition challenges in dense Cloud-RANs, we further present a low-rank matrix completion approach and a chance constrained optimization method to manage the interference only based on the network topology information and to deal with the CSI uncertainty, respectively. To enable scalable algorithm design in dense Cloud-RANs, we further propose a two-stage framework to solve general large-scale convex optimization problems, which is amenable to parallel implementation in the cloud data center.

In particular, the goals of this thesis are to

- Develop a holistic approach for green Cloud-RAN design by minimizing the network power consumption.

- Develop efficient CSI acquisition and exploration methods to resolve the massive CSI challenges in dense Cloud-RANs.

- Develop scalable and parallel optimization algorithms to address the computing issues in dense Cloud-RANs.

Specifically, the main contributions of this thesis are as follows:

1. We propose a group sparse optimization framework to design a green Cloud-RAN,

which is formulated as a joint RRH selection and transmit power minimization beamforming problem. To efficiently solve the group sparse beamforming problem with the *non-convex combinatorial objective function*, a weighted mixed ℓ_1/ℓ_2-norm minimization algorithm is proposed to induce the group-sparsity of beamformers, thereby guiding the RRH selection.

2. We propose a smoothed ℓ_p-minimization framework to design a green Cloud-RAN by taking the imperfect CSI and multicast service into consideration. To solve the robust multicast group sparse beamforming problem with the *non-convex combinatorial objective function and non-convex quadratic QoS constraints*, we develop an iterative reweighted-ℓ_2 algorithm based on the majorization-minimization (MM) method.

3. We develop a compressive CSI acquisition method to reduce the CSI signaling overhead in dense Cloud-RANs, followed by a generic stochastic coordinated beamforming framework to deal with stochastic CSI uncertainty in the available CSI. To solve the stochastic coordinated beamforming problem with the *non-convex probabilistic QoS constraints*, we propose a novel stochastic DC (difference-of-convex) programming algorithm with optimality guarantee, which can serve as the benchmark for evaluating heuristic and sub-optimal algorithms. A flexible low-rank matrix completion approach is presented to investigate the topological interference management problem for any network topology.

4. We develop a novel two-stage approach to solve general large-scale convex optimization problems for dense Cloud-RAN, which can effectively detect infeasibility and enjoy modeling flexibility. This is achieved by transforming the original convex problem into a standard cone programming form via matrix stuffing technique and then solving the homogeneous self-dual embedding of the primal-dual pair of the standard form using the operator splitting method.

A common theme in all of our developed algorithms is that they exploit the problem structures (e.g., group sparsity and low-rankness) to address the design challenges in terms of non-convexity and scalability in dense green Cloud-RAN. Specifically, in the group sparse beamforming case, this comes from the observation that network power minimization can be achieved by adaptively selecting active RRHs via controlling the group-sparsity structure of the beamforming vector. In the robust multicast group sparse beamforming case, this consists of observing that the smoothed ℓ_p-norm can promote sparsity by introducing quadratic forms in the beamforming vectors. In the stochastic coordinated beamforming case, this comes from the key observation that the highly intractable probability constraint can be equivalently reformulated as a stochastic DC constraint. In the low-rank matrix completion for topological interference management case, this comes from the key observation that the achievable DoF equals the inverse of the rank of the design matrix. In large-scale convex optimization case, this comes in the salient feature of the Cartesian product of closed convex cones (e.g., second-order cone and semidefinite cone) in the transformed standard cone programming form, which can be exploited to enable parallel and scalable computing.

1.4 Organization and Published Materials

The remainder of this thesis is organized as follows.

Chapter 2 This chapter investigates the network power minimization problem to design a green Cloud-RAN based on the assumptions of unicast service and perfectly instantaneous CSI. The material in this chapter has been presented in part in [J7, C5].

Chapter 3 This chapter studies the green Cloud-RAN design problem in multicast transmission setting with imperfect CSI. The material in this chapter has been presented in part in [J2, J3, C2].

Chapter 4 This chapter deals with the CSI acquisition overhead and the stochastic CSI uncertainty in dense Cloud-RANs. The material in this chapter has been presented in part in [J1, J6, C1, C4].

Chapter 5 This chapter presents a unified two-stage approach to solve general large-scale convex optimization problems in dense Cloud-RANs. The material in this chapter has been presented in part in [J4, C3].

Chapter 6 This chapter closes this thesis with a brief summary and discussions of future search directions.

In particular, the underlying idea of scalable sparse optimization in ultra-dense green Cloud-RAN in this chapter has been presented in part in [J5].

Journal Articles

J1. **Y. Shi**, J. Zhang, and K. B. Letaief, “Low-rank matrix completion for topological interference management by Riemannian pursuit,” submitted to *IEEE Trans. Wireless Commun.*, Jul. 2015.

J2. **Y. Shi**, J. Cheng, J. Zhang, B. Bai, W. Chen and K. B. Letaief, “Smoothed L_p-minimization for green Cloud-RAN with user admission control,” submitted to *IEEE J. Select. Areas Commun.*, under second-round revision.

J3. **Y. Shi**, J. Zhang, and K. B. Letaief, “Robust group sparse beamforming for multicast green Cloud-RAN with imperfect CSI,” *IEEE Trans. Signal Process.*, vol. 63, no. 17, pp. 4647-4659, Sept. 2015.

J4. **Y. Shi**, J. Zhang, B. O’Donoghue, and K. B. Letaief, “Large-scale convex optimization for dense wireless cooperative networks,” *IEEE Trans. Signal Process.*, vol. 63, no. 18, pp. 4729- 4743, Sept. 2015.

J5. **Y. Shi**, J. Zhang, K. B. Letaief, B. Bai and W. Chen,“Large-scale convex optimization for ultra-dense Cloud-RAN,” *IEEE Wireless Commun. Mag.*, pp. 84-91, Jun. 2015.

J6. **Y. Shi**, J. Zhang, and K. B. Letaief, “Optimal stochastic coordinated beamforming for wireless cooperative networks with CSI uncertainty,” *IEEE Trans. Signal Process.*, vol. 63,, no. 4, pp. 960-973, Feb. 2015.

J7. **Y. Shi**, J. Zhang, and K. B. Letaief, “Group sparse beamforming for green Cloud-RAN,” *IEEE Trans. Wireless Commun.*, vol. 13, no. 5, pp. 2809-2823, May 2014.

Conference Papers

C1. **Y. Shi**, J. Zhang, and K. B. Letaief, “Low-rank matrix completion via Riemannian pursuit for topological interference management,” in *Proc. IEEE Int. Symp. Inform. Theory (ISIT)*, Hong Kong, Jun. 2015.

C2. J. Cheng, **Y. Shi**, B. Bai, W. Chen, J. Zhang, and K. B. Letaief, “Group sparse beamforming for multicast green Cloud-RAN via parallel semidefinite programming,” in *Proc. IEEE Int. Conf. Commun. (ICC)*, London, UK, Jun. 2015.

C3. **<u>Y. Shi</u>**, J. Zhang, and K. B. Letaief, "Scalable coordinated beamforming for dense wireless cooperative networks," in *Proc. IEEE Globecom*, Austin, TX, Dec. 2014.

C4. **<u>Y. Shi</u>**, J. Zhang, and K. B. Letaief, "CSI overhead reduction with stochastic beamforming for cloud radio access networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Sydney, Australia, Jun. 2014.

C5. **<u>Y. Shi</u>**, J. Zhang, and K. B. Letaief, "Group sparse beamforming for green cloud radio access networks," in *Proc. IEEE Globecom*, Atlanta, GA, Dec. 2013.

C6. **<u>Y. Shi</u>**, J. Zhang, and K. B. Letaief, "Coordinated relay beamforming for amplify-and-forward two-hop interference networks," in *Proc. IEEE Globecom*, Anaheim, CA, Dec. 2012.

Chapter 2

Group Sparse Beamforming for Green Cloud-RAN

In this chapter we present a new framework to design a green Cloud-RAN, which is formulated as a joint RRH and fronthaul link selection and transmit power minimization beamforming problem. To efficiently solve this problem, we first propose a greedy selection algorithm in Section 2.4, which is shown to provide near-optimal performance. To further reduce the complexity, in Section 2.5, a novel group sparse beamforming method is proposed by inducing the group-sparsity of beamformers using the weighted ℓ_1/ℓ_2-norm minimization, where the group sparsity pattern indicates those RRHs that can be switched off. These results provide new insights for other network performance optimization problems consist of discrete and continuous variables. Proofs and preliminaries are relegated to Appendix A. The material in this chapter has been presented in part in [28, 29].

2.1 Introduction

Mobile data traffic has been growing enormously in recent years, and it is expected that cellular networks will have to offer a 1000x increase in capacity in the following decade to meet this demand [4]. Massive MIMO [3] and heterogeneous and small cell networks (HetSNets) [4] are regarded as two most promising approaches to achieve this goal. By deploying a large number of antennas at each base station (BS), massive MIMO can exploit spatial multiplexing gain in a large scale and also improve energy efficiency. However, the performance of massive MIMO is limited by correlated scattering with the antenna spacing constraints, which also brings high deployment cost to maintain the minimum spacing [4]. HetSNets exploit the spatial reuse by deploying more and more access points (APs). Meanwhile, as stated in [30], placing APs based on the traffic demand is an effective way for compensating path-loss, resulting in energy efficient cellular networks. However, efficient interference management is challenging for dense small-cell networks. Moreover, deploying more and more small-cells will cause significant cost and operating challenges for operators.

Cloud radio access network (Cloud-RAN) has recently been proposed as a promising network architecture to unify the above two technologies in order to jointly manage the interference (via coordinated multiple-point process (CoMP)), increase network capacity and energy efficiency (via network densification), and reduce both the network capital expenditure (CAPEX) and operating expense (OPEX) (by moving baseband processing to the baseband unit (BBU) pool) [10, 31]. A large-scale distributed cooperative MIMO system will thus be formed. Cloud-RAN can therefore be regarded as the ultimate solution to the "spectrum crunch" problem of cellular networks.

There are three key components in a Cloud-RAN: (i) a pool of BBUs in a *cloud data center*, supported by the real-time virtualization and high performance processors, where all the baseband processing is performed; (ii) a high-bandwidth low-latency optical fronthaul network connecting the BBU pool and the remote radio heads (RRHs); and (iii) distributed transmission/reception points (i.e., RRHs). The key feature of Cloud-RAN is that RRHs and BBUs are separated, resulting a centralized BBU pool, which enables efficient cooperation of the transmission/reception among different RRHs. As a result, significant performance improvements through joint scheduling and joint signal processing such as coordinated beamforming or multi-cell processing [7] can be achieved. With efficient interference suppression, a network of RRHs with a very high density can be deployed. This will also reduce the communication distance to the mobile terminals and can thus significantly reduce the transmission power. Moreover, as baseband signal processing is shifted to the BBU pool, RRHs only need to support basic transmission/reception functionality, which further reduces their energy consumption and deployment cost.

The new architecture of Cloud-RAN also indicates a paradigm shift in the network design, which causes some technical challenges for implementation. For instance, as the data transmitted between the RRHs and the BBU pool is typically oversampled real-time I/Q digital data streams in the order of Gbps, high-bandwidth optical fronthaul links with low latency will be needed. To support CoMP and enable computing resource sharing among BBUs, new virtualization technologies need to be developed to distribute or group the BBUs into a centralized entity [10]. Another important aspect is the energy efficiency consideration, due to the increased power consumption of a large number of RRHs and also of the fronthaul links.

Conventionally, the fronthaul network (i.e., backhaul links between the core network and base stations (BSs)) power consumption can be ignored as it is negligible compared to the power consumption of macro BSs. Therefore, all the previous works investigating the energy efficiency of cellular networks only consider the BS power consumption [18, 32]. Recently, the impact of the backhaul power consumption in cellular networks was investigated in [33], where it was shown through simulations that the backhaul power consumption will affect the energy efficiency of different cellular network deployment scenarios. Subsequently, Rao *et al.* in [34] investigated the spectral efficiency and energy efficiency tradeoff in homogeneous cellular networks when taking the backhaul power consumption into consideration.

In Cloud-RAN, the fronthaul network power consumption will have a more significant impact on the network energy efficiency. Hence, allowing the fronthaul links and the corresponding RRHs to support the sleep mode will be essential to reduce the network power consumption for the Cloud-RAN. Moreover, with the spatial and temporal variation of the mobile traffic, it would be feasible to switch off some RRHs while still maintaining the quality of service (QoS) requirements. It will be also practical to implement such an idea in the Cloud-RAN with the help of centralized signal processing at the BBU pool. As energy efficiency is one of the major objectives for future cellular networks [31], in this chapter we will focus on the design of green Cloud-RAN by jointly considering the power consumption of the fronthaul network and RRHs.

2.1.1 Contributions

The main objective of this chapter is to minimize the network power consumption of Cloud-RAN, including the fronthaul network and radio access network power consumption, with a QoS constraint at each user. Specifically, we formulate the design problem as a joint RRH selection and power minimization beamforming problem, where the fronthaul network power consumption is determined by the set of active RRHs, while the transmit power consumption of the active RRHs is minimized through coordinated beamforming. This is a mixed-integer non-linear programming (MINLP) problem, which is NP-hard. We will focus on designing low-complexity algorithms for practical implementation. The major contributions of the chapter are summarized as follows:

1. We formulate the network power consumption minimization problem for the Cloud-RAN by enabling both the fronthaul links and RRHs to support the sleep mode. In particular, we provide a group sparse beamforming (GSBF) formulation of the design problem, which assists the problem analysis and algorithm design.

2. We first propose a greedy selection (GS) algorithm, which selects one RRH to switch off at each step. It turns out that the RRH selection rule is critical, and we propose to switch off the RRH that *maximizes the reduction in the network power consumption* at each step. From the simulations, the proposed GS algorithm often yields optimal or near-optimal solutions, but its complexity may still be prohibitive for a large-sized network.

3. To further reduce the complexity, we propose a three-stage group sparse beamforming (GSBF) framework, by adopting the weighted mixed ℓ_1/ℓ_p-norm to induce the group sparsity for the beamformers. In contrast to all the previous works applying the mixed ℓ_1/ℓ_p-norm to induce group sparsity, we exploit the additional prior information (i.e., fronthaul link power consumption, power amplifier efficiency, and instantaneous effective channel gains) to design the weights for different beamformer coefficient groups, resulting in a significant performance gain. Two GSBF algorithms with different complexities are proposed: namely, a bi-section GSBF algorithm and an iterative GSBF algorithm.

4. We shall show that the GS algorithm always provides near-optimal performance. Hence, it would be a good option if the number of RRHs is relatively small, such as in clustered deployment. With a very low computational complexity, the bi-section GSBF algorithm is an attractive option for a large-scale Cloud-RAN. The iterative GSBF algorithm provides a good tradeoff between the complexity and performance, which makes it a good candidate for a medium-size network.

2.1.2 Related Works

A main design tool applied in this chapter is optimization with the group sparsity induced norm. With the recent theoretical breakthrough in compressed sensing [35, 36], the sparsity patterns in different applications in signal processing and communications have been exploited for more efficient system design, e.g., for pilot aided sparse channel estimation [37].

The sparsity inducing norms have been widely applied in high-dimensional statistics, signal processing, and machine learning in the last decade [38]. The ℓ_1-norm regularization has been successfully applied in compressed sensing [35, 36]. More recently, mixed ℓ_1/ℓ_p-norms are widely investigated in the case where some variables forming a group will be selected or removed simultaneously, where the mixed ℓ_1/ℓ_2-norm [39] and mixed ℓ_1/ℓ_∞-norm [40] are two commonly used ones to induce group sparsity for their computational and analytical convenience.

In Cloud-RAN, one RRH will be switched off only when all the coefficients in its beamformer are set to zeros. In other words, all the coefficients in the beamformer at one RRH should be selected or ignored simultaneously, which requires group sparsity rather than individual sparsity for the coefficients as commonly used in compressed sensing. In this chapter, we will adopt the mixed ℓ_1/ℓ_p-norm to promote group sparsity for the beamformers instead of ℓ_1-norm, which only promotes individual sparsity. Recently, there are some works [41–43] adopting the mixed ℓ_1/ℓ_p-norm to induce group-sparsity in a large-scale cooperative wireless cellular network. Specifically, Hong *et al.* [41] adopted the mixed ℓ_1/ℓ_2-norm and Zhao *et al.* [42] used the ℓ_2-norm to induce the group sparsity of the beamformers, which reduce the amount of the shared user data among different BSs. The squared mixed ℓ_1/ℓ_∞-norm was investigated in [43] for antenna selection.

All of the above works simply adopted the un-weighted mixed ℓ_1/ℓ_p-norms to induce group-sparsity, in which, no prior information of the unknown signal is assumed other than the fact that it is sufficiently sparse. By exploiting the prior information in terms of system parameters, the weights for different beamformer coefficient groups can be more rigorously designed and performance can be enhanced. We demonstrate through simulations that the proposed three-stage GSBF framework, which is based on the weighted mixed ℓ_1/ℓ_p-norm minimization, outperforms the conventional unweighted mixed ℓ_1/ℓ_p-norm minimization based algorithms substantially.

2.1.3 Organization

The remainder of the chapter is organized as follows. Section 2.2 presents the system and power model. In section 2.3, the network power consumption minimization problem is formulated, followed by some analysis. Section 2.4 presents the GS algorithm, which yields near-optimal solutions. The three-stage GSBF framework is presented in Section 2.5. Simulation results will be presented in Section 2.6. Finally, conclusions and discussions are presented in Section 2.7.

Notations: $\|\cdot\|_{\ell_p}$ is the ℓ_p-norm. Boldface lower case and upper case letters represent vectors and matrices, respectively. $(\cdot)^T$, $(\cdot)^\dagger$, $(\cdot)^{\mathsf{H}}$ and $\text{Tr}(\cdot)$ denote the transpose, conjugate, Hermitian and trace operators, respectively. $\Re(\cdot)$ denotes the real part.

2.2 System and Power Model

2.2.1 System Model

We consider a Cloud-RAN with L remote radio heads (RRHs), where the l-th RRH is equipped with N_l antennas, and K single-antenna mobile users (MUs), as shown in Fig. 1.1. In this network architecture, all the base band units (BBUs) are moved into a single BBU pool, creating a set of shared processing resources, and enabling efficient interference management and mobility management. With the baseband signal processing functionality migrated to the BBU pool, the RRHs can be deployed in a large scale with low-cost. The BBU pool is connected to the RRHs using the common public radio interface (CPRI) fronthaul technology via a high-bandwidth, low-latency optical fronthaul network, i.e., fronthaul network [10]. In order to enable full cooperation among RRHs, it is assumed that all the user data are routed to the BBU pool from the core network through the backhaul links [10], i.e., all users can access all the RRHs. The digitized baseband complex inphase (I) and quadrature (Q) samples of the radio signals are transported over the fronthaul links between the BBUs and RRHs. The key technical and economic issue of the Cloud-RAN is that this architecture requires significant fronthaul network resources. As the focus of this chapter is on network power consumption, we will assume all the fronthaul links have sufficiently high capacity and negligible latency[1].

Due to the high density of RRHs and the joint transmission among them, the energy used

[1] The impact of limited-capacity fronthaul links on compression in Cloud-RAN was recently investigated in [44, 45], and its impact in our setting is left to future work.

for signal transmission will be reduced significantly. However, the power consumption of the fronthaul network becomes enormous and cannot be ignored. Therefore, it is highly desirable to switch off some fronthaul links and the corresponding RRHs to reduce the network power consumption based on the data traffic requirements, which forms the main theme of this work.

Let $\mathcal{L} = \{1, ..., L\}$ denote the set of RRH indices, $\mathcal{A} \subseteq \mathcal{L}$ denote the active RRH set, $\mathcal{Z}$ denote the inactive RRH set with $\mathcal{A} \cup \mathcal{Z} = \mathcal{L}$, and $\mathcal{S} = \{1, ..., K\}$ denote the index set of scheduled users. In a beamforming design framework, the baseband transmit signals are of the form:

$$\mathbf{x}_l = \sum_{k=1}^{K} \mathbf{w}_{lk} s_k, \forall l \in \mathcal{A}, \tag{2.2.1}$$

where s_k is a complex scalar denoting the data symbol for user k and $\mathbf{w}_{lk} \in \mathbb{C}^{N_l}$ is the beamforming vector at RRH l for user k. Without loss of generality, we assume that $E[|s_k|^2] = 1$ and s_k's are independent with each other. The baseband signals $\mathbf{x}_l$'s will be transmitted to the corresponding RRHs, but not the data information s_k's [10, 45]. The baseband received signal at user k is given by

$$y_k = \sum_{l \in \mathcal{A}} \mathbf{h}_{kl}^{\mathsf{H}} \mathbf{w}_{lk} s_k + \sum_{i \neq k} \sum_{l \in \mathcal{A}} \mathbf{h}_{kl}^{\mathsf{H}} \mathbf{w}_{li} s_i + z_k, k \in \mathcal{S}, \tag{2.2.2}$$

where $\mathbf{h}_{kl} \in \mathbb{C}^{N_l}$ is the channel vector from RRH l to user k, and $z_k \sim \mathcal{CN}(0, \sigma_k^2)$ is the additive Gaussian noise.

We assume that all the users are employing single user detection (i.e., treating interference as noise), so that they can use the receivers with a low-complexity and energy-efficient structure. Moreover, in the low interference region, treating interference as noise can be optimal [46]. The corresponding signal-to-interference-plus-noise ratio (SINR) for user k is hence given by

$$\mathrm{SINR}_k = \frac{|\sum_{l \in \mathcal{A}} \mathbf{h}_{kl}^{\mathsf{H}} \mathbf{w}_{lk}|^2}{\sum_{i \neq k} |\sum_{l \in \mathcal{A}} \mathbf{h}_{kl}^{\mathsf{H}} \mathbf{w}_{li}|^2 + \sigma_k^2}, \forall k \in \mathcal{S}. \tag{2.2.3}$$

Each RRH has its own transmit power constraint

$$\sum_{k=1}^{K} \|\mathbf{w}_{lk}\|_{\ell_2}^2 \leq P_l, \forall l \in \mathcal{A}. \tag{2.2.4}$$

2.2.2 Power Model

The network power model is critical for the investigation of the energy efficiency of Cloud-RAN, which is described as follows.

2.2.2.1 RRH Power Consumption Model

We will adopt the following empirical linear model [47] for the power consumption of an RRH:

$$P_l^{\mathrm{rrh}} = \begin{cases} P_{a,l}^{\mathrm{rrh}} + \frac{1}{\eta_l} P_l^{\mathrm{out}}, & \text{if } P_l^{\mathrm{out}} > 0, \\ P_{s,l}^{\mathrm{rrh}}, & \text{if } P_l^{\mathrm{out}} = 0. \end{cases} \tag{2.2.5}$$

where $P_{a,l}^{\mathrm{rrh}}$ is the active power consumption, which depends on the number of antennas N_l, $P_{s,l}^{\mathrm{rrh}}$ is the power consumption in the sleep mode, P_l^{out} is the transmit power, and η_l is the drain efficiency of the radio frequency (RF) power amplifier. For the Pico-BS, the typical values are $P_{a,l}^{\mathrm{rrh}} = 6.8W$, $P_{s,l}^{\mathrm{rrh}} = 4.3W$, and $\eta_l = 1/4$ [47]. Based on this power consumption model, we conclude that it is essential to put the RRHs into sleep whenever possible.

2.2.2.2 Fronthaul Network Power Consumption Model

Although there is no superior solution to meet the low-cost, high-bandwidth, low-latency requirement of fronthaul networks for the Cloud-RAN, the future passive optical network (PON) can provide cost-effective connections between the RRHs and the BBU pool [48]. PON comprises an optical line terminal (OLT) that connects a set of associated optical network units (ONUs) through a single fiber. Implementing a sleep mode in the optical network unit (ONU) has been considered as the most cost-effective and promising power-saving method [49] for the PON, but the OLT cannot go into the sleep mode and its power consumption is fixed [49]. Hence, the total power consumption of the fronthaul network is given by [49]

$$P^{\mathrm{tn}} = P_{\mathrm{olt}} + \sum_{l=1}^{L} P_l^{\mathrm{tl}}, \tag{2.2.6}$$

where P_{olt} is the OLT power consumption, $P_l^{\mathrm{tl}} = P_{a,l}^{\mathrm{tl}}$ and $P_l^{\mathrm{tl}} = P_{s,l}^{\mathrm{tl}}$ denote the power consumed by the ONU l (or the fronthaul link l) in the active mode and sleep mode, respectively.

The typical values are $P_{\text{olt}} = 20W$, $P_{a,l}^{\text{tl}} = 3.85W$ and $P_{s,l}^{\text{tl}} = 0.75W$ [49]. Thus, we conclude that putting some fronthaul links into the sleep mode is a promising way to reduce the power consumption of Cloud-RAN.

2.2.2.3 Network Power Consumption

Based on the above discussion, we define $P_l^a \triangleq P_{a,l}^{\text{rrh}} + P_{a,l}^{\text{tl}}$ ($P_l^s \triangleq P_{s,l}^{\text{rrh}} + P_{s,l}^{\text{tl}}$) as the active (sleep) power consumption when both the RRH and the corresponding fronthaul link are switched on (off). Therefore, the network power consumption of the Cloud-RAN is given by

$$\begin{aligned}\hat{p}(\mathcal{A}) &= \sum_{l\in\mathcal{A}} \frac{1}{\eta_l} P_l^{\text{out}} + \sum_{l\in\mathcal{A}} P_l^a + \sum_{l\in\mathcal{Z}} P_l^s + P_{\text{olt}} \\ &= \sum_{l\in\mathcal{A}} \frac{1}{\eta_l} P_l^{\text{out}} + \sum_{l\in\mathcal{A}} (P_l^a - P_l^s) + \sum_{l\in\mathcal{L}} P_l^s + P_{\text{olt}} \\ &= \sum_{l\in\mathcal{A}} \sum_{k=1}^{K} \frac{1}{\eta_l} \|\mathbf{w}_{lk}\|_{\ell_2}^2 + \sum_{l\in\mathcal{A}} P_l^c + \sum_{l\in\mathcal{L}} P_l^s + P_{\text{olt}}, \end{aligned} \tag{2.2.7}$$

where $P_l^{\text{out}} = \sum_{k=1}^{K} \|\mathbf{w}_{lk}\|_{\ell_2}^2$ and $P_l^c = P_l^a - P_l^s$, and the second equality in (2.2.7) is based on the fact $\sum_{l\in\mathcal{Z}} P_l^s = \sum_{l\in\mathcal{L}} P_l^s - \sum_{l\in\mathcal{A}} P_l^s$. Given a Cloud-RAN with the RRH set $\mathcal{L}$, the term $(\sum_{l\in\mathcal{L}} P_l^s + P_{\text{olt}})$ in (2.2.7) is a constant. Therefore, minimizing the total network power consumption $\hat{p}(\mathcal{A})$ (2.2.7) is equivalent to minimizing the following *re-defined* network power consumption by omitting the constant term $(\sum_{l\in\mathcal{L}} P_l^s + P_{\text{olt}})$:

$$p(\mathcal{A}, \mathbf{w}) = \sum_{l\in\mathcal{A}} \sum_{k=1}^{K} \frac{1}{\eta_l} \|\mathbf{w}_{lk}\|_{\ell_2}^2 + \sum_{l\in\mathcal{A}} P_l^c, \tag{2.2.8}$$

where $\mathbf{w} = [\mathbf{w}_{11}^T, \ldots, \mathbf{w}_{1K}^T, \ldots, \mathbf{w}_{L1}^T, \ldots, \mathbf{w}_{LK}^T]^T$. The advantage of introducing the term P_l^c is that we can rewrite the network power consumption model (2.2.7) in a more compact form as in (2.2.8) and extract the relevant parameters for our system design. In the following discussion, we refer to P_l^c as the *relative fronthaul link power consumption* for simplification. Therefore, the first part of (2.2.8) is the total transmit power consumption and the second part is the total relative fronthaul link power consumption.

Note 1. *The re-defined network power consumption model (2.2.8) reveals two key design parameters: the transmit power consumption ($\frac{1}{\eta_l} \sum_{k=1}^{K} \|\mathbf{w}_{lk}\|_{\ell_2}^2$) and the relative fronthaul link power consumption P_l^c. With the typical values provided in Section 2.2.2.1 and Section*

2.2.2.2, the maximum transmit power consumption, i.e., $\frac{1}{\eta_l}P_l^{out} = 4W$*, is comparable with the relative fronthaul link power consumption, i.e.,* $P_l^c = P_l^a - P_l^s = (P_{a,l}^{\mathrm{rrh}} + P_{a,l}^{\mathrm{tl}}) - (P_{s,l}^{\mathrm{rrh}} + P_{s,l}^{\mathrm{tl}}) = 5.6W$*. This implies that a joint RRH selection (and the corresponding fronthaul link selection) and power minimization beamforming is required to minimize the network power consumption.*

2.3 Problem Formulation and Analysis

Based on the power consumption model, we will formulate the network power consumption minimization problem in this section.

2.3.1 Power Saving Strategies and Problem Formulation

The network power consumption model $(2.2.8)$ indicates the following two strategies to reduce the network power consumption:

- Reduce the transmission power consumption;
- Reduce the number of active RRHs and the corresponding fronthaul links.

However, the two strategies conflict with each other. Specifically, in order to reduce the transmission power consumption, more RRHs are required to be active to exploit a higher beamforming gain. On the other hand, allowing more RRHs to be active will increase the power consumption of fronthaul links. As a result, the network power consumption minimization problem requires a joint design of RRH (and the corresponding fronthaul link) selection and coordinated transmit beamforming.

In this work, we assume perfect channel state information (CSI) available at the BBU pool. With target SINRs $\boldsymbol{\gamma} = (\gamma_1, \dots, \gamma_K)$, the network power consumption minimization problem can be formulated as

$$\begin{aligned} \mathscr{P}: \underset{\{\mathbf{w}_{lk}\},\mathcal{A}}{\text{minimize}} \quad & p(\mathcal{A}, \mathbf{w}) \\ \text{subject to} \quad & \frac{|\sum_{l\in\mathcal{A}} \mathbf{h}_{kl}^{\mathsf{H}}\mathbf{w}_{lk}|^2}{\sum_{i\neq k}|\sum_{l\in\mathcal{A}} \mathbf{h}_{kl}^{\mathsf{H}}\mathbf{w}_{li}|^2 + \sigma_k^2} \geq \gamma_k, \\ & \sum_{k=1}^{K} \|\mathbf{w}_{lk}\|_{\ell_2}^2 \leq P_l, l \in \mathcal{A}. \end{aligned} \qquad (2.3.1)$$

Problem $\mathscr{P}$ is a joint RRH set selection and transmit beamforming problem, which is difficult to solve in general. In the following, we will analyze and reformulate it.

2.3.2 Problem Analysis

We first consider the case with a given active RRH set $\mathcal{A}$ for problem $\mathscr{P}$, resulting a network power minimization problem $\mathscr{P}(\mathcal{A})$. Let $\mathbf{w}_k = [\mathbf{w}_{lk}^T]^T \in \mathbb{C}^{\sum_{l\in\mathcal{A}} N_l}$ indexed by $l \in \mathcal{A}$, and $\mathbf{h}_k = [\mathbf{h}_{lk}^T]^T \in \mathbb{C}^{\sum_{l\in\mathcal{A}} N_l}$ indexed by $l \in \mathcal{A}$, such that $\mathbf{h}_k^{\mathsf{H}}\mathbf{w}_k = \sum_{l\in\mathcal{A}} \mathbf{h}_{kl}^{\mathsf{H}}\mathbf{w}_{lk}$. Since the phases of $\mathbf{w}_k$ will not change the objective function and constraints of $\mathscr{P}(\mathcal{A})$ [50], the SINR constraints are equivalent to the following second order cone (SOC) constraints:

$$\mathcal{C}_1(\mathcal{A}) : \sqrt{\sum\nolimits_{i\neq k} |\mathbf{h}_k^{\mathsf{H}}\mathbf{w}_i|^2 + \sigma_k^2} \leq \frac{1}{\sqrt{\gamma_k}}\Re(\mathbf{h}_k^{\mathsf{H}}\mathbf{w}_k), k \in \mathcal{S}. \tag{2.3.2}$$

The per-RRH power constraints (2.2.4) can be rewritten as

$$\mathcal{C}_2(\mathcal{A}) : \sqrt{\sum\nolimits_{k=1}^{K} \|\mathbf{A}_{lk}\mathbf{w}_k\|_{\ell_2}^2} \leq \sqrt{P_l}, l \in \mathcal{A}, \tag{2.3.3}$$

where $\mathbf{A}_{lk} \in \mathbb{C}^{\sum_{l\in\mathcal{A}} N_l \times \sum_{l\in\mathcal{A}} N_l}$ is a block diagonal matrix with the identity matrix $\mathbf{I}_{N_l}$ as the l-th main diagonal block square matrix and zeros elsewhere. Therefore, given the active RRH set $\mathcal{A}$, the network power minimization problem is given by

$$\begin{aligned} \mathscr{P}(\mathcal{A}) : \underset{\mathbf{w}_1,\ldots,\mathbf{w}_K}{\text{minimize}} \quad & \sum_{l\in\mathcal{A}} \left(\sum_{k=1}^{K} \frac{1}{\eta_l} \|\mathbf{A}_{lk}\mathbf{w}_k\|_{\ell_2}^2 + P_l^c \right) \\ \text{subject to} \quad & \mathcal{C}_1(\mathcal{A}), \mathcal{C}_2(\mathcal{A}), \end{aligned} \tag{2.3.4}$$

with the optimal value denoted as $p^\star(\mathcal{A})$. This is a second-order cone programming (SOCP) problem, and can be solved efficiently, e.g., via interior point methods [22].

Based on the solution of $\mathscr{P}(\mathcal{A})$, the network power minimization problem $\mathscr{P}$ can be solved by searching over all the possible RRH sets, i.e.,

$$p^\star = \underset{Q\in\{J,\ldots,L\}}{\text{minimize}}\ p^\star(Q), \tag{2.3.5}$$

where $J \geq 1$ is the minimum number of RRHs that makes the network support the QoS requirements, and $p^\star(Q)$ is determined by

$$p^\star(Q) = \underset{\mathcal{A}\subseteq\mathcal{L},|\mathcal{A}|=Q}{\text{minimize}}\ p^\star(\mathcal{A}), \tag{2.3.6}$$

where $p^\star(\mathcal{A})$ is the optimal value of the problem $\mathscr{P}(\mathcal{A})$ in (2.3.4) and $|\mathcal{A}|$ is the cardinality of set $\mathcal{A}$. The number of subsets $\mathcal{A}$ of size m is $\binom{L}{m}$, which can be very large. Thus, in general, the overall procedure will be exponential in the number of RRHs L and thus cannot be applied in practice. Therefore, we will reformulate this problem to develop more efficient algorithms to solve it.

2.3.3 Group Sparse Beamforming Formulation

One way to solve problem $\mathscr{P}$ is to reformulate it as a MINLP problem [28], and the generic algorithms for solving MINLP can be applied. Unfortunately, due to the high complexity, such an approach can only provide a performance benchmark for a simple network setting. In the following, we will pursue a different approach, and try to exploit the problem structure.

We will exploit the group sparsity of the optimal aggregative beamforming vector $\mathbf{w}$, which can be written as a partition:

$$\mathbf{w} = [\underbrace{\mathbf{w}_{11}^T, \ldots, \mathbf{w}_{1K}^T}_{\tilde{\mathbf{w}}_1^T}, \ldots, \underbrace{\mathbf{w}_{L1}^T, \ldots, \mathbf{w}_{LK}^T}_{\tilde{\mathbf{w}}_L^T}]^T, \tag{2.3.7}$$

where all the coefficients in a given vector $\tilde{\mathbf{w}}_l = [\mathbf{w}_{l1}^T, \ldots, \mathbf{w}_{lK}^T]^T \in \mathbb{C}^{KN_l}$ form a *group*. When the RRH l is switched off, the corresponding coefficients in the vector $\tilde{\mathbf{w}}_l$ will be set to zeros simultaneously. Overall there may be multiple RRHs being switched off and the corresponding beamforming vectors will be set to zeros. That is, $\mathbf{w}$ has a group sparsity structure, with the priori knowledge that the blocks of variables in $\tilde{\mathbf{w}}_l$'s should be selected (the corresponding RRH will be switched on) or ignored (the corresponding RRH will be switched off) simultaneously.

Define $N = K\sum_{l=1}^{L} N_l$ and an index set $\mathcal{V} = \{1, 2, \ldots, N\}$ with its power set as $2^{\mathcal{V}} = \{\mathcal{I}, \mathcal{I} \subseteq \mathcal{V}\}$. Furthermore, define the sets $\mathcal{G}_l = \{K\sum_{i=1}^{l-1} N_i + 1, \ldots, K\sum_{i=1}^{l} N_i\}, l = 1, \ldots, L$, as a partition of $\mathcal{V}$, such that $\tilde{\mathbf{w}}_l = [w_i]$ is indexed by $i \in \mathcal{G}_l$. Define the support of beamformer $\mathbf{w}$ as

$$\mathcal{T}(\mathbf{w}) = \{i | w_i \neq 0\}, \tag{2.3.8}$$

where $\mathbf{w} = [w_i]$ is indexed by $i \in \mathcal{V}$. Hence, the total relative fronthaul link power consumption can be written as

$$F(\mathcal{T}(\mathbf{w})) = \sum_{l=1}^{L} P_l^c I(\mathcal{T}(\mathbf{w}) \cap \mathcal{G}_l \neq \emptyset), \tag{2.3.9}$$

where $I(\mathcal{T} \cap \mathcal{G}_l \neq \emptyset)$ is an indicator function that takes value 1 if $\mathcal{T} \cap \mathcal{G}_l \neq \emptyset$ and 0 otherwise. Therefore, the network power minimization problem $\mathscr{P}$ is equivalent to the following group sparse beamforming (GSBF) formulation

$$\begin{aligned} \mathscr{P}_{\text{sparse}} : & \underset{\mathbf{w}}{\text{minimize}} \;\; T(\mathbf{w}) + F(\mathcal{T}(\mathbf{w})) \\ & \text{subject to} \;\; \mathcal{C}_1(\mathcal{L}), \mathcal{C}_2(\mathcal{L}), \end{aligned} \tag{2.3.10}$$

where $T(\mathbf{w}) = \sum_{l=1}^{L} \sum_{k=1}^{K} \frac{1}{\eta_l} \|\mathbf{w}_{lk}\|_{\ell_2}^2$ represents the total transmit power consumption. The equivalence means that if $\mathbf{w}^\star$ is a solution to $\mathscr{P}_{\text{sparse}}$, then $(\{\mathbf{w}_{lk}^\star\}, \mathcal{A}^\star)$ with $\mathcal{A}^\star = \{l : \mathcal{T}(\mathbf{w}^\star) \cap \mathcal{G}_l \neq \emptyset\}$ is a solution to $\mathscr{P}$, and vice versa.

Note that the group sparsity of $\mathbf{w}$ is fundamentally different from the conventional sparsity measured by the ℓ_0-norm of $\mathbf{w}$, which is often used in compressed sensing [35, 36]. The reason is that although the ℓ_0-norm of $\mathbf{w}$ will result in a sparse solution for $\mathbf{w}$, the zero entries of $\mathbf{w}$ will not necessarily align to a same group $\tilde{\mathbf{w}}_l$ to lead to switch off one RRH. As a result, the conventional ℓ_1-norm relaxation [35, 36] to the ℓ_0-norm will not work for our problem. Therefore, we will adopt the mixed ℓ_1/ℓ_p-norm [38] to induce group sparsity for $\mathbf{w}$. The details will be presented in Section V. Note that the "group" in this work refers to the collection of beamforming coefficients associated with each RRH, but not a subset of RRHs.

Since obtaining the global optimization solutions to problem $\mathscr{P}$ is computationally difficult, in the following sections, we will propose two low-complexity algorithms to solve it. We will first propose a greedy algorithm in Section IV, which can be viewed as an approximation to the iteration procedure of (2.3.5). In order to further reduce the complexity, based on the GSBF formulation $\mathscr{P}_{\text{sparse}}$, a three-stage GSBF framework will then be developed based on the group-sparsity inducing norm minimization in Section V.

2.4 Greedy Selection Algorithm

In this section, we develop a heuristic algorithm to solve $\mathscr{P}$ based on the backward greedy selection, which was successfully applied in spare filter design [51] and has been shown to often yield optimal or near-optimal solutions. The backward greedy selection algorithm iteratively selects one RRH to switch off at each step, while re-optimizing the coordinated transmit beamforming for the remaining active RRH set. The key design element for this algorithm is the selection rule of the RRHs to determine which one should be switched off at each step.

2.4.1 Greedy Selection Procedure

Denote the iteration number as $i = 0, 1, 2, \ldots$. At the ith iteration, $\mathcal{A}^{[i]} \subseteq \mathcal{L}$ shall denote the set of active RRHs, and $\mathcal{Z}^{[i]}$ denotes the inactive RRH set with $\mathcal{Z}^{[i]} \cup \mathcal{A}^{[i]} = \mathcal{L}$. At iteration i, an additional RRH $r^{[i]} \in \mathcal{A}^{[i]}$ will be added to $\mathcal{Z}^{[i]}$, resulting in a new set $\mathcal{Z}^{[i+1]} = \mathcal{Z}^{[i]} \cup \{r^{[i]}\}$ after this iteration. We initialize by setting $\mathcal{Z}^{[0]} = \emptyset$. In our algorithm, once an RRH is added to the set $\mathcal{Z}$, it cannot be removed. This procedure is a simplification of the exact search method described in Section III-B. At iteration i, we need to solve the network power minimization problem $\mathscr{P}(\mathcal{A}^{[i]})$ in (2.3.4) with the given active RRH set $\mathcal{A}^{[i]}$.

2.4.1.1 RRH Selection Rule

How to select $r^{[i]}$ at the ith iteration is critical for the performance of the greedy selection algorithm. Based on our objective, we propose to select $r^{[i]}$ to maximize the decrease in the network power consumption. Specifically, at iteration i, we obtain the network power consumption $p^\star(\mathcal{A}_m^{[i]})$ with $\mathcal{A}_m^{[i]} \cup \{m\} = \mathcal{A}^{[i]}$ by removing any $m \in \mathcal{A}^{[i]}$ from the active RRH set $\mathcal{A}^{[i]}$. Thereafter, $r^{[i]}$ is chosen to yield the smallest network power consumption after switching off the corresponding RRH, i.e.,

$$r^{[i]} = \arg\min_{m \in \mathcal{A}^{[i]}} p^\star(\mathcal{A}_m^{[i]}). \tag{2.4.1}$$

We assume that $p^\star(\mathcal{A}_m^{[i]}) = +\infty$ if problem $\mathscr{P}(\mathcal{A}_m^{[i]})$ is infeasible. The impact of switching off one RRH is reducing the fronthaul network power consumption while increasing the total transmit power consumption. Thus, the proposed selection rule actually aims at minimizing the impact of turning off one RRH at each iteration.

Denote $\mathcal{J}$ as the set of candidate RRHs that can be turned off, the greedy selection algorithm is described as follows:

```
Algorithm 1: The Greedy Selection Algorithm
    Step 0: Initialize $\mathcal{Z}^{[0]} = \emptyset$, $\mathcal{A}^{[0]} = \{1, \ldots, L\}$ and $i = 0$;
    Step 1: Solve the optimization problem $\mathscr{P}(\mathcal{A}^{[i]})$ (2.3.4);
        1. If (2.3.4) is feasible, obtain $p^\star(\mathcal{A}^{[i]})$;
            • If $\forall m \in \mathcal{A}^{[i]}$, problem $\mathscr{P}(\mathcal{A}_m^{[i]})$ is infeasible, obtain $\mathcal{J} = \{0, \ldots, i\}$, go to Step 2;
            • If $\exists m \in \mathcal{A}^{[i]}$ makes problem $\mathscr{P}(\mathcal{A}_m^{[i]})$ feasible, find the $r^{[i]}$ according to (2.4.1)
              and update the set $\mathcal{Z}^{[i+1]} = \mathcal{Z}^{[i]} \cup \{r^{[i]}\}$ and the iteration number $i \leftarrow i + 1$, go
              to Step 1;
        2. If (2.3.4) is infeasible, when $i = 0$, $p^\star = \infty$, go to End; when $i > 0$, obtain
           $\mathcal{J} = \{0, 1, \ldots, i - 1\}$,
           go to Step 2;
    Step 2: Obtain the optimal active RRH set $\mathcal{A}^{[j^\star]}$ with $j^\star = \arg\min_{j \in \mathcal{J}} p^\star(\mathcal{A}^{[j]})$ and the
    transmit beamformers minimizing $\mathscr{P}(\mathcal{A}^{[j^\star]})$;
    End
```

2.4.2 Complexity Analysis

At the i-th iteration, we need to solve $|\mathcal{A}^{[i]}|$ SCOP problems $\mathscr{P}(\mathcal{A}_m^{[i]})$ by removing the RRH m from the set $\mathcal{A}^{[i]}$ to determine which RRH should be selected. For each of the SOCP problem $\mathscr{P}(\mathcal{A})$, using the interior-point method, the computational complexity is $\mathcal{O}((K\sum_{l\in\mathcal{A}} N_l)^{3.5})$ [22]. The total number of iterations is bounded by L. As a result, the total number of SOCP problems required to be solved grows *quadratically* with L. Although this reduces the computational complexity significantly compared with the mixed-integer conic programming based algorithms in [52] and [53], the complexity is still prohibitive for large-scale networks. Therefore, in the next section we will propose a group sparse beamforming framework to further reduce the complexity.

2.5 Group Sparse Beamforming Framework

In this section, we will develop two low-complexity algorithms based on the GSBF formulation $\mathscr{P}_{\text{sparse}}$, namely a bi-section GSBF algorithm and an iterative GSBF algorithm, for which, the overall number of SOCP problems to solve grows *logarithmically* and *linearly* with L, respectively. The main motivation is to induce group sparsity in the beamformer, which corresponds to switching off RRHs.

In the bi-section GSBF algorithm, we will minimize the *weighted* mixed ℓ_1/ℓ_2-norm to induce group-sparsity for the beamformer. By exploiting the additional prior information (i.e., power amplifier efficiency, relative fronthaul link power consumption, and channel power gain) available in our setting, the proposed bi-section GSBF algorithm will be demonstrated through rigorous analysis and simulations to outperform the conventional *unweighted* mixed ℓ_1/ℓ_p-norm minimization substantially [41–43]. By minimizing the *re-weighted* mixed ℓ_1/ℓ_2-norm iteratively to enhance the group sparsity for the beamformer, the proposed iterative GSBF algorithm will further improve the performance.

The proposed GSBF framework is a three-stage approach, as shown in Fig. 2.1. Specifically, in the first stage, we minimize a weighted (or re-weighted) group-sparsity inducing norm to induce the group-sparsity in the beamformer. In the second stage, we propose an ordering rule to determine the priority for the RRHs that should be switched off, based on not only the (approximately) sparse beamformer obtained in the first stage, but also some key system parameters. Following the ordering rule, a selection procedure is performed to determine the optimal active RRH set, followed by the coordinated beamforming. The details will be presented in the following subsections.

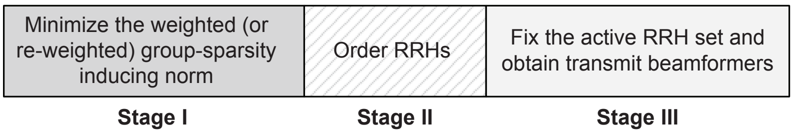
Minimize the weighted (or re-weighted) group-sparsity inducing norm
Order RRHs
Fix the active RRH set and obtain transmit beamformers
Stage I
Stage II
Stage III

Figure 2.1: A three-stage GSBF framework.

2.5.1 Preliminaries on Group-Sparsity Inducing Norms

The mixed ℓ_1/ℓ_p-norm has recently received lots of attention and is shown to be effective to induce group sparsity [38], which is defined as follows:

Definition 1. *Consider the vector* $\mathbf{w} = [\mathbf{w}_{lk}]$ *indexed by* $l \in \mathcal{L}$ *and* $k \in \mathcal{S}$ *as define in (3.3.1). Its mixed* ℓ_1/ℓ_p*-norm is defined as follows:*

$$\mathcal{R}(\mathbf{w}) = \sum_{l=1}^{L} \beta_l \|\tilde{\mathbf{w}}_l\|_{\ell_p}, \; p > 1, \tag{2.5.1}$$

where $\beta_1, \beta_2, \ldots, \beta_L$ are positive weights.

Define the vector $\mathbf{r} = [\|\tilde{\mathbf{w}}_1\|_{\ell_p}, \ldots, \|\tilde{\mathbf{w}}_L\|_{\ell_p}]^T$, then the mixed ℓ_1/ℓ_p-norm behaves as the ℓ_1-norm on the vector $\mathbf{r}$, and therefore, inducing group sparsity (i.e., each vector $\tilde{\mathbf{w}}_l$ is encouraged to be set to zero) for $\mathbf{w}$. Note that, within the group $\tilde{\mathbf{w}}_l$, the ℓ_p-norm does not promote sparsity as $p > 1$. By setting $p = 1$, the mixed ℓ_1/ℓ_p-norm becomes a weighted ℓ_1-norm, which will not promote group sparsity. The mixed ℓ_1/ℓ_2-norm and ℓ_1/ℓ_∞-norm are two commonly used norms for inducing group sparsity. For instance, the mixed ℓ_1/ℓ_2-norm is used with the name *group least-absolute selection* and *shrinkage operator* (or *Group-Lasso*) in machine learning [39]. In high dimensional statistics, the mixed ℓ_1/ℓ_∞-norm is adopted as a regularizer in the linear regression problems with sparsity constraints for its computational convenience [40].

2.5.2 Bi-Section GSBF Algorithm

In this section, we propose a binary search based GSBF algorithm, in which, the overall number of SOCP problems required to be solved grows logarithmically with L, instead of quadratically for the GS algorithm.

2.5.2.1 Group-Sparsity Inducing Norm Minimization

With the combinatorial function $F(\cdot)$ in the objective function $p(\mathbf{w}) = T(\mathbf{w}) + F(\mathcal{T}(\mathbf{w}))$, the problem $\mathscr{P}_{\text{sparse}}$ becomes computationally intractable. Therefore, we first construct an appropriate convex relaxation for the objective function $p(\mathbf{w})$ as a surrogate objective function, resulting a weighted mixed ℓ_1/ℓ_2-norm minimization problem to induce group sparsity for the beamformer. Specifically, we first derive its tightest positively homogeneous lower bound $p_h(\mathbf{w})$, which has the property $p_h(\lambda\mathbf{w}) = \lambda p_h(\mathbf{w}), 0 < \lambda < \infty$. Since $p_h(\mathbf{w})$ is still not convex, we further calculate its Fenchel-Legendre biconjugate $p_h^{**}(\mathbf{w})$ to provide a tightest convex lower bound for $p_h(\mathbf{w})$. We call $p_h^{**}(\mathbf{w})$ as the *convex positively homogeneous lower bound* (the details can be found in [54]) of function $p(\mathbf{w})$, which is provided in the following proposition:

Proposition 1. *The tightest convex positively homogeneous lower bound of the objective function in $\mathscr{P}_{sparse}$, denoted as $p(\mathbf{w})$, is given by*

$$\Omega(\mathbf{w}) = 2\sum_{l=1}^{L}\sqrt{\frac{P_l^c}{\eta_l}}\|\tilde{\mathbf{w}}_l\|_{\ell_2}. \tag{2.5.2}$$

Proof. Please refer to Appendix A. □

This proposition indicates that the group-sparsity inducing norm (i.e., the weighted mixed ℓ_1/ℓ_2-norm) can provide a convex relaxation for the objective function $p(\mathbf{w})$. Furthermore, it encapsulates the additional prior information in terms of system parameters into the weights for the groups. Intuitively, the weights indicate that the RRHs with a higher fronthaul link power consumption and lower power amplifier efficiency will have a higher chance being forced to be switched off. Using the weighted mixed ℓ_1/ℓ_2-norm as a surrogate for the objective function, we minimize the weighted mixed ℓ_1/ℓ_2-norm $\Omega(\mathbf{w})$ to induce the group-sparsity for the beamformer $\mathbf{w}$:

$$\begin{aligned}\mathscr{P}_{\text{GSBF}}: \underset{\mathbf{w}}{\text{minimize}}\ & \Omega(\mathbf{w})\\ \text{subject to}\ & \mathcal{C}_1(\mathcal{L}), \mathcal{C}_2(\mathcal{L}),\end{aligned} \tag{2.5.3}$$

which is an SOCP problem and can be solved efficiently.

2.5.2.2 RRH Ordering

After obtaining the (approximately) sparse beamformer $\hat{\mathbf{w}}$ via solving the weighted group-sparsity inducing norm minimization problem $\mathscr{P}_{\text{GSBF}}$, the next question is how to determine the active RRH set. We will first give priorities to different RRHs, so that an RRH with a higher priority should be switched off before the one with a lower priority. Most previous works [41–43] applying the idea of group-sparsity inducing norm minimization directly to map the sparsity to their application, e.g., in [43], the transmit antennas corresponding to the smaller coefficients in the group (measured by the ℓ_∞-norm) will have a higher priority to be switched off. In our setting, one might be tempted to give a higher priority for an RRH l with a smaller coefficient $r_l = (\sum_{k=1}^{K}\|\hat{\mathbf{w}}_{lk}\|_{\ell_2}^2)^{1/2}$, as it may provide a lower beamforming gain and should be encouraged to be turned off. It turns out that such an ordering rule is not a good option and will bring performance degradation.

To get a better performance, the priority of the RRHs should be determined by not only the beamforming gain but also other key system parameters that indicate the impact of the RRHs on the network performance. In particular, the channel power gain $\kappa_l = \sum_{k=1}^{K} \|\mathbf{h}_{kl}\|_{\ell_2}^2$ should be taken into consideration. Specifically, by the broadcast channel (BC)-multiple-access channel (MAC) duality [55], we have the sum capacity of the Cloud-RAN as:

$$C_{\text{sum}} = \log\det(\mathbf{I}_N + \mathsf{snr}\sum_{k=1}^{K} \mathbf{h}_k\mathbf{h}_k^{\mathsf{H}}), \tag{2.5.4}$$

where we assume equal power allocation to simplify the analysis, i.e., $\mathsf{snr} = P/\sigma^2, \forall k = 1, \ldots, K$. One way to upper-bound C_{sum} is through upper-bounding the capacity by the total receive SNR, i.e., using the following relation

$$\log\det(\mathbf{I}_N + \mathsf{snr}\sum_{k=1}^{K} \mathbf{h}_k\mathbf{h}_k^{\mathsf{H}}) \;\leq\; \mathrm{Tr}(\mathsf{snr}\sum_{k=1}^{K} \mathbf{h}_k\mathbf{h}_k^{\mathsf{H}}) = \mathsf{snr}\sum_{l=1}^{L} \kappa_l, \tag{2.5.5}$$

which relies on the inequality $\log(1+x) \leq x$. Therefore, from the capacity perspective, the RRH with a higher channel power gain κ_l contributes more to the sum capacity, i.e., it provides a higher power gain and should not be encouraged to be switched off.

Therefore, different from the previous democratic assumptions (e.g., [41–43]) on the mapping between the sparsity and their applications directly, we exploit the prior information in terms of system parameters to refine the mapping on the group-sparsity. Specifically, considering the key system parameters, we propose the following ordering criterion to determine which RRHs should be switched off, i.e.,

$$\theta_l := \sqrt{\frac{\kappa_l \eta_l}{P_l^c}} \left(\sum_{k=1}^{K} \|\hat{\mathbf{w}}_{lk}\|_{\ell_2}\right)^{1/2}, \forall l = 1, \ldots, L, \tag{2.5.6}$$

where the RRH with a smaller θ_l will have a higher priority to be switched off. This ordering rule indicates that the RRH with a lower beamforming gain, lower channel power gain, lower power amplifier efficiency, and higher relative fronthaul link power consumption should have a higher priority to be switched off. The proposed ordering rule will be demonstrated to significantly improve the performance of the GSBF algorithm through simulations.

2.5.2.3 Binary Search Procedure

Based on the ordering rule (3.4.22), we sort the coefficients in the ascending order: $\theta_{\pi_1} \leq \theta_{\pi_2} \leq \cdots \leq \theta_{\pi_L}$ to fix the final active RRH set. We set the first J smallest coefficients to zero, as a result, the corresponding RRHs will be turned. Denote J_0 as the maximum number of RRHs that can be turned off, i.e., the problem $\mathscr{P}(\mathcal{A}^{[i]})$ is infeasible if $i > J_0$, where $\mathcal{A}^{[i]} \cup \mathcal{Z}^{[i]} = \mathcal{L}$ with $\mathcal{Z}^{[i]} = \{\pi_0, \pi_1, \ldots, \pi_i\}$ and $\pi_0 = \emptyset$. A binary search procedure can be adopted to determine J_0, which only needs to solve no more than $(1 + \lceil \log(1 + L) \rceil)$ SOCP problems. In this algorithm, we regard $\mathcal{A}^{[J_0]}$ as the final active RRH set and the solution of $\mathscr{P}(\mathcal{A}^{[J_0]})$ is the final transmit beamformer.

Therefore, the bi-section GSBF algorithm is presented as follows:

Algorithm 2: The Bi-Section GSBF Algorithm

```
Step 0: Solve the weighted group-sparsity inducing norm minimization problem
$\mathscr{P}_{\text{GSBF}}$;
    1. If it is infeasible, set $p^\star = \infty$, go to End;
    2. If it is feasible, obtain the solution $\hat{\mathbf{w}}$, calculate ordering criterion (3.4.22), and sort
       them in the
       ascending order: $\theta_{\pi_1} \leq \cdots \leq \theta_{\pi_L}$, go to Step 1;
Step 1: Initialize $J_{\text{low}} = 0$, $J_{\text{up}} = L$, $i = 0$;
Step 2: Repeat
    1. Set $i \leftarrow \lfloor \frac{J_{\text{low}} + J_{\text{up}}}{2} \rfloor$;
    2. Solve the optimization problem $\mathscr{P}(\mathcal{A}^{[i]})$ (2.3.4): if it is infeasible, set $J_{\text{low}} = i$;
       otherwise, set $J_{\text{up}} = i$;
Step 3: Until $J_{\text{up}} - J_{\text{low}} = 1$, obtain $J_0 = J_{\text{low}}$ and obtain the optimal active RRH set
$\mathcal{A}^\star$ with $\mathcal{A}^\star \cup \mathcal{J} = \mathcal{L}$ and $\mathcal{J} = \{\pi_1, \ldots, \pi_{J_0}\}$;
Step 4: Solve the problem $\mathscr{P}(\mathcal{A}^\star)$, obtain the minimum network power consumption
and the corresponding transmit beamformers;
End
```

2.5.3 Iterative GSBF Algorithm

Under the GSBF framework, the main task of the first two stages is to order the RRHs according to the criterion (3.4.22), which depends on the sparse solution to $\mathscr{P}_{\text{GSBF}}$, i.e., $\{\hat{\mathbf{w}}_{lk}\}$. However, when the minimum of $r_l = (\sum_{k=1}^{K} \|\hat{\mathbf{w}}_{lk}\|_{\ell_2}^2)^{1/2} > 0$ is not close to zero, it will introduce large bias in estimating which RRHs can be switched off. To resolve this issue, we will apply the idea from the majorization-minimization (MM) algorithm [56] (please refer to appendix B for details on this algorithm), to enhance group-sparsity for the beamformer to better estimate which RRHs can be switched off.

The MM algorithms have been successfully applied in the re-weighted ℓ_1-norm (or mixed ℓ_1/ℓ_2-norm) minimization problem to enhance sparsity [42, 43, 57]. However, these algorithms failed to exploit the additional system prior information to improve the performance. Specifically, they used the un-weighted ℓ_1-norm (or mixed ℓ_1/ℓ_p-norm) minimization as the start point of the iterative algorithms and re-weighted the ℓ_1-norm (or mixed ℓ_1/ℓ_p-norm) only using the estimate of the coefficients obtained in the last minimization step. Different from the above conventional re-weighted algorithms, we exploit the additional system prior information at each step (including the start step) to improve the estimation on the group sparsity of the beamformer.

2.5.3.1 Re-weighted Group-Sparsity Inducing Norm Minimization

One way to enhance the group-sparsity compared with using the weighted mixed ℓ_1/ℓ_2 norm $\Omega(\mathbf{w})$ in (3.4.1) is to minimize the following combinatorial function directly:

$$\mathcal{R}(\mathbf{w}) = 2\sum_{l=1}^{L}\sqrt{\frac{P_l^c}{\eta_l}}I(\|\tilde{\mathbf{w}}_l\|_{\ell_2} > 0), \tag{2.5.7}$$

for which the convex function $\Omega(\mathbf{w})$ in (3.4.1) can be regarded as an ℓ_1-norm relaxation. Unfortunately, minimizing $\mathcal{R}(\mathbf{w})$ will lead to a non-convex optimization problem. In this subsection, we will provide a sub-optimal algorithm to solve (25) by adopting the idea from the MM algorithm to enhance sparsity.

Based on the following fact in [58]

$$\lim_{\epsilon\to 0}\frac{\log(1+x\epsilon^{-1})}{\log(1+\epsilon^{-1})} = \begin{cases} 0 & \text{if } x = 0, \\ 1 & \text{if } x > 0, \end{cases} \tag{2.5.8}$$

we rewrite the indicator function in (2.5.7) as

$$I(\|\tilde{\mathbf{w}}_l\|_{\ell_2} > 0) = \lim_{\epsilon\to 0}\frac{\log(1+\|\tilde{\mathbf{w}}_l\|_{\ell_2}\epsilon^{-1})}{\log(1+\epsilon^{-1})}, \forall l \in \mathcal{L}. \tag{2.5.9}$$

The surrogate objective function $\mathcal{R}(\mathbf{w})$ can then be approximated as

$$f(\mathbf{w}) = \lambda_\epsilon \sum_{l=1}^{L} \sqrt{\frac{P_l^c}{\eta_l}} \log(1 + \|\tilde{\mathbf{w}}_l\|_{\ell_2}\epsilon^{-1}), \tag{2.5.10}$$

by neglecting the limit in (2.5.9) and choosing an appropriate $\epsilon > 0$, where $\lambda_\epsilon = \frac{2}{\log(1+\epsilon^{-1})}$. Compared with $\Omega(\mathbf{w})$ in (3.4.1), the log-sum penalty function $f(\mathbf{w})$ has the potential to be much more sparsity-encouraging. The detailed explanations can be found in [57].

Since $\log(1 + x), x \geq 0$, is a concave function, we can construct a majorization function for f by the first-order approximation of $\log(1 + \|\tilde{\mathbf{w}}_l\|_{\ell_2}\epsilon^{-1})$, i.e.,

$$f(\mathbf{w}) \leq \lambda_\epsilon \sum_{l=1}^{L} \sqrt{\frac{P_l^c}{\eta_l}} \left(\underbrace{\frac{\|\tilde{\mathbf{w}}_l\|_{\ell_2}}{\|\tilde{\mathbf{w}}_l^{[m]}\|_{\ell_2} + \epsilon} + c(\mathbf{w}^{[m]})}_{g(\mathbf{w}|\mathbf{w}^{[m]})} \right), \tag{2.5.11}$$

where $\mathbf{w}^{[m]}$ is the minimizer at the $(m-1)$-th iteration, and $c(\mathbf{w}^{[m]}) = \log(1+\|\tilde{\mathbf{w}}_l^{[m]}\|_{\ell_2}) - \|\tilde{\mathbf{w}}_l^{[m]}\|_{\ell_2}/(\|\tilde{\mathbf{w}}_l^{[m]}\|_{\ell_2}+\epsilon)$ is a constant provided that $\mathbf{w}^{[m]}$ is already known at the current m-th iteration.

By omitting the constant part of $g(\mathbf{w}|\mathbf{w}^{[m]})$ at the m-th iteration, which will not affect the solution, we propose a re-weighted GSBF framework to enhance the group-sparsity:

$$\begin{aligned} \mathscr{P}_{\text{iGSBF}}^{[m]} : \{\tilde{\mathbf{w}}_l^{[m+1]}\}_{l=1}^{L} = \arg\min \quad & \sum_{l=1}^{L} \beta_l^{[m]} \|\tilde{\mathbf{w}}_l\|_{\ell_2} \\ \text{subject to} \quad & \mathcal{C}_1(\mathcal{L}), \mathcal{C}_2(\mathcal{L}), \end{aligned} \tag{2.5.12}$$

where

$$\beta_l^{[m]} = \sqrt{\frac{P_l^c}{\eta_l}} \frac{1}{(\|\tilde{\mathbf{w}}_l^{[m]}\|_{\ell_2} + \epsilon)}, \forall l = 1, \dots, L, \tag{2.5.13}$$

are the weights for the groups at the m-th iteration. At each step, the mixed ℓ_1/ℓ_2-norm optimization is re-weighted using the estimate of the beamformer obtained in the last minimization step.

As this iterative algorithm cannot guarantee the global minimum, it is important to choose a suitable starting point to obtain a good local optimum. As suggested in [42, 43, 57], this algorithm can be initiated with the solution of the unweighted ℓ_1-norm minimization, i.e., $\beta_l^{[0]} = 1, \forall l = 1, \ldots, L$. In our setting, however, the prior information on the system parameters can help us generate a high quality stating point for the iterative GSBF framework. Specifically, with the available channel state information, we choose the ℓ_2-norm of the initial beamformer at the l-th RRH $\|\tilde{\mathbf{w}}_l^{[0]}\|_{\ell_2}$ to be proportional to its corresponding channel power gain κ_l, arguing that the RRH with a low channel power gain should be encouraged to be switched off as justified in section V-B. Therefore, from (2.5.13), we set the following weights as the initiation weights for $\mathscr{P}_{\text{iGSBF}}^{[0]}$:

$$\beta_l^{[0]} = \sqrt{\frac{P_l^c}{\eta_l \kappa_l}}, \forall l = 1, \ldots, L. \tag{2.5.14}$$

The weights indicate that the RRHs with a higher relative fronthaul link consumption, lower power amplifier efficiency and lower channel power gain should be penalized more heavily.

As observed in the simulations, this algorithm converges very fast (typically within 20 iterations). We set the maximum number of iterations as $m_{\text{max}} = L$ in our simulations.

2.5.3.2 Iterative Search Procedure

After obtaining the (approximately) sparse beamformers using the above re-weighted GSBF framework, we still adopt the same ordering criterion (3.4.22) to fix the final active RRH set.

Different from the aggressive strategy in the bi-section GSBF algorithm, which assumes that the RRH should be switched off as many as possible and thus results a minimum fronthaul network power consumption, we adopt a conservative strategy to determine the final active RRH set by realizing that the minimum network power consumption may not be attained when the fronthaul network power consumption is minimized.

Specifically, denote J_0 as the maximum number of RRHs that can be switched off, the corresponding inactive RRH set is $\mathcal{J} = \{\pi_0, \pi_1, \ldots, \pi_{J_0}\}$. The minimum network power consumption should be searched over all the values of $\mathscr{P}^*(\mathcal{A}^{[i]})$, where $\mathcal{A}^{[i]} = \mathcal{L} \backslash \{\pi_0, \pi_1, \ldots, \pi_i\}$ and $0 \leq i \leq J_0$. This can be accomplished using an iterative search procedure that requires to solve no more than L SOCP problems.

Therefore, the overall iterative GSBF algorithm is presented as Algorithm 3.

Algorithm 3: The Iterative GSBF Algorithm

```
Step 0: Initialize the weights $\beta_l^{[0]}, l = 1, \ldots, L$ as in (2.5.14) and the iteration counter as $m = 0$;
Step 1: Solve the weighted GSBF problem $\mathscr{P}_{\text{iGSBF}}^{[m]}$ (2.5.12): if it is infeasible, set $p^\star = \infty$ and go to End; otherwise, set $m = m + 1$, go to Step 2;
Step 2: Update the weights using (2.5.13);
Step 3: If converge or $m = m_{\max}$, obtain the solution $\hat{\mathbf{w}}$ and calculate the selection criterion (3.4.22), and sort them in the ascending order: $\theta_{\pi_1} \leq \cdots \leq \theta_{\pi_L}$, go to Step 4; otherwise, go to Step 1;
Step 4: Initialize $\mathcal{Z}^{[0]} = \emptyset$, $\mathcal{A}^{[0]} = \{1, \ldots, L\}$, and $i = 0$;
Step 5: Solve the optimization problem $\mathscr{P}(\mathcal{A}^{[i]})$ (2.3.4);
    1. If (2.3.4) is feasible, obtain $p^*(\mathcal{A}^{[i]})$, update the set $\mathcal{Z}^{[i+1]} = \mathcal{Z}^{[i]} \cup \{\pi_{i+1}\}$ and $i = i + 1$, go to Step 5;
    2. If (2.3.4) is infeasible, obtain $\mathcal{J} = \{0, 1, \ldots, i - 1\}$, go to Step 6;
Step 6: Obtain optimal RRH set $\mathcal{A}^{[j^\star]}$ and beamformers minimizing $\mathscr{P}(\mathcal{A}^{[j^\star]})$ with $j^\star = \arg\min_{j \in \mathcal{J}} p^*(\mathcal{A}^{[j]})$;
End
```

2.5.4 Complexity Analysis and Optimality Discussion

We have demonstrated that the maximum number of iterations is linear and logarithmical to L for the "Iterative GSBF Algorithm" and the "Bi-Section GSBF Algorithm", respectively. Therefore, the convergence speed of the proposed GSBF algorithms scales well for large-scale Cloud-RAN (e.g., with $L = 100$). However, the main computational complexity of the proposed algorithms is related to solving an SOCP problem at each iteration. In particular, with a large number of RRHs, the computational complexity of solving an SOCP problem using the interior-point method is proportional to $\mathcal{O}(L^{3.5})$. Therefore, in order to solve a large-sized SOCP problem, other approaches need to be explored (e.g., the *alternating direction method of multipliers* (ADMM) method [59]). This is an on-going research topic, and we will leave it as our future research direction.

Furthermore, the proposed group sparse beamforming algorithm is a convex relaxation to the original combinatorial optimization problem using the group-sparsity inducing norm, i.e., the mixed ℓ_1/ℓ_2-norm. It is very challenging to quantify the performance gap due to the convex relaxation, for which normally specific prior information is needed, e.g., in compressive sensing, the sparse signal is assumed to obey a power law (see Eq.(1.8) in [36]). However, we do not have any prior information about the optimal solution. This is the fundamental difference between our problem and the existing ones in the field of sparse signal processing.

Table 2.1: Simulation Parameters

Parameter	Value
Path-loss at distance d_{kl} (km)	148.1+37.6 $\log_2(d_{kl})$ dB
Standard deviation of log-norm shadowing σ_s	8 dB
Small-scale fading distribution $\mathbf{g}_{kl}$	$\mathcal{CN}(\mathbf{0}, \mathbf{I})$
Noise power σ_k^2 [4] (10 MHz bandwidth)	-102 dBm
Maximum transmit power of RRH P_l [4]	1 W
Power amplifier efficiency η_l [47]	25%
Transmit antenna power gain	9 dBi

The optimality analysis of the group sparse beamforming algorithms will be left to our future work.

2.6 Simulation Results

In this section, we simulate the performance of the proposed algorithms. We consider the following channel model

$$\mathbf{h}_{kl} = 10^{-L(d_{kl})/20}\sqrt{\varphi_{kl}s_{kl}}\mathbf{g}_{kl}, \tag{2.6.1}$$

where $L(d_{kl})$ is the path-loss at distance d_{kl}, , as given in Table 2.1, s_{kl} is the shadowing coefficient, φ_{kl} is the antenna gain and $\mathbf{g}_{kl}$ is the small scale fading coefficient. We use the standard cellular network parameters as showed in Table 2.1. Each point of the simulation results is averaged over 50 randomly generated network realizations. The network power consumption is given in (2.2.7). We set $P_{s,l}^{\text{rrh}} = 4.3W$ and $P_{s,l}^{\text{tl}} = 0.7W, \forall l$, and $P_{\text{olt}} = 20W$.

The proposed algorithms are compared to the following algorithms:

- **Coordinated beamforming (CB) algorithm**: In this algorithm, all the RRHs are active and only the total transmit power consumption is minimized [18].
- **Mixed-integer nonlinear programming (MINLP) algorithm**: This algorithm [52, 53] can obtain the global optimum. Since the complexity of the algorithm grows exponentially with the number of RRHs L, we only run it in a small-size network.
- **Conventional sparsity pattern (SP) based algorithm**: In this algorithm, the unweighted mixed ℓ_1/ℓ_p-norm is adopted to induce group sparsity as in [41] and [43]. The ordering of RRHs is determined only by the group-sparsity of the beamformer,

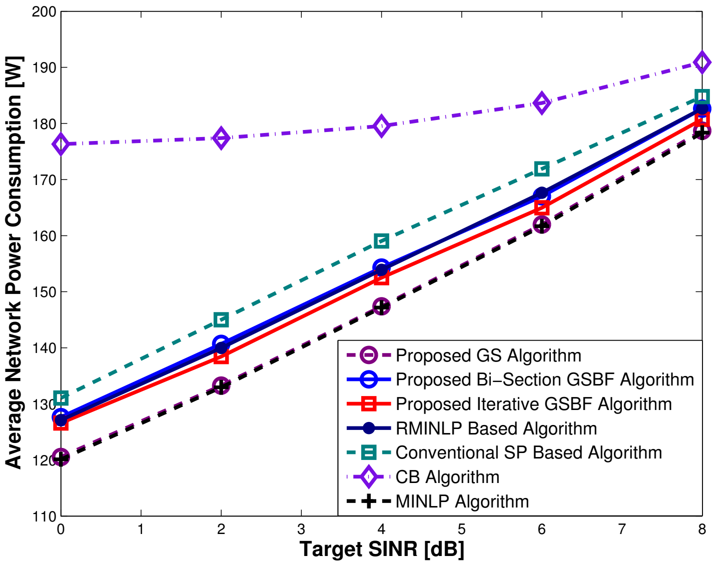

Figure 2.2: Average network power consumption versus target SINR.

i.e., $\theta_l = (\sum_{k=1}^{K} \|\hat{\mathbf{w}}_{lk}\|_{\ell_2})^{1/2}, \forall l = 1, \ldots, L$, instead of (3.4.22). The complexity of the algorithm grows logarithmically with L.

- **Relaxed mixed-integer nonlinear programming (RMINLP) based algorithm**: In this algorithm, a deflation procedure is performed to switch off RRHs one-by-one based on the solutions obtained via solving the relaxed MINLP by relaxing the integers to the unit intervals [53]. The complexity of the algorithm grows linearly with L.

2.6.1 Network Power Consumption versus Target SINR

Consider a network with $L = 10$ 2-antenna RRHs and $K = 15$ single-antenna MUs uniformly and independently distributed in the square region $[-1000\ 1000] \times [-1000\ 1000]$ meters. We set all the relative fronthaul link power consumption to be $P_l^c = (5 + l)W, l = 1, \ldots, L$, which is to indicate the inhomogeneous power consumption on different fronthaul links and RRHs. Fig. 2.2 demonstrates the average network power consumption with different target SINRs.

This figure shows that the proposed GS algorithm can always achieve global optimum (i.e., the optimal value from the MINLP algorithm), which confirms the effectiveness of the proposed RRH selection rule for the greedy search procedure. With only logarithmic complexity, the proposed bi-section GSBF algorithm achieves almost the same performance as the RMINLP algorithm, which has a linear complexity. Moreover, with the same complexity, the gap between the conventional SP based algorithm and the proposed bi-section GSBF algorithm is large. Furthermore, the proposed iterative GSBF algorithm always outperforms the RMINLP algorithm, while both of them have the same computational complexity. These confirm the effectiveness of the proposed GSBF framework to minimize the network power consumption. Overall, this figure shows that our proposed schemes have the potential to reduce the power consumption by 40% in the low QoS regime, and by 20% in the high QoS regime.

This figure also demonstrates that, when the target SINR increases[2], the performance gap between the CB algorithm and the other algorithms becomes smaller. In particular, when the target SINR is relatively high (e.g., 8 dB), all the other algorithms achieve almost the same network power consumption as the CB algorithm. This implies that almost all the RRHs need to be switched on when the QoS requirements are extremely high. In the extreme case with all the RRHs active, all the algorithms will yield the same network power consumption, as all of them will perform coordinated beamforming with all the RRHs active, resulting in the same total transmit power consumption.

2.6.1.1 Impact of Different Components of Network Power Consumption

Consider the same network setting as in Fig. 2.2. The corresponding average total transmit power consumption $p_1(\mathcal{A}) = \sum_{l\in\mathcal{A}} \frac{1}{\eta_l} \sum_{k=1}^{K} \|\mathbf{w}_{lk}\|_{\ell_2}^2$ is demonstrated in Fig. 2.3, and the corresponding average total relative fronthaul link power consumption $p_2(\mathcal{A}) = \sum_{l\in\mathcal{A}} P_l^c$ is shown in Fig. 2.4. Table 2.2 shows the average numbers of RRHs that are switched off with different algorithms. From Fig. 2.3 and Fig. 2.4, we see that the CB algorithm, which intends to minimize the total transmit power consumption, achieves the lowest total transmit power consumption due to the highest beamforming gain with all the RRH active, but it has the highest total relative fronthaul link power consumption. This implies that a joint RRH selection and power minimization beamforming is required to minimize the network power consumption.

From Table 2.2, we see that the proposed GS algorithm can switch off almost the same number of RRHs as the MINLP algorithm. Furthermore, the proposed GSBF algorithms can

[2] We will show, in Table 2.2 and Fig. 2.3, both the number of active RRHs and the total transmit power consumption will increase simultaneously to meet the QoS requirements.

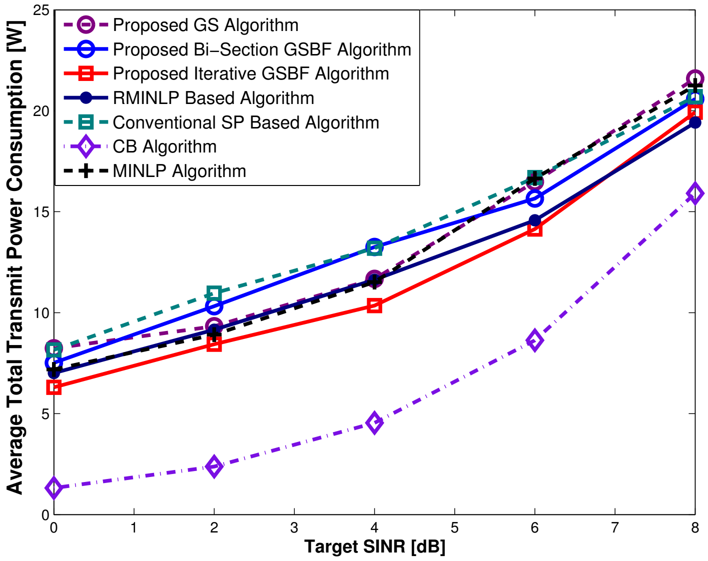

Figure 2.3: Average total transmit power consumption versus target SINR.

Table 2.2: The Average Number of Inactive RRHs with Different Algorithms

Target SINR [dB]	0	2	4	6	8
Proposed GS Algorithm	5.00	4.00	3.02	2.35	1.40
Proposed Bi-Section GSBF Algorithm	4.92	3.98	2.96	2.04	1.13
Proposed Iterative GSBF Algorithm	4.94	4.00	2.94	2.15	1.25
RMINLP Based Algorithm	4.88	3.90	2.79	1.85	1.00
Conventional SP Based Algorithm	4.88	3.90	2.81	1.94	1.10
CB Algorithm	0.00	0.00	0.00	0.00	0.00
MINLP Algorithm	5.00	4.00	3.08	2.42	1.44

switch off more RRHs than the RMINLP based algorithm and the conventional SP based algorithm on average. Overall, the proposed algorithms achieve a lower total relative fronthaul link power consumption, as shown in Fig. 2.4. In particular, the proposed iterative GSBF algorithm can achieve a higher beamforming gain to minimize the total transmit power consumption, as shown in Fig. 2.3. Therefore, the results in Fig. 2.3, Fig. 2.4, and Table 2.2 demonstrate the effectiveness of our proposed RRH selection rule and RRH ordering rule for the GS algorithm and the GSBF algorithms, respectively. Furthermore, the results in Table 2.2 verify the group sparsity assumption in the GSBF algorithms.

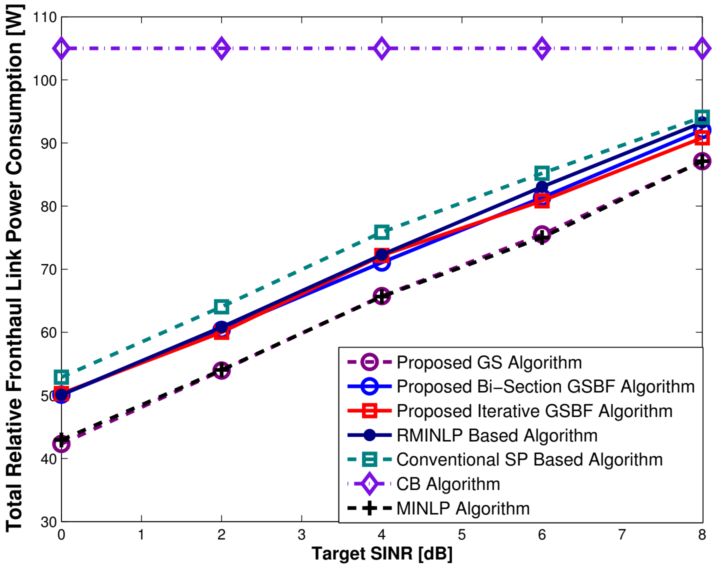

Figure 2.4: Average total relative fronthaul link power consumption versus target SINR.

2.6.2 Network Power Consumption versus Fronthaul Links Power Consumption

Consider a network involving[3] $L = 20$ 2-antenna RRHs and $K = 15$ single-antenna MUs uniformly and independently distributed in the square region $[-2000\ 2000] \times [-2000\ 2000]$ meters. We set all the relative fronthaul link power consumption to be the same, i.e., $P_c = P_l^c, \forall l = 1, \dots, L$ and set the target SINR as 4 dB. Fig. 2.5 presents average network power consumption with different relative fronthaul link power consumption.

This figure shows that both the GS algorithm and the iterative GSBF algorithm significantly outperform other algorithms, especially in the high fronthaul link power consumption regime. Moreover, the proposed bi-section GSBF algorithm provides better performance than the conventional SP based algorithm and is close to the RMINLP based algorithm, while with a lower complexity. This result clearly indicates the importance of considering the key system parameters when applying the group sparsity beamforming framework.

Furthermore, this figure shows that all the algorithms achieve almost the same network power consumption when the relative fronthaul link power consumption is relatively low (e.g., $2W$). This implies that almost all the RRHs need to be switched on to get a high beamforming gain to minimize the total transmit power consumption when the relative fronthaul

[3] In [10, Section 6.1], some field trials were demonstrated to verify the feasibility of Cloud-RAN, in which, a BBU pool can typically support 18 RRHs.

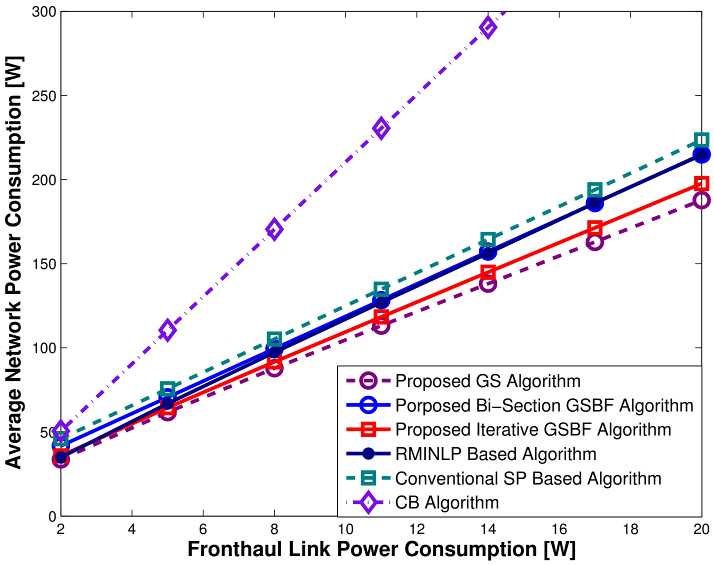

Figure 2.5: Average network power consumption versus relative fronthaul links power consumption.

link power consumption can be ignored, compared to the RRH transmit power consumption.

2.6.3 Network Power Consumption versus the Number of Mobile Users

Consider a network with $L = 20$ 2-antenna RRHs uniformly and independently distributed in the square region $[-2000\ 2000] \times [-2000\ 2000]$ meters. We set all the relative fronthaul link power consumption to be the same, i.e., $P_l^c = 20W, \forall l = 1, \ldots, L$ and set the target SINR as 4 dB. Fig. 2.6 presents the average network power consumption with different numbers of MUs, which are uniformly and independently distributed in the same region.

Overall, this figure further confirms the following conclusions:

1. With the $\mathcal{O}(L^2)$ computational complexity, the proposed GS algorithm has the best performance among all the low-complexity algorithms.

2. With the $\mathcal{O}(L)$ computational complexity, the proposed iterative GSBF algorithm outperforms the RMINLP algorithm, which has the same complexity.

3. With $\mathcal{O}(\log(L))$ computational complexity, the proposed bi-section GSBF algorithm has almost the same performance with the RMINLP algorithm and outperforms the conventional SP based algorithm, which has the same complexity. Therefore, the bi-section GSBF algorithm is very attractive for practical implementation in large-scale

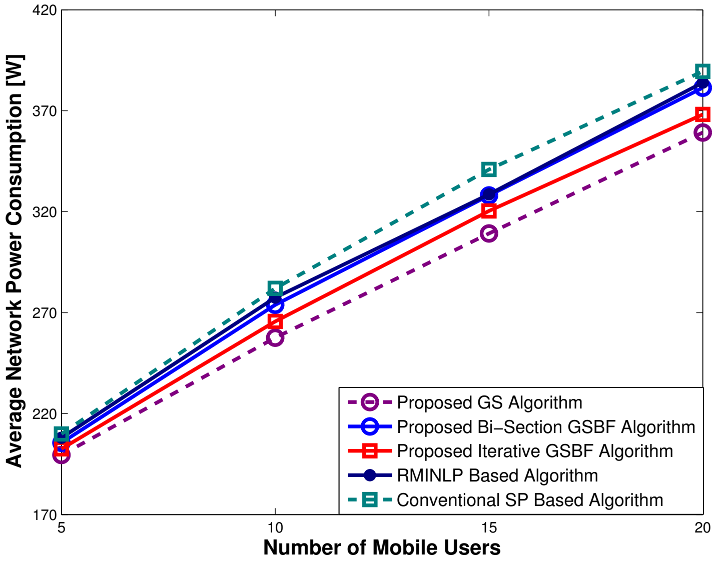

Figure 2.6: Average network power consumption versus the number of mobile users.

Cloud-RAN.

2.7 Discussions

We proposed a group sparse beamforming framework to improve the energy efficiency of cellular networks with the new architecture of Cloud-RAN. It was shown that the fronthaul network power consumption can not be ignored when designing a green Cloud-RAN. By jointly selecting the active RRHs and minimizing the transmit power consumption through coordinated beamforming, the overall network power consumption can be significantly reduced, especially in the low QoS regime. The proposed group sparse formulation $\mathscr{P}_{\text{sparse}}$ serves as a powerful design tool for developing low complexity GSBF algorithms. Through rigorous analysis and careful simulations, the proposed GSBF framework was demonstrated to be very effective to provide near-optimal solutions. Especially, for the large-scale Cloud-RAN, the proposed bi-section GSBF algorithm will be a prior option due to its low complexity, while the iterative GSBF algorithm can be applied to provide better performance in a medium-size network. Simulation also showed that the proposed GS algorithm can always achieve nearly optimal performance, which makes it very attractive in the small-size clustered deployment of Cloud-RAN.

This initial investigation demonstrated the advantage of Cloud-RAN in terms of the network energy efficiency. More works will be needed to exploit the full benefits and overcome the main challenges of Cloud-RAN. Future research directions include theoretical analysis of the optimality of the proposed group sparse beamforming algorithms, more efficient beamforming algorithms for very large-scale Cloud-RAN deployment, joint beamforming and compression when considering the limited-capacity fronthaul links, joint user scheduling, and effective CSI acquisition methods.

Chapter 3

Smoothed L_p-Minimization for Multicast Green Cloud-RAN with Imperfect CSI

In this chapter we propose a holistic sparse optimization framework to design a green Cloud-RAN by taking into consideration the power consumption of the fronthaul links, multicast services, and imperfect CSI. Specifically, we first identify the group sparsity structures in the solutions of the network power minimization problem, which calls for adaptive remote radio head (RRH) selection. However, finding the optimal sparsity structures turns out to be NP-hard with the coupled challenges of the ℓ_0-norm based combinatorial objective function and the nonconvex quadratic QoS constraints due to multicast beamforming and imperfect CSI. In contrast to the previous works on convex but non-smooth group sparsity inducing approaches, e.g., the mixed ℓ_1/ℓ_2-norm relaxation, we adopt the nonconvex but smoothed ℓ_p-minimization ($0 < p \leq 1$) approach to promote group sparsity in the multicast setting, thereby enabling efficient algorithm design based on the principle of the majorization-minimization (MM) algorithm and the semidefinite relaxation (SDR) technique. In particular, an iterative reweighted-ℓ_2 algorithm is developed, which will converge to a Karush-Kuhn-Tucker (KKT) point of the resulting smoothed ℓ_p-minimization problems. Proofs and preliminaries are relegated to Appendix B. The material in this chapter has been presented in part in [60–62].

3.1 Introduction

The great success of wireless industry is driving the proposal of new services and innovative applications, such as Internet of Things (IoT) and mobile Cyber-Physical applications, which yield an exponential growth of wireless traffic with billions of connected devices [11]. To handle orders of magnitude mobile data traffic, network densification and heterogeneity supported by various radio access technologies (e.g., massive MIMO [3] and millimeter-wave communications [63]) have become an irreversible trend in 5G wireless networks [1]. However, this will have a profound impact and bring formidable challenges to the design of 5G wireless communication systems in terms of energy efficiency, capital expenditure (CAPEX), operating expenditure (OPEX), and interference management [11]. In particular, the energy consumption will become prohibitively high in such dense wireless networks in the era of mobile data deluge. Therefore, to accommodate the upcoming diversified and high-volume data services in a cost-effective and energy-efficient way, a paradigm shift is required in the design of 5G wireless networks.

By leveraging the cloud computing technology [12], the cloud radio access network (Cloud-RAN) [10, 64] becomes a disruptive technology to address the key challenges of energy efficiency in 5G wireless networks. Specifically, by moving the baseband units (BBUs) into a single BBU pool (i.e., a cloud data center) with shared computation resources, scalable and parallel signal processing, coordinated resource allocation and cooperative interference management algorithms [7, 65] can be enabled among a large number of radio access points, thereby significantly improving the energy efficiency [29, 66] and spectral efficiency [67]. As the conventional compact base stations are replaced by low-cost and low-power remote radio heads (RRHs), which are connected to the BBU pool through high-capacity and low-latency fronthaul links, Cloud-RAN provides a cost-effective and energy-efficient way to densify the radio access networks [11].

While Cloud-RAN has a great potential to reduce the energy consumption of each RRH, with additional fronthaul link components and dense deployment of RRHs, new challenges arise for designing green Cloud-RAN. In particular, instead of only minimizing the total transmit power consumption via coordinated beamforming [18], the network power consumption consisting of the fronthaul link power consumption and the RRH power consumption should be adopted as the performance metric for designing green Cloud-RAN [29, 66]. To minimize the network power consumption, a group sparse beamforming framework was proposed in [29] to adaptively select the active RRHs and the corresponding fronthaul links via controlling the group sparsity structures of the beamforming vectors. Such an idea of exploiting sparsity structures in the solutions has also demonstrated its effectiveness in solving other mixed combinatorial optimization problems in Cloud-RAN, e.g., the data assignment problem [68] and the joint uplink and downlink network power minimization problem [66].

The effectiveness of group sparse beamforming has been demonstrated in [5], but with certain limitations in the network model, e.g., perfect CSI is assumed at the BBU pool, and only unicast services are considered. In practice, inevitably there will be uncertainty in the available CSI, originating from various sources, e.g., limited feedback [69], channel estimation errors [70], partial CSI acquisition [71, 72] and delay in the obtained CSI [73, 74]. In terms of transmission services from the RRHs, it has been well recognized that the physical layer integration technique [23] can effectively improve the network performance. In particular, the RRHs should not only transmit data to individual users [75] (i.e., broadcast/unicast services) but also integrate additional multicast services [76], where the RRHs transmit a common message in such a way that all the MUs in the same group can decode it. Such multigroup multicast transmission is promising to provide high capacity services and content-aware applications in next generation wireless networks. For instance, with physical layer caching for wireless video delivery [77], it is common that multiple users are interested in the same video stream, which creates multicast groups.

In this chapter, we will thus focus on the design of green Cloud-RAN by jointly minimizing the RRH power consumption and fronthaul link power consumption, considering the practical scenarios with imperfect CSI and multigroup multicast services. We adopt the robust optimization approach to address the CSI uncertainty, such that the QoS requirements are satisfied for *any* realization of the uncertainty in a predefined set [78]. The unique challenges of the network power minimization problem arise from both the infinite number of the non-convex quadratic QoS constraints (due to the robust design criteria and multicast transmission) and the combinatorial composite objective function (due to the consideration of both the relative fronthaul link power consumption and the RRH transmit power consumption).

3.1.1 Related Works

3.1.1.1 Robust Multicast Beamforming

Although the integration of multicast, individual services and cooperative transmission can significantly improve the capacity of wireless networks [23], it will bring significant challenges from both the information theoretic [79] and signal processing perspectives [76, 80].

In particular, the physical-layer multicast beamforming problem is in general NP-hard due to the non-convex quadratic QoS constraints [76]. Furthermore, to address the CSI uncertainty, one may either adopt the stochastic optimization formulation [81] or the robust optimization formulation [82]. However, the stochastic optimization formulations often yield highly intractable problems, e.g., the stochastic coordinated beamforming problem based on the chance constrained programming [72]. The worst-case based robust optimization, on the other hand, has the advantage of computational tractability [78]. Although the original robust and/or multicast beamforming design problems may be non-convex due to the infinite number of non-convex quadratic QoS constraints [83], the convex optimization based SDR technique [84] with S-lemma [22] has recently been applied to provide a principled way to develop polynomial time complexity algorithms to find an approximate solution [85].

However, we cannot directly apply such SDR technique to solve the network power minimization problem due to the non-convex combinatorial composite objective function, which represents the network power consumption.

3.1.1.2 Group Sparse Beamforming

The convex sparsity-inducing penalty approach [38] has recently been widely used to develop polynomial time complexity algorithms for the mixed combinatorial optimization problems in wireless networks, e.g., joint base station clustering and transmit beamforming [41], joint antenna [43] or RRH [29] selection and transmit beamforming. The main idea of this approach is that the sparsity pattern of the beamforming vector, which can be induced by minimizing a sparsity penalty function (e.g., the mixed ℓ_1/ℓ_2-norm minimization can induce the group-sparsity), can provide guidelines for, e.g., antenna selection [43], where the antennas with smaller beamforming coefficients (measured by the ℓ_∞-norm) have a higher priority to be switched off. However, most works only consider the ideal scenario (e.g., perfect CSI and broadcast services [29]), which usually yield convex constraints (e.g., second-order cone constraints [29]).

Unfortunately, we cannot directly adopt the *non-smooth* weighted mixed ℓ_1/ℓ_2-norm developed in [29] to induce the group-sparsity for the robust multicast beamforming vector. This is because the resultant group-sparsity inducing optimization problem will be highly intractable, due to the non-smooth sparsity-inducing objective function and the infinite number of non-convex quadratic QoS constraints.

Based on above discussion and in contrast to the previous work [29] on group sparse beamforming with a non-convex combinatorial composite objective function but convex QoS constraints in the unicast Cloud-RAN, we need to address the following coupled challenges in order to solve the network power minimization problem for multicast green Cloud-RAN with imperfect CSI:

- An infinite number of non-convex quadratic QoS constraints;
- The combinatorial composite objective function.

Thus, to apply the computationally efficient group sparse beamforming approach [29] to more practical scenarios, unique challenges arise. We need to redesign the group-sparsity inducing norm, and then deal with the non-convex group-sparsity inducing optimization problem with an infinite number of non-convex quadratic QoS constraints. We should also develop efficient algorithms for non-convex feasibility problems for the adaptive RRH selection, and for non-convex robust multicast beamforming design after determining the active RRHs.

3.1.2 Contributions

In this chapter, we provide a convex relaxation based robust group sparse beamforming framework for network power minimization in multicast Cloud-RAN with imperfect CSI. The major contributions are summarized as follows:

1. A group sparse beamforming formulation is proposed to minimize the network power consumption for Cloud-RAN. It will simultaneously control the group-sparsity structure and the magnitude of the beamforming coefficients, thereby minimizing the relative fronthaul link power consumption and the transmit power consumption, respectively. The group sparse beamforming modeling framework lays the foundation for developing the three-stage robust group sparse beamforming algorithm based on the convex relaxation.

2. In the first stage, a smoothed ℓ_p-minimization approach is adopted to induce the group-sparsity structure for the robust multicast beamforming vector, thereby guiding the RRH selection. The main motivation for smoothed ℓ_p-minimization formulation is to make the group-sparsity inducing penalty function compatible with the quadratic QoS

constraints. Based on the principle of the MM algorithm and the SDR technique, an iterative reweighted-ℓ_2 algorithm is proposed to solve the resulting nonconvex smoothed ℓ_p-minimization problem. This algorithm is proven to converge to a KKT point of the convexified smoothed ℓ_p-minimization problem.

3. In the second stage, a *PhaseLift* approach based algorithm is proposed to solve the non-convex feasibility problems, based on which the active RRHs can be determined with a binary search. Finally, the SDR technique is adopted to solve the non-convex robust multicast beamforming optimization problem to determine the transmit beamformers for the active RRHs.

4. Simulation results will demonstrate the effectiveness of the proposed robust group sparse beamforming algorithm to minimize the network power consumption.

3.1.3 Organization

The remainder of the chapter is organized as follows. Section 3.2 presents the system model and problem formulation, followed by the problem analysis. In Section 3.3, the group sparse beamforming modeling framework is proposed to formulate the the network power minimization problem. The smoothed ℓ_p-minimization and semidefinite programming (SDP) based robust group sparse beamforming algorithm is developed in Section 3.4. Simulation results will be illustrated in Section 3.5. Finally, conclusions and discussions are presented in Section 3.6.

3.2 System Model and Problem Formulation

3.2.1 System Model

Consider a multicast Cloud-RAN with L RRHs and K single-antenna mobile users (MUs), where the l-th RRH is equipped with N_l antennas, as shown in Fig. 1.1. The centralized signal processing is performed at the baseband unit (BBU) pool [10, 29]. Define $\mathcal{S} = \{1, \ldots, K\}$ as the set of all the MUs and $\mathcal{L} = \{1, \ldots, L\}$ as the set of all the RRHs. We focus on the downlink transmission, for which the signal processing is more challenging. Assume that there are M ($1 \leq M \leq K$) multicast groups, i.e., $\{\mathcal{G}_1, \ldots, \mathcal{G}_M\}$, where $\mathcal{G}_m$ is the set of

MUs in the multicast group m with $1 \leq m \leq M$. Let $\mathcal{M} = \{1, \ldots, M\}$ be the set of the multicast groups. Each MU only belongs to a single multicast group, i.e., $\mathcal{G}_i \cap \mathcal{G}_j = \emptyset$ such that $\cup_i \mathcal{G}_i = \mathcal{S}$ and $\sum_i |\mathcal{G}_i| = K$. Let $\mathbf{v}_{lm} \in \mathbb{C}^{N_l}$ be the transmit beamforming vector from the l-th RRH to the k-th MU in group $\mathcal{G}_m$. The encoded transmission information symbol of the multicast group m is denoted as $s_m \in \mathbb{C}$ with $\mathbb{E}[|s_m|^2] = 1$. The channel propagation between MU k and RRH l is denoted as $\mathbf{h}_{kl} \in \mathbb{C}^{N_l}$. Therefore, the received signal $y_{k,m} \in \mathbb{C}$ at MU k in the multicast group m is given by

$$y_{k,m} = \sum_{l=1}^{L} \mathbf{h}_{kl}^{\mathsf{H}} \mathbf{v}_{lm} s_m + \sum_{i \neq m} \sum_{l=1}^{L} \mathbf{h}_{kl}^{\mathsf{H}} \mathbf{v}_{li} s_i + n_k, \forall k \in \mathcal{G}_m, \tag{3.2.1}$$

where $n_k \sim \mathcal{CN}(0, \sigma_k^2)$ is the additive Gaussian noise at MU k. We assume that s_m's and n_k's are mutually independent and all the MUs apply single user detection. The signal-to-interference-plus-noise ratio (SINR) for MU $k \in \mathcal{G}_m$ is given by

$$\Gamma_{k,m} = \frac{|\mathbf{h}_k^{\mathsf{H}} \mathbf{v}_m|^2}{\sum_{i \neq m} |\mathbf{h}_k^{\mathsf{H}} \mathbf{v}_i|^2 + \sigma_k^2}, \forall k \in \mathcal{G}_m, \tag{3.2.2}$$

where $\mathbf{h}_k \triangleq [\mathbf{h}_{k1}^T, \ldots, \mathbf{h}_{kL}^T]^T \in \mathbb{C}^N$ with $N = \sum_{l=1}^{L} N_l$, and $\mathbf{v}_m \triangleq [\mathbf{v}_{1m}^T, \mathbf{v}_{2m}^T, \ldots, \mathbf{v}_{Lm}^T]^T \in \mathbb{C}^N$ is the aggregative beamforming vector for the multicast group m from all the RRHs. The transmit signal at RRH l is given by

$$\mathbf{x}_l = \sum_{m=1}^{M} \mathbf{v}_{lm} s_m, \forall l. \tag{3.2.3}$$

Each RRH has its own transmit power constraint, i.e.,

$$\sum_{m=1}^{M} \|\mathbf{v}_{lm}\|_2^2 \leq P_l, \forall l, \tag{3.2.4}$$

where $P_l > 0$ is the maximum transmit power of RRH l.

3.2.2 Problem Formulation

3.2.2.1 Imperfect CSI

In practice, the CSI at the BBU pool will be imperfect, which may originate from a variety of sources. For instance, in frequency-division duplex (FDD) systems, the CSI imperfection may originate from downlink training based channel estimation [70] and uplink limited feedback [69]. It could also be due to the hardware deficiencies, partial CSI acquisition [71, 72] and delays in CSI acquisition [73, 74]. In this chapter, we adopt the following additive error model [83, 86, 87] to model the channel imperfection from all the RRHs to MU k, i.e.,

$$\mathbf{h}_k = \hat{\mathbf{h}}_k + \mathbf{e}_k, \forall k, \tag{3.2.5}$$

where $\hat{\mathbf{h}}_k$ is the estimated channel vector and $\mathbf{e}_k$ is the estimation error vector. There are mainly two ways to model the CSI uncertainty: one is the stochastic modeling based on the probabilistic description, and the other is the deterministic and set-based modeling. However, the stochastic CSI uncertainty modeling will yield probabilistic QoS constraints. The resulting chance constrained programming problems are highly intractable in general [72]. Therefore, to seek a computationally tractable formulation, we further assume that the error vectors satisfy the following elliptic model [83, 86, 87]:

$$\mathbf{e}_k^{\mathsf{H}} \mathbf{\Theta}_k \mathbf{e}_k \leq 1, \forall k, \tag{3.2.6}$$

where $\mathbf{\Theta}_k \in \mathbb{H}^{N \times N}$ with $\mathbf{\Theta}_k \succeq \mathbf{0}$ is the shape of the ellipsoid. This model is motivated by viewing the channel estimation as the main source of CSI uncertainty [87, Section 4.1].

3.2.2.2 Network Power Consumption

In Cloud-RAN, it is vital to minimize the network power consumption, consisting of RRH transmit power and relative fronthaul network power [29], in order to design a green wireless network. RRH selection will be adopted for this purpose. Specifically, let $\mathcal{A}$ be the set of active RRHs, the network power consumption is given by

$$p(\mathcal{A}) = \sum_{l \in \mathcal{A}} P_l^c + \sum_{l \in \mathcal{A}} \sum_{m=1}^{M} \frac{1}{\eta_l} \|\mathbf{v}_{lm}\|_2^2, \tag{3.2.7}$$

where $P_l^c \geq 0$ is the relative fronthaul link power consumption [29] (i.e., the static power saving when both the fronthaul link and the corresponding RRH are switched off) and $\eta_l > 0$ is the drain inefficiency coefficient of the radio frequency power amplifier. The typical values are $P_l^c = 5.6W$ and $\eta_l = 25\%$ [29], respectively.

Given the QoS thresholds $\boldsymbol{\gamma} = (\gamma_1, \ldots, \gamma_K)$, in this chapter, we aim at minimizing the network power consumption while guaranteeing the worst-case QoS requirements in the presence of CSI uncertainty and the per-RRH power constraints, i.e., we will consider the following non-convex mixed combinatorial robust multicast beamforming optimization problem,

$$\mathscr{P} : \underset{\mathbf{v},\mathcal{A},\mathcal{Z}}{\text{minimize}} \sum_{l\in\mathcal{A}} P_l^c + \sum_{l\in\mathcal{A}}\sum_{m=1}^{M} \frac{1}{\eta_l}\|\mathbf{v}_{lm}\|_2^2 \tag{3.2.8}$$

$$\text{subject to} \sum_{m=1}^{M} \|\mathbf{v}_{lm}\|_2^2 \leq P_l, \forall l \in \mathcal{A} \tag{3.2.9}$$

$$\sum_{m=1}^{M} \|\mathbf{v}_{lm}\|_2^2 = 0, \forall l \in \mathcal{Z} \tag{3.2.10}$$

$$\frac{|(\hat{\mathbf{h}}_k + \mathbf{e}_k)^{\mathsf{H}}\mathbf{v}_m|^2}{\sum_{i\neq m}|(\hat{\mathbf{h}}_k + \mathbf{e}_k)^{\mathsf{H}}\mathbf{v}_i|^2 + \sigma_k^2} \geq \gamma_k \tag{3.2.11}$$

$$\mathbf{e}_k^{\mathsf{H}}\mathbf{\Theta}_k\mathbf{e}_k \leq 1, \forall k \in \mathcal{G}_m, m \in \mathcal{M}, \tag{3.2.12}$$

where $\mathcal{Z}$ is the set of inactive RRHs such that $\mathcal{A} \cup \mathcal{Z} = \mathcal{L}$ and $\mathbf{v} = [\mathbf{v}_{lm}]$ is the aggregated beamforming vector from all the RRHs to all the MUs. The constraints in (3.2.10) indicate that the transmit powers of the inactive RRHs are enforced to be zero. That is, the beamforming coefficients at the inactive RRHs are set to be zero simultaneously. Constraints (3.2.11) and (3.2.12) indicate that all the QoS requirements in (3.2.11) should be satisfied for *all* realizations of the errors $\mathbf{e}_k$'s within the feasible set formed by the constraint (3.2.12).

The network power minimization problem $\mathscr{P}$ imposes the following challenges:

1. For a given set of CSI error vectors $\mathbf{e}_k$'s, the corresponding network power minimization problem is highly intractable, due to the combinatorial composite objective function (3.2.8) and the non-convex quadratic constraints (3.2.10) and (3.2.11).

2. There are an infinite number of non-convex quadratic QoS constraints due to the worst-case design criterion.

To efficiently address the above unique challenges in a unified fashion, in this chapter, we will propose a systematic convex relaxation approach based on SDP optimization to solve problem $\mathscr{P}$. In particular, the combinatorial challenge will be addressed by the sparsity-inducing penalty approach in Section 3.4.1, based on the smoothed ℓ_p-minimization. The convex optimization technique based on PhaseLift, SDR and S-lemma will be adopted to cope with the infinite number of non-convex quadratic constraints in Sections 3.4.2 and 3.4.3.

In the next subsection, we will provide a detailed analysis of problem $\mathscr{P}$. In particular, the connections with the formulations in existing literatures will be discussed, which will reveal the generality of the formulation $\mathscr{P}$ for practical design problems in Cloud-RAN.

3.2.3 Problem Analysis

While problem $\mathscr{P}$ incorporates most of the practical elements in Cloud-RAN, i.e., imperfect CSI and multigroup multicast transmission, it raises unique challenges compared with the existing works. Following is a list of key aspects of the difficulty of problem $\mathscr{P}$, accompanied with potential solutions.

- *Robust Beamforming Design*: Suppose that all the RRHs are active, i.e., $\mathcal{A} = \mathcal{L}$, with broadcast/unicast transmission, i.e., $|\mathcal{G}_m| = 1, \forall m$ and $M = K$. Then problem $\mathscr{P}$ reduces to the conventional worst-case non-convex robust beamforming design problems [83, 86]. For this special case, the SDR technique [84] combined with the S-lemma [22] is proven to be powerful to find good approximation solutions to such problems.

- *Multicast Beamforming Design*: Physical-layer multicast beamforming design problems [76] prove to be non-convex quadratically constrained problems (QCQP) [22], even with perfect CSI and all the RRHs active. Again, the SDR technique can relax this problem to a convex one, yielding efficient approximation solutions.

- *Quadratically Constrained Feasibility Problem*: Suppose that the inactive RRH set $\mathcal{Z}$ with $|\mathcal{Z}| > 0$ is fixed, then we have the quadratic equation constraints (3.2.10) in problem $\mathscr{P}$. *PhaseLift* [88] is a convex programming technique to relax the non-convex feasibility problem with such quadratic equation constraints to a convex one by lifting the problem to higher dimensions and relaxing the rank-one constraints by the convex surrogates, i.e., the trace norms or nuclear norms.

- *Non-convex Mixed-integer Nonlinear Programming Optimization Problem*: Problem $\mathscr{P}$ can be easily reformulated as a mixed-integer non-linear programming (MINLP) problem as shown in [29]. However, the MINLP problem has exponential complexity [52]. Therefore, such a reformulation cannot bring algorithmic design advantages. One thus has to resort to some global optimization techniques [28, 53] (e.g, branch-and-bound method) or greedy algorithms [29]. Instead, the group-sparsity inducing penalty approach has recently received enormous attention to seek effective convex relaxation for the MINLP problems, e.g., for jointly designing transmit beamformers and selecting bases stations [41], transmit antennas [43], or RRHs [29]. However, with multicast transmission and imperfect CSI, we cannot directly adopt the group-sparsity inducing penalty developed in [29] with the weighted mixed ℓ_1/ℓ_2-norm, as we have seen that we need to lift the problem $\mathscr{P}$ to higher dimensions to cope with the non-convexity of the robust multicast beamforming problem. This requires to develop a new group-sparsity inducing penalty function, which needs to be compatible with quadratic forms, as the beamforming coefficients will be lifted to higher dimensions.

The above discussions show that problem $\mathscr{P}$ cannot be directly solved by existing methods. Thus, we will propose a new robust group sparse beamforming algorithm in this chapter, to solve the highly intractable problem $\mathscr{P}$. Specifically, in Section 3.3, we will propose a group sparse beamforming modeling framework to reformulate the original problem $\mathscr{P}$. The algorithmic advantages of working with the group sparse beamforming formulation will be revealed in Section 3.4, where an iterative reweighted-ℓ_2 algorithm will be developed.

3.3 A Group Sparse Beamforming modeling framework

In this section, we propose a group sparse beamforming modeling framework to reformulate the network power minimization problem $\mathscr{P}$ by controlling the group-sparsity structure and the magnitude of the beamforming coefficients simultaneously. The main advantage of such a modeling framework is the capability of enabling polynomial time complexity algorithm design via convex relaxation.

3.3.1 Network Power Consumption Modeling

We observe that the network power consumption (3.2.7) can be modeled by a composite function parameterized by the aggregative beamforming coefficients $\mathbf{v} \in \mathbb{C}^{NM}$, which can be written as a partition

$$\mathbf{v} = [\underbrace{\mathbf{v}_{11}^T, \dots, \mathbf{v}_{1M}^T}_{\tilde{\mathbf{v}}_1^T}, \dots, \underbrace{\mathbf{v}_{L1}^T, \dots, \mathbf{v}_{LM}^T}_{\tilde{\mathbf{v}}_L^T}]^T, \tag{3.3.1}$$

where all the coefficients in a given vector $\tilde{\mathbf{v}}_l = [\mathbf{v}_{l1}^T, \ldots, \mathbf{v}_{lM}^T]^T \in \mathbb{C}^{MN_l}$ form a beamforming coefficient group. Specifically, observe that the optimal aggregative beamforming vector $\mathbf{v}$ in problem $\mathscr{P}$ should have the group-sparsity structure. That is, when the RRH l is switched off, the corresponding coefficients in the beamforming vector $\tilde{\mathbf{v}}_l$ will be set to zero simultaneously. Overall there may be multiple RRHs being switched off and the corresponding beamforming vectors will be set to zero, yielding a group-sparsity structure in the beamforming vector $\mathbf{v}$.

Define the support of the beamforming vector $\mathbf{v}$ as

$$\mathcal{T}(\mathbf{v}) = \{i | v_i \neq 0\}, \tag{3.3.2}$$

where $\mathbf{v} = [v_i]$ is indexed by $i \in \mathcal{V}$ with $\mathcal{V} = \{1, \ldots, MN\}$. Furthermore, define the sets $\mathcal{V}_l = \{M\sum_{i=1}^{l-1} N_i + 1, \ldots, M\sum_{i=1}^{l} N_i\}, l = 1, \ldots, L$, as a partition of $\mathcal{V}$, such that $\tilde{\mathbf{v}}_l = [v_i]$ is indexed by $i \in \mathcal{V}_l$. The network power consumption in the first term of (3.2.7) thus can be defined by the following *combinatorial function* with respect to the support of the beamforming vector, i.e.,

$$F(\mathcal{T}(\mathbf{v})) = \sum_{l=1}^{L} P_l^c I(\mathcal{T}(\mathbf{v}) \cap \mathcal{V}_l \neq \emptyset), \tag{3.3.3}$$

where $I(\mathcal{T} \cap \mathcal{V}_l \neq \emptyset)$ is an indicator function that takes value 1 if $\mathcal{T} \cap \mathcal{V}_l \neq \emptyset$ and 0 otherwise. Therefore, the total relative fronthaul link power consumption can be reduced by encouraging the group-sparsity structure of the beamforming vector $\mathbf{v}$.

Furthermore, the total transmit power consumption in the second term of (3.2.7) can be defined by the *continuous function* with respect to the ℓ_2-norms of the beamforming vector, i.e.,

$$T(\mathbf{v}) = \sum_{l=1}^{L}\sum_{m=1}^{M} \frac{1}{\eta_l}\|\mathbf{v}_{lm}\|_2^2, \tag{3.3.4}$$

which implicates that the transmit powers of the inactive RRHs are zero, i.e., the corresponding beamforming coefficients are zero. Therefore, the transmit power consumption can be minimized by controlling the magnitude of the beamforming coefficients. As a result, the network power consumption in (3.2.7) can be rewritten as the following combinatorial composite function parameterized by the beamforming vector coefficients $\mathbf{v}$, i.e.,

$$P(\mathbf{v}) = F(\mathcal{T}(\mathbf{v})) + T(\mathbf{v}). \tag{3.3.5}$$

Thus, it requires to *simultaneously* control both the combinatorial function F and the continuous function T to minimize the network power consumption. Such a composite function in (3.3.5) captures the unique property of the network power consumption that involves two parts (i.e., relative fronthaul network power consumption and transmit power consumption) only through the beamforming coefficients $\mathbf{v}$.

3.3.2 Group Sparse Beamforming Modeling

Based on (3.3.5), problem $\mathscr{P}$ can be reformulated as the following robust group sparse beamforming problem

$$\begin{aligned} \mathscr{P}_{\text{sparse}} : \underset{\mathbf{v}}{\text{minimize}} \;& F(\mathcal{T}(\mathbf{v})) + T(\mathbf{v}) \\ \text{subject to} \;& \sum_{m=1}^{M} \|\mathbf{v}_{lm}\|_2^2 \leq P_l, \forall l \in \mathcal{L} \\ & \frac{|(\hat{\mathbf{h}}_k + \mathbf{e}_k)^{\mathsf{H}}\mathbf{v}_m|^2}{\sum_{i \neq m} |(\hat{\mathbf{h}}_k + \mathbf{e}_k)^{\mathsf{H}}\mathbf{v}_i|^2 + \sigma_k^2} \geq \gamma_k \\ & \mathbf{e}_k^{\mathsf{H}}\mathbf{\Theta}_k\mathbf{e}_k \leq 1, \forall k \in \mathcal{G}_m, m \in \mathcal{M}, \end{aligned} \tag{3.3.6}$$

via optimizing the beamforming coefficients $\mathbf{v}$. We will show that the special structure of the objective function in $\mathscr{P}_{\text{sparse}}$ yields computationally efficient algorithm design. In particular, the weighted mixed ℓ_1/ℓ_2-norm will be derived as a convex surrogate to control both parts in (3.3.5) by inducing the group-sparsity structure for the robust multicast beamforming vector $\mathbf{v}$, thereby providing guidelines for RRH selection.

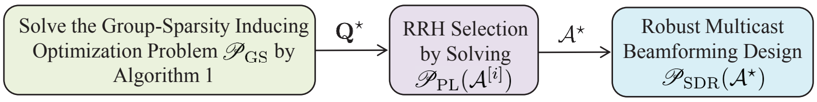
Solve the Group-Sparsity Inducing Optimization Problem $\mathscr{P}_{\text{GS}}$ by Algorithm 1
$\mathbf{Q}^\star$
RRH Selection by Solving $\mathscr{P}_{\text{PL}}(\mathcal{A}^{[i]})$
$\mathcal{A}^\star$
Robust Multicast Beamforming Design $\mathscr{P}_{\text{SDR}}(\mathcal{A}^\star)$

Figure 3.1: The proposed three-stage robust group sparse beamforming frame- work.

3.4 A Smoothed ℓ_p-Minimization Framework for Network Power Minimization

In this section, we will present the semidefinite programming technique for the robust group sparse beamforming problem $\mathscr{P}_{\text{sparse}}$ by lifting the problem to higher dimensions. The general idea is to relax the combinatorial composite objective function by the smoothed ℓ_p-norm to induce the group-sparsity structure for the beamforming vector $\mathbf{v}$. Unfortunately, the resultant group sparse inducing optimization problem is still non-convex. We thus propose an iterative reweighted-ℓ_2 algorithm to find a stationary point to the convexified smoothed ℓ_p-minimization problem, thereby providing the information on determining the priority for the RRHs that should be switched off. Based on the ordering result, a selection procedure is then performed to determine active RRH sets, followed by the robust multicast coordinated beamforming for the active RRHs in the final stage. The proposed three-stage robust group sparse beamforming framework is presented in Fig. 3.1.

3.4.1 Stage One: Smoothed ℓ_p-Minimization for Group Sparsity Inducing

In this section, we describe a systematic way to address the combinatorial challenge in problem $\mathscr{P}_{\text{sparse}}$ by deriving a convex surrogate to approximate the composite objective function in problem $\mathscr{P}_{\text{sparse}}$. Specifically, we first derive the tightest convex positively homogeneous lower bound for the network power consumption function (3.3.5) in the following proposition.

Proposition 2. *The tightest convex positively homogeneous lower bound of the objective function in problem $\mathscr{P}_{sparse}$ is given by*

$$\Omega(\mathbf{v}) = 2\sum_{l=1}^{L}\sqrt{\frac{P_l^c}{\eta_l}}\|\tilde{\mathbf{v}}_l\|_2, \tag{3.4.1}$$

which is a group-sparsity inducing norm for the aggregative robust multicast beamformer vector $\mathbf{v}$.

Proof. Please refer to [29, Appendix A] for the proof. □

Based on proposition 2, one way is to minimize the weighted mixed ℓ_1/ℓ_2-norm to induce the group-sparsity structure for the aggregative robust multicast beamforming vector $\mathbf{v}$:

$$
\begin{aligned}
\mathscr{P}_{\text{GSBF}}: \underset{\mathbf{v}}{\text{minimize}}\ & \Omega(\mathbf{v}) \\
\text{subject to}\ & \sum_{m=1}^{M} \|\mathbf{v}_{lm}\|_2^2 \leq P_l, \forall l \in \mathcal{L} \\
& \frac{|(\hat{\mathbf{h}}_k + \mathbf{e}_k)^{\mathsf{H}} \mathbf{v}_m|^2}{\sum_{i \neq m} |(\hat{\mathbf{h}}_k + \mathbf{e}_k)^{\mathsf{H}} \mathbf{v}_i|^2 + \sigma_k^2} \geq \gamma_k \qquad (3.4.2) \\
& \mathbf{e}_k^{\mathsf{H}} \mathbf{\Theta}_k \mathbf{e}_k \leq 1, \forall k \in \mathcal{G}_m, m \in \mathcal{M}. \qquad (3.4.3)
\end{aligned}
$$

This is, however, a non-convex optimization problem due to the non-convex worst-case QoS constraints (3.4.2) and (3.4.3).

To seek computationally efficient algorithms to solve the non-convex problem $\mathscr{P}_{\text{GSBF}}$, we first propose to lift the problem to higher dimensions with optimization variables as $\mathbf{Q}_m = \mathbf{v}_m \mathbf{v}_m^{\mathsf{H}} \in \mathbb{C}^{N \times N}, \forall m$. To further enhance sparsity in problem $\mathscr{P}_{\text{GSBF}}$ and extract the variables $\mathbf{Q}_m$'s, in Section 3.4.1.1, a smoothed-ℓ_p formulation is proposed to turn the non-smooth group-sparsity inducing norm $\Omega(\mathbf{v})$ into a smooth one with quadratic forms. We then "linearize" the non-convex worst-case QoS constraints with the S-lemma in Section 3.4.1.2. In Section 3.4.1.3, an iterative reweighted-ℓ_2 algorithm is proposed based on the principle of MM algorithm.

3.4.1.1 Smoothed ℓ_p-Minimization

To promote sparse solutions, instead of applying the ℓ_1-norm based convex approximation approach, we adopt a nonconvex approach based on the ℓ_p-norm $(0 < p \leq 1)$ to seek a tighter approximation of the ℓ_0-norm in the objective functions in problem (3.3.5) [89]. The ℓ_p-norm is defined as $\|\boldsymbol{z}\|_p = \left(\sum_{i=1}^{n} |z_i|^p\right)^{1/p}$ with $\boldsymbol{z} \in \mathbb{C}^m$. Furthermore, to enable efficient algorithm design as well as induce the quadratic forms in the resulting approximation problems, we instead adopt the following smoothed version of the ℓ_p-norm to induce sparsity:

$$f_p(\boldsymbol{z};\epsilon) := \sum_{i=1}^{m}(z_i^2+\epsilon^2)^{p/2}, \tag{3.4.4}$$

for $\boldsymbol{z} \in \mathbb{R}^m$ and some small fixed regularizing parameter $\epsilon > 0$.

3.4.1.2 Linearize the Non-convex Worst-case QoS Constraints

Define $\mathbf{G}_m = (\mathbf{Q}_m - \gamma_k \sum_{i\neq m}\mathbf{Q}_i)$, and then the worst-case QoS constraints (3.4.2) and (3.4.3) can be rewritten as

$$\min_{\mathbf{e}_k^{\mathsf{H}}\boldsymbol{\Theta}_k\mathbf{e}_k\leq 1} (\hat{\mathbf{h}}_k+\mathbf{e}_k)^{\mathsf{H}}\mathbf{G}_m(\hat{\mathbf{h}}_k+\mathbf{e}_k) \geq \gamma_k\sigma_k^2, \forall k \in \mathcal{G}_m. \tag{3.4.5}$$

As the number of choices of $\mathbf{e}_k$'s in the worst-case QoS constraint (3.4.5) is infinite, there are an infinite number of such "linearized" QoS constraints. Fortunately, using the S-lemma [22, Appendix B.2], the worst-case QoS constraints (3.4.5) can be equivalently written as the following finite number of convex constraints:

$$\mathcal{C}_1: \left[\begin{array}{c|c}\mathbf{G}_m & \mathbf{G}_m\hat{\mathbf{h}}_k \\ \hline \hat{\mathbf{h}}_k^{\mathsf{H}}\mathbf{G}_m & \hat{\mathbf{h}}_k^{\mathsf{H}}\mathbf{G}_m\hat{\mathbf{h}}_k-\gamma_k\sigma_k^2\end{array}\right] + \lambda_k\left[\begin{array}{c|c}\boldsymbol{\Theta}_k & \mathbf{0} \\ \hline \mathbf{0}^{\mathsf{H}} & -1\end{array}\right] \succeq \mathbf{0}, \tag{3.4.6}$$

where $\lambda_k \geq 0$ and $k \in \mathcal{G}_m$ with $m \in \mathcal{M}$.

Based on the above discussions and utilizing the principle of SDR technique [84] by dropping the rank-one constraints for $\mathbf{Q}_k$'s, we propose to solve the following smoothed ℓ_p-minimization problem to induce the group-sparsity structure for the beamforming vector $\mathbf{v}$:

$$\begin{aligned}\mathscr{P}_{\text{GS}}(\epsilon): \underset{\mathbf{Q},\boldsymbol{\lambda}}{\text{minimize}} & \sum_{l=1}^{L}\rho_l\left(\sum_{m=1}^{M}\text{Tr}(\mathbf{C}_{lm}\mathbf{Q}_m)+\epsilon^2\right)^{p/2} \\ \text{subject to } & \mathcal{C}_1, \mathcal{C}_2(\mathcal{L}), \lambda_k \geq 0, \mathbf{Q}_m \succeq \mathbf{0} \\ & \forall k \in \mathcal{G}_m, m \in \mathcal{M},\end{aligned} \tag{3.4.7}$$

where $\rho_l = P_l^c/\eta_l$, $\boldsymbol{\lambda} = [\lambda_k]$ and $\mathcal{C}_2(\mathcal{A})$ is the set of linearized per-RRH transmit power constraints,

$$\mathcal{C}_2(\mathcal{A}) : \sum_{m=1}^{M} \mathrm{Tr}(\mathbf{C}_{lm}\mathbf{Q}_m) \leq P_l, l \in \mathcal{A}. \tag{3.4.8}$$

Although problems $\mathscr{P}_{\text{GS}}$ is still nonconvex due to the nonconvex objective function, the resulting smoothed ℓ_p-minimization problem preserve the algorithmic advantages, as will be presented in the next subsection.

3.4.1.3 Iterative Reweighted-ℓ_2 Algorithm

Consider the following smoothed ℓ_p-minimization problem,

$$\mathscr{P}_{\text{sm}}(\epsilon) : \underset{\boldsymbol{z}\in\mathcal{C}}{\text{minimize}} \ f_p(\boldsymbol{z};\epsilon) := \sum_{i=1}^{m}(z_i^2+\epsilon^2)^{p/2}, \tag{3.4.9}$$

where $\mathcal{C}$ is an arbitrary convex set, $\boldsymbol{z} \in \mathbb{R}^m$ and $\epsilon > 0$ is some fixed regularizing parameter. In the following, we first prove that the optimal solution of the smoothed ℓ_p-minimization problem $\mathscr{P}_{\text{sm}}(\epsilon)$ is also optimal for the original non-smooth ℓ_p-minimization problem (i.e., $\mathscr{P}_{\text{sm}}(0)$) when ϵ is small. We then demonstrate the algorithmic advantages of the smoothness in the procedure of developing the iterative reweighted-ℓ_2 algorithm.

3.4.1.4 Optimality of Smoothing the ℓ_p-Norm

The set of KKT paris of problem $\mathscr{P}_{\text{sm}}(\epsilon)$ is given as

$$\Omega(\epsilon) = \{\boldsymbol{z} \in \mathcal{C} : 0 \in \nabla_{\boldsymbol{z}} f_p(\boldsymbol{z};\epsilon) + \mathcal{N}_{\mathcal{C}}(\boldsymbol{z})\}, \tag{3.4.10}$$

where $\mathcal{N}_{\mathcal{C}}(\boldsymbol{z})$ is the normal cone of a convex set $\mathcal{C}$ at point $\boldsymbol{z}$ consisting of the outward normals to all hyperplanes that support $\mathcal{C}$ at $\boldsymbol{z}$, i.e.,

$$\mathcal{N}_{\mathcal{C}}(\boldsymbol{z}) := \{\boldsymbol{s} : \langle \boldsymbol{s}, \mathbf{x} - \boldsymbol{z}\rangle \leq 0, \forall \mathbf{x} \in \mathcal{C}\}. \tag{3.4.11}$$

Define the deviation of a given set $\mathcal{Z}_1$ from another set $\mathcal{Z}_2$ as [81],

$$\mathbb{D}(\mathcal{Z}_1, \mathcal{Z}_2) = \sup_{z_1\in\mathcal{Z}_1}\left(\inf_{z_2\in\mathcal{Z}_2} \|z_1 - z_2\|\right). \tag{3.4.12}$$

We then have the following theorem on the relationship between the smoothed ℓ_p-minimization problem $\mathscr{P}_{\text{sm}}(\epsilon)$ and the original non-smooth ℓ_p-minimization problem $\mathscr{P}_{\text{sm}}(0)$.

Theorem 1. *Let Ω_ϵ be the set of KKT points of problem $\mathscr{P}_{sm}(\epsilon)$. Then, we have*

$$\lim_{\epsilon \searrow 0} \mathbb{D}(\Omega(\epsilon), \Omega(0)) = 0. \tag{3.4.13}$$

Proof. Please refer to Appendix B.1 for details. □

This theorem indicates that any limit of the sequence of KKT paris of problem $\mathscr{P}_{\text{sm}}(\epsilon)$ is a KKT pair of problem $\mathscr{P}_{\text{sm}}(0)$ when ϵ is small enough. That is, at least a local optimal solution can be achieved. In the sequel, we will focus on finding a KKT point of problem $\mathscr{P}_{\text{sm}}(\epsilon)$ with a small ϵ, yielding good approximations to the KKT points of the ℓ_p-minimization problem $\mathscr{P}_{\text{sm}}(0)$ to induce sparsity in the solutions.

3.4.1.5 The MM Algorithm for the Smoothed ℓ_p-Minimization

With the established asymptotic optimality, we then leverage the principle of the MM algorithm to solve problem (3.4.9). Basically, this algorithm generates the iterates $\{\boldsymbol{z}_n\}_{n=1}^{\infty}$ by successively minimizing upper bounds $Q(\boldsymbol{z}; \boldsymbol{z}^{[n]})$ of the objective function $f_p(\boldsymbol{z}; \epsilon)$. The quality of the upper bounds will control the convergence (rate) and optimality of the resulting algorithms. Inspired by the results in the expectation-maximization (EM) algorithm [90, 91], we adopt the upper bounds in the following proposition to approximate the smoothed ℓ_p-norm.

Proposition 3. *Given the iterate $\boldsymbol{z}^{[n]}$ at the n-th iteration, an upper bound for the objective function of the smoothed ℓ_p-norm $f_p(\boldsymbol{z}; \epsilon)$ can be constructed as follows,*

$$Q(\boldsymbol{z}; \boldsymbol{\omega}^{[n]}) := \sum_{i=1}^{m} \omega_i^{[n]} z_i^2, \tag{3.4.14}$$

where

$$\omega_i^{[n]} = \frac{p}{2}\left[\left(z_i^{[n]}\right)^2 + \epsilon^2\right]^{\frac{p}{2}-1}, \forall i = 1, \ldots, m. \tag{3.4.15}$$

From the weights given in (3.4.15), it is clear that, by adding the regularizer parameter $\epsilon > 0$, we can avoid yielding infinite values when some z_i's become zeros in the iterations.

Proof. Define the approximation error as

$$f_p(\boldsymbol{z};\epsilon) - Q(\boldsymbol{z};\boldsymbol{\omega}^{[n]}) = \sum_{i=1}^{m}[\kappa(z_i^2) - \kappa'((z_i^{[n]})^2)z_i^2], \tag{3.4.16}$$

where $\kappa(s) = (s+\epsilon^2)^{p/2}$ with $s \geq 0$. The sound property of the Q-function (3.4.14) is that the approximation error (3.4.16) attains its maximum at $\boldsymbol{z} = \boldsymbol{z}^{[n]}$. In particular, we only need to prove that the function $g(s) = \kappa(s) - \kappa'(s^{[n]})s$ with $s \geq 0$ attains the maximum at $s = s^{[n]}$. This is true based on the facts that $g'(s^{[n]}) = 0$ and $\kappa(s)$ is strictly concave. □

Let $\boldsymbol{z}^{[n+1]}$ be the minimizer of the upper bound function $Q(\boldsymbol{z};\boldsymbol{\omega}^{[n]})$ at the n-th iteration, i.e.,

$$\boldsymbol{z}^{[n+1]} := \arg\min_{\boldsymbol{z}\in\mathcal{C}} \; Q(\boldsymbol{z};\boldsymbol{\omega}^{[n]}). \tag{3.4.17}$$

Based on Proposition 3 and (3.4.17), we have

$$\begin{aligned} f_p(\boldsymbol{z}^{[n+1]};\epsilon) &= Q(\boldsymbol{z}^{[n+1]};\boldsymbol{\omega}^{[n]}) + f_p(\boldsymbol{z}^{[n+1]};\epsilon) - Q(\boldsymbol{z}^{[n+1]};\boldsymbol{\omega}^{[n]}) \\ &\leq Q(\boldsymbol{z}^{[n+1]};\boldsymbol{\omega}^{[n]}) + f_p(\boldsymbol{z}^{[n]};\epsilon) - Q(\boldsymbol{z}^{[n]};\boldsymbol{\omega}^{[n]}) \\ &\leq Q(\boldsymbol{z}^{[n]};\boldsymbol{\omega}^{[n]}) + f_p(\boldsymbol{z}^{[n]};\epsilon) - Q(\boldsymbol{z}^{[n]};\boldsymbol{\omega}^{[n]}) \\ &= f_p(\boldsymbol{z}^{[n]};\epsilon), \end{aligned} \tag{3.4.18}$$

where the first inequality is based on the fact that function $(f_p(\boldsymbol{z};\epsilon) - Q(\boldsymbol{z};\boldsymbol{\omega}^{[n]}))$ attains its maximum at $\boldsymbol{z} = \boldsymbol{z}^{[n]}$, and the second inequality follows from (3.4.17). Therefore, minimizing the upper bound, i.e., the Q-function in (3.4.14), can reduce the objective function $f_p(\boldsymbol{z};\epsilon)$ successively.

Remark 1. *In the context of the EM algorithm [92] for computing the maximum likelihood estimator of latent variable models, the functions $-f_p(\boldsymbol{z};\epsilon)$ and $-Q(\boldsymbol{z};\boldsymbol{\omega}^{[n]})$ can be regarded as the log-likelihood and comparison functions (i.e., the lower bound of the log-likelihood), respectively [90].*

The MM algorithm for the smoothed ℓ_p-minimization problem is presented in Algorithm 4.

The convergence of the iterates $\{\boldsymbol{z}^{[n]}\}_{n=1}^{\infty}$ (3.4.19) is presented in the following theorem.

Algorithm 4: Iterative Reweighted-ℓ_2 Algorithm

```
input: Initialize $\boldsymbol{\omega}^{[0]} = (1, \ldots, 1)$; $I$ (the maximum number of iterations)
Repeat
  1) Solve problem
      $$\boldsymbol{z}^{[n+1]} := \arg\min_{\boldsymbol{z} \in \mathcal{C}} \sum_{i=1}^{m} \omega_i^{[n]} z_i^2. \quad (3.4.19)$$
  If it is feasible, go to 2); otherwise, stop and return output 2.
  2) Update the weights as
      $$\omega_i^{[n+1]} = \frac{p}{2}\left[\left(z_i^{[n+1]}\right)^2 + \epsilon^2\right]^{\frac{p}{2}-1}, \forall i = 1, \ldots, m. \quad (3.4.20)$$
Until convergence or attain the maximum iterations and return output 1.
output 1: $\boldsymbol{z}^\star$; output 2: Infeasible.
```

Theorem 2. *Let $\{\boldsymbol{z}^{[n]}\}_{n=1}^{\infty}$ be the sequence generated by the iterative reweighted-ℓ_2 algorithm (3.4.19). Then, every limit point $\bar{\boldsymbol{z}}$ of $\{\boldsymbol{z}^{[n]}\}_{n=1}^{\infty}$ has the following properties*

1. *$\bar{\boldsymbol{z}}$ is a KKT point of problem $\mathscr{P}(\epsilon)$;*

2. *$f_p(\boldsymbol{z}^{[n]}; \epsilon)$ converges monotonically to $f_p(\boldsymbol{z}^\star; \epsilon)$ for some KKT point $\boldsymbol{z}^\star$.*

Proof. Please refer to Appendix B.2 for details. □

Remark 2. *The algorithm consisting of the iterate (3.4.19) accompanied with weights (3.4.20) is known as the iterative reweighted least squares [92–94] in the fields of statistics, machine learning and compressive sensing. In particular, with a simple constraint $\mathcal{C}$, the iterates often yield closed-forms with better computational efficiency. For instance, for the noiseless compressive sensing problem [93], the iterates have closed-form solutions [93, (1.9)]. Therefore, this method has better performance in terms of computational efficiency and also better signal recovery capability compared with the conventional ℓ_1-minimization approach.*

In contrast to the existing works on the iterative reweighted least squares methods, we provide a new perspective to develop the iterative reweighted-ℓ_2 algorithm to solve the smoothed ℓ_p-minimization problem with convergence guarantees based on the principle of the MM algorithm. Furthermore, the main motivation and advantages for developing the iterates (3.4.19) is to induce the quadratic forms in the objective function in problem $\mathscr{P}_{\text{GSBF}}$ to make it compliant with the SDR technique, thereby inducing the group sparsity structure in the multicast beamforming vectors via convex programming. Applying algorithm 4 to problem $\mathscr{P}_{\rm GS}(\epsilon)$ is straightforward.

3.4.2 Stage Two: RRH Selection

Given the solution $\mathbf{Q}^{\star}$ to the group sparse inducing optimization problem $\mathscr{P}_{\rm GS}(\epsilon)$, the group-sparsity structure information for the beamformer $\mathbf{v}$ can be extracted from the following relation:

$$\|\tilde{\mathbf{v}}_l\|_{\ell_2} = \left(\sum_{m=1}^{M} \operatorname{Tr}(\mathbf{C}_{lm}\mathbf{Q}_m)\right)^{1/2}, \forall l. \tag{3.4.21}$$

Based on the (approximated) group-sparsity information in (3.4.21), the following ordering criterion [29] incorporating the key system parameters is adopted to determine which RRHs should be switched off, i.e.,

$$\theta_l = \sqrt{\frac{\kappa_l \eta_l}{P_l^c}} \left(\sum_{m=1}^{M} \operatorname{Tr}(\mathbf{C}_{lm}\mathbf{Q}_m^{\star}\right)^{1/2}, \forall l \in \mathcal{L}, \tag{3.4.22}$$

where $\kappa_l = \sum_{k=1}^{K} \|\hat{\mathbf{h}}_{kl}\|_2^2$ is the channel gain for the estimated channel coefficients between RRH l and all the MUs. Therefore, the RRH with a smaller parameter θ_l will have a higher priority to be switched off. Note that most previous works applying the idea of sparsity inducing norm minimization approach directly map the sparsity pattern to their applications. For instance, in [43], the transmit antenna with smaller coefficients in the beamforming coefficient group (measured by the ℓ_∞-norm) will have a higher priority to be switched off. In [29], however, we show that the ordering rule (3.4.22), which incorporates the key system parameters, yields much better performance than the pure sparsity pattern based selection rule in terms of network power minimization.

In this chapter, we adopt a simple RRH selection procedure, i.e., binary search, due to its low-complexity. Specifically, based on the ordering rule (3.4.22), we sort the coefficients in the ascending order: $\theta_{\pi_1} \le \theta_{\pi_2} \le \cdots \le \theta_{\pi_L}$ to determine the active RRH set. Denote J_0 as the maximum number of RRHs that can be switched off. That is, problem $\mathscr{F}(\mathcal{A}^{[i]})$ is feasible for any $i \leq J_0$,

$$\begin{aligned} \mathscr{F}(\mathcal{A}^{[i]}) : & \text{ find } \mathbf{v} \\ & \text{subject to } (3.2.9), (3.2.10), (3.2.11), (3.2.12), \end{aligned} \tag{3.4.23}$$

where $\mathcal{A}^{[i]} \cup \mathcal{Z}^{[i]} = \mathcal{L}$ with $\mathcal{Z}^{[i]} = \{\pi_0, \pi_1, \ldots, \pi_i\}$ and $\pi_0 = \emptyset$. Likewise, problem $\mathscr{F}(\mathcal{A}^{[i]})$ with $\mathcal{A}^{[i]} = \{\pi_{i+1}, \ldots, \pi_L\}$ is infeasible for any $i > J_0$. A binary search procedure can be adopted to determine J_0, which only needs to solve no more than $(1+\lceil\log(1+L)\rceil)$ feasibility problems (3.4.23) as will be presented in Algorithm 5. Denote $\mathcal{A}^{[J_0]}$ as the final active RRH set, we thus need to solve the following transmit power minimization problem

$$\begin{aligned} \mathscr{P}(\mathcal{A}) : \underset{\mathbf{v}}{\text{minimize}} & \sum_{l \in \mathcal{A}} \left(\frac{1}{\eta_l} \sum_{m=1}^{M} \|\mathbf{v}_{lm}\|_2^2 + P_l^c \right) \\ \text{subject to } & (3.2.9), (3.2.10), (3.2.11), (3.2.12), \end{aligned} \tag{3.4.24}$$

with the fixed active RRH set $\mathcal{A} = \mathcal{A}^{[J_0]}$ to determine the transmit beamformer coefficients for the active RRHs. Unfortunately, both problems $\mathscr{F}(\mathcal{A})$ and $\mathscr{P}(\mathcal{A})$ are non-convex and intractable. Thus, in the chapter, we resort to the computationally efficient semidefinite programming technique to find approximate solutions to feasibility problem $\mathscr{F}(\mathcal{A})$ and optimization problem $\mathscr{P}(\mathcal{A})$.

Notice that, with perfect CSI assumptions as in [29, 43], given the active RRH set $\mathcal{A}$, the size of the corresponding optimization problem $\mathscr{P}(\mathcal{A})$ (e.g., [29, (12)] and [43, (13)]) will be reduced. The key observation is that we only need to consider the channel links from the active RRHs. However, with imperfect CSI, we still need to consider the channel links from all the RRHs due to the lack of the knowledge of the exact values of the CSI errors $\mathbf{e}_k$'s. As a result, the sizes of corresponding optimization problems $\mathscr{P}(\mathcal{A}^{[i]})$'s cannot be reduced with imperfect CSI.

3.4.2.1 PhaseLift to the Non-convex Feasibility Problem

In this subsection, we use the *PhaseLift* technique [88] to find approximate solutions to the non-convex feasibility problem $\mathscr{F}(\mathcal{A})$. Specifically, we first lift the problem to higher dimensions such that the feasibility problem $\mathscr{F}(\mathcal{A})$ can be reformulated as

$$\begin{aligned} &\text{find } \mathbf{Q}_1, \ldots, \mathbf{Q}_M \\ &\text{subject to } \mathcal{C}_1, \mathcal{C}_2(\mathcal{A}), \mathcal{C}_3(\mathcal{Z}), \lambda_k \geq 0, \mathbf{Q}_m \succeq \mathbf{0} \\ &\qquad\qquad \text{rank}(\mathbf{Q}_m) = 1, \forall k \in \mathcal{G}_m, m \in \mathcal{M}, \end{aligned} \tag{3.4.25}$$

where

$$\mathcal{C}_3(\mathcal{Z}) : \sum_{m=1}^{M} \text{Tr}(\mathbf{C}_{lm}\mathbf{Q}_m) = 0, \forall l \in \mathcal{Z}. \tag{3.4.26}$$

The main idea of the PhaseLift technique is to approximate the non-convex rank functions in problem (3.4.25) using the convex surrogates, yielding the following convex feasibility problem

$$\begin{aligned} \mathscr{P}_{\text{PL}}(\mathcal{A}) : &\text{find } \mathbf{Q}_1, \ldots, \mathbf{Q}_M \\ &\text{subject to } \mathcal{C}_1, \mathcal{C}_2(\mathcal{A}), \mathcal{C}_3(\mathcal{Z}), \lambda_k \geq 0, \mathbf{Q}_m \succeq \mathbf{0} \\ &\qquad\qquad \forall k \in \mathcal{G}_m, m \in \mathcal{M}, \end{aligned} \tag{3.4.27}$$

which is an SDP problem and can be solved using the interior-point method [22] efficiently. Furthermore, the Gaussian randomization procedure [84] can be applied to obtain a rank-one approximate solution from the solution of $\mathscr{P}_{\text{PL}}(\mathcal{A})$.

Remark 3. *The PhaseLift technique, serving as one promising application of the SDR method, was proposed in [88] to solve the phase retrieval problem [95], which is mathematically a feasibility problem with multiple quadratic equation constraints. Various conditions are presented in [88, 95] for the phase retrieval problem, under which the corresponding solution of the PhaseLift relaxation problem yields a rank-one solution with a high probability. However, for our problem $\mathscr{P}_{PL}$ with additional complicated constraints, it is challenging to perform such rank-one solution analysis. Thus, in this chapter, we only focus on developing computationally efficient approximation algorithms based on the SDR technique.*

3.4.3 Stage Three: SDR to the Robust Multicast Beamforming Problem

Once we have selected active RRHs, i.e., fix the set $\mathcal{A}$, we need to finalize the beamforming vector by solving problem $\mathscr{P}(\mathcal{A})$. We lift the non-convex optimization problem $\mathscr{P}(\mathcal{A})$ to higher dimensions and adopt the SDR technique by dropping the rank-one constraints, yielding the following convex relaxation problem

$$\begin{aligned} \mathscr{P}_{\text{SDR}}(\mathcal{A}) : \underset{\mathbf{Q},\boldsymbol{\lambda}}{\text{minimize}} \quad & \sum_{l\in\mathcal{A}} \left(\frac{1}{\eta_l} \sum_{m=1}^{M} \text{Tr}(\mathbf{C}_{lm}\mathbf{Q}_m) + P_l^c \right) \\ \text{subject to} \quad & \mathcal{C}_1, \mathcal{C}_2(\mathcal{A}), \mathcal{C}_3(\mathcal{Z}), \lambda_k \geq 0, \mathbf{Q}_m \succeq \mathbf{0} \\ & \forall k \in \mathcal{G}_m, m \in \mathcal{M}, \end{aligned} \tag{3.4.28}$$

which is an SDP problem and can be solved using the interior-point method [22]. It is important to investigate whether the solution to the problem $\mathscr{P}_{\text{SDR}}(\mathcal{A})$ yields a rank-one solution. This is, however, an on-going research topic [85]. In this chapter, if the rank-one solution is failed to be obtained, the Gaussian randomization method [84] will be employed to obtain a rank-one approximation solution to $\mathscr{P}_{\text{SDR}}(\mathcal{A})$.

Finally, we arrive at the robust group sparse beamforming algorithm as shown in Algorithm 5.

Algorithm 5: Robust Group Sparse Beamforming Algorithm

```
Step 0: Solve the group-sparsity inducing optimization problem $\mathscr{P}_{\text{GS}}$ (3.4.7) using
Algorithm 1.

    1. If it is infeasible, go to End.

    2. If it is feasible, obtain the solutions $\mathbf{Q}_m^{\star}$'s, calculate the ordering criterion (3.4.22),
       and sort them in the ascending order: $\theta_{\pi_1} \leq \cdots \leq \theta_{\pi_L}$, go to Step 1.

Step 1: Initialize $J_{\text{low}} = 0$, $J_{\text{up}} = L$, $i = 0$.
Step 2: Repeat

    1. Set $i \leftarrow \lfloor \frac{J_{\text{low}}+J_{\text{up}}}{2} \rfloor$.

    2. Solve problem $\mathscr{P}_{\text{PL}}(\mathcal{A}^{[i]})$ (3.4.27): if it is infeasible, set $J_{\text{up}} = i$; otherwise, set
       $J_{\text{low}} = i$.

Step 3: Until $J_{\text{up}} - J_{\text{low}} = 1$, obtain $J_0 = J_{\text{low}}$ and obtain the optimal active RRH set
$\mathcal{A}^{\star}$ with $\mathcal{A}^{\star} \cup \mathcal{J} = \mathcal{L}$ and $\mathcal{J} = \{\pi_1, \ldots, \pi_{J_0}\}$.
Step 4: Solve problem $\mathscr{P}_{\text{SDR}}(\mathcal{A}^{\star})$ (3.4.28), obtain the robust multicast beamforming
coefficients for the active RRHs.
End
```

Remark 4. *The proposed robust group sparse beamforming algorithm consists of three stages. In the first stage, we observe that the iterative reweighted-ℓ_2 algorithm converges in 20 iterations on average in all the simulated settings in this chapter, while it is interesting to analyze the convergence rate for this algorithm. In the second stage, to find the set of active RRHs, we only need to solve no more than $(1 + \lceil \log(1 + L) \rceil)$ convex feasibility problems (3.4.27) using the bi-section method. Finally, we need to solve problem (3.4.28) to determine the transmit beamforming coefficients for the fixed active RRHs.*

3.5 Simulation Results

In this section, we analyze the performance of the proposed robust group sparse beamforming algorithm. For illustration purposes, all the estimated channels $\hat{\mathbf{h}}_k$'s are modeled as spatially uncorrelated Rayleigh fading and the CSI errors are modeled as the elliptic model (3.2.6) with $\mathbf{Q}_k = \varepsilon_k^{-2}\mathbf{I}_N, \forall k$. We assume that each multicast group has the same number of MUs, i.e., $|\Omega_1| = |\Omega_2| = \cdots = |\Omega_M|$. The power amplifier efficiency coefficients are set to be $\eta_l = 25\%, \forall l$. The perturbed parameter ϵ in the iterative reweighted-ℓ_2 algorithm is set to be 10^{-3} and the algorithm will stop if either the difference between the objective values of consecutive iterations is less than 10^{-3} or it exceeds the predefined maximum iterations 20. Each point of the simulation results is averaged over 50 randomly generated channel realizations, except for Fig. 3.2, where we only report one typical channel realization.

3.5.1 Convergence of the Iterative Reweighted-ℓ_2 Algorithm

Consider a network with $L = 10$ 2-antennas RRHs and 3 multicast groups with 2 single-antenna MUs in each group, i.e., $|\Omega_m| = 2, \forall m$. All error radii ε_k's are set to be 0.05. The convergence of the iterative reweighted-ℓ_2 algorithm is demonstrated in Fig. 3.2 for a typical channel realization. This figure shows that the iterative reweighted-ℓ_2 algorithm converges very fast (less 20 iterations) in the simulated network size.

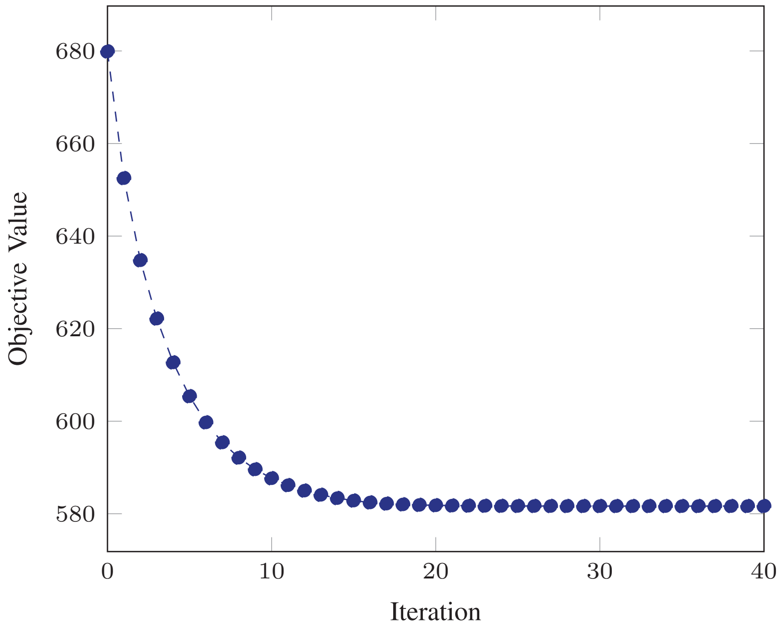

Figure 3.2: Convergence of the iterative reweighted-ℓ_2 algorithm.

Table 3.1: The Average Number of Active RRHs with Different Algorithms for Scenario One

Target SINR [dB]	0	2	4	6	8
Coordinated Beamforming	5.00	5.00	5.00	5.00	5.00
ℓ_1/ℓ_∞-Norm Algorithm	2.00	2.33	2.73	3.30	4.10
Proposed Algorithm	2.00	2.13	2.63	3.13	4.00
Exhaustive Search	2.00	2.07	2.60	3.10	4.00

3.5.2 Network Power Minimization

3.5.2.1 Scenario One

We first consider a network with $L = 5$ 2-antenna RRHs and $M = 2$ multicast groups each has 2 single-antenna MUs, i.e., $|\Omega_m| = 2, \forall m$. The relative fronthaul links power consumption are set to be $P_l^c = 5.6W, \forall l$. All error radii ε_k's are set to be 0.01. Fig. 3.3 demonstrates the average network power consumption with different target SINRs. The corresponding average number of active RRHs and average total transmit power consumption are showed in Table 3.1 and Table 3.2, respectively.

Specifically, Fig. 3.3 shows that the proposed robust group sparse beamforming algorithm achieves near-optimal values of network power consumption compared with the ones obtained by the exhaustive search algorithm via solving a sequence of problems (3.4.28). Furthermore, it is observed that the proposed algorithm outperforms the square of ℓ_1/ℓ_∞-norm

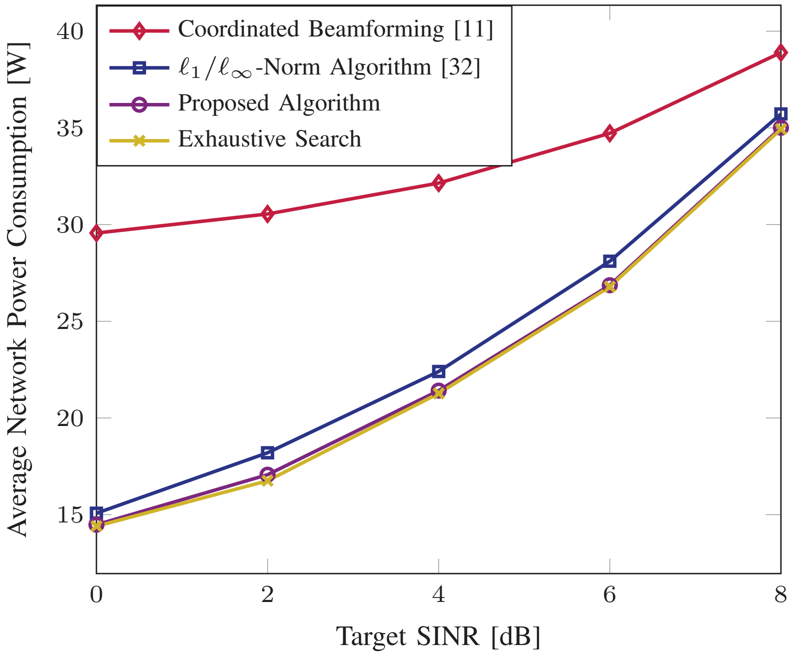

Figure 3.3: Average network power consumption versus target SINR for scenario one.

Table 3.2: The Average Total Transmit Power Consumption with Different Algorithms for Scenario One

Target SINR [dB]	0	2	4	6	8
Coordinated Beamforming	1.56	2.55	4.15	6.72	10.89
ℓ_1/ℓ_∞-Norm Algorithm	3.88	5.13	7.10	9.63	12.76
Proposed Algorithm	3.28	5.12	6.67	9.32	12.61
Exhaustive Search	3.20	5.18	6.71	9.43	12.54

based algorithm with sparsity pattern ordering rule in [43] in terms of network power minimization. Specifically, the objective function of the group-sparsity inducing optimization problem (3.4.7) will be replaced by $\mathcal{R} = \sum_{l_1=1}^{L}\sum_{l_2=1}^{L}\max_m \max_{n_{l_1}} \max_{n_{l_2}} |\mathbf{Q}_m(n_{l_1}, n_{l_2})|$ with $\mathbf{Q}_m(i,j)$ being the entry indexed by (i,j) in $\mathbf{Q}_m$. Then the RRH with smaller beamforming coefficients measured by the ℓ_∞-norm will have a higher priority to be switched off. In particular, Table 3.1 shows that the proposed algorithm can switch off more RRHs than the ℓ_1/ℓ_∞-norm based algorithm, which is almost the same as the exhaustive search algorithm. Besides, this table also verifies the group-sparsity assumption for the aggregative transmit beamformer v, i.e., the beamforming coefficients of the switched off RRHs are set to be zeros simultaneously. Meanwhile, Table 3.2 shows that the proposed algorithm can achieve higher transmit beamforming gains, yielding lower total transmit power consumption compared with the ℓ_1/ℓ_∞-norm based algorithm. The coordinated beamforming algorithm [18], which aims at only minimizing the total transmit power consumption with all the RRHs

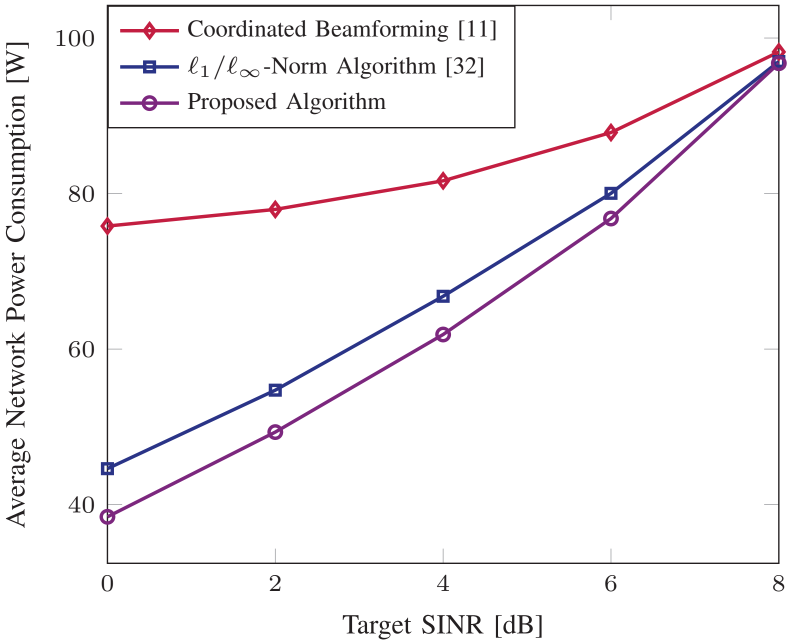

Figure 3.4: Average network power consumption versus target SINR for scenario two.

Table 3.3: The Average Relative Fronthaul Links Power Consumption with Different Algorithms for Scenario Two

Target SINR [dB]	0	2	4	6	8
Coordinated Beamforming	72.80	72.80	72.80	72.80	72.80
ℓ_1/ℓ_∞-Norm Algorithm	36.08	43.76	52.36	60.16	69.56
Proposed Algorithm	30.40	38.08	45.56	56.76	70.48

active, achieves the highest beamforming gain but with the highest relative fronthaul links power consumption.

Overall, Fig. 3.3, Table 3.1 and Table 3.2 show the effectiveness of the proposed robust group sparse beamforming algorithm to minimize the network power consumption.

3.5.2.2 Scenario Two

We then consider a larger-sized network with $L = 8$ 2-antenna RRHs and $M = 5$ multicast groups each has 2 single-antenna MUs, i.e., $|\Omega_m| = 2, \forall m$. The relative fronthaul links power consumption are set to be $P_l^c = [5.6 + (l-1)]W, \forall l$. All error radii ε_k's are set to be 0.05. Due to the high computational cost of the exhaustive search algorithm, we only simulate the ℓ_1/ℓ_∞-norm based algorithm and the proposed robust group sparse beamforming algorithm. Fig. 3.4, Table 3.3 and Table 3.4 show the average network power consumption, the average relative fronthaul link power consumption and the average total transmit power

Table 3.4: The Average Total Transmit Power Consumption with Different Algorithms for Scenario Two

Target SINR [dB]	0	2	4	6	8
Coordinated Beamforming	3.02	5.16	8.84	15.05	25.41
ℓ_1/ℓ_∞-Norm Algorithm	8.54	10.96	14.43	19.87	27.42
Proposed Algorithm	8.03	11.25	16.32	20.03	26.28

consumption versus SINRs with different algorithms, respectively. From Fig. 3.4, we see that the proposed robust beamforming algorithm achieves lower network power consumption compared with the ℓ_1/ℓ_∞-norm algorithm and the coordinated beamforming algorithm. In particular, Table 3.3 shows that proposed algorithm achieves much lower relative fronthaul links power consumption, thought with a little higher transmit power consumption at the moderate target SINR regimes. Compared with the ℓ_1/ℓ_∞-norm algorithm, the performance gain of the proposed algorithm is more prominent with low target SINRs.

Overall, all the simulation results illustrate the effectiveness of the proposed robust group sparse beamforming algorithm to control both the relative fronthaul power consumption and the RRH transmit power consumption with different network configurations.

3.6 Discussions

This chapter described a systematic way to develop computationally efficient algorithms based on the group-sparsity inducing penalty approach for the highly intractable network power minimization problem for multicast Cloud-RAN with imperfect CSI. A novel smoothed ℓ_p-minimization approach was proposed to induce the group-sparsity structure for the robust multicast beamformer, thereby guiding the RRH selection. The iterative reweighted-ℓ_2 minimization, PhaseLift method, and SDR technique based algorithms were developed to solve the group-sparsity inducing optimization problem, the feasibility problems in RRH selection procedure and the transmit beamformer design problem in the final stage, respectively. Simulation results illustrated the effectiveness of the proposed robust group sparse beamforming algorithm to minimize the network power consumption.

Several future directions of interest are listed as follows:

- Although the proposed SDP based robust group sparse beamforming algorithm has a polynomial time complexity, the computational cost of the interior-point method will the prohibitive when the dimensions of the SDP problems are large, such as in dense wireless networks. One may use the first-order method, e.g., the alternating direction method of multipliers (ADMM) [59, 96, 97] to seek modest accuracy solutions within reasonable time for the large-scale SDP problems [60].

- It is desirable to lay the theoretical foundations for the tightness of the group-sparsity inducing penalty approach for finding approximate solutions to the network power minimization problem as a mixed-integer non-linear optimization problem, and also for the tightness of PhaseLift method and SDR technique.
- It is interesting to apply the sparsity modeling framework to more mixed-integer nonlinear optimization problems, i.e., the joint user scheduling or admission and beamforming problems, which are essentially required to control the sparsity structure and the magnitude of the beamforming coefficients.

Chapter 4

Chance Constrained Programming for Dense Cloud-RAN with CSI Uncertainty

In this chapter we first propose a novel CSI acquisition method, called compressive CSI acquisition, to resolve the CSI challenge for Cloud-RAN. This new method can effectively reduce the CSI signaling overhead by obtaining instantaneous coefficients of only a subset of all the channel links. As a result, the BBU pool will obtain mixed CSI consisting of (imperfect) instantaneous values of some links and statistical CSI for the others. We then establish a generic stochastic coordinated beamforming (SCB) framework to deal with CSI uncertainty in the available mixed CSI. It provides flexibility in the channel uncertainty modeling (e.g., general stochastic model), while guaranteeing optimality in the transmission strategies. The SCB problem turns out to be a joint chance constrained program (JCCP) and is known to be highly intractable. In contrast to all of the previous algorithms for JCCP that can only find feasible but sub-optimal solutions, we propose a novel stochastic DC (difference-of-convex) programming algorithm based on successive convex approximation with optimality guarantee, which can serve as the benchmark for evaluating heuristic and sub-optimal algorithms as well as help investigate the effectiveness of the proposed CSI acquisition strategy. To further reduce the CSI acquisition overhead, a low rank matrix completion approach via Riemannian optimization is presented for topological interference management in the partially connected wireless networks. Proofs are deferred in Appendix C. The material in this chapter has been presented in part in [25, 71, 72].

4.1 Introduction

Network cooperation is a promising way to improve both the energy efficiency and spectral efficiency of wireless networks by sharing control information and/or user data [7]. Among all the cooperation strategies, jointly processing the user data can achieve the best performance by exploiting the benefits of a large-scale virtual MIMO system [65, 98]. This inspires a recent proposal of a new network architecture, i.e., cloud radio access network (Cloud-RAN) [10, 29], which will enable fully cooperative transmission/reception by moving all the baseband signal processing to a cloud data center. In order to fully exploit the benefits of cooperative networks and develop efficient transmission strategies (i.e., coordinated beamforming), channel state information (CSI) is often required. However, as the BBU pool can typically support hundreds of RRHs, obtaining full CSI in Cloud- RAN will deplete the radio resources, which can be regarded as the curse of dimensionality of Cloud-RAN. In particular, Lozano *et al.* [8] showed that the full cooperation gain is limited by the overhead of the orthogonal pilot-assisted channel estimation for uplink transmission in large-scale cooperative cellular networks. Huh *et al.* [9] quantified the downlink training overhead for large-scale network MIMO, which is regarded as the system overhead bottleneck even if the uplink feedback overhead is ignored. Therefore, the development of novel and effective CSI acquisition methods is critical for the practical implementation of the fully cooperative Cloud-RAN. We thus propose a novel CSI acquisition method, called *compressive CSI acquisition*, which can systematically reduce both the pilot training overhead and uplink feedback overhead. Specifically, it is achieved by exploiting the sparsity of the large-scale fading coefficients and determining, before the training phase, the channel coefficients needed to obtain their instantaneous values. As a result, the BBU pool will obtain the mixed CSI, including a subset of instantaneous CSI and statistical CSI for the other channel coefficients.

However, the channel knowledge uncertainty in the mixed CSI due to the partial and imperfect CSI brings technical challenges in system performance optimization. To address such challenges brought by the channel knowledge uncertainty, one may either adopt a robust optimization formulation [82] or stochastic optimization formulation [81]. Specifically, for the robust formulation, the channel knowledge uncertainty model is deterministic and set-based [86]. Thus, the corresponding transmission strategies aim at guaranteeing the worst-case performance over the entire uncertainty set. The primary advantage of robust formulation is the computational tractability [78]. However, the worst-case formulation might be over-conservative [78], as the probability of the worst case could be very small [17]. Meanwhile, how to model the uncertainty set is also challenging [87]. On the other hand, in the stochastic optimization formulation, the channel knowledge is modeled by a probabilistic description. Thus, the corresponding transmission strategies seek to immunize a solution against the stochastic uncertainty in a probabilistic sense [99–103]. The freedom of the probabilistic robustness can provide improved system performance [102] and provide a tradeoff between the conservativeness and probability guarantee [78].

Motivated by the fact that most wireless systems can tolerate occasional outages in the quality-of-service (QoS) requirements [99–101], in this chapter, we propose a stochastic coordinated beamforming (SCB) framework to minimize the total transmit power while guaranteeing the system probabilistic QoS requirements. In this framework, we only assume that the distribution information of the channel uncertainty is available, but without any further structural modeling assumptions (e.g., adopting the ellipsoidal error model for robust design [86] or assuming complex Gaussian random distribution for the channel errors [101–103] for stochastic design). In spite of the distinct advantages, including the design flexibility and the insights obtained by applying the SCB framework to handle the CSI uncertainty, it falls into a joint chance constrained program (JCCP) [81], which is known to be highly intractable [104]. All the available algorithms (e.g., the scenario approach [71, 100, 105] and the Bernstein approximation method [99, 101]) can only find *feasible but suboptimal* solutions without any optimality guarantee.

In contrast, in this chapter, we propose a novel stochastic DC programming algorithm, which can find the globally optimal solution if the original SCB problem is convex and find a locally optimal solution if the problem is non-convex. The main idea of the algorithm is to reformulate the system probabilistic QoS constraint as a DC constraint, producing an equivalent stochastic DC program. Although the DC programming problem is still non-convex, it has the algorithmic advantage and can be efficiently solved by the successive convex approximation algorithm [104, 106].

The main computational complexity of the proposed algorithm comes from solving a large-sized sample problem with the Monte Carlo approach at each iteration. This makes such an approach inapplicable in large-size networks. However, the proposed stochastic DC programming algorithm gives a first attempt to solve a highly-intractable and highly-complicated problem with optimality guarantee, while existing algorithms fail to possess the optimality feature. Therefore, it can serve as a performance benchmark for evaluating other suboptimal and heuristic algorithms.

4.1.1 Related Works

The chance constrained programming has recently received emerging interests in designing efficient resource allocation strategies in communication networks by leveraging the distribution information of uncertain channel knowledge [71, 99–103, 107]. However, due to the high intractability of the underlying chance or probabilistic constraints (e.g., it is difficult to justify the convexity or provide analytical expressions), even finding a feasible solution is challenging. Therefore, it is common to approximate the probability constraint to yield computationally tractable and deterministic formulations. One way is to approximate the chance constraints using analytical functions, which, however, often requires further assumptions on the distribution of the uncertain channel knowledge (e.g., complex Gaussian distributions for Bernstein-type inequality approximation [101, 103] or the affine constraint functions in perturbations for Bernstein approximation [99, 102, 108]). The other way is to use the Monte Carlo simulation approach to approximate the chance constraints (e.g., the scenario approach [71, 100, 105] and the conditional-value-at-risk (CVaR) [109]). However, all the above approaches only seek conservative approximations to the original problem. Thus, it is difficult to prove the optimality and quantify the conservativeness of the obtained solutions.

Hong *et. al* [104] recently made a breakthrough on providing optimality of the highly intractable joint chance constrained programming problems for the first time. However, the convexity of the functions in the chance constraint is required. Our proposed stochastic DC programming algorithm is inspired by the ideas in [104]. Unfortunately, the functions in the chance constraint in our problem are non-convex, and thus, we cannot directly apply the algorithm in [104]. Instead, by exploiting the special structure of the functions in the chance constraint, we equivalently reformulate the chance constraint into a DC constraint. The resulting DC program is further supported by efficient algorithms. Thus, we extend the work [104] by removing the convexity assumption on the functions in the chance constraint. Furthermore, to improve the convergence rate, instead of fixing the approximation parameter as in [104], a joint approximation method is proposed.

4.1.2 Contributions

In this chapter, we provide a general framework to design optimal transmission strategies with CSI uncertainty for wireless cooperative networks. The major contributions are summarized as follows:

1. We propose a novel compressive CSI acquisition method that can effectively reduce both the pilot training overhead and uplink feedback overhead by exploiting the sparsity of the large-scale fading coefficients.

2. We establish a general SCB framework to cope with the uncertainty in the available channel knowledge, which intends to minimize the total transmit power with a system probabilistic QoS guarantee. This framework only requires the distribution information of the uncertain channel coefficients. Thus, it enjoys the flexibility in modeling channel knowledge uncertainty without any further structural assumptions. The SCB problem is then formulated as a JCCP problem.

3. We develop a novel stochastic DC programming algorithm to solve the SCB problem, which will converge to the globally optimal solution if the SCB problem is convex or a locally optimal solution if it is non-convex. The proposed stochastic DC programming algorithm can be regarded as the first attempt to guarantee the optimality for the solutions of JCCP without the convexity assumption on functions in the chance constraint [104], while the available algorithms (i.e., the scenario approach and the Bernstein approximation method) for JCCP can only find a feasible solution without any optimality guarantee.

4. The proposed SCB framework is simulated in Section 4.4. In particular, the convergence, conservativeness, stability and performance gains of the proposed algorithm are illustrated.

4.1.3 Organization

The remainder of the chapter is organized as follows. Section 4.2 presents the system model and problem formulation, followed by the problem analysis. In Section 4.3, the stochastic DC programming algorithm is developed. Simulation results will be presented in Section

4.4. Finally, conclusions and discussions are presented in Section 4.5. To keep the main text clean and free of technical details, we divert most of the proofs to the appendix.

4.2 System Model and Problem Formulation

We consider a fully cooperative Cloud-RAN[1] with L remote radio heads (RRHs), where the l-th RRH is equipped with N_l antennas, and there are K single-antenna mobile users (MUs). The centralized signal processing is performed at a central processor, e.g., at the baseband unit (BBU) pool in Cloud-RAN [29]. The propagation channel from the l-th RRH to the k-th MU is denoted as $\mathbf{h}_{kl} \in \mathbb{C}^{N_l}, 1 \leq k \leq K, 1 \leq l \leq L$. We focus on the downlink transmission, for which the joint signal processing is more challenging. The received signal $y_k \in \mathbb{C}$ at MU k is given by

$$y_k = \sum_{l=1}^{L} \mathbf{h}_{kl}^{\mathsf{H}} \mathbf{v}_{lk} s_k + \sum_{i \neq k} \sum_{l=1}^{L} \mathbf{h}_{kl}^{\mathsf{H}} \mathbf{v}_{li} s_i + n_k, \forall k, \tag{4.2.1}$$

where s_k is the encoded information symbol for MU k with $\mathbb{E}[|s_k|^2] = 1$, $\mathbf{v}_{lk} \in \mathbb{C}^{N_l}$ is the transmit beamforming vector from the l-th RRH to the k-th MU, and $n_k \sim \mathcal{CN}(0, \sigma_k^2)$ is the additive Gaussian noise at MU k. We assume that s_k's and n_k's are mutually independent and all the users apply single user detection. The corresponding signal-to-interference-plus-noise ratio (SINR) for MU k is given by

$$\Gamma_k(\mathbf{v}, \mathbf{h}_k) = \frac{|\mathbf{h}_k^{\mathsf{H}} \mathbf{v}_k|^2}{\sum_{i \neq k} |\mathbf{h}_k^{\mathsf{H}} \mathbf{v}_i|^2 + \sigma_k^2}, \forall k, \tag{4.2.2}$$

where $\mathbf{h}_k \triangleq [\mathbf{h}_{k1}^T, \mathbf{h}_{k2}^T, \ldots, \mathbf{h}_{kL}^T]^T = [h_{kn}]_{1 \leq n \leq N} \in \mathbb{C}^N$ with $N = \sum_{l=1}^{L} N_l$, $\mathbf{v}_k \triangleq [\mathbf{v}_{1k}^T, \mathbf{v}_{2k}^T, \ldots, \mathbf{v}_{Lk}^T]^T \in \mathbb{C}^N$ and $\mathbf{v} \triangleq [\mathbf{v}_k]_{k=1}^K \in \mathbb{C}^{NK}$. The beamforming vectors $\mathbf{v}_{lk}$'s are designed to minimize the total transmit power while satisfying the QoS requirements for all the MUs. The beamformer design problem can be formulated as

$$\begin{aligned} \mathscr{P}_{\text{Full}} : \underset{\mathbf{v} \in \mathcal{V}}{\text{minimize}} \ & \sum_{l=1}^{L} \sum_{k=1}^{K} \|\mathbf{v}_{lk}\|^2 \\ \text{subject to } & \Gamma_k(\mathbf{v}, \mathbf{h}_k) \geq \gamma_k, \forall k, \end{aligned} \tag{4.2.3}$$

[1] The proposed framework can be easily extended to more general cooperation scenarios as shown in [86].

where γ_k is the target SINR for MU k, and the convex set $\mathcal{V}$ is the feasible set of $\mathbf{v}_{lk}$'s that satisfy the per-RRH power constraints:

$$\mathcal{V} \triangleq \left\{ \mathbf{v}_{lk} \in \mathbb{C}^{N_l} : \sum_{k=1}^{K} \|\mathbf{v}_{lk}\|^2 \leq P_l, \forall l, k \right\}, \tag{4.2.4}$$

with P_l as the maximum transmit power of the RRH l.

The problem $\mathscr{P}_{\text{Full}}$ can be reformulated as a second-order conic programming (SOCP) problem, which is convex and can be solved efficiently (e.g., via the interior-point method). Please refer to [29] for details. Such coordinated beamforming can significantly improve the network energy efficiency. However, solving problem $\mathscr{P}_{\text{Full}}$ requires full and perfect CSI available at the central processor. In practice, inevitably there will be uncertainty in the available channel knowledge. Such uncertainty may originate from various sources, e.g., training based channel estimation [70], limited feedback [69], delays [73, 74], hardware deficiencies [86] and partial CSI acquisition [71, 107]. In the next subsection, we will provide a generic stochastic model for the CSI uncertainty.

4.2.1 Compressive CSI Acquisition

In this subsection, we will present the proposed compressive CSI acquisition method. The main idea is to determine the most "relevant" channel links that are critical for performance before the training phase, and then only the coefficients of these links will be obtained during CSI training.

4.2.1.1 CSI Overhead Reduction

We will first quantify the CSI overhead for both training and feedback phases. The discussion will be general, as we do not make any assumption on the duplexing mode. With compressive CSI acquisition, only part of the channel coefficients will be obtained. For user k, define a set Ω_k of size $\mathcal{D}_k$ ($0 \leq \mathcal{D}_k \leq N$) such that the channel coefficients h_{kn} will be obtained during CSI training if and only if $(k, n) \in \Omega_k$. Given $\Omega_k, \forall k \in \mathcal{K}$, and assuming that orthogonal pilot symbols are used for downlink training, then the *training overhead* is proportional to $\max_{1 \leq k \leq K} \mathcal{D}_k$. This is justified by modeling the orthogonal pilot allocation problem as a graph coloring problem on an unweighted bipartite graph $G = (\mathcal{N}, \mathcal{K}, \mathcal{E})$, where $\mathcal{N} \triangleq \{1, 2, \ldots, N\}$ is the set of transmit antennas, $\mathcal{K}$ is the set of MUs. In this case, an edge

$e \in \mathcal{E}$ exists if $(n, k) \in \Omega$ where $\Omega \triangleq \Omega_1 \cup \cdots \cup \Omega_K$, $n \in \mathcal{N}$ and $k \in \mathcal{K}$. By the Vizing's theorem [110], the minimum number of colors assigned to the edges of a bipartite graph so that no two adjacent edges have the same color (corresponding to no mutual interference for pilot training) is its maximum degree. Therefore, the required number of orthogonal pilots is $\max_{1\leq k\leq K} \mathcal{D}_k$, which quantifies the training overhead. For the *uplink feedback overhead*, which is needed for the FDD system, given $\Omega_k, \forall k \in \mathcal{K}$, in order to guarantee a constant CSI distortion d, the total CSI feedback bits should scale as $\mathcal{O}(\sum_{k=1}^{K} \mathcal{D}_k \log(1/d))$ [111]. Therefore, $\sum_{k=1}^{K} \mathcal{D}_k$ is a good indicator for the feedback overhead.

From the above discussion, the CSI overhead is controlled by the sizes of Ω_k's. We propose to determine the sets Ω_k's before the pilot training phase, so that the CSI overhead can be effectively controlled, especially compared to the channel coherence time. This approach is fundamentally different from the conventional limited feedback wireless systems, which require knowledge of all the channel coefficients before the feedback. We will refer to this method as *compressive CSI acquisition*, as it is similar to "compressive sensing" [35], where useful information can be extracted with much fewer samples than obtaining complete data. This "compression" idea will be critical for the design of large-scale wireless cooperative networks, as there is no way to collect all the side information before actual processing. We should rather try to directly extract the relevant information so that efficient communication can be achieved.

4.2.1.2 CSI Selection Rule

Given the size constraints for the sets Ω_k's (i.e, $|\Omega_k| = \mathcal{D}_k, \forall k \in \mathcal{K}$), how to determine the indices of each set is a combinatorial optimization problem, which is intractable in general. In this chapter, we propose a practical CSI selection rule by exploiting the *sparsity* of the large-scaling fading coefficients. Denote the support of the channel vector $\mathbf{h}_k = [h_{kn}]_{1\leq n\leq N} \in \mathbb{C}^N$ as $\|\mathbf{h}_k\|_{\ell_0}(\lambda) \triangleq |\{n \in \mathbb{Z}^N : |h_{kn}| \geq \lambda\}|$, where $\lambda > 0$ is a pre-chosen threshold. Due to path loss and large-scale fading, the size of the support of each channel vector can be much smaller than N, which is the number of coefficients if full CSI is to be obtained. This property was exploited in [24] to measure the partial connectivity of the channel links for topological interference management, where the receivers will compare powers of the estimated channel links with a pre-chosen threshold to determine which channel links are strong. However, the approach in [24] requires that each receiver obtains all the instantaneous channel coefficients to measure the channel sparsity, which cannot reduce the downlink training overhead.

By Chebyshev's inequality, $\Pr\{|h_{kn}| \geq \lambda\} \leq \frac{\theta_{kn}^2}{\lambda^2}$ with $\theta_{kn} = \sqrt{\mathbb{E}[|h_{kn}|^2]}$ representing the large-scale fading coefficient of the channel link h_{kn}, the following support of the large-scaling fading coefficient vector $\boldsymbol{\theta}_k = [\theta_{kn}]_{1\leq n\leq N}$ can be regarded as a good estimate of the sparsity of the channel coefficients,

$$\|\boldsymbol{\theta}_k\|_{\ell_0}(\bar{\lambda}) \triangleq |\{n \in \mathbb{Z}^N : |\theta_{kn}| \geq \bar{\lambda}\}|, \tag{4.2.5}$$

where $\bar{\lambda}$ is a pre-chosen parameter. A similar idea on exploiting the sparsity of the large-scaling fading coefficients was presented in [112] under a linear equal-spaced transmit antenna topology model.

Based on the above discussion, we propose in this chapter, the following sparsity based CSI selection rule to determine the sets Ω_k's.

Sparsity Based CSI Selection Rule Given the CSI overhead constraints $|\Omega_k| = \mathcal{D}_k, \forall k \in \mathcal{K}$, rearranging the entries of the vector $\boldsymbol{\theta}_k = [\theta_{kn}]_{1\leq n\leq N}$ with decreased magnitudes $|\theta_{k(1)}| \geq |\theta_{k(2)}| \geq \cdots \geq |\theta_{k(N)}|$, then the set Ω_k is determined by including the indices of the $\mathcal{D}_k$ largest entries of the vector $\boldsymbol{\theta}_k$.

Remark 5. *The proposed sparsity based CSI selection rule is easy to implement. It is possible to improve performance by developing more sophisticated selection rules. For example, a different selection rule based on statistical CSI was proposed in [71], which, however, does not have any performance guarantee and is with higher implementation complexity. A full investigation on this aspect will be left to our future work, while, in this chapter, we focus on stochastic coordinated beamforming to handle mixed CSI.*

4.2.2 Stochastic Coordinated Beamforming with Probability QoS Guarantee

With compressive CSI acquisition, the BBU pool will obtain mixed CSI, i.e., with (imperfect) instantaneous channel coefficients for links indexed in the set Ω and statistical CSI for the other channel links. The uncertainty in the available CSI brings a new technical challenge for the system design. To guarantee performance, we impose a probabilistic QoS constraint, specified as follows

$$\Pr\left\{\Gamma_k(\mathbf{v}, \mathbf{h}_k) \geq \gamma_k, \forall k\right\} \geq 1 - \epsilon, \tag{4.2.6}$$

where the distribution information of $\mathbf{h}_k$'s is known, $0 < \epsilon < 1$ indicates that the system should guarantee the QoS requirements for all the MUs simultaneously with probability of at least $1 - \epsilon$. The probability is calculated over all the random vectors $\mathbf{h}_k$'s. The SCB is thus formulated to minimize the total transmit power while satisfying the system probabilistic QoS constraint (4.2.6):

$$\begin{aligned} \mathscr{P}_{\text{SCB}} : \underset{\mathbf{v}\in\mathcal{V}}{\text{minimize}} & \sum_{l=1}^{L}\sum_{k=1}^{K} \|\mathbf{v}_{lk}\|^2 \\ \text{subject to} & \Pr\left\{\Gamma_k(\mathbf{v}, \mathbf{h}_k) \geq \gamma_k, \forall k\right\} \geq 1 - \epsilon, \end{aligned} \tag{4.2.7}$$

which is a joint chance constrained program (JCCP) [81, 104] and is known to be intractable in general.

4.2.2.1 Problem Analysis

There are two major challenges in solving $\mathscr{P}_{\text{SCB}}$. Firstly, the chance (or probabilistic) constraint (4.2.6) has no closed-form expression in general and thus is difficult to evaluate. Secondly, the convexity of the feasible set formed by the probabilistic constraint is difficult to verify. The general idea to handle such a constraint is to seek a *safe and tractable approximation*. "Safe" means that the feasible set formed by the approximated constraint is a subset of the original feasible set, while "tractable" means that the optimization problem over the approximated feasible set should be computationally efficient (e.g., relaxed to a convex program).

A natural way to form a computationally tractable approximation is the scenario approach [105]. Specifically, the chance constraint (4.2.6) will be approximated by the following KJ sampling constraints:

$$\Gamma_k(\mathbf{v}, \mathbf{h}_k^j) \geq \gamma_k, 1 \leq j \leq J, \forall k, \tag{4.2.8}$$

where $\mathbf{h}_k^j \in \mathbb{C}^N$ is the j-th realization of the random vector $\mathbf{h}_k \in \mathbb{C}^N$. Let $\mathbf{h}^j = [\mathbf{h}_k^j]_{1\leq k\leq K}$, then $\mathbf{h}^1, \mathbf{h}^2, \ldots, \mathbf{h}^J$ are J independent realizations of the random vector $\mathbf{h} \in \mathbb{C}^{NK}$. The

SCB problem $\mathscr{P}_{\text{SCB}}$ thus can be approximated by a convex program based on the constraints (4.2.8). This approach can find a feasible solution with a high probability, for which more details can be found in [71]. An alternative way is to derive an analytical upper bound for the chance constraint based on the Bernstein-type inequality [101, 103, 108], resulting in a deterministic convex optimization problem. The Bernstein approximation based approach thus can find a feasible but suboptimal solution.

Although the above methods have the advantage of computational efficiency due to the convex approximation, the common drawback of all these algorithms is the conservativeness due to the "safe" approximation. Furthermore, it is also difficult to quantify the qualities of the solutions generated by the algorithms. This motivates us to seek a novel approach to find a more reliable solution to the problem $\mathscr{P}_{\text{SCB}}$. In this chapter, we will propose a stochastic DC programming algorithm to find the globally optimal solution to $\mathscr{P}_{\text{SCB}}$ if the problem is convex and a locally optimal solution if it is non-convex, which can be regarded as the first attempt to guarantee the optimality for the solutions of the JCCP (4.2.7).

4.3 Stochastic DC Programming Algorithm

In this section, we propose a stochastic DC programming algorithm to solve the problem $\mathscr{P}_{\text{SCB}}$. We will first propose a DC programming reformulation for the problem $\mathscr{P}_{\text{SCB}}$, which will then be solved by stochastic successive convex optimization.

4.3.1 DC Programming Reformulation for the SCB Problem

The main challenge of the SCB problem $\mathscr{P}_{\text{SCB}}$ is the intractable chance constraint. In order to overcome the difficulty, we will propose a DC programming reformulation that is different from all the previous conservative approximation methods. We first propose a DC approximation to the chance constraint (4.2.6). Specifically, the QoS constraints $\Gamma_k(\mathbf{v}, \mathbf{h}_k) \geq \gamma_k$ can be rewritten as the following DC constraints [113]

$$d_k(\mathbf{v}, \mathbf{h}_k) \triangleq c_{k,1}(\mathbf{v}_{-k}, \mathbf{h}_k) - c_{k,2}(\mathbf{v}_k, \mathbf{h}_k) \leq 0, \forall k, \tag{4.3.1}$$

where $\mathbf{v}_{-k} \triangleq [\mathbf{v}_i]_{i \neq k}$, and both $c_{k,1}(\mathbf{v}_{-k}, \mathbf{h}_k) \triangleq \sum_{i \neq k} \mathbf{v}_i^{\mathsf{H}} \mathbf{h}_k \mathbf{h}_k^{\mathsf{H}} \mathbf{v}_i + \sigma_k^2$ and $c_{k,2}(\mathbf{v}_k, \mathbf{h}_k) \triangleq \frac{1}{\gamma_k} \mathbf{v}_k^{\mathsf{H}} \mathbf{h}_k \mathbf{h}_k^{\mathsf{H}} \mathbf{v}_k$ are convex quadratic functions in $\mathbf{v}$. Therefore, $d_k(\mathbf{v}, \mathbf{h}_k)$'s are DC functions in $\mathbf{v}$. Then, the chance constraint (4.2.6) can be rewritten as $f(\mathbf{v}) \leq \epsilon$, with $f(\mathbf{v})$ given by

$$f(\mathbf{v}) = 1 - \Pr\left\{\Gamma_k(\mathbf{v}, \mathbf{h}_k) \geq \gamma_k, \forall k\right\} = \Pr\left\{\left(\max_{1\leq k\leq K} d_k(\mathbf{v}, \mathbf{h}_k)\right) > 0\right\}$$
$$= \mathbb{E}\left[1_{(0,+\infty)}\left(\max_{1\leq k\leq K} d_k(\mathbf{v}, \mathbf{h}_k)\right)\right], \tag{4.3.2}$$

where $1_{\mathcal{A}}(z)$ is an indicator of set $\mathcal{A}$. That is, $1_{\mathcal{A}}(z) = 1$ if $z \in \mathcal{A}$ and $1_{\mathcal{A}}(z) = 0$, otherwise. The indicator function makes $f(\mathbf{v})$ non-convex in general.

The conventional approach to deal with the non-convex indicator function is to approximate it by a convex function, yielding a conservative convex approximation. For example, using $\exp(z) \geq 1_{(0,+\infty)}(z)$ will yield the Bernstein approximation [108]. Applying $[\nu + z]^+/\nu \geq 1_{(0,+\infty)}(z)$, $\nu > 0$ will obtain a conditional-value-at-risk (CVaR) type approximation [108]. Although these approximations might enjoy the advantage of being convex, all of them are conservative and will lose optimality for the solution of the original problem. More specifically, only the feasibility of the solutions can be guaranteed with these approximations.

To find a better approximation to $f(\mathbf{v})$ in (4.3.2), in this chapter, we propose to use the following non-convex function [104, Fig. 2] to approximate the indicator function $1_{(0,+\infty)}(z)$ in (4.3.2):

$$\psi(z, \nu) = \frac{1}{\nu}[(\nu + z)^+ - z^+], \nu > 0, \tag{4.3.3}$$

which is a DC function [113] in z. Although the DC function is not convex, it does have many advantages. In particular, Hong *et al.* [104] proposed to use this DC function to approximate the chance constraint assuming that the functions in the chance constraint are convex, resulting in a DC program reformulation. However, we cannot directly extend their results for our problem, since the functions $d_k(\mathbf{v}, \mathbf{h}_k)$'s in (4.3.1) are non-convex. Fortunately, we can still adopt the DC function $\psi(z, \nu)$ in (4.3.3) to approximate the chance constraint based on the following lemma.

Lemma 1 (DC Approximation for the Chance Constraint)**.** *The non-convex function $f(\mathbf{v})$ in (4.3.2) has the following conservative DC approximation for any $\nu > 0$,*

$$\hat{f}(\mathbf{v}, \nu) = \mathbb{E}\left[\psi\left(\max_{1\leq k\leq K} d_k(\mathbf{v}, \mathbf{h}_k), \nu\right)\right] = \frac{1}{\nu}[u(\mathbf{v}, \nu) - u(\mathbf{v}, 0)], \nu > 0, \tag{4.3.4}$$

where

$$u(\mathbf{v}, \nu) = \mathbb{E}\left[\max_{1\le k\le K+1} s_k(\mathbf{v}, \mathbf{h}, \nu)\right], \tag{4.3.5}$$

is a convex function and the convex quadratic functions $s_k(\mathbf{v}, \mathbf{h}, \nu)$'s are given by

$$s_k(\mathbf{v}, \mathbf{h}, \nu) \triangleq \nu + c_{k,1}(\mathbf{v}_{-k}, \mathbf{h}_k) + \sum_{i\neq k} c_{i,2}(\mathbf{v}_i, \mathbf{h}_i), \forall k, \tag{4.3.6}$$

and $s_{K+1}(\mathbf{v}, \mathbf{h}, \nu) \triangleq \sum_{i=1}^{K} c_{i,2}(\mathbf{v}_i, \mathbf{h}_i)$ is a convex quadratic function too.

Proof. Please refer to Appendix C.1 for details. □

Based on the DC approximation function $\hat{f}(\mathbf{v}, \nu)$, we propose to solve the following problem to approximate the original SCB problem $\mathscr{P}_{\text{SCB}}$:

$$\begin{aligned} \mathscr{P}_{\text{DC}} : \underset{\mathbf{v}\in\mathcal{V}}{\text{minimize}} & \sum_{l=1}^{L}\sum_{k=1}^{K} \|\mathbf{v}_{lk}\|^2 \\ \text{subject to} & \inf_{\nu>0} \hat{f}(\mathbf{v}, \nu) \le \epsilon, \end{aligned} \tag{4.3.7}$$

where $\inf_{\nu>0} \hat{f}(\mathbf{v}, \nu)$ is the most accurate approximation function to $f(\mathbf{v})$. Program $\mathscr{P}_{\text{DC}}$ is a DC program with the convex set $\mathcal{V}$, the convex objective function, and the DC constraint function [113]. One major advantage of the DC approximation $\mathscr{P}_{\text{DC}}$ is the equivalence to the original problem $\mathscr{P}_{\text{SCB}}$. That is, the DC approximation will not lose any optimality of the solution of the SCB problem $\mathscr{P}_{\text{SCB}}$, as stated in the following theorem.

Theorem 3 (DC Programming Reformulation)**.** *The DC programming problem $\mathscr{P}_{DC}$ in (4.3.7) is equivalent to the original SCB problem $\mathscr{P}_{SCB}$.*

Proof. Please refer to Appendix C.2 for details. □

Based on this theorem, in the sequel, we focus on how to solve the problem $\mathscr{P}_{\text{DC}}$.

4.3.2 Optimality of Joint Optimization over $\mathbf{v}$ and κ

As the constraint in $\mathscr{P}_{\text{DC}}$ itself is an optimization problem, it is difficult to be solved directly. To circumvent this difficulty, by observing that $\hat{f}(\mathbf{v}, \nu)$ is nondecreasing in ν for $\nu > 0$, as indicated in (C.2.6), one way is to solve the following κ-approximation problem [104]

$$\begin{aligned} &\underset{\mathbf{v}\in\mathcal{V}}{\text{minimize}} \sum_{l=1}^{L}\sum_{k=1}^{K}\|\mathbf{v}_{lk}\|^2 \\ &\text{subject to } u(\mathbf{v},\kappa)-u(\mathbf{v},0)\leq\kappa\epsilon, \end{aligned} \tag{4.3.8}$$

for any fixed small enough parameter $\kappa > 0$ to approximate the original problem $\mathscr{P}_{\text{DC}}$. However, an extremely small κ might cause numerical stability issues and might require more time to solve the subproblems that will be developed later [104].

We notice that, by regarding κ as an optimization variable, problem (4.3.8) is still a DC program, as the function $\mu(\mathbf{v},\kappa)$ is jointly convex in $(\mathbf{v},\kappa)$. Therefore, we propose to solve the following joint approximation optimization problem by treating κ as an optimization variable

$$\begin{aligned} \tilde{\mathscr{P}}_{\text{DC}} : &\underset{\mathbf{v}\in\mathcal{V},\kappa>0}{\text{minimize}} \sum_{l=1}^{L}\sum_{k=1}^{K}\|\mathbf{v}_{lk}\|^2 \\ &\text{subject to } [u(\mathbf{v},\kappa)-\kappa\epsilon]-u(\mathbf{v},0)\leq 0. \end{aligned} \tag{4.3.9}$$

The following proposition implies that the joint approximation problem $\tilde{\mathscr{P}}_{\text{DC}}$ can enhance the performance of problem (4.3.8).

Proposition 4 (Effectiveness of Joint Approximation)**.** *Denote the optimal value of the problem (4.3.8) with a fixed $\kappa = \hat{\kappa}$ and that of the problem $\tilde{\mathscr{P}}_{DC}$ as $V^\star(\hat{\kappa})$ and $\tilde{V}^\star$, respectively, then we have $\tilde{V}^\star \leq V^\star(\hat{\kappa})$.*

Proof. Define the feasible region of problem $\tilde{\mathscr{P}}_{\text{DC}}$ as

$$\mathcal{D} \triangleq \{\mathbf{v}\in\mathcal{V},\kappa>0 : [u(\mathbf{v},\kappa)-\kappa\epsilon]-u(\mathbf{v},0)\leq 0\}. \tag{4.3.10}$$

The projection of $\mathcal{D}$ on the set $\mathcal{V}$ is given by

$$\bar{\mathcal{D}} = \{\mathbf{v}\in\mathcal{D} : \exists\kappa>0, \text{s.t.}(\mathbf{v},\kappa)\in\mathcal{D}\}. \tag{4.3.11}$$

Therefore, by fixing $\kappa = \hat{\kappa}$, any feasible solution in problem (4.3.8) belongs to the set $\mathcal{D}$. Hence, the feasible set of the optimization problem (4.3.8) is a subset of $\bar{\mathcal{D}}$. As a result, solving $\tilde{\mathscr{P}}_{\text{DC}}$ can achieve a smaller minimum value with a larger feasible region. □

Define the deviation of a given set $\mathcal{A}_1$ from another set $\mathcal{A}_2$ as [81]

$$\mathbb{D}(\mathcal{A}_1, \mathcal{A}_2) = \sup_{x_1 \in \mathcal{A}_1} \left(\inf_{x_2 \in \mathcal{A}_2} \|x_1 - x_2\| \right), \tag{4.3.12}$$

then we have the following theorem indicating the optimality of the joint approximation program $\tilde{\mathscr{P}}_{\text{DC}}$.

Theorem 4 (Optimality of Joint Approximation)**.** *Denote the set of the optimal solutions and optimal values of problems $\tilde{\mathscr{P}}_{DC}$, $\mathscr{P}_{SCB}$ and the problem (4.3.8) with a fixed $\kappa = \hat{\kappa}$ as $(\tilde{\mathcal{P}}^\star, \tilde{V}^\star)$, $(\mathcal{P}^\star, V^\star)$ and $(\mathcal{P}^\star(\hat{\kappa}), V^\star(\hat{\kappa}))$, respectively, then*

$$\lim_{\hat{\kappa} \searrow 0} (V^\star(\hat{\kappa}) - \tilde{V}^\star) = \lim_{\hat{\kappa} \searrow 0} (V^\star(\hat{\kappa}) - V^\star) = 0, \tag{4.3.13}$$

and

$$\lim_{\hat{\kappa} \searrow 0} \mathbb{D}(\mathcal{P}^\star(\hat{\kappa}), \tilde{\mathcal{P}}^\star) = \lim_{\hat{\kappa} \searrow 0} \mathbb{D}(\mathcal{P}^\star(\hat{\kappa}), \mathcal{P}^\star) = 0. \tag{4.3.14}$$

Proof. Based on Proposition 4, the proof follows [104, Theorem 2]. □

Based on Theorem 4, we can thus focus on solving program $\tilde{\mathscr{P}}_{\text{DC}}$. Although $\tilde{\mathscr{P}}_{\text{DC}}$ is still a non-convex DC program, it has an algorithmic advantage, as will be presented in the next subsection.

4.3.3 Successive Convex Approximation Algorithm

In this subsection, we will present a successive convex approximation algorithm [104, 106] to solve the non-convex joint approximation program $\tilde{\mathscr{P}}_{\text{DC}}$. We will prove in Theorem 5 that this algorithm still preserves the optimality properties, i.e., achieving the Karush-Kuhn-Tucker (KKT) pair of the non-convex program $\tilde{\mathscr{P}}_{\text{DC}}$. The main idea is to upper bound the non-convex DC constraint function in $\tilde{\mathscr{P}}_{\text{DC}}$ by a convex function at each iteration. Specifically, at the j-th iteration, given the vector $(\mathbf{v}^{[j]}, \kappa^{[j]}) \in \mathcal{D}$, for the convex function $u(\mathbf{v}, 0)$, we have

$$u(\mathbf{v}, 0) \geq u(\mathbf{v}^{[j]}, 0) + 2\langle \nabla_{\mathbf{v}^*} u(\mathbf{v}^{[j]}, 0), \mathbf{v} - \mathbf{v}^{[j]} \rangle, \tag{4.3.15}$$

where $\langle \mathbf{a}, \mathbf{b} \rangle \triangleq \Re(\mathbf{a}^{\mathsf{H}}\mathbf{b})$ for any $\mathbf{a}, \mathbf{b} \in \mathbb{C}$ and the gradient of function $u(\mathbf{v}, 0)$ is given as follows.

Lemma 2. *The complex gradient of $u(\mathbf{v}, 0)$ with respect to $\mathbf{v}^*$ (the complex conjugate of $\mathbf{v}$) is given by*

$$\nabla_{\mathbf{v}^*} u(\mathbf{v}, 0) = \mathbb{E}[\nabla_{\mathbf{v}^*} s_{k^\star}(\mathbf{v}, \mathbf{h}, 0)], \tag{4.3.16}$$

where $k^\star = \arg \max_{1 \le k \le K+1} s_k(\mathbf{v}, \mathbf{h}, 0)$, and $\nabla_{\mathbf{v}^*} s_k(\mathbf{v}, \mathbf{h}, 0) = [\boldsymbol{\nu}_{k,i}]_{1 \le i \le K} (1 \le k \le K)$ with $\boldsymbol{\nu}_{k,i} \in \mathbb{C}^N$ given by

$$\boldsymbol{\nu}_{k,i} = \begin{cases} \left(\mathbf{h}_k \mathbf{h}_k^{\mathsf{H}} + \frac{1}{\gamma_i} \mathbf{h}_i \mathbf{h}_i^{\mathsf{H}}\right) \mathbf{v}_i, \textit{if } i \neq k, 1 \le k \le K, \\ 0, \qquad\qquad\qquad\qquad \textit{otherwise}, \end{cases}$$

and $\nabla_{\mathbf{v}^*} s_{K+1}(\mathbf{v}, \mathbf{h}, \kappa) = [\boldsymbol{\nu}_{K+1,i}]_{1 \le i \le K}$ with $\boldsymbol{\nu}_{K+1,i} = \frac{1}{\gamma_i} \mathbf{h}_i \mathbf{h}_i^{\mathsf{H}} \mathbf{v}_i, \forall i$. Furthermore, the gradient of $u(\mathbf{v}, 0)$ with respect to κ is zero, as $\kappa = 0$ is a constant in the function $u(\mathbf{v}, 0)$.

Proof. Please refer to Appendix C.3 for details. □

Therefore, at the j-th iteration, the non-convex DC constraint function $[u(\mathbf{v}, \kappa) - \kappa\epsilon] - u(\mathbf{v}, 0)$ in $\tilde{\mathscr{P}}_{\text{DC}}$ can be upper bounded by the convex function $l(\mathbf{v}, \kappa; \mathbf{v}^{[j]}, \kappa^{[j]}) - \kappa\epsilon$ with

$$l(\mathbf{v}, \kappa; \mathbf{v}^{[j]}, \kappa^{[j]}) = u(\mathbf{v}, \kappa) - u(\mathbf{v}^{[j]}, 0) - 2\langle \nabla_{\mathbf{v}^*} u(\mathbf{v}^{[j]}, 0), \mathbf{v} - \mathbf{v}^{[j]} \rangle. \tag{4.3.17}$$

Based on the convex approximation (4.3.17) to the DC constraint in $\tilde{\mathscr{P}}_{\text{DC}}$, we will then solve the following stochastic convex programming problem at the j-th iteration:

$$\begin{aligned} \tilde{\mathscr{P}}_{\text{DC}}(\mathbf{v}^{[j]}, \kappa^{[j]}) : \underset{\mathbf{v} \in \mathcal{V}, \kappa > 0}{\text{minimize}} & \sum_{l=1}^{L} \sum_{k=1}^{K} \|\mathbf{v}_{lk}\|^2 \\ \text{subject to } & l(\mathbf{v}, \kappa; \mathbf{v}^{[j]}, \kappa^{[j]}) - \kappa\epsilon \le 0. \end{aligned} \tag{4.3.18}$$

The proposed stochastic DC programming algorithm to the SCB problem $\mathscr{P}_{\text{SCB}}$ is thus presented in Algorithm 1.

Algorithm 6: Stochastic DC Programming Algorithm

```
Step 0: Find the initial solution $(\mathbf{v}^{[0]}, \kappa^{[0]}) \in \mathcal{D}$ and set the iteration counter $j = 0$;
Step 1: If $(\mathbf{v}^{[j]}, \kappa^{[j]})$ satisfies the termination criterion, go to End;
Step 2: Solve problem $\tilde{\mathscr{P}}_{\text{DC}}(\mathbf{v}^{[j]}, \kappa^{[j]})$ and obtain the optimal solution $(\mathbf{v}^{[j+1]}, \kappa^{[j+1]})$;
Step 3: Set $j = j + 1$ and go to Step 1;
End.
```

Based on Theorem 4 on the optimality of the joint approximation, the convergence of the stochastic DC programming algorithm is presented in the following theorem, which reveals the main advantage compared with all the previous algorithms for the JCCP problem, i.e., it guarantees optimality.

Theorem 5 (Convergence of Stochastic DC Programming)**.** *Denote* $\{\mathbf{v}^{[j]}, \kappa^{[j]}\}$ *as the sequence generated by the stochastic DC programming algorithm. Suppose that the limit of the sequence exists, i.e.,* $\lim_{j\to+\infty}(\mathbf{v}^{[j]}, \kappa^{[j]}) = (\mathbf{v}^\star, \kappa^\star)$*, which satisfies the Slater's condition*[2]*, then* $\mathbf{v}^\star$ *is the globally optimal solution of the SCB problem* $\mathscr{P}_{\mathrm{SCB}}$ *if it is convex. Otherwise,* $\mathbf{v}^\star$ *is a locally optimal solution. Furthermore,* κ *converges to zero for most scenarios, except that*

$$\Pr\left\{\max_{1\le k\le K} d_k(\mathbf{v}^\star, \mathbf{h}) \in (-\kappa^\star, 0]\right\} = 0, \tag{4.3.19}$$

if $\kappa^\star \neq 0$.

Proof. Please refer to Appendix C.4 for details. □

Based on Theorem 5, in the sequel, we focus on how to efficiently implement the stochastic DC programming algorithm.

4.3.4 Sample Average Approximation Method for the Stochastic DC Programming Algorithm

In order to implement the stochastic DC programming algorithm, we need to address the problem on how to solve the stochastic convex program $\tilde{\mathscr{P}}_{\mathrm{DC}}(\mathbf{v}^{[j]}, \kappa^{[j]})$ (4.3.18) efficiently at each iteration.

We propose to use the sample average approximation (SAA) based algorithm [81] to solve the stochastic convex problem $\tilde{\mathscr{P}}_{\mathrm{DC}}(\mathbf{v}^{[j]}, \kappa^{[j]})$ at the j-th iteration. Specifically, the SAA estimate of $u(\mathbf{v}, \kappa)$ is given by

$$\bar{u}(\mathbf{v}, \kappa) = \frac{1}{M}\sum_{m=1}^{M} \max_{1\le k\le K+1} s_k(\mathbf{v}, \mathbf{h}^m, \kappa), \tag{4.3.20}$$

[2] Slater's condition is a commonly used constraint qualification to ensure the existence of KKT pairs in convex optimization [22].

where $\mathbf{h}^1, \mathbf{h}^2, \ldots, \mathbf{h}^M$ are M independent realizations of the random vector $\mathbf{h} \in \mathbb{C}^{NK}$. Similarly, the SAA estimate of the gradient $\nabla_{\mathbf{v}^*} u(\mathbf{v}, 0)$ is given by

$$\bar{\nabla}_{\mathbf{v}^*} u(\mathbf{v}, 0) = \frac{1}{M} \sum_{m=1}^{M} \nabla_{\mathbf{v}^*} s_{k_m^\star}(\mathbf{v}, \mathbf{h}^m, 0), \tag{4.3.21}$$

where $k_m^\star = \arg \max_{1 \leq k \leq K+1} s_k(\mathbf{v}, \mathbf{h}^m, 0)$. Therefore, the SAA estimate of the convex function $l(\mathbf{v}, \kappa; \mathbf{v}^{[j]}, \kappa^{[j]})$ (4.3.17) is given by

$$\bar{l}(\mathbf{v}, \kappa; \mathbf{v}^{[j]}, \kappa^{[j]}) = \bar{u}(\mathbf{v}, \kappa) - \bar{u}(\mathbf{v}^{[j]}, 0) - 2\langle \bar{\nabla}_{\mathbf{v}^*} u(\mathbf{v}^{[j]}, 0), \mathbf{v} - \mathbf{v}^{[j]} \rangle, \tag{4.3.22}$$

which is jointly convex in $\mathbf{v}$ and κ. We will thus solve the following SAA based convex optimization problem

$$\begin{aligned} \bar{\mathscr{P}}_{\text{DC}}(\mathbf{v}^{[j]}, \kappa^{[j]}; M) : \underset{\mathbf{v} \in \mathcal{V}, \kappa > 0}{\text{minimize}} & \sum_{l=1}^{L} \sum_{k=1}^{K} \|\mathbf{v}_{lk}\|^2 \\ \text{subject to} \, & \bar{l}(\mathbf{v}, \kappa; \mathbf{v}^{[j]}, \kappa^{[j]}) - \kappa\epsilon \leq 0, \end{aligned} \tag{4.3.23}$$

to approximate the stochastic convex optimization problem $\tilde{\mathscr{P}}_{\text{DC}}(\mathbf{v}^{[j]}, \kappa^{[j]})$, which can be reformulated as the following convex quadratically constraint quadratic program (QCQP) [22]:

$$\begin{aligned} \mathscr{P}_{\text{QCQP}}^{[j]} : \underset{\mathbf{v} \in \mathcal{V}, \kappa > 0, \mathbf{x}}{\text{minimize}} \; & \sum_{l=1}^{L} \sum_{k=1}^{K} \|\mathbf{v}_{lk}\|^2 \\ \text{subject to} \; & \frac{1}{M} \sum_{m=1}^{M} x_m - \bar{u}(\mathbf{v}^{[j]}, 0) - 2\langle \bar{\nabla}_{\mathbf{v}^*} u(\mathbf{v}^{[j]}, 0), \mathbf{v} - \mathbf{v}^{[j]} \rangle \leq \kappa\epsilon \\ & s_k(\mathbf{v}, \mathbf{h}^m, \kappa) \leq x_m, x_m \geq 0, \forall k, m, \end{aligned} \tag{4.3.24}$$

which can then be solved efficiently using the interior-point method [22], where $\mathbf{x} = [x_m]_{1 \leq m \leq M} \in \mathbb{R}^M$ is the collection of the slack variables.

The following theorem indicates that the SAA based program $\bar{\mathscr{P}}_{\text{DC}}(\mathbf{v}^{[j]}, \kappa^{[j]}; M)$ for the stochastic convex optimization $\tilde{\mathscr{P}}_{\text{DC}}(\mathbf{v}^{[j]}, \kappa^{[j]})$ will not lose any optimality in the asymptotic regime.

Theorem 6. *Denote the set of the optimal solutions and optimal values of problems* $\tilde{\mathscr{P}}_{DC}(\mathbf{v}^{[j]}, \kappa^{[j]})$

and $\bar{\mathscr{P}}_{DC}(\mathbf{v}^{[j]}, \kappa^{[j]}; M)$ as $(\mathcal{P}^{\star}(\mathbf{v}^{[j]}, \kappa^{[j]}), V^{\star}(\mathbf{v}^{[j]}, \kappa^{[j]}))$ and $(\mathcal{P}_M^{\star}(\mathbf{v}^{[j]}, \kappa^{[j]}), V_M^{\star}(\mathbf{v}^{[j]}, \kappa^{[j]}))$, respectively, then we have

$$\mathbb{D}(\mathcal{P}_M^{\star}(\mathbf{v}^{[j]}, \kappa^{[j]}), \mathcal{P}^{\star}(\mathbf{v}^{[j]}, \kappa^{[j]})) \to 0, \tag{4.3.25}$$

and

$$V_M^{\star}(\mathbf{v}^{[j]}, \kappa^{[j]}) \to V^{\star}(\mathbf{v}^{[j]}, \kappa^{[j]}), \tag{4.3.26}$$

with probability one, as the sample size increases, i.e., as $M \to +\infty$.

Proof. Please refer to Appendix C.5 for details. □

Based on Theorems 1-4, we conclude that the proposed stochastic DC programming algorithm converges to the globally optimal solution of the SCB problem if it is convex and to a locally optimal solution if the problem is non-convex, in the asymptotic regime, i.e., $M \to +\infty$.

Remark 6. *Although the scenario approach based on constraints (4.2.8) is also a Monte Carlo algorithm, its performance can not be improved by generating more samples of the channel vector $\mathbf{h}$ [104], which is in contrast to our proposed stochastic DC programming algorithm. The reason is that increasing the sample size will make the resultant optimization problem more conservative, as more constraints need to be satisfied. This might result in worse solutions, i.e., beamformers with a higher transmit power.*

4.3.5 Complexity Analysis and Discussions

To implement the stochastic DC programming algorithm, at each iteration, we need to solve the convex QCQP program $\mathscr{P}_{\text{QCQP}}^{[j]}$ with $m = (L + KM + 1)$ (M is the number of independent realizations of the random vector $\mathbf{h}$) constraints and $n = (NK + M + 1)$ optimization variables. The convex QCQP problem can be solved with a worst-case complexity of $\mathcal{O}((mn^2 + n^3)m^{1/2} \log(1/\varepsilon))$ given a solution accuracy $\varepsilon > 0$ using the interior-point method [114]. As the Monte Carlo sample size M could be very large in order to reduce the approximation bias [104], the computational complexity of the stochastic DC programming algorithm could be higher than other deterministic approximation methods, e.g., the Bernstein approximation method.

In order to further improve the computational efficiency of the stochastic DC programming algorithm, other approaches can be explored (e.g., the alternating direction method of multipliers (ADMM) method [59]) to solve the large-scale conic program $\mathscr{P}_{\text{QCQP}}^{[j]}$ in (4.3.24) at each iteration. This is an on-going research topic, and we will leave it as our future work.

Furthermore, as the stochastic DC programming algorithm only requires distribution information of the random vector $\mathbf{h}$ to generate the Monte Carlo samples, this approach can be widely applied for any channel uncertainty model. As the proposed stochastic DC programming algorithm provides optimality guarantee, it can serve as the performance benchmark in various beamforming design problems with CSI uncertainty and probabilistic QoS guarantees, and thus it will find wide applications in future wireless networks.

4.4 Simulation Results

In this section, we simulate the proposed stochastic DC algorithm for coordinated beamforming design. We consider the following channel model for the link between the k-th user and the l-th RRH [18, 115]:

$$\begin{aligned}\mathbf{h}_{kl} &= \underbrace{10^{-L(d_{kl})/20}\sqrt{\varphi_{kl}s_{kl}}}_{D_{kl}}\left(\sqrt{1-\tau_{kl}^2}\hat{\mathbf{c}}_{kl}+\tau_{kl}\mathbf{e}_{kl}\right)\\ &= \sqrt{1-\tau_{kl}^2}D_{kl}\hat{\mathbf{c}}_{kl}+\tau_{kl}D_{kl}\mathbf{e}_{kl}, \forall k,l, \qquad (4.4.1)\end{aligned}$$

where $L(d_{kl})$ is the path-loss at distance d_{kl}, as given in [29, Table I], s_{kl} is the shadowing coefficient, φ_{kl} is the antenna gain, $\hat{\mathbf{c}}_{kl} \in \mathcal{CN}(\mathbf{0}, \mathbf{I}_{N_l})$ is the estimated imperfect small-scale fading coefficient and $\mathbf{e}_{kl}$ is the CSI error. We assume that the BBU pool can accurately track the large-scale fading coefficients D_{kl}'s [107]. The error vector is modeled as $\mathbf{e}_{kl} \in \mathcal{CN}(\mathbf{0}, \mathbf{I}_{N_l})$. The parameters τ_{kl}'s depend on the CSI acquisition schemes, e.g., channel estimation errors using MMSE. We use the standard cellular network parameters as shown in [29, Table I]. The maximum outage probability that the system can tolerate is set as $\epsilon = 0.1$. The proposed stochastic DC programming algorithm will stop if the difference between the objective values of $\tilde{\mathscr{P}}_{\text{DC}}(\mathbf{v}^{[j]}, \kappa^{[j]})$ (4.3.18) of two consecutive iterations is less than 10^{-4}.

The proposed stochastic DC programming algorithm is compared to the following two algorithms:

- **The scenario approach**: The main idea of this algorithm is to approximate the probabilistic QoS constraint by multiple "sampling" QoS constraints [100, 105]. This algorithm can only find a feasible solution for problem $\mathscr{P}_{\text{SCB}}$ with a high probability. Please refer to [71] for more details.

- **The Bernstein approximation method**: The main idea of this algorithm is to use the Bernstein-type inequality to find a closed-form approximation for the chance constraint (4.2.6) [101, 103]. The original stochastic optimization problem $\mathscr{P}_{\text{SCB}}$ can be conservatively approximated by a deterministic optimization problem. Therefore, the computational complexity of the deterministic approximation method is normally much lower than the Monte Carlo approaches, e.g., the scenario approach and the stochastic DC programming algorithm. Nevertheless, the Bernstein approximation method can also only find a feasible but suboptimal solution, and the conservativeness of this method is difficult to quantify. Moreover, to derive closed-form expressions, the Bernstein approximation method restricts the distribution of the random vector $\mathbf{h}$ to be complex Gaussian distribution. Therefore, this method is not robust against the distribution of the random vector $\mathbf{h}$.

Due to the computational complexity of solving large-size sample problems for both the stochastic DC programming algorithm and the scenario approach, we only consider a simple and particular network realization to demonstrate the performance benchmarking capability of the proposed stochastic DC programming algorithm. Specifically, consider a network with $L = 5$ single-antenna RRHs and $K = 3$ single-antenna MUs uniformly and independently distributed in the square region $[-400, 400] \times [-400, 400]$ meters. In this scenario, we consider a mixed CSI uncertainty model [71, 107], i.e., partial and imperfect CSI. Specifically, for MU k, we set $\tau_{kn} = 0.01, \forall n \in \Omega_k$ (i.e., the obtained channel coefficients are imperfect) and $\tau_{kn} = 1, \forall \neq \Omega_k$, where Ω_k includes the indices of the 2 largest entries of the vector consisting of all the large-scale fading coefficients for MU k. That is, only 40% of the channel coefficients are obtained in this scenario. The QoS requirements are set as $\gamma_k = 3\text{dB}, \forall k$. The sample size for the scenario approach is 308 [105], which yields a solution that satisfies the probability constraint (4.2.7) (i.e., a feasible solution to problem $\mathscr{P}_{\text{SCB}}$) with probability at least 99%. The sample size for the stochastic DC programming algorithm is set to be 1000. The simulated channel data is given in (4.4.2), where $\hat{\mathbf{H}} = [D_{kl}\hat{\mathbf{c}}_{kl}]$ and $\mathbf{D} = [D_{kl}]$. In the following, we will illustrate the convergence, conservativeness, stability and performance

$$\hat{\mathbf{H}} = \begin{bmatrix} -2.24+0.96i & -1.03+2.03i & 3.66+11.33i \\ -0.57-0.16i & 8.47+19.50i & -0.01-1.38i \\ 28.90-13.22i & 4.35-10.15i & 1.65-4.81i \\ -1.68+1.26i & -2.67-2.01i & 42.98-5.68i \\ 3.46-2.08i & 4.13+1.87i & -2.31+1.34i \end{bmatrix}, \mathbf{D} = \begin{bmatrix} 2.80 & 4.46 & 26.89 \\ 2.48 & 9.56 & 1.92 \\ 29.97 & 24.34 & 13.83 \\ 2.11 & 4.09 & 38.80 \\ 2.87 & 3.92 & 3.59 \end{bmatrix}. \quad (4.4.2)$$

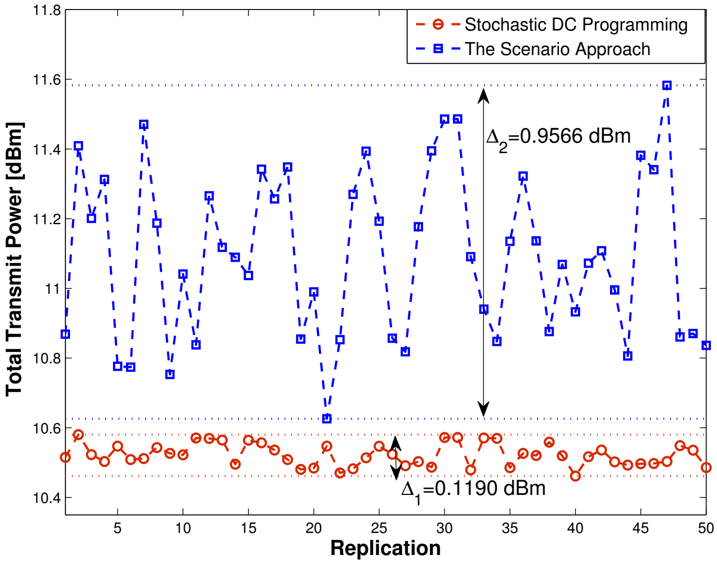

Figure 4.1: Optimal value versus different Monte Carlo replications.

gains of the stochastic DC programming algorithm.

4.4.1 Stability of the Algorithms

As both the stochastic DC programming and scenario approach use Monte Carlo samples to obtain the solutions, the corresponding solutions should depend on the particular samples. Therefore, it is essential to investigate the stability of solutions obtained by the stochastic algorithms. We thus run the algorithms 50 replications with different Monte Carlo samples for each replication to illustrate the stability of the algorithms.

From Fig. 4.1 and Fig. 4.2, we can see that the solutions and the estimated probability constraints obtained from the stochastic DC programming algorithm are very stable, as they converge to a similar solution. In particular, the average total transmit power is 10.5228 dBm, with the lowest being 10.4614 dBm and the highest being 10.5804 dBm. The corresponding average probability constraint is 0.9010, with the range of 0.8933 to 0.9067.

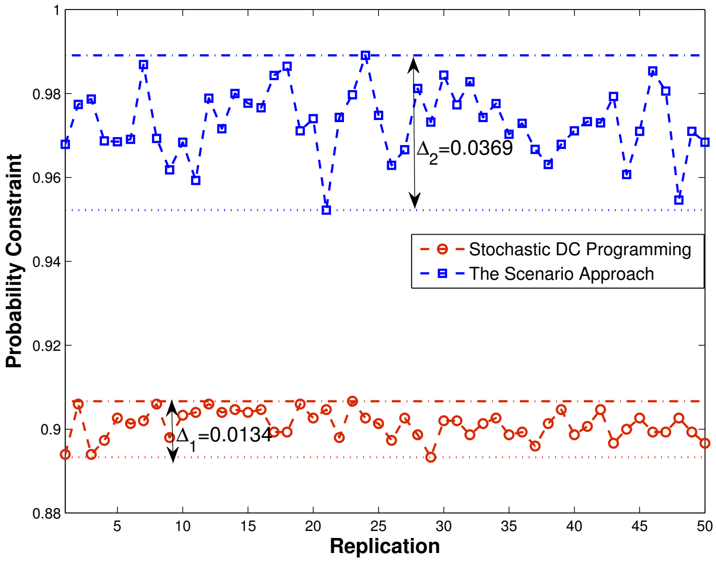

Figure 4.2: Probability constraint versus different Monte Carlo replications.

However, the solutions and the estimated probability constraints obtained from the scenario approach drastically differ from replication to replication due to the randomness in the Monte Carlo samples. In particular, the average total transmit power is 11.1004 dBm, with the lowest being 10.6260 dBm and the highest being 11.5826 dBm. The corresponding average probability constraint is 0.9731, and is in the range between 0.9522 and 0.9891.

We can see that the stochastic DC programming algorithm can achieve a lower transmit power than the scenario approach on average. The scenario approach yields a much more conservative approximation for the probability constraint. Furthermore, the performance of the scenario approach cannot be improved by increasing the sampling size as this will cause more conservative solutions. This is in contrast to the proposed stochastic DC programming algorithm, as Theorem 6 indicates that more samples can improve the Monte Carlo approximation performance and most Monte Carlo approach based stochastic algorithms possess such a property.

Finally, the average value of the parameter κ is 1.5×10^{-3} and is in the rang between 7.8×10^{-4} and 2.6×10^{-3} when the stochastic DC programming algorithm terminates. This justifies the conclusion that the parameter κ will converge to zero as presented in Theorem 5.

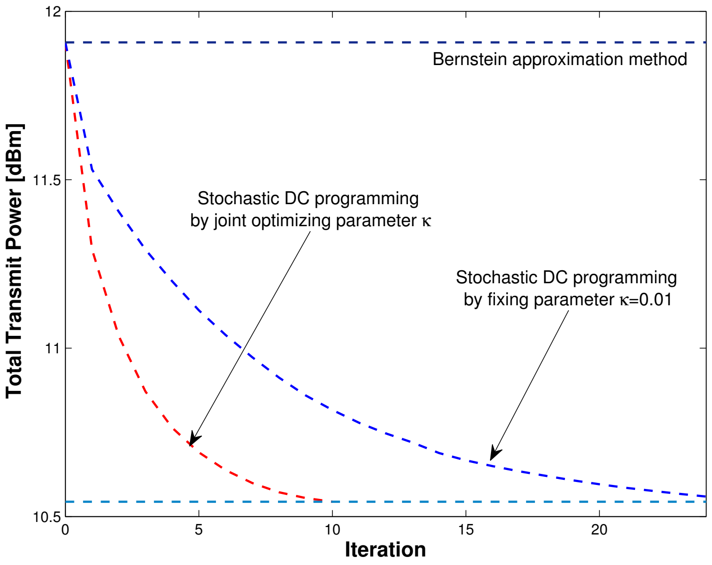

Figure 4.3: Convergence of the stochastic DC programming algorithm.

4.4.2 Convergence of the Stochastic DC Programming Algorithm

We report a typical performance on the convergence of the stochastic DC programming algorithm, as shown in Fig. 4.3, with the initial point being the solution from the Bernstein approximation method. This figure shows that the convergence rate of the proposed stochastic DC programming is very fast for the simulated scenario. We can see that the stochastic DC programming algorithm can achieve a much lower transmit power than the Bernstein approximation method. This figure also demonstrates the effectiveness of jointly optimizing over the parameter κ and beamforming vector $\mathbf{v}$, as this can significantly improve the convergence rate. Furthermore, the parameter κ is 1.3×10^{-3} when the proposed stochastic DC programming algorithm terminates under this scenario.

4.4.3 Conservativeness of the Algorithms

We also report the typical performances of all the algorithms on the conservativeness of approximating probability constraints in the SCB problem under the same scenario as the above subsection. The estimated probability constraint in $\mathscr{P}_{\text{SCB}}$ is shown in Fig. 4.4, which is 0.988 using the Bernstein approximation. On the other hand, for the stochastic DC programming algorithm, we can see that the probability constraint becomes tight when it terminates, and thus the Bernstein approximation is too conservative. This coincide with the fact that the suboptimal algorithms only seek conservative approximations to the chance constraint.

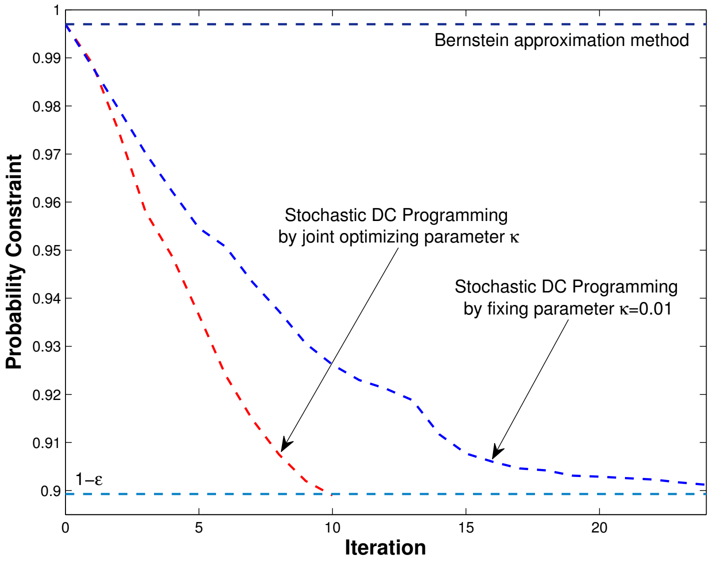

Figure 4.4: Probability constraint.

4.5 Discussions

We presented a unified framework consisting of a novel compressive CSI acquisition method and a generic stochastic coordinated beamforming framework for the optimal transmission strategy design with a probabilistic model for the CSI uncertainty in dense Cloud-RAN. This framework frees us from the structural modeling assumptions and distribution types assumptions for the uncertain channel knowledge, and thus it provides modeling flexibility. With the optimality guarantee, the proposed stochastic DC programming algorithm can serve as the benchmark for evaluating suboptimal and heuristic algorithms. The benchmarking capability was demonstrated numerically in terms of conservativeness, stability and optimal values by comparing with the Bernstein approximation method and scenario approach. Furthermore, the proposed algorithm has a better convergence rate by jointly optimizing the approximation parameter κ. As the proposed stochastic DC programming algorithm provides optimality guarantee, we believe this algorithm can be applied in various beamforming design problems with probabilistic QoS guarantees due to the CSI uncertainty, and it will find wide applications in future wireless networks.

Several future research directions are listed as follows:

- Although our framework only requires the distribution information of the uncertain channel knowledge, so as to generate Monte Carlo samples for the stochastic DC programming algorithm, it might be challenging to obtain the exact information in some scenarios. Therefore, one may either seek more sophisticated measuring methods to estimate the distribution information or adopt the distributionally robust optimization approaches to deal with the ambiguous distributions, e.g., [116].

- The main drawback of the stochastic DC programming algorithm is the highly computational complexity with the sample problem $\mathscr{P}_{\text{QCQP}}^{[j]}$ at each iteration, one may either resort to ADMM [59] based algorithms to solve the large-sized sample problem in parallel or reduce the optimization dimensions by fixing the directions of the beamformers and only optimizing the transmit power allocation (e.g., in [102], the corresponding power allocation problem is a linear program and can be solved with a much lower computational complexity).
- The optimality of the compressive CSI acquisition should be characterized by establishing relation between the performance loss and the acquired number of channel links.
- The partial connectivity in wireless networks provides great opportunities for massive CSI overhead reduction. In particular, some preliminary results on the topological interference management based on the low rank matrix completion with Riemannian optimization [117] were presented in [25]. In particular, the low rank matrix completion approach has the potential for the applications in index coding, network coding, and distributed caching and storage problems.

Chapter 5

Large-Scale Convex Optimization for Dense Cloud-RAN

In this chapter we present a two-stage approach to solve large-scale convex optimization problems in dense Cloud-RAN, which can effectively detect infeasibility and enjoy modeling flexibility. In the proposed approach, the original large-scale convex problem is transformed into a standard cone programming form in the first stage via matrix stuffing, which only needs to copy the problem parameters to the pre-stored structure of the standard form. The capability of yielding infeasibility certificates and enabling parallel computing is achieved by solving the homogeneous self-dual embedding of the primal-dual pair of the standard form. In the solving stage, the operator splitting method, namely, the alternating direction method of multipliers (ADMM), is adopted to solve the large-scale homogeneous self-dual embedding in parallel. These results will serve the purpose of providing practical and theoretical guidelines on designing algorithms for generic large-scale optimization problems in dense wireless networks. The derivations of the standard conic programming form transformation are presented in Appendix D. The material in this chapter has been presented in part in [67, 97].

5.1 Introduction

The proliferation of smart mobile devices, coupled with new types of wireless applications, has led to an exponential growth of wireless and mobile data traffic. In order to provide high-volume and diversified data services, ultra-dense wireless cooperative network architectures have been proposed for next generation wireless networks [11], e.g., Cloud-RAN [10, 29], and distributed antenna systems [118]. To enable efficient interference management and resource allocation, large-scale multi-entity collaboration will play pivotal roles in dense wireless networks. For instance, in Cloud-RAN, all the baseband signal processing is shifted to a single cloud data center with very powerful computational capability. Thus the centralized signal processing can be performed to support large-scale cooperative transmission/reception among the remote radio heads (RRHs).

Convex optimization serves as an indispensable tool for resource allocation and signal processing in wireless communication systems [17, 84, 87]. For instance, coordinated beamforming [18] often yields a direct convex optimization formulation, i.e., second-order cone programming (SOCP) [22]. The network max-min fairness rate optimization [119] can be solved through the bi-section method [22] in polynomial time, wherein a sequence of convex subproblems are solved. Furthermore, convex relaxation provides a principled way of developing polynomial-time algorithms for non-convex or NP-hard problems, e.g., group-sparsity penalty relaxation for the NP-hard mixed integer nonlinear programming problems [29], semidefinite relaxation [84] for NP-hard robust beamforming [61, 83] and multicast beamforming [60], and sequential convex approximation to the highly intractable stochastic coordinated beamforming [72].

Nevertheless, in dense Cloud-RAN [11], which may possibly need to simultaneously handle hundreds of RRHs, resource allocation and signal processing problems will be dramatically scaled up. The underlying optimization problems will have high dimensions and/or large numbers of constraints (e.g., per-RRH transmit power constraints and per-MU (mobile user) QoS constraints). For instance, for a Cloud-RAN with 100 single-antenna RRHs and 100 single-antenna MUs, the dimension of the aggregative coordinated beamforming vector (i.e., the optimization variables) will be 10^4. Most advanced off-the-shelf solvers (e.g., SeDuMi [120], SDPT3 [121] and MOSEK [122]) are based on the interior-point method. However, the computational burden of such second-order method makes it inapplicable for large-scale problems. For instance, solving convex quadratic programs has cubic complexity [114]. Furthermore, to use these solvers, the original problems need to be transformed to the standard forms supported by the solvers. Although the parser/solver modeling frameworks like CVX [123] and YALMIP [124] can automatically transform the original problem instances into standard forms, it may require substantial time to perform such transformation [125], especially for problems with a large number of constraints [67].

One may also develop custom algorithms to enable efficient computation by exploiting the structures of specific problems. For instance, the uplink-downlink duality [18] is exploited to extract the structures of the optimal beamformers [126] and enable efficient algorithms. However, such an approach still has the cubic complexity to perform matrix inversion at each iteration [127]. First-order methods, e.g., the ADMM algorithm [59], have recently attracted attention for their distributed and parallelizable implementation, as well as the capability of scaling to large problem sizes. However, most existing ADMM based algorithms cannot provide the certificates of infeasibility [83, 127, 128]. Furthermore, some of them may still fail to scale to large problem sizes, due to the SOCP subproblems [128] or semidefinite programming (SDP) subproblems [83] needed to be solved at each iteration.

Without efficient and scalable algorithms, previous studies of wireless cooperative networks either only demonstrate performance in small-size networks, typically with less than 10 RRHs, or resort to sub-optimal algorithms, e.g., zero-forcing based approaches [65, 129]. Meanwhile, from the above discussion, we see that the large-scale optimization algorithms to be developed should possess the following two features:

- To scale well to large problem sizes with parallel computing capability;
- To effectively detect problem infeasibility, i.e., provide certificates of infeasibility.

To address these two challenges in a unified way, in this chapter, we shall propose a two-stage approach as shown in Fig. 5.1. The proposed framework is capable to solve large-scale convex optimization problems in parallel, as well as providing certificates of infeasibility. Specifically, the original problem $\mathscr{P}$ will be first transformed into a standard cone programming form $\mathscr{P}_{\text{cone}}$ [114] based on the Smith form reformulation [130], via introducing a new variable for each subexpression in the disciplined convex programming form [131] of the original problem. This will eventually transform the coupled constraints in the original problem into the constraint only consisting of two convex sets: a subspace and a convex set formed by a Cartesian product of a finite number of standard convex cones. Such a structure helps to develop efficient parallelizable algorithms and enable the infeasibility detection capability *simultaneously* via solving the homogeneous self-dual embedding [132] of the primal-dual pair of the standard form by the ADMM algorithm.

As the mapping between the standard cone program and the original problem only depends on the network size (i.e., the numbers of RRHs, MUs and antennas at each RRH),

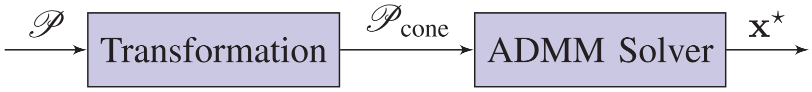

Figure 5.1: The proposed two-stage approach for large-scale convex optimization. The optimal solution or the certificate of infeasibility can be extracted from $\mathbf{x}^\star$ by the ADMM solver.

we can pre-generate and store the structures of the standard forms with different candidate network sizes. Then for each problem instance, that is, given the channel coefficients, QoS requirements, and maximum RRH transmit powers, we only need to copy the original problem parameters to the standard cone programming data. Thus, the transformation procedure can be very efficient and can avoid repeatedly parsing and re-generating problems [123, 124]. This technique is called *matrix stuffing* [67, 125], which is essential for the proposed framework to scale well to large problem sizes. It may also help rapid prototyping and testing for practical equipment development.

5.1.1 Contributions

The major contributions of the chapter are summarized as follows:

1. We formulate main performance optimization problems in dense wireless cooperative networks into a general framework. It is shown that all of them can essentially be solved through solving one or a sequence of large-scale convex optimization or convex feasibility problems.

2. To enable both the infeasibility detection capability and parallel computing capability, we propose to transform the original convex problem to an equivalent standard cone program. The transformation procedure scales very well to large problem sizes with the matrix stuffing technique. Simulation results will demonstrate the effectiveness of the proposed fast transformation approach over the state-of-art parser/solver modeling frameworks.

3. The operator splitting method is then adopted to solve the large-scale homogeneous self-dual embedding of the primal-dual pair of the transformed standard cone program in parallel. This first-order optimization algorithm makes the second stage scalable. Simulation results will show that it can speedup several orders of magnitude over the state-of-art interior-point solvers.

4. The proposed framework enables evaluating various cooperation strategies in dense wireless networks, and helps reveal new insights *numerically*. For instance, simulation results demonstrate a significant performance gain of optimal beamforming over sub-optimal schemes, which shows the importance of developing large-scale optimal beamforming algorithms.

This work will serve the purpose of providing practical and theoretical guidelines on designing algorithms for generic large-scale optimization problems in dense wireless networks.

5.1.2 Organization

The remainder of the chapter is organized as follows. Section 5.2 presents the system model and problem formulations. In Section 5.3, a systematic cone programming form transformation procedure is developed. The operator splitting method is presented in Section 5.4. The practical implementation issues are discussed in Section 5.5. Numerical results will be demonstrated in Section 5.6. Finally, conclusions and discussions are presented in Section 5.7. To keep the main text clean and free of technical details, we divert most of the proofs, derivations to the appendix.

5.2 Large-Scale Optimization in Dense Cloud-RAN

In this section, we will first present two representative optimization problems in Cloud-RAN, i.e., the network power minimization problem and the network utility maximization problem. We will then provide a unified formulation for large-scale optimization problems in dense Cloud-RAN.

5.2.1 Signal Model

Consider a dense fully cooperative Cloud-RAN[1] with L RRHs and K single-antenna MUs, where the l-th RRH is equipped with N_l antennas. The centralized signal processing is performed at a central processor, e.g., the baseband unit pool in Cloud-RAN [10, 29]. The propagation channel from the l-th RRH to the k-th MU is denoted as $\mathbf{h}_{kl} \in \mathbb{C}^{N_l}, \forall k, l$. We focus on

[1] This is mainly for notation simplification. The proposed framework can be easily extended to more general cooperation scenarios as presented in [87].

the downlink transmission, for which the signal processing is more challenging. The received signal $y_k \in \mathbb{C}$ at MU k is given by

$$y_k = \sum_{l=1}^{L} \mathbf{h}_{kl}^{\mathsf{H}} \mathbf{v}_{lk} s_k + \sum_{i \neq k} \sum_{l=1}^{L} \mathbf{h}_{kl}^{\mathsf{H}} \mathbf{v}_{li} s_i + n_k, \forall k, \tag{5.2.1}$$

where s_k is the encoded information symbol for MU k with $\mathbb{E}[|s_k|^2] = 1$, $\mathbf{v}_{lk} \in \mathbb{C}^{N_l}$ is the transmit beamforming vector from the l-th RRH to the k-th MU, and $n_k \sim \mathcal{CN}(0, \sigma_k^2)$ is the additive Gaussian noise at MU k. We assume that s_k's and n_k's are mutually independent and all the users apply single user detection. Thus the signal-to-interference-plus-noise ratio (SINR) of MU k is given by

$$\Gamma_k(\mathbf{v}) = \frac{|\mathbf{h}_k^{\mathsf{H}} \mathbf{v}_k|^2}{\sum_{i \neq k} |\mathbf{h}_k^{\mathsf{H}} \mathbf{v}_i|^2 + \sigma_k^2}, \forall k, \tag{5.2.2}$$

where $\mathbf{h}_k \triangleq [\mathbf{h}_{k1}^T, \dots, \mathbf{h}_{kL}^T]^T \in \mathbb{C}^N$ with $N = \sum_{l=1}^{L} N_l$, $\mathbf{v}_k \triangleq [\mathbf{v}_{1k}^T, \mathbf{v}_{2k}^T, \dots, \mathbf{v}_{Lk}^T]^T \in \mathbb{C}^N$ and $\mathbf{v} \triangleq [\mathbf{v}_1^T, \dots, \mathbf{v}_K^T]^T \in \mathbb{C}^{NK}$. We assume that each RRH has its own power constraint,

$$\sum_{k=1}^{K} \|\mathbf{v}_{lk}\|_2^2 \leq P_l, \forall l, \tag{5.2.3}$$

where $P_l > 0$ is the maximum transmit power of the l-th RRH. In this chapter, we assume that the full and perfect CSI is available at the central processor and all RRHs only provide unicast/broadcast services.

5.2.2 Network Power Minimization

Network power consumption is an important performance metric for the energy efficiency design in wireless cooperative networks. Coordinated beamforming is an efficient way to design energy-efficient systems [18], in which, beamforming vectors $\mathbf{v}_{lk}$'s are designed to minimize the total transmit power among RRHs while satisfying the QoS requirements for all the MUs. Specifically, given the target SINRs $\boldsymbol{\gamma} = (\gamma_1, \dots, \gamma_K)$ for all the MUs with $\gamma_k > 0, \forall k$, we will solve the following total transmit power minimization problem:

$$\mathscr{P}_1(\boldsymbol{\gamma}) : \underset{\mathbf{v} \in \mathcal{V}}{\text{minimize}} \ \sum_{l=1}^{L} \sum_{k=1}^{K} \|\mathbf{v}_{lk}\|_2^2, \tag{5.2.4}$$

where $\mathcal{V}$ is the intersection of the sets formed by QoS constraints and transmit power constraints, i.e.,

$$\mathcal{V} = \mathcal{P}_1 \cap \mathcal{P}_2 \cap \cdots \cap \mathcal{P}_L \cap \mathcal{Q}_1 \cap \mathcal{Q}_2 \ldots, \cap \mathcal{Q}_K, \tag{5.2.5}$$

where $\mathcal{P}_l$'s are feasible sets of $\mathbf{v}$ that satisfy the per-RRH transmit power constraints, i.e.,

$$\mathcal{P}_l = \left\{ \mathbf{v} \in \mathbb{C}^{NK} : \sum_{k=1}^{K} \|\mathbf{v}_{lk}\|_2^2 \leq P_l \right\}, \forall l, \tag{5.2.6}$$

and $\mathcal{Q}_k$'s are the feasible sets of $\mathbf{v}$ that satisfy the per-MU QoS constraints, i.e.,

$$\mathcal{Q}_k = \{\mathbf{v} \in \mathbb{C}^{NK} : \Gamma_k(\mathbf{v}) \geq \gamma_k\}, \forall k. \tag{5.2.7}$$

As all the sets $\mathcal{Q}_k$'s and $\mathcal{P}_l$'s can be reformulated into second-order cones as shown in [29], problem $\mathscr{P}_1(\boldsymbol{\gamma})$ can be reformulated as an SOCP problem.

However, in dense wireless cooperative networks, the mobile hauling network consumption can not be ignored. In [29], a two-stage group sparse beamforming (GSBF) framework is proposed to minimize the network power consumption for Cloud-RAN, including the power consumption of all optical fronthaul links and the transmit power consumption of all RRHs. Specially, in the first stage, the group-sparsity structure of the aggregated beamformer $\mathbf{v}$ is induced by minimizing the weighted mixed ℓ_1/ℓ_2-norm of $\mathbf{v}$, i.e.,

$$\mathscr{P}_2(\boldsymbol{\gamma}) : \underset{\mathbf{v} \in \mathcal{V}}{\text{minimize}} \ \sum_{l=1}^{L} \omega_l \|\tilde{\mathbf{v}}_l\|_2, \tag{5.2.8}$$

where $\tilde{\mathbf{v}}_l = [\mathbf{v}_{l1}^T, \ldots, \mathbf{v}_{lK}^T]^T \in \mathbb{C}^{N_l K}$ is the aggregated beamforming vector at RRH l, and $\omega_l > 0$ is the corresponding weight for the beamformer coefficient group $\tilde{\mathbf{v}}_l$. Based on the (approximated) group sparse beamformer $\mathbf{v}^\star$, which is the optimal solution to $\mathscr{P}_2(\boldsymbol{\gamma})$, in the second stage, an RRH selection procedure is performed to switch off some RRHs so as to minimize the network power consumption. In this procedure, we need to check if the remaining RRHs can support the QoS requirements for all the MUs, i.e., check the feasibility of problem $\mathscr{P}_1(\boldsymbol{\gamma})$ given the active RRHs. Please refer to [29] for more details on the group sparse beamforming algorithm.

5.2.3 Network Utility Maximization

Network utility maximization is a general approach to optimize network performance. We consider maximizing an arbitrary network utility function $U(\Gamma_1(\mathbf{v}), \ldots, \Gamma_K(\mathbf{v}))$ that is strictly increasing in the SINR of each MU [87], i.e.,

$$\mathscr{P}_3 : \underset{\mathbf{v}\in\mathcal{V}_1}{\text{maximize}} \;\; U(\Gamma_1(\mathbf{v}), \ldots, \Gamma_K(\mathbf{v})), \tag{5.2.9}$$

where $\mathcal{V}_1 = \cap_{l=1}^{L}\mathcal{P}_l$ is the intersection of the sets of the per-RRH transmit power constraints (5.2.6). It is generally very difficult to solve, though there are tremendous research efforts on this problem [87]. In particular, Liu *et al.* in [133] proved that $\mathscr{P}_3$ is NP-hard for many common utility functions, e.g., weighted sum-rate. Please refer to [87, Table 2.1] for details on classification of the convexity of utility optimization problems.

Assume that we have the prior knowledge of SINR values $\Gamma_1^\star, \ldots, \Gamma_K^\star$ that can be achieved by the optimal solution to problem $\mathscr{P}_3$. Then the optimal solution to problem $\mathscr{P}_1(\boldsymbol{\gamma})$ with target SINRs as $\boldsymbol{\gamma} = (\Gamma_1^\star, \ldots, \Gamma_K^\star)$ is an optimal solution to problem $\mathscr{P}_3$ as well [126]. The difference between problem $\mathscr{P}_1(\boldsymbol{\gamma})$ and problem $\mathscr{P}_3$ is that the SINRs in $\mathscr{P}_1(\boldsymbol{\gamma})$ are pre-defined, while the optimal SINRs in $\mathscr{P}_3$ need to be searched. For the max-min fairness maximization problem, optimal SINRs can be searched by the bi-section method [67], which can be accomplished in polynomial time. For the general increasing utility maximization problem $\mathscr{P}_3$, the corresponding optimal SINRs can be searched as follows

$$\underset{\boldsymbol{\gamma}\in\mathcal{R}}{\text{maximize}} \;\; U(\gamma_1, \ldots, \gamma_K), \tag{5.2.10}$$

where $\mathcal{R} \in \mathbb{R}_+^K$ is the achievable performance region

$$\mathcal{R} = \{(\Gamma_1(\mathbf{v}), \ldots, \Gamma_K(\mathbf{v})) : \mathbf{v} \in \mathcal{V}_1\}. \tag{5.2.11}$$

Problem (5.2.10) is a monotonic optimization problem [134] and thus can be solved by the polyblock outer approximation algorithm [134] or the branch-reduce-and-bound algorithm [87]. The general idea of both algorithms is iteratively improving the lower-bound U_{min} and upper-bound U_{max} of the objective function of problem (5.2.10) such that

$$U_{\text{max}} - U_{\text{min}} \leq \epsilon, \tag{5.2.12}$$

for a given accuracy ϵ in finite iterations. In particular, at the m-iteration, we need to check the convex feasibility problem of $\mathscr{P}_1(\boldsymbol{\gamma}^{[m]})$ given the target SINRs $\boldsymbol{\gamma}^{[m]} = (\Gamma_1^{[m]}, \dots, \Gamma_K^{[m]})$. However, the number of iterations scales exponentially with the number of MUs [87]. Please refer to the tutorial [87, Section 2.3] for more details. Furthermore, the network achievable rate region [135] can also be characterized by the rate profile method [136] via solving a sequence of such convex feasibility problems $\mathscr{P}_1(\boldsymbol{\gamma})$.

5.2.4 A Unified Framework of Large-Scale Network Optimization

In dense wireless cooperative networks, the central processor can support hundreds of RRHs for simultaneously transmission/reception [10]. Therefore, all the above optimization problems are shifted into a new domain with a high problem dimension and a large number of constraints. As presented previously, to solve the performance optimization problems, we essentially need to solve a sequence of the following convex optimization problem with different problem instances (e.g., different channel realizations, network sizes and QoS targets)

$$\mathscr{P} : \underset{\mathbf{v} \in \mathcal{V}}{\text{minimize}} \ f(\mathbf{v}), \tag{5.2.13}$$

where $f(\mathbf{v})$ is convex in $\mathbf{v}$ as shown in $\mathscr{P}_1(\boldsymbol{\gamma})$ and $\mathscr{P}_2(\boldsymbol{\gamma})$. Solving problem $\mathscr{P}$ means that the corresponding algorithm should return the optimal solution or the certificate of infeasibility.

For all the problems discussed above, problem $\mathscr{P}$ can be reformulated as an SOCP problem, and thus it can be solved in polynomial time via the interior-point method, which is implemented in most advanced off-the-shelf solvers, e.g., public software packages like SeDuMi [120] and SDPT3 [121] and commercial software packages like MOSEK [122]. However, the computational cost of such second-order methods will be prohibitive for large-scale problems. On the other hand, most custom algorithms, e.g., the uplink-downlink approach [18] and the ADMM based algorithms [83, 127, 128], however, fail to either scale well to large problem sizes or detect the infeasibility effectively.

To overcome the limitations of the scalability of the state-of-art solvers and the capability of infeasibility detection of the custom algorithms, in this chapter, we propose to solve the homogeneous self-dual embedding [132] (which aims at providing necessary certificates) of problem $\mathscr{P}$ via a first-order optimization method [59] (i.e., the operator splitting method). This will be presented in Section 5.4. To arrive at the homogeneous self-dual embedding and enable parallel computing, the original problem will be first transformed into a standard cone programming form as will be presented in Section 5.3. This forms the main idea of the two-stage based large-scale optimization framework as shown in Fig. 5.1.

5.3 Matrix Stuffing for Fast Standard Cone Programming Transformation

Although the parser/solver modeling language framework, like CVX [123] and YALMIP [124], can automatically transform the original problem instance into a standard form, it requires substantial time to accomplish this procedure [67, 125]. In particular, for each problem instance, the parser/solver modeling frameworks need to repeatedly parse and canonicalize it. To avoid such modeling overhead of reading problem data and repeatedly parsing and canonicalizing, we propose to use the matrix stuffing technique [67, 125] to perform fast transformation by exploiting the problem structures. Specifically, we will first generate the mapping from the original problem to the cone program, and then the structure of the standard form will be stored. This can be accomplished offline. Therefore, for each problem instance, we only need to stuff its parameters to data of the corresponding pre-stored structure of the standard cone program. Similar ideas were presented in the emerging parse/generator modeling frameworks like CVXGEN [137] and QCML [125], which aim at embedded applications for some specific problem families. In this chapter, we will demonstrate in Section 5.6 that matrix stuffing is essential to scale to large problem sizes for fast transformation at the first stage of the proposed framework.

5.3.1 Conic Formulation of Convex Programs

In this section, we describe a systematic way to transform the original problem $\mathscr{P}$ to the standard cone program. To enable parallel computing, a common way is to replicate some variables through either exploiting problem structures [83, 127] or using the consensus formulation [59, 128]. However, directly working on these reformulations is difficult to provide computable mathematical certificates of infeasibility. Therefore, heuristic criteria are often adopted to detect the infeasibility, e.g., the underlying problem instance is reported to be infeasible when the algorithm exceeds the pre-defined maximum iterations without convergence [127]. To unify the requirements of parallel and scalable computing and to provide computable mathematical certificates of infeasibility, in this chapter, we propose to transform the original problem $\mathscr{P}$ to the following equivalent cone program $\mathscr{P}_{\text{cone}}$:

$$\begin{aligned}\mathscr{P}_{\text{cone}} : \underset{\boldsymbol{\nu},\boldsymbol{\mu}}{\text{minimize}}\ & \mathbf{c}^T\boldsymbol{\nu} \\ \text{subject to}\ & \mathbf{A}\boldsymbol{\nu} + \boldsymbol{\mu} = \mathbf{b} && (5.3.1) \\ & (\boldsymbol{\nu}, \boldsymbol{\mu}) \in \mathbb{R}^n \times \mathcal{K}, && (5.3.2)\end{aligned}$$

where $\boldsymbol{\nu} \in \mathbb{R}^n$ and $\boldsymbol{\mu} \in \mathbb{R}^m$ are the optimization variables, $\mathcal{K} = \{0\}^r \times \mathcal{S}^{m_1} \times \cdots \times \mathcal{S}^{m_q}$ with $\mathcal{S}^p$ as the standard second-order cone of dimension p

$$\mathcal{S}^p = \{(y, \mathbf{x}) \in \mathbb{R} \times \mathbb{R}^{p-1} | \|\mathbf{x}\| \leq y\}, \tag{5.3.3}$$

and $\mathcal{S}^1$ is defined as the cone of nonnegative reals, i.e., $\mathbb{R}_+$. Here, each $\mathcal{S}^i$ has dimension m_i such that $(r + \sum_{i=1}^q m_i) = m$, $\mathbf{A} \in \mathbb{R}^{m\times n}$, $\mathbf{b} \in \mathbb{R}^m$, $\mathbf{c} \in \mathbb{R}^n$. The equivalence means that the optimal solution or the certificate of infeasibility of the original problem $\mathscr{P}$ can be extracted from the solution to the equivalent cone program $\mathscr{P}_{\text{cone}}$. To reduce the storage and memory overhead, we store the matrix $\mathbf{A}$, vectors $\mathbf{b}$ and $\mathbf{c}$ in the sparse form [138] by only storing the non-zero entries.

The general idea of such transformation is to rewrite the original problem $\mathscr{P}$ into a Smith form by introducing a new variable for each subexpression in disciplined convex programming form [131] of problem $\mathscr{P}$. The details are presented in the Appendix. Working with this transformed standard cone program $\mathscr{P}_{\text{cone}}$ has the following two advantageous:

- The homogeneous self-dual embedding of the primal-dual pair of the standard cone program can be induced, thereby providing certificates of infeasibility. This will be presented in Section 5.4.1.

- The feasible set $\mathcal{V}$ (5.2.5) formed by the intersection of a finite number of constraint sets $\mathcal{P}_l$'s and $\mathcal{Q}_k$'s in the original problem $\mathscr{P}$ can be transformed into two sets in $\mathscr{P}_{\text{cone}}$: a subspace (5.3.1) and a convex cone $\mathcal{K}$, which is formed by the Cartesian product of second-order cones. This salient feature will be exploited to enable parallel and scalable computing, as will be presented in Section 5.4.2.

5.3.2 Matrix Stuffing for Fast Transformation

Inspired by the work [125] on fast optimization code deployment for embedding second-order cone program, we propose to use the matrix stuffing technique [67, 125] to transform the original problem into the standard cone program quickly. Specifically, for any given network size, we first generate and store the structure that maps the original problem $\mathscr{P}$ to the standard form $\mathscr{P}_{\text{cone}}$. Thus, the pre-stored standard form structure includes the problem dimensions (i.e., m and n), the description of $\mathcal{V}$ (i.e., the array of the cone sizes $[r, m_1, m_2, \ldots, m_q]$), and the symbolic problem parameters $\mathbf{A}$, $\mathbf{b}$ and $\mathbf{c}$. This procedure can be done offline.

Based on the pre-stored structure, for a given problem instance $\mathscr{P}$, we only need to copy its parameters (i.e., the channel coefficients $\mathbf{h}_K$'s, maximum transmit powers P_l's, SINR targets γ_k's) to the corresponding data in the standard form $\mathscr{P}_{\text{cone}}$ (i.e., $\mathbf{A}$ and $\mathbf{b}$). Details of the exact description of copying data for transformation are presented in the Appendix. As the procedure for transformation only needs to copy memory, it thus is suitable for fast transformation and can avoid repeated parsing and generating as in parser/solver modeling frameworks like CVX.

Remark 7. *As shown in the Appendix, the dimension of the transformed standard cone program $\mathscr{P}_{\text{cone}}$ becomes $m = (L + K) + (2NK + 1) + \sum_{l=1}^{L}(2KN_l + 1) + K(2K + 2)$, which is much larger than the dimension of the original problem, i.e., $2NK$ in the equivalent real-field. But as discussed above, there are unique advantages of working with this standard form, which compensate for the increase in the size, as will be explicitly presented in later sections.*

5.4 The Operator Splitting Method For Large-Scale Homogeneous Self-Dual Embedding

Although the standard cone program $\mathscr{P}_{\text{cone}}$ itself is suitable for parallel computing via the operator splitting method [139], directly working on this problem may fail to provide certificates of infeasibility. To address this limitation, based on the recent work by O'Donoghue *et. al* [96], we propose to solve the homogeneous self-dual embedding [132] of the primal-dual pair of the cone program $\mathscr{P}_{\text{cone}}$. The resultant homogeneous self-dual embedding is further solved via the operator splitting method, a.k.a. the ADMM algorithm [59].

5.4.1 Homogeneous Self-Dual Embedding of Cone Programming

The basic idea of the homogeneous self-dual embedding is to embed the primal and dual problems of the cone program $\mathscr{P}_{\text{cone}}$ into a single feasibility problem (i.e., finding a feasible point of the intersection of a subspace and a convex set) such that either the optimal solution or the certificate of infeasibility of the original cone program $\mathscr{P}_{\text{cone}}$ can be extracted from the solution of the embedded problem.

The dual problem of $\mathscr{P}_{\text{cone}}$ is given by [96]

$$\begin{aligned} \mathscr{D}_{\text{cone}} : \underset{\boldsymbol{\eta},\boldsymbol{\lambda}}{\text{maximize}} \; & -\mathbf{b}^T\boldsymbol{\eta} \\ \text{subject to} \; & -\mathbf{A}^T\boldsymbol{\eta} + \boldsymbol{\lambda} = \mathbf{c} \\ & (\boldsymbol{\lambda}, \boldsymbol{\eta}) \in \{0\}^n \times \mathcal{K}^*, \end{aligned} \tag{5.4.1}$$

where $\boldsymbol{\lambda} \in \mathbb{R}^n$ and $\boldsymbol{\eta} \in \mathbb{R}^m$ are the dual variables, $\mathcal{K}^*$ is the dual cone of the convex cone $\mathcal{K}$. Note that $\mathcal{K} = \mathcal{K}^*$, i.e., $\mathcal{K}$ is self dual. Define the optimal values of the primal program $\mathscr{P}_{\text{cone}}$ and dual program $\mathscr{P}_{\text{cone}}$ are $p^\star$ and $d^\star$, respectively. Let $p^\star = +\infty$ and $p^\star = -\infty$ indicate primal infeasibility and unboundedness, respectively. Similarly, let $d^\star = -\infty$ and $d^\star = +\infty$ indicate the dual infeasibility and unboundedness, respectively. We assume strong duality for the convex cone program $\mathscr{P}_{\text{cone}}$, i.e., $p^\star = d^\star$, including cases when they are infinite. This is a standard assumption for practically designing solvers for conic programs, e.g., it is assumed in [96, 120–122, 132]. Besides, we do not make any regularity assumption on the feasibility and boundedness assumptions on the primal and dual problems.

5.4.1.1 Certificates of Infeasibility

Given the cone program $\mathscr{P}_{\text{cone}}$, a main task is to detect feasibility. In [50, Theorem 1], a sufficient condition for the existence of strict feasible solution was provided for the transmit power minimization problem without power constraints. However, for the general problem $\mathscr{P}$ with per-MU QoS constraints and per-RRH transmit power constraints, it is difficult to obtain such a feasibility condition analytically. Therefore, most existing works either assume that the underlying problem is feasible [18] or provide heuristic ways to handle infeasibility [127].

Nevertheless, the only way to detect infeasibility effectively is to provide a certificate or proof of infeasibility as presented in the following proposition.

Proposition 5. *[Certificates of Infeasibility] The following system*

$$\mathbf{A}\boldsymbol{\nu} + \boldsymbol{\mu} = \mathbf{b}, \boldsymbol{\mu} \in \mathcal{K}, \tag{5.4.2}$$

is infeasible if and only if the following system is feasible

$$\mathbf{A}^T\boldsymbol{\eta} = \mathbf{0}, \boldsymbol{\eta} \in \mathcal{K}^\star, \mathbf{b}^T\boldsymbol{\eta} < 0. \tag{5.4.3}$$

Therefore, any dual variable $\boldsymbol{\eta}$ satisfying the system (5.4.3) provides a certificate or proof that the primal program $\mathscr{P}_{cone}$ (equivalently the original problem $\mathscr{P}$) is infeasible.

Similarly, any primal variable $\boldsymbol{\nu}$ satisfying the following system

$$-\mathbf{A}\boldsymbol{\nu} \in \mathcal{K}, \mathbf{c}^T\boldsymbol{\nu} < 0, \tag{5.4.4}$$

is a certificate of the dual program $\mathscr{D}_{cone}$ infeasibility.

Proof. This result directly follows the theorem of strong alternatives [22, Section 5.8.2]. □

5.4.1.2 Optimality Conditions

If the transformed standard cone program $\mathscr{P}_{\text{cone}}$ is feasible, then $(\boldsymbol{\nu}^\star, \boldsymbol{\mu}^\star, \boldsymbol{\lambda}^\star, \boldsymbol{\eta}^\star)$ are optimal if and only if they satisfy the following Karush-Kuhn-Tucker (KKT) conditions

$$\mathbf{A}\boldsymbol{\nu}^\star + \boldsymbol{\mu}^\star - \mathbf{b} = \mathbf{0} \tag{5.4.5}$$

$$\mathbf{A}^T\boldsymbol{\eta}^\star - \boldsymbol{\lambda}^\star + \mathbf{c} = \mathbf{0} \tag{5.4.6}$$

$$(\boldsymbol{\eta}^\star)^T\boldsymbol{\mu}^\star = 0 \tag{5.4.7}$$

$$(\boldsymbol{\nu}^\star, \boldsymbol{\mu}^\star, \boldsymbol{\lambda}^\star, \boldsymbol{\eta}^\star) \in \mathbb{R}^n \times \mathcal{K} \times \{0\}^n \times \mathcal{K}^*. \tag{5.4.8}$$

In particular, the complementary slackness condition (C.4.8) can be rewritten as

$$\mathbf{c}^T\boldsymbol{\nu}^\star + \mathbf{b}^T\boldsymbol{\eta}^\star = 0, \tag{5.4.9}$$

which explicitly forces the duality gap to be zero.

5.4.1.3 Homogeneous Self-Dual Embedding

We can first detect feasibility by Proposition 5, and then solve the KKT system if the problem is feasible and bounded. However, the disadvantage of such a two-phase method is that two related problems (i.e., checking feasibility and solving KKT conditions) need to be solved sequentially [132]. To avoid such inefficiency, we propose to solve the following homogeneous self-dual embedding [132]:

$$\mathbf{A}\boldsymbol{\nu} + \boldsymbol{\mu} - \mathbf{b}\tau = \mathbf{0} \tag{5.4.10}$$

$$\mathbf{A}^T\boldsymbol{\eta} - \boldsymbol{\lambda} + \mathbf{c}\tau = \mathbf{0} \tag{5.4.11}$$

$$\mathbf{c}^T\boldsymbol{\nu} + \mathbf{b}^T\boldsymbol{\eta} + \kappa = \mathbf{0} \tag{5.4.12}$$

$$(\boldsymbol{\nu}, \boldsymbol{\mu}, \boldsymbol{\lambda}, \boldsymbol{\eta}, \tau, \kappa) \in \mathbb{R}^n \times \mathcal{K} \times \{0\}^n \times \mathcal{K}^* \times \mathbb{R}_+ \times \mathbb{R}_+, \tag{5.4.13}$$

to embed all the information on the infeasibility and optimality into a single system by introducing two new nonnegative variables τ and κ, which encode different outcomes. The homogeneous self-dual embedding thus can be rewritten as the following compact form

$$\begin{aligned} \mathscr{F}_{\text{HSD}} &: \text{find } (\mathbf{x}, \mathbf{y}) \\ &\text{subject to } \mathbf{y} = \mathbf{Q}\mathbf{x} \\ &\qquad\qquad \mathbf{x} \in \mathcal{C}, \mathbf{y} \in \mathcal{C}^*, \end{aligned} \tag{5.4.14}$$

where

$$\underbrace{\begin{bmatrix} \boldsymbol{\lambda} \\ \boldsymbol{\mu} \\ \kappa \end{bmatrix}}_{\mathbf{y}} = \underbrace{\begin{bmatrix} \mathbf{0} & \mathbf{A}^T & \mathbf{c} \\ -\mathbf{A} & \mathbf{0} & \mathbf{b} \\ -\mathbf{c}^T & -\mathbf{b}^T & \mathbf{0} \end{bmatrix}}_{\mathbf{Q}} \underbrace{\begin{bmatrix} \boldsymbol{\nu} \\ \boldsymbol{\eta} \\ \tau \end{bmatrix}}_{\mathbf{x}}, \tag{5.4.15}$$

$\mathbf{x} \in \mathbb{R}^{m+n+1}$, $\mathbf{y} \in \mathbb{R}^{m+n+1}$, $\mathbf{Q} \in \mathbb{R}^{(m+n+1)\times(m+n+1)}$, $\mathcal{C} = \mathbb{R}^n \times \mathcal{K}^* \times \mathbb{R}_+$ and $\mathcal{C}^* = \{0\}^n \times \mathcal{K} \times \mathbb{R}_+$. This system has a trivial solution with all variables as zeros.

The homogeneous self-dual embedding problem $\mathscr{F}_{\text{HSD}}$ is thus a feasibility problem finding a nonzero solution in the intersection of a subspace and a convex cone. Let $(\boldsymbol{\nu}, \boldsymbol{\mu}, \boldsymbol{\lambda}, \boldsymbol{\eta}, \tau, \kappa)$ be a non-zero solution of the homogeneous self-dual embedding. We then have the following remarkable trichotomy derived in [132]:

- **Case 1**: $\tau > 0, \kappa = 0$, then

$$\hat{\boldsymbol{\nu}} = \boldsymbol{\nu}/\tau, \hat{\boldsymbol{\eta}} = \boldsymbol{\eta}/\tau, \hat{\boldsymbol{\mu}} = \boldsymbol{\mu}/\tau \tag{5.4.16}$$

  are the primal and dual solutions to the cone program $\mathscr{P}_{\text{cone}}$.

- **Case 2**: $\tau = 0, \kappa > 0$; this implies $\mathbf{c}^T\boldsymbol{\nu} + \mathbf{b}^T\boldsymbol{\eta} < 0$, then

  1. If $\mathbf{b}^T\boldsymbol{\eta} < 0$, then $\hat{\boldsymbol{\eta}} = \boldsymbol{\eta}/(-\mathbf{b}^T\boldsymbol{\eta})$ is a certificate of the primal infeasibility as

$$\mathbf{A}^T\hat{\boldsymbol{\eta}} = \mathbf{0}, \hat{\boldsymbol{\eta}} \in \mathcal{V}^\star, \mathbf{b}^T\hat{\boldsymbol{\eta}} = -1. \tag{5.4.17}$$

  2. If $\mathbf{c}^T\boldsymbol{\nu} < 0$, then $\hat{\boldsymbol{\nu}} = \boldsymbol{\nu}/(-\mathbf{c}^T\hat{\boldsymbol{\nu}})$ is a certificate of the dual infeasibility as

$$-\mathbf{A}\hat{\boldsymbol{\nu}} \in \mathcal{V}, \mathbf{c}^T\hat{\boldsymbol{\nu}} = -1. \tag{5.4.18}$$

- **Case 3**: $\tau = \kappa = 0$; no conclusion can be made about the cone problem $\mathscr{P}_{\text{cone}}$.

Therefore, from the solution to the homogeneous self-dual embedding, we can extract either the optimal solution (based on (D.1.17)) or the certificate of infeasibility for the original problem. Furthermore, as the set (5.4.13) is a Cartesian product of a finite number of sets, this will enable parallelizable algorithm design. With the distinct advantages of the homogeneous self-dual embedding, in the sequel, we focus on developing efficient algorithms to solve the large-scale feasibility problem $\mathscr{F}_{\text{HSD}}$ via the operator splitting method.

5.4.2 The Operator Splitting Method

Conventionally, the convex homogeneous self-dual embedding $\mathscr{F}_{\text{HSD}}$ can be solved via the interior-point method, e.g., [120–122, 132]. However, such second-order method has cubic computational complexity for the scend-order cone programs [114], and thus the computational cost will be prohibitive for large-scale problems. Instead, O'Donoghue *et al.* [96] develop a first-order optimization algorithm based on the operator splitting method, i.e., the ADMM algorithm [59], to solve the large-scale homogeneous self-dual embedding. The key observation is that the convex cone constraint in $\mathscr{F}_{\text{HSD}}$ is the Cartesian product of standard convex cones (i.e., second-order cones, nonnegative reals and free variables), which enables parallelizable computing. Furthermore, we will show that the computation of each iteration in the operator splitting method is very cheap and efficient.

Specifically, the homogeneous self-dual embedding $\mathscr{F}_{\text{HSD}}$ can be rewritten as

$$\text{minimize } I_{\mathcal{C}\times\mathcal{C}^*}(\mathbf{x}, \mathbf{y}) + I_{\mathbf{Qx}=\mathbf{y}}(\mathbf{x}, \mathbf{y}), \tag{5.4.19}$$

where $I_{\mathcal{S}}$ is the indicator function of the set $\mathcal{S}$, i.e., $I_{\mathcal{S}}(z)$ is zero for $z \in \mathcal{S}$ and it is $+\infty$ otherwise. By replicating variables $\mathbf{x}$ and $\mathbf{y}$, problem (5.4.19) can be transformed into the following consensus form [59, Section 7.1]

$$\begin{aligned} \mathscr{P}_{\text{ADMM}} : \text{minimize } & I_{\mathcal{C}\times\mathcal{C}^*}(\mathbf{x}, \mathbf{y}) + I_{\mathbf{Q}\tilde{\mathbf{x}}=\tilde{\mathbf{y}}}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) \\ \text{subject to } & (\mathbf{x}, \mathbf{y}) = (\tilde{\mathbf{x}}, \tilde{\mathbf{y}}), \end{aligned} \tag{5.4.20}$$

which is readily to be solved by the operator splitting method.

Applying the ADMM algorithm [59, Section 3.1] to problem $\mathscr{P}_{\text{ADMM}}$ and eliminating the dual variables by exploiting the self-dual property of the problem $\mathscr{F}_{\text{HSD}}$ (Please refer to [96, Section 3] on how to simplify the ADMM algorithm), the final algorithm is shown as follows:

$$\mathcal{OS}_{\text{ADMM}} : \begin{cases} \tilde{\mathbf{x}}^{[i+1]} = (\mathbf{I} + \mathbf{Q})^{-1}(\mathbf{x}^{[i]} + \mathbf{y}^{[i]}) \\ \mathbf{x}^{[i+1]} = \Pi_{\mathcal{C}}(\tilde{\mathbf{x}}^{[i+1]} - \mathbf{y}^{[i]}) \\ \mathbf{y}^{[i+1]} = \mathbf{y}^{[i]} - \tilde{\mathbf{x}}^{[i+1]} + \mathbf{x}^{[i+1]}, \end{cases} \tag{5.4.21}$$

where $\Pi_{\mathcal{C}}(\mathbf{x})$ denotes the Euclidean projection of $\mathbf{x}$ onto the set $\mathcal{C}$. This algorithm has the $\mathcal{O}(1/k)$ convergence rate [140] with k as the iteration counter (i.e., the ϵ accuracy can be achieved in $\mathcal{O}(1/\epsilon)$ iterations) and will not converge to zero if a nonzero solution exists [96, Section 3.4]. Empirically, this algorithm can converge to modest accuracy within a reasonable amount of time. As the last step is computationally trivial, in the sequel, we will focus on how to solve the first two steps efficiently.

5.4.2.1 Subspace Projection via Factorization Caching

The first step in the algorithm $\mathcal{OS}_{\text{ADMM}}$ is a subspace projection. After simplification [96, Section 4], we essentially need to solve the following linear equation at each iteration, i.e.,

$$\underbrace{\begin{bmatrix} \mathbf{I} & -\mathbf{A}^T \\ -\mathbf{A} & -\mathbf{I} \end{bmatrix}}_{\mathbf{S}} \underbrace{\begin{bmatrix} \boldsymbol{\nu} \\ -\boldsymbol{\eta} \end{bmatrix}}_{\mathbf{x}} = \underbrace{\begin{bmatrix} \boldsymbol{\nu}^{[i]} \\ \boldsymbol{\eta}^{[i]} \end{bmatrix}}_{\mathbf{b}}, \tag{5.4.22}$$

for the given $\boldsymbol{\nu}^{[i]}$ and $\boldsymbol{\eta}^{[i]}$ at iteration i, where $\mathbf{S} \in \mathbb{R}^{d\times d}$ with $d = m + n$ is a *symmetric quasidefinite* matrix [141]. To enable quicker inversions and reduce memory overhead via exploiting the sparsity of the matrix $\mathbf{S}$, the sparse permuted LDL^T factorization [138] method can be adopted. Specifically, such factor-solve method can be carried out by first computing the sparse permuted LDL^T factorization as follows

$$\mathbf{S} = \mathbf{PLDL}^T\mathbf{P}^T, \tag{5.4.23}$$

where $\mathbf{L}$ is a lower triangular matrix, $\mathbf{D}$ is a diagonal matrix [139] and $\mathbf{P}$ with $\mathbf{P}^{-1} = \mathbf{P}^T$ is a permutation matrix to fill-in of the factorization [138], i.e., the number of nonzero entries in $\mathbf{L}$. Such factorization exists for any permutation $\mathbf{P}$, as the matrix $\mathbf{S}$ is symmetric quasidefinite [141, Theorem 2.1]. Computing the factorization costs much less than $\mathcal{O}(1/3d^3)$ flops, while the exact value depends on d and the sparsity pattern of $\mathbf{S}$ in a complicated way. Note that such factorization only needs to be computed once in the first iteration and can be cached for re-using in the sequent iterations for subspace projections. This is called the *factorization caching* technique [96].

Given the cached factorization (5.4.23), solving subsequent projections $\mathbf{x} = \mathbf{S}^{-1}\mathbf{b}$ (5.4.22) can be carried out by solving the following much easier equations:

$$\mathbf{P}\mathbf{x}_1 = \mathbf{b}, \mathbf{L}\mathbf{x}_2 = \mathbf{x}_1, \mathbf{D}\mathbf{x}_3 = \mathbf{x}_2, \mathbf{L}^T\mathbf{x}_4 = \mathbf{x}_3, \mathbf{P}^T\mathbf{x} = \mathbf{x}_4, \tag{5.4.24}$$

which cost zero flops, $\mathcal{O}(sd)$ flops by forward substitution with s as the number of nonzero entries in $\mathbf{L}$, $\mathcal{O}(d)$ flops, $\mathcal{O}(sd)$ flops by backward substitution, and zero flops, respectively [22, Appendix C].

5.4.2.2 Cone Projection via Proximal Operator Evaluation

The second step in the algorithm $\mathcal{OS}_{\text{ADMM}}$ is to project a point $\boldsymbol{\omega}$ onto the cone $\mathcal{C}$. As $\mathcal{C}$ is the Cartesian product of the finite number of convex cones $\mathcal{C}_i$, we can perform projection onto $\mathcal{C}$ by projecting onto $\mathcal{C}_i$ separately and in parallel. Furthermore, the projection onto each convex cone can be done with closed-forms. Specifically, for nonnegative real $\mathcal{C}_i = \mathbb{R}_+$, we have that [142, Section 6.3.1]

$$\Pi_{\mathcal{C}_i}(\boldsymbol{\omega}) = \boldsymbol{\omega}_+, \tag{5.4.25}$$

where the nonnegative part operator $(\cdot)_+$ is taken elementwise. For the second-order cone $\mathcal{C}_i = \{(y, \mathbf{x}) \in \mathbb{R} \times \mathbb{R}^{p-1} | \|\mathbf{x}\| \leq y\}$, we have that [142, Section 6.3.2]

$$\Pi_{\mathcal{C}_i}(\boldsymbol{\omega}, \tau) = \begin{cases} 0, \|\boldsymbol{\omega}\|_2 \leq -\tau \\ (\boldsymbol{\omega}, \tau), \|\boldsymbol{\omega}\|_2 \leq \tau \\ (1/2)(1 + \tau/\|\boldsymbol{\omega}\|_2)(\boldsymbol{\omega}, \|\boldsymbol{\omega}\|_2), \|\boldsymbol{\omega}\|_2 \geq |\tau|. \end{cases} \tag{5.4.26}$$

In summary, we have presented that each step in the algorithm $\mathcal{OS}_{\text{ADMM}}$ can be computed efficiently. In particular, from both (5.4.25) and (5.4.26), we see that the cone projection can be carried out very efficiently with closed-forms, leading to parallelizable algorithms.

5.5 Practical Implementation Issues

In previous sections, we have presented the unified two-stage framework for large-scale convex optimization in dense wireless cooperative networks. In this section, we will focus on the implementation issues of the proposed framework.

5.5.1 Automatic Code Generation for Fast Transformation

In the Appendix, we describe a systematic way to transform the original problem to the standard cone programming form. The resultant structure that maps the original problem to the standard form can be stored and re-used for fast transforming via matrix stuffing. This can significantly reduce the modeling overhead compared with the parse/solver modeling frameworks like CVX. However, it requires tedious manual works to find the mapping and may not be easy to verify the correctness of the generated mapping. Chu *et al.* [125] gave such an attempt intending to automatically generate the code for matrix stuffing. However, the corresponding software package QCML [125], so far, is far from complete and may not be suitable for our applications. Extending the numerical-based transformation modeling frameworks like CVX to the symbolic-based transformation modeling frameworks like QCML is not trivial and requires tremendous mathematical and technical efforts. In this chapter, we derive the mapping in the Appendix manually and verify the correctness by comparing with CVX through extensive simulations.

5.5.2 Implementation of the Operator Splitting Algorithm

Theoretically, the presented operator splitting algorithm $\mathcal{OS}_{\text{ADMM}}$ is compact, parameter-free, with parallelizable computing and linear convergence. Practically, there are typically several ways to improve the efficiency of the algorithm. In particular, there are various tricks that can be employed to improve the convergence rate, e.g, over-relaxation, warm-staring and problem data scaling as described in [96]. In the dense wireless cooperative networks with multi-entity collaborative architecture, we are interested in two particular ways to speed up the subspace projection of the algorithm $\mathcal{OS}_{\text{ADMM}}$, which is the main computational bottleneck. Specifically, one way is to use the parallel algorithms for the factorization (5.4.23) by utilizing the distributed computing and memory resources [143]. For instance, in the cloud computing environments in Cloud-RAN, all the baseband units share the computing, memory and storage resources in a single baseband unit pool [10, 29]. Another way is to leverage *symbolic* factorization (5.4.23) to speed up the numerical factorization for each problem instance, which is a general idea for the code generation system CVXGEN [137] for realtime convex quadratic optimization [144] and the interior-point method based SOCP solver [145] for embedded systems. Eventually, the ADMM solver in Fig. 5.1 can be symbolic based so as to provide numerical solutions for each problem instance extremely fast and in a realtime way. However, this requires further investigation.

5.6 Numerical Results

In this section, we simulate the proposed two-stage based large-scale convex optimization framework for performance optimization in dense wireless cooperative networks. We consider the following channel model for the link between the k-th MU and the l-th RRH:

$$\mathbf{h}_{kl} = 10^{-L(d_{kl})/20}\sqrt{\varphi_{kl}s_{kl}}\mathbf{f}_{kl}, \forall k, l, \tag{5.6.1}$$

where $L(d_{kl})$ is the path-loss in dB at distance d_{kl} as shown in [29, Table I], s_{kl} is the shadowing coefficient, φ_{kl} is the antenna gain and $\mathbf{f}_{kl}$ is the small-scale fading coefficient. We use the standard cellular network parameters as showed in [29, Table I]. All the simulations are carried out on a personal computer with 3.2 GHz quad-core Intel Core i5 processor and 8 GB of RAM running Linux. The reference implementation of the operator splitting algorithm SCS is available at (`https://github.com/cvxgrp/scs`), which is a general software package for solving large-scale convex cone problems based on [96] and can be called by the modeling frameworks CVX and CVXPY [146]. The settings (e.g., the stopping criteria) of SCS can be found in [96].

The proposed two-stage approach framework, termed "Matrix Stuffing+SCS", is compared with the following state-of-art frameworks:

- CVX+SeDuMi/SDPT3/MOSEK: This category adopts second-order methods. The modeling framework CVX will first automatically transform the original problem instance (e.g., the problem $\mathscr{P}$ written in the disciplined convex programming form) into the standard cone programming form and then call an interior-point solver, e.g., SeDuMi [120], SDPT3 [121] or MOSEK [122].

- CVX+SCS: In this first-order method based framework, CVX first transforms the original problem instance into the standard form and then calls the operator splitting solver SCS.

We define the "**modeling time**" as the transformation time for the first stage, the "**solving time**" as the time spent for the second stage, and the "**total time**" as the time of the two stages for solving one problem instance. As the large-scale convex optimization algorithm should scale well to both the modeling part and the solving part simultaneously, the time comparison of each individual stage will demonstrate the effectiveness of the proposed two-stage approach.

Given the network size, we first generate and store the problem structure of the standard form $\mathscr{P}_{\text{cone}}$, i.e., the structure of $\mathbf{A}$, $\mathbf{b}$, $\mathbf{c}$ and the descriptions of $\mathcal{V}$. As this procedure can be done offline for all the candidate network sizes, we thus ignore this step for time comparison. We repeat the following procedures to solve the large-scale convex optimization problem $\mathscr{P}$ with different parameters and sizes using the proposed framework "Matrix Stuffing+SCS":

1. Copy the parameters in the problem instance $\mathscr{P}$ to the data in the pre-stored structure of the standard cone program $\mathscr{P}_{\text{cone}}$.

2. Solve the resultant standard cone programming instance $\mathscr{P}_{\text{cone}}$ using the solver SCS.

3. Extract the optimal solutions of $\mathscr{P}$ from the solutions to $\mathscr{P}_{\text{cone}}$ by the solver SCS.

Finally, note that all the interior-point solvers are multiple threaded (i.e., they can utilize multiple threads to gain extra speedups), while the operator splitting algorithm solver SCS is single threaded. Nevertheless, we will show that SCS performs much faster than the interior-point solvers. We also emphasize that the operator splitting method aims at scaling well to large problem sizes and thus provides solutions to modest accuracy within reasonable time, while the interior-point method intends to provide highly accurate solutions. Furthermore, the modeling framework CVX aims at rapid prototyping and providing a user-friendly tool for automatically transformations for general problems, while the matrix-stuffing technique targets at scaling to large-scale problems for the specific problem family $\mathscr{P}$. Therefore, these frameworks and solvers are not really comparable with different purposes and application capabilities. We mainly use them to verify the effectiveness and reliability of our proposed framework in terms of the solution time and the solution quality.

5.6.1 Effectiveness and Reliability of the Proposed Large-Scale Convex Optimization Framework

Consider a network with L 2-antenna RRHs and K single-antenna MUs uniformly and independently distributed in the square region $[-3000, 3000] \times [-3000, 3000]$ meters with $L = K$. We consider the total transmit power minimization problem $\mathscr{P}_1(\boldsymbol{\gamma})$ with the QoS requirements for each MU as $\gamma_k = 5$ dB, $\forall k$. Table 5.1 demonstrates the comparison of the running

Table 5.1: Time and Solution Results for Different Convex Optimization Frameworks

Network Size ($L = K$)		20	50	100	150	200
CVX+SeDuMi	**Total Time** [sec]	**8.1164**	N/A	N/A	N/A	N/A
	Objective [W]	12.2488	N/A	N/A	N/A	N/A
CVX+SDPT3	**Total Time** [sec]	**5.0398**	**330.6814**	N/A	N/A	N/A
	Objective [W]	12.2488	6.5216	N/A	N/A	N/A
CVX+MOSEK	**Total Time** [sec]	**1.2072**	**51.6351**	N/A	N/A	N/A
	Objective [W]	12.2488	6.5216	N/A	N/A	N/A
CVX+SCS	**Total Time** [sec]	**0.8501**	**5.6432**	**51.0472**	**227.9894**	**725.6173**
	Modeling Time [sec]	0.7563	4.4301	38.6921	178.6794	534.7723
	Objective [W]	12.2505	6.5215	3.1303	2.0693	1.5404
Matrix Stuffing+SCS	**Total Time** [sec]	**0.1137**	**2.7222**	**26.2242**	**90.4190**	**328.2037**
	Modeling Time [sec]	0.0128	0.2401	2.4154	9.4167	29.5813
	Objective [W]	12.2523	6.5193	3.1296	2.0689	1.5403

time and solutions using different convex optimization frameworks. Each point of the simulation results is averaged over 100 randomly generated network realizations (i.e., one small scaling fading realization for each large-scale fading realization).

For the modeling time comparisons, this table shows that the value of the proposed matrix stuffing technique ranges between 0.01 and 30 seconds[2] for different network sizes and can speedup about 15x to 60x compared to the parser/solver modeling framework CVX. In particular, for large-scale problems, the transformation using CVX is time consuming and becomes the bottleneck, as the "**modeling time**" is comparable and even larger than the "**solving time**". For example, when $L = 150$, the "**modeling time**" using CVX is about 3 minutes, while the matrix stuffing only requires about 10 seconds. Therefore, the matrix stuffing for fast transformation is essential for solving large-scale convex optimization problems quickly.

For the solving time (which can be easily calculated by subtracting the "**modeling time**" from the "**total time**") using different solvers, this table shows that the operator splitting solver can speedup by several orders of magnitude over the interior-point solvers. For example, for $L = 50$, it can speedup about 20x and 130x over MOSEK[3] and SDPT3, respectively, while SeDuMi is inapplicable for this problem size as the running time exceeds the predefined maximum value, i.e., one hour. In particular, all the interior-point solvers fail to solve large-scale problems (i.e., $L = 100, 150, 200$), denoted as "N/A", while the operator splitting solver SCS can scale well to large problem sizes. For the largest problems with $L = 200$, the

[2] This value can be significantly reduced in practical implementations, e.g., at the BBU pool in Cloud-RAN, which, however, requires substantial further investigation. Meanwhile, the results effectively confirm that the proposed matrix stuffing technique scales well to large-scale problems.

[3] Although SeDuMi, SDPT3 and MOSEK (commercial software) are all based on the interior-point method, the implementation efficiency of the corresponding software packages varies substantially. In the following simulations, we mainly compare with the state-of-art public solver SDPT3.

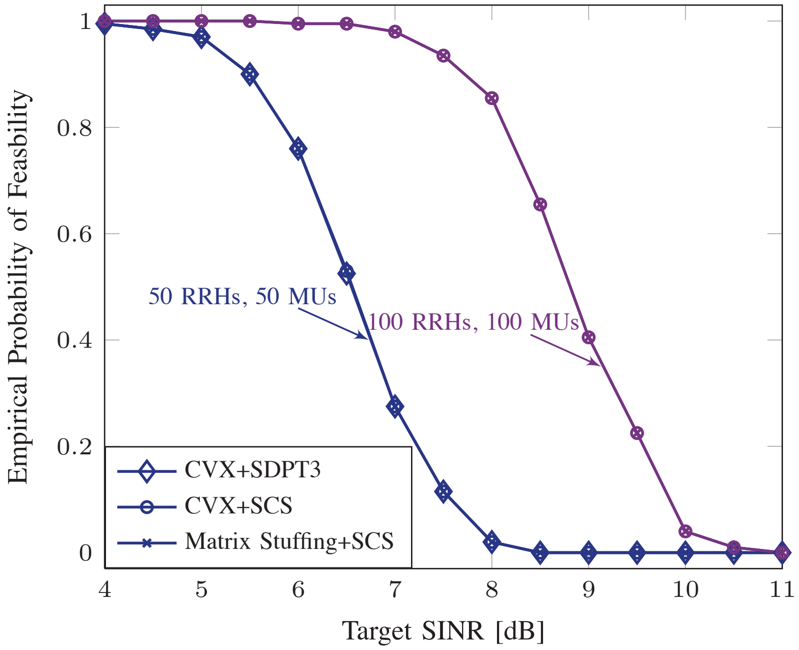

Figure 5.2: The empirical probability of feasibility versus target SINR with different network sizes.

operator splitting solver can solve them in about 5 minutes.

For the quality of the solutions, this table shows that the propose framework can provide a solution to modest accuracy within much less time. For the two problem sizes, i.e., $L = 20$ and $L = 50$, which can be solved by the interior-point method based frameworks, the optimal values attained by the proposed framework are within 0.03% of that obtained via the second-order method frameworks.

In summary, the proposed two-stage based large-scale convex optimization framework scales well to large-scale problem modeling and solving simultaneously. Therefore, it provides an effective way to evaluate the system performance via large-scale optimization in dense wireless networks. However, its implementation and performance in practical systems still need further investigation. In particular, this set of results indicate that the scale of cooperation in dense wireless networks may be fundamentally constrained by the computation complexity/time.

5.6.2 Infeasibility Detection Capability

A unique property of the proposed framework is its infeasibility detection capability, which will be verified in this part. Consider a network with $L = 50$ single-antenna RRHs and

$K = 50$ single-antenna MUs uniformly and independently distributed in the square region $[-2000, 2000] \times [-2000, 2000]$ meters. The empirical probabilities of feasibility in Fig. 5.2 show that the propose framework can detect the infeasibility accurately compared with the second-order method framework "CVX+SDPT3" and the first-order method framework "CVX+SCS". Each point of the simulation results is averaged over 200 randomly generated network realizations. The average ("**total time**", "**solving time**") for obtaining a single point with "CVX+SDPT3", "CVX+SCS" and "Matrix Stuffing+SCS" are (101.7635, 99.1140) seconds, (5.0754, 2.3617) seconds and (1.8549, 1.7959) seconds, respectively. This shows that the operator splitting solver can speedup about 50x over the interior-point solver.

We further consider a larger-sized network with $L = 100$ single-antenna RRHs and $K = 100$ single-antenna MUs uniformly and independently distributed in the square region $[-2000, 2000] \times [-2000, 2000]$ meters. As the second-order method framework fails to scale to this size, we only compare with the first-order method framework. Fig. 5.2 demonstrates that the proposed framework has the same infeasibility detection capability as the first-order method framework. This verifies the correctness and the reliability of the proposed fast transformation via matrix stuffing. Each point of the simulation results is averaged over 200 randomly generated network realizations. The average ("**solving time**", "**modeling time**") for obtaining a single point with "CVX+SCS" and "Matrix Stuffing+SCS" are $(41.9273, 18.6079)$ seconds and $(31.3660, 0.5028)$ seconds, respectively. This shows that the matrix stuffing technique can speedup about 40x over the numerical based parser/solver modeling framework CVX. We also note that the solving time of the proposed framework is smaller than the framework "CVX+SCS", the speedup is due to the warm-staring [96, Section 4.2].

5.6.3 Group Sparse Beamforming for Network Power Minimization

In this part, we simulate the network power minimization problem using the group sparse beamforming algorithm [29, Algorithm 2]. We set each fronthaul link power consumption as $5.6W$ and set the power amplifier efficiency coefficient for each RRH as 25%. In this algorithm, a sequence of convex feasibility problems need to be solved to determine the active RRHs and one convex optimization problem needs to be solved to determine the transmit beamformers. This relies on the infeasibility detection capability of the proposed framework.

Consider a network with $L = 20$ 2-antenna RRHs and $K = 20$ single-antenna MUs

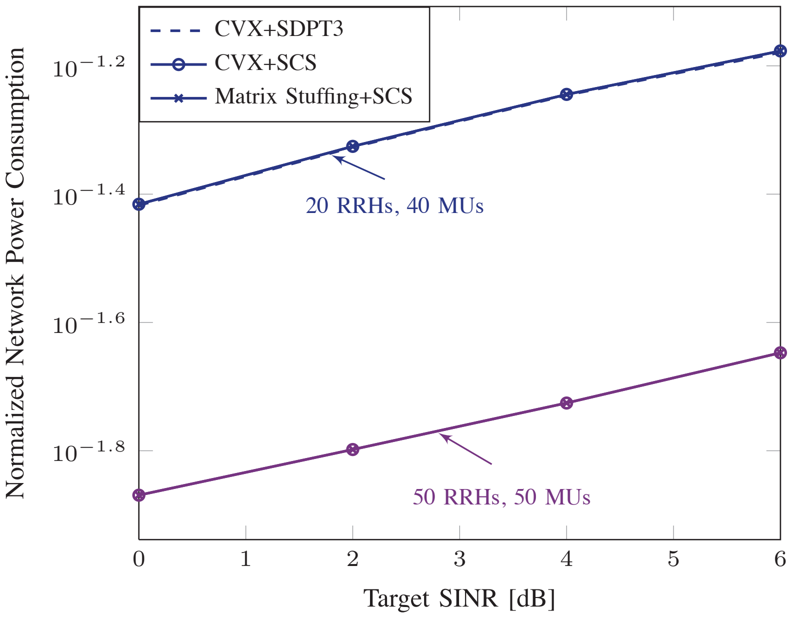

Figure 5.3: Average normalized network power consumption versus target SINR with different network sizes.

uniformly and independently distributed in the square region $[-1000, 1000] \times [-1000, 1000]$ meters. Each point of the simulation results is averaged over 50 randomly generated network realizations. Fig. 5.3 demonstrates the accuracy of the solutions in the network power consumption obtained by the proposed framework compared with the second-order method framework "CVX+SDPT3" and the first-order method framework "CVX+SCS". The average ("**total time**", "**solving time**") for obtaining a single point with "CVX+SDPT3", "CVX+SCS" and "Matrix Stuffing+SCS" are $(19.1822, 10.8831)$ seconds, $(8.4101, 0.4255)$ seconds and $(0.5857, 0.5222)$ seconds, respectively. This shows that the operator splitting solver can speedup about 25x over the interior-point solver.

We further consider a larger-sized network with $L = 50$ 2-antenna RRHs and $K = 50$ single-antenna MUs uniformly and independently distributed in the square region $[-3000, 3000] \times [-3000, 3000]$ meters. As the second-order method framework is not applicable to this problem size, we only compare with the first-order method framework. Each point of the simulation results is averaged over 50 randomly generated network realizations. Fig. 5.3 shows that the proposed framework can achieve the same solutions in network power consumption as the first-order method framework "CVX+SCS". The average ("**solving time**", "**modeling time**") for obtaining a single point with "CVX+SCS" and "Matrix Stuffing+SCS" are $(11.9643, 69.0520)$ seconds and $(14.6559, 2.1567)$ seconds, respectively. This shows that

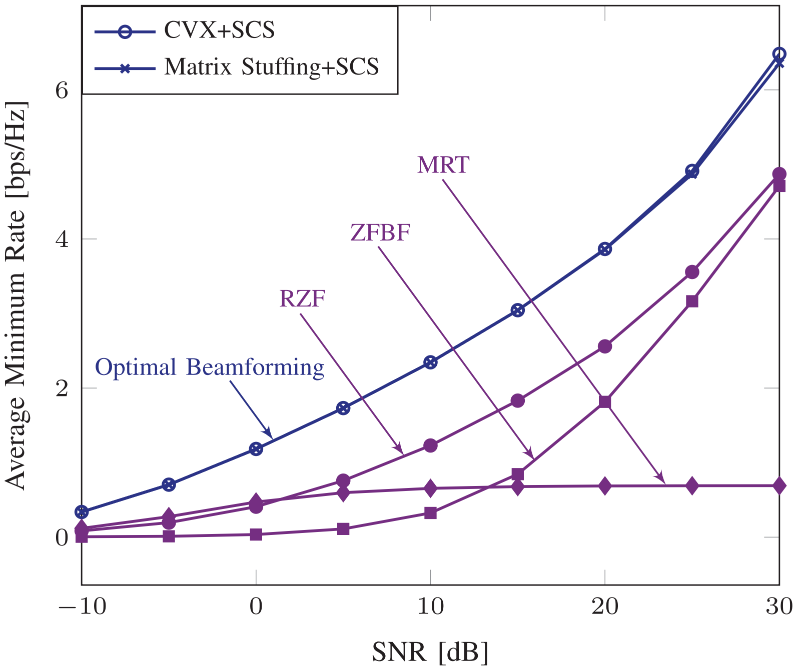

Figure 5.4: The minimum network-wide achievable versus transmit SNR with 55 single-antenna RRHs and 50 single-antenna MUs.

the matrix stuffing technique can speedup about 30x over the numerical based parser/solver modeling framework CVX.

In summary, Fig. 5.3 demonstrates the capability of infeasibility detection (as a sequence of convex feasibility problems need to be solved in the RRH selection procedure), the accuracy of the solutions, and speedups provided by the proposed framework over the existing frameworks.

5.6.4 Max-min Rate Optimization

We will simulate the minimum network-wide achievable rate maximization problem using the max-min fairness optimization algorithm in [67, Algorithm 1] via the bi-section method, which requires to solve a sequence of convex feasibility problems. We will not only show the quality of the solutions and speedups provided by the proposed framework, but also demonstrate that the optimal coordinated beamformers significantly outperform the low-complexity and heuristic transmission strategies, i.e., zero-forcing beamforming (ZFBF) [129, 147], regularized zero-forcing beamforming (RZF) [148] and maximum ration transmission (MRT) [3].

Consider a network with $L = 55$ single-antenna RRHs and $K = 50$ single-antenna MUs uniformly and independently distributed in the square region $[-5000, 5000] \times [-5000, 5000]$

meters. Fig. 5.4 demonstrates the minimum network-wide achievable rate versus different SNRs (which is defined as the transmit power at all the RRHs over the receive noise power at all the MUs) using different algorithms. Each point of the simulation results is averaged over 50 randomly generated network realizations. For the optimal beamforming, this figure shows the accuracy of the solutions obtained by the proposed framework compared with the first-order method framework "CVX+SCS". The average ("**solving time**", "**modeling time**") for obtaining a single point for the optimal beamforming with "CVX+SCS" and "Matrix Stuffing+SCS" are $(176.3410,\ 55.1542)$ seconds and $(82.0180,\ 1.2012)$ seconds, respectively. This shows that the proposed framework can reduce both the solving time and modelling time via warm-starting and matrix stuffing, respectively.

Furthermore, this figure also shows that the optimal beamforming can achieve quite an improvement for the per-user rate compared to suboptimal transmission strategies RZF, ZFBF and MRT, which clearly shows the importance of developing optimal beamforming algorithms for such networks. The average ("**solving time**", "**modeling time**") for a single point using "CVX+SDPT3" for the RZF, ZFBF and MRT are (2.6210, 30.2053) seconds, (2.4592, 30.2098) seconds and (2.5966, 30.2161) seconds, respectively. Note that the solving time is very small, which is because we only need to solve a sequence of linear programming problems for power control when the directions of the beamformers are fixed during the bi-section search procedure. The main time consuming part is from transformation using CVX.

5.7 Discussions

We proposed a unified two-stage framework for large-scale optimization in dense Cloud-RAN. We showed that various performance optimization problems can be essentially solved by solving one or a sequence of convex optimization or feasibility problems. The proposed framework only requires the convexity of the underlying problems (or subproblems) without any other structural assumptions, e.g., smooth or separable functions. This is achieved by first transforming the original convex problem to the standard form via matrix stuffing and then using the ADMM algorithm to solve the homogeneous self-dual embedding of the primal-dual pair of the transformed standard cone program. Simulation results demonstrated the infeasibility detection capability, the modeling flexibility and computing scalability, and the reliability of the proposed framework.

In principle, one may apply the proposed framework to any large-scale convex optimization problems and only needs to focus on the standard form reformulation as shown in Appendix, as well as to compute the proximal operators for different cone projections in (5.4.26). However, in practice, we need to address the following issues to provide a user-friendly framework and to assist practical implementation:

- Although the parse/solver frameworks like CVX can automatically transform an original convex problem into the standard form *numerically* based on the graph implementation, extending such an idea to the *automatic and symbolic* transformation, thereby enabling matrix stuffing, is desirable but challenging in terms of reliability and correctness verification.

- Efficient projection algorithms are highly desirable. For the subspace projection, as discussed in Section 5.5.2, parallel factorization and symbolic factorization are especially suitable for the cloud computing environments as in Cloud-RAN [10, 29]. For the cone projection, although the projection on the second-order cone is very efficient, as shown in (5.4.26), projecting on the semidefinite cone (which is required to solve the semidefinite programming problems) is computationally expensive, as it requires to perform eigenvalue decomposition. The structure of the cone projection should be exploited to make speedups.

- It is interesting to apply the proposed framework to various non-convex optimization problems. For instance, the well-known majorization-minimization optimization provides a principled way to solve the general non-convex problems, whereas a sequence of convex subproblems need to be solved at each iteration. Enabling scalable computation at each iteration will hopefully lead to scalability of the overall algorithm.

Chapter 6

Summary and Future Directions

6.1 Summary

The central theme of the thesis has been to exploit the problem structures (e.g., group sparsity and low-rankness) based on the convex optimization and Riemannian optimization to address the networking issues (i.e., network power consumption, massive CSI acquisition and exploration) and computing issues in dense Cloud-RANs. We end this thesis by summarizing our main findings and discussing several future research directions.

6.1.1 Group Sparse Optimization for Network Adaptation

Network power minimization in green dense Cloud-RAN is a difficult non-convex mixed combinatorial optimization problem. Group sparse optimization offers a principled way to investigate the green dense Cloud-RAN design problems by enabling network adaptation. The underlying idea is that the network entity selection (e.g., RRHs and fronthaul links) can be achieved via controlling the group-sparsity structure of the aggregative beamforming vector. This key observation enables efficient group sparse beamforming algorithm design. Specifically, the mixed ℓ_1/ℓ_2-norm was proposed to convexify the combinatorial composite function of network power consumption. A novel quadratic variational formulation of the weighted mixed ℓ_1/ℓ_2 -norm was further developed to induce the group-sparsity structure for the robust multicast beamformer, thereby guiding the RRH selection. Extensive numerical results have demonstrated the computational efficiency and near-optimal performance of the group sparse beamforming algorithms.

It is critical but also challenging to establish the optimality of the group sparse beamforming algorithms. Finding more applications of the group sparse optimization techniques are very interesting, e.g., computation offloading in mobile edge computing and networking [11].

6.1.2 Massive CSI Acquisition in Dense Cloud-RAN

We proposed a unified framework consisting of a novel compressive CSI acquisition method and stochastic coordinated beamforming (SCB) with mixed CSI for dense Cloud-RAN, thereby significantly reducing the CSI overhead. In particular, the The proposed SCB framework provides modeling flexibility in the channel knowledge uncertainty, while the stochastic DC programming algorithm guarantees to find a stationary point to the resulting chance constrained programming problem. The key finding is that the challenge of CSI overhead reduction may be overcome by exploiting the sparsity (partial connectivity) in wireless channels.

This initial investigation demonstrated that modeling the wireless channels, CSI acquisition and utilization are critical to address the massive CSI challenges in dense Cloud-RAN. In particular, the underlying channel structures assumptions (e.g., sparsity) need to be verified by real-world measurements. The CSI exploitation methods need to be algorithmic efficient. Our recent results on the low-rank matrix completion for interference management based on the network topology information provides a promising example [25].

6.1.3 Large-Scale Convex Optimization for Dense Cloud-RAN

Large-scale convex optimization is essential for scalable and flexible networking in dense Cloud-RAN. We developed a unified two-stage framework to solve general large-scale convex optimization problems. This is achieved by first transforming the original convex problem to the standard form via matrix stuffing and then using the ADMM algorithm to solve the homogeneous self-dual embedding of the primal-dual pair of the transformed standard cone program. The key findings suggest that convex optimization can not only handle highly intractable design problems in dense Cloud-RAN but also can scale very well to large problem sizes. The proposed framework will offer a promising way to solve general convex problems with wide applications in wireless networking, operation research, finance and machine learning.

To further improve the computational efficiency of the two-stage approach framework, serval issues need to be further addressed including efficient semidefinite cone projection algorithm for SDP problems as well as communication-efficient parallel and distributed implementations.

6.2 Future Directions

6.2.1 High-Dimensional Channel Estimation

CSI plays a pivotal role for effective interference management and resource allocation in dense Cloud-RAN. With the dense deployment of RRHs, CSI acquisition becomes a formidable task. In particular, due to the limited radio resources for CSI training, the training pilot length is typically smaller than the dimension of the channel. Conventional methods, such as the least square estimate, become inapplicable in such settings, and novel CSI acquisition methodologies are needed. A unique property of Cloud-RAN with geographically distributed RRHs is the sparsity of the large-scale fading coefficients due to pathloss and shadowing. That is, the channel links between the RRHs and MUs that are far away will have negligible channel gains and contribute little to system performance. A practical way to reduce the CSI acquisition overhead in terms of training and feedback is to only obtain a subset of the strongest channel links. This is called compressive CSI acquisition [72], in which only a subset of "relevant" channel links will be obtained. Therefore, it is expected that exploiting the "low-complexity" structures of channels can reduce the training overhead, thereby overcoming the curse of dimensionality of channel estimation in dense Cloud-RAN.

Furthermore, in terms of CSI exploitation, the proposed stochastic coordinated beamforming framework has demonstrated its modeling flexibility to deal with CSI uncertainty. But the knowledge of channel distribution is still required, which may not be applicable in some scenarios. Based on the fact of the partial connectivity in dense wireless networks due to pathloss and shadowing, a practical way to manage the interference with the minimum channel acquisition overhead is only based on the network topology information without CSI at the transmitters. This forms the basic CSI assumption for the recent studies of topological interference management [24]. It is thus interesting to develop efficient algorithms in dense Cloud-RAN to exploit the network topology knowledge, e.g., the low-rank matrix completion approach [25].

6.2.2 Fronthaul Compression

Most of the algorithms developed in this thesis assume that the Cloud-RAN is deployed in the scenarios with abundant fronthaul resources, i.e., the capacity of the fronthaul links is high enough. However, it is critical to consider the limited fronthaul capacity constrained scenarios [14]. It is obvious that this will complicate the design problems, e.g., in the downlink case, the limited fronthaul capacity will yield non-convex DC constraints [45]. The developed stochastic DC programming algorithm in Chapter 4 will have the potential to accommodate the DC constraints in the limited fronthaul capacity case. However, to scale well to large problem sizes, it is not trivial to extend the large-scale parallel convex optimization framework in Chapter 5 to the non-convex optimization problems. A interesting future research direction will be developing large-scale parallel algorithms for general non-convex optimization problems.

Appendix A

Proofs and Preliminaries in Chapter 2

A.1 Proof of Proposition 1

We begin by deriving the tightest positively homogeneous lower bound of $p(\mathbf{w})$, which is given by [54, 149]

$$p_h(\mathbf{w}) = \inf_{\lambda>0} \frac{p(\lambda \mathbf{w})}{\lambda} = \inf_{\lambda>0} \lambda T(\mathbf{w}) + \frac{1}{\lambda} F(\mathcal{T}(\mathbf{w})). \tag{A.1.1}$$

Setting the gradient of the objective function to zero, the minimum is obtained at $\lambda = \sqrt{F(\mathcal{T}(\mathbf{w}))/T(\mathbf{w})}$. Thus, the positively homogeneous lower bound of the objective function becomes

$$p_h(\mathbf{w}) = 2\sqrt{T(\mathbf{w})F(\mathcal{T}(\mathbf{w}))}, \tag{A.1.2}$$

which combines two terms multiplicatively.

Define diagonal matrices $\mathbf{U} \in \mathbb{R}^{N\times N}$, $\mathbf{V} \in \mathbb{R}^{N\times N}$ with $N = K\sum_{l=1}^{L} N_l$, for which the l-th block elements are $\eta_l \mathbf{I}_{KN_l}$ and $\frac{1}{\eta_l}\mathbf{I}_{KN_l}$, respectively. Next, we calculate the convex envelope of $p_h(\mathbf{w})$ via computing its conjugate:

$$\begin{aligned} p_h^*(\mathbf{y}) &= \sup_{\mathbf{w}\in\mathbb{C}^N} \left(\mathbf{y}^T\mathbf{U}^T\mathbf{V}\mathbf{w} - 2\sqrt{T(\mathbf{w})F(\mathcal{T}(\mathbf{w}))}\right), \\ &= \sup_{\mathcal{I}\subseteq\mathcal{V}} \sup_{\mathbf{w}_\mathcal{I}\in\mathbb{C}^{|\mathcal{I}|}} \left(\mathbf{y}_\mathcal{I}^T \mathbf{U}_{\mathcal{I}\mathcal{I}}^T \mathbf{V}_{\mathcal{I}\mathcal{I}} \mathbf{w}_\mathcal{I} - 2\sqrt{T(\mathbf{w}_\mathcal{I})F(\mathcal{I})}\right) \\ &= \begin{cases} 0 & \text{if } \Omega^*(\mathbf{y}) \leq 1 \\ \infty, \text{otherwise.} \end{cases} \end{aligned} \tag{A.1.3}$$

where $\mathbf{y}_{\mathcal{I}}$ is the $|\mathcal{I}|$-dimensional vector formed with the entries of $\mathbf{y}$ indexed by $\mathcal{I}$ (similarly for $\mathbf{w}$), and $\mathbf{U}_{\mathcal{I}\mathcal{I}}$ is the $|\mathcal{I}| \times |\mathcal{I}|$ matrix formed with the rows and columns of $\mathbf{U}$ indexed by $\mathcal{I}$ (similarly for $\mathbf{V}$), and $\Omega^*(\mathbf{y})$ defines a dual norm of $\Omega(\mathbf{w})$:

$$\Omega^*(\mathbf{y}) = \sup_{\mathcal{I}\subseteq\mathcal{V},\mathcal{I}\neq\emptyset} \frac{\|\mathbf{y}_{\mathcal{I}}\mathbf{U}_{\mathcal{I}}\|_{\ell_2}}{2\sqrt{F(\mathcal{I})}} = \frac{1}{2}\max_{l=1,\ldots,L}\sqrt{\frac{\eta_l}{P_l^c}}\|\mathbf{y}_{\mathcal{G}_l}\|_{\ell_2}. \tag{A.1.4}$$

The first equality in (A.1.4) can be obtained by the Cauchy-Schwarz inequality:

$$\mathbf{y}_{\mathcal{I}}^T\mathbf{U}_{\mathcal{I}\mathcal{I}}^T\mathbf{V}_{\mathcal{I}\mathcal{I}}\mathbf{w}_{\mathcal{I}} \leq \|\mathbf{y}_{\mathcal{I}}\mathbf{U}_{\mathcal{I}}\|_{\ell_2}\cdot\|\mathbf{V}_{\mathcal{I}\mathcal{I}}\mathbf{w}_{\mathcal{I}}\|_{\ell_2} = \|\mathbf{y}_{\mathcal{I}}\mathbf{U}_{\mathcal{I}}\|_{\ell_2}\cdot\sqrt{T(\mathbf{w}_{\mathcal{I}})}. \tag{A.1.5}$$

The second equality in (A.1.4) can be justified by

$$\begin{aligned}\Omega^*(\mathbf{y}) \geq & \sup_{\mathcal{I}\subseteq\mathcal{V},\mathcal{I}\neq\emptyset}\left(\frac{1}{2\sqrt{F(\mathcal{I})}}\max_{l=1,\ldots,L}\|\mathbf{y}_{\mathcal{I}\cap\mathcal{G}_l}\mathbf{U}_{\mathcal{I}\cap\mathcal{G}_l}\|_{\ell_2}\right)\\ = & \frac{1}{2}\max_{l=1,\ldots,L}\sqrt{\frac{\eta_l}{P_l^c}}\|\mathbf{y}_{\mathcal{G}_l}\|_{\ell_2},\end{aligned} \tag{A.1.6}$$

and

$$\Omega^*(\mathbf{y}) \leq \sup_{\mathcal{I}\subseteq\mathcal{V},\mathcal{I}\neq\emptyset}\left(\frac{\|\mathbf{y}_{\mathcal{I}}\mathbf{U}_{\mathcal{I}}\|_{\ell_2}}{2\min_{l=1,\ldots,L}\sqrt{F(\mathcal{I}\cap\mathcal{G}_l)}}\right) = \frac{1}{2}\max_{l=1,\ldots,L}\sqrt{\frac{\eta_l}{P_l^c}}\|\mathbf{y}_{\mathcal{G}_l}\|_{\ell_2}. \tag{A.1.7}$$

Therefore, the tightest convex positively homogeneous lower bound of the function $p(\mathbf{w})$ is

$$\begin{aligned}\Omega(\mathbf{w}) = & \sup_{\Omega^*(\mathbf{y})\leq 1}\mathbf{w}^T\mathbf{y} \leq \sup_{\Omega^*(\mathbf{y})\leq 1}\sum_{l=1}^{L}\|\mathbf{w}_{\mathcal{G}_l}\|_{\ell_2}\|\mathbf{y}_{\mathcal{G}_l}\|_{\ell_2}\\ \leq & \sup_{\Omega^*(\mathbf{y})\leq 1}\left(\sum_{l=1}^{L}\sqrt{\frac{P_l^c}{\eta_l}}\|\mathbf{w}_{\mathcal{G}_l}\|_{\ell_2}\right)\left(\max_{l=1,\ldots,L}\sqrt{\frac{\eta_l}{P_l^c}}\|\mathbf{y}_{\mathcal{G}_l}\|_{\ell_2}\right)\\ = & 2\sum_{l=1}^{L}\sqrt{\frac{P_l^c}{\eta_l}}\|\mathbf{w}_{\mathcal{G}_l}\|_{\ell_2}.\end{aligned} \tag{A.1.8}$$

This upper bound actually holds with equality. Specifically, we let $\bar{\mathbf{y}}_{\mathcal{G}_l} = 2\sqrt{\frac{P_l^c}{\eta_l}}\frac{\mathbf{w}_{\mathcal{G}_l}^{\dagger}}{\|\mathbf{w}_{\mathcal{G}_l}^{\dagger}\|_{\ell_2}}$, such that $\Omega^*(\bar{\mathbf{y}}) = 1$. Therefore,

$$\Omega(\mathbf{w}) = \sup_{\Omega^*(\mathbf{y})\leq 1}\mathbf{w}^T\mathbf{y} \geq \sum_{l=1}^{L}\mathbf{w}_{\mathcal{G}_l}^T\bar{\mathbf{y}}_{\mathcal{G}_l} = 2\sum_{l=1}^{L}\sqrt{\frac{P_l^c}{\eta_l}}\|\mathbf{w}_{\mathcal{G}_l}\|_{\ell_2}. \tag{A.1.9}$$

A.2 Preliminaries on Majorization-Minimization Algorithms

The majorization-minimization (MM) algorithm, being a powerful tool to find a local optimum by minimizing a surrogate function that majorizes the objective function iteratively, has been widely used in statistics, machine learning, etc., [56]. We introduce the basic idea of MM algorithms, which allows us to derive our main results.

Consider the problem of minimizing $f(\mathbf{x})$ over $\mathcal{F}$. The idea of MM algorithms is as follows. First, we construct a majorization function $g(\mathbf{x}|\mathbf{x}^{[m]})$ for $f(\mathbf{x})$ such that

$$g(\mathbf{x}|\mathbf{x}^{[m]}) \geq f(\mathbf{x}), \forall\, \mathbf{x} \in \mathcal{F}, \tag{A.2.1}$$

and the equality is attained when $\mathbf{x} = \mathbf{x}^{[m]}$. In an MM algorithm, we will minimize the majorization function $g(\mathbf{x}|\mathbf{x}^{[m]})$ instead of the original function $f(\mathbf{x})$. Let $\mathbf{x}^{[m+1]}$ denote the minimizer of the function $g(\mathbf{x}|\mathbf{x}^{[m]})$ over $\mathcal{F}$ at the m-th iteration, i.e.,

$$\mathbf{x}^{[m+1]} = \arg\min_{\mathbf{x}\in\mathcal{F}}\ g(\mathbf{x}|\mathbf{x}^{[m]}), \tag{A.2.2}$$

then we can see that this iterative procedure will decrease the value of $f(\mathbf{x})$ monotonically after each iteration, i.e.,

$$f(\mathbf{x}^{[m+1]}) \leq g(\mathbf{x}^{[m+1]}|\mathbf{x}^{[m]}) \leq g(\mathbf{x}^{[m]}|\mathbf{x}^{[m]}) = f(\mathbf{x}^{[m]}), \tag{A.2.3}$$

which is a direct result from the definitions (A.2.1) and (A.2.2). The decreasing property makes an MM algorithm numerically stable. More details can be found in a tutorial on MM algorithms [56] and references therein.

Appendix B

Proofs in Chapter 3

B.1 Proof of Theorem 1

We first need to prove that

$$\limsup_{\epsilon \searrow 0} \Omega(\epsilon) \subset \Omega(0). \tag{B.1.1}$$

For any $\bar{\boldsymbol{z}} \in \limsup_{\epsilon \searrow 0} \Omega(\epsilon^{[n]})$, there exists $\boldsymbol{z}^{[n]} \in \Omega(\epsilon^{[n]})$ such that $\boldsymbol{z}^{[n]} \to \bar{\boldsymbol{z}}$ and $\epsilon^{[n]} \searrow 0$. To prove (B.1.1), we only need to prove that $\bar{\boldsymbol{z}} \in \Omega(0)$. Specifically, $\boldsymbol{z}^{[n]} \in \Omega(\epsilon^{[n]})$ indicates that

$$0 \in \nabla_{\boldsymbol{z}} f_p(\boldsymbol{z}^{[n]}; \epsilon^{[n]}) + \mathcal{N}_{\mathcal{C}}(\boldsymbol{z}^{[n]}). \tag{B.1.2}$$

As $f_p(\boldsymbol{z}; \epsilon)$ is continuously differentiable in both $\boldsymbol{z}$ and ϵ, we have

$$\begin{aligned}\lim_{n \to \infty} \nabla_{\boldsymbol{z}} f_p(\boldsymbol{z}^{[n]}; \epsilon^{[n]}) &= \lim_{\boldsymbol{z}^{[n]} \to \bar{\boldsymbol{z}}} \lim_{\epsilon^{[n]} \searrow 0} \nabla_{\boldsymbol{z}} f_p(\boldsymbol{z}^{[n]}; \epsilon^{[n]}) \\ &= \nabla_{\boldsymbol{z}} f_p(\bar{\boldsymbol{z}}; 0).\end{aligned} \tag{B.1.3}$$

Furthermore, based on [150, Proposition 6.6], we have

$$\limsup_{\boldsymbol{z}^{[n]} \to \bar{\boldsymbol{z}}} \mathcal{N}_{\mathcal{C}}(\boldsymbol{z}^{[n]}) = \mathcal{N}_{\mathcal{C}}(\bar{\boldsymbol{z}}). \tag{B.1.4}$$

Based on (B.1.3) and (B.1.4) and taking $n \to \infty$ in (B.1.2), we thus prove (B.1.1). Based on [104, Theorem 4], we complete the proof for (C.4.5).

B.2 Convergence of the Iterative Reweighted-ℓ_2 Algorithm

1) We will show that any convergent subsequence $\{\boldsymbol{z}^{[n_k]}\}_{k=1}^{\infty}$ of $\{\boldsymbol{z}^{[n]}\}_{n=1}^{\infty}$ satisfies the definition of the KKT points of problem $\mathscr{P}_{\text{sm}}(\epsilon)$ (3.4.10). Specifically, let $\boldsymbol{z}^{[n_k]} \rightarrow \bar{\boldsymbol{z}}$ be one such convergent subsequence with

$$\lim_{k\rightarrow\infty} \boldsymbol{z}^{[n_k+1]} = \lim_{k\rightarrow\infty} \boldsymbol{z}^{[n_k]} = \bar{\boldsymbol{z}}. \tag{B.2.1}$$

As

$$\boldsymbol{z}^{[n_k+1]} := \arg\min_{\boldsymbol{z}\in\mathcal{C}} \; Q(\boldsymbol{z};\boldsymbol{\omega}^{[n_k]}), \tag{B.2.2}$$

which is a convex optimization problem, the KKT condition holds at $\boldsymbol{z}^{[n_k+1]}$, i.e.,

$$0 \in \nabla_{\boldsymbol{z}} Q(\boldsymbol{z}^{[n_k+1]};\boldsymbol{\omega}^{[n_k]}) + \mathcal{N}_{\mathcal{C}}(\boldsymbol{z}^{[n_k+1]}). \tag{B.2.3}$$

Based on [150, Proposition 6.6] and (B.2.1), we have

$$\limsup_{\boldsymbol{z}^{[n_k+1]}\rightarrow\bar{\boldsymbol{z}}} \mathcal{N}_{\mathcal{C}}(\boldsymbol{z}^{[n_k+1]}) = \mathcal{N}_{\mathcal{C}}(\bar{\boldsymbol{z}}). \tag{B.2.4}$$

Furthermore, based on (B.2.1), we also have

$$\begin{aligned}
\lim_{k\rightarrow\infty} \nabla_{\boldsymbol{z}} Q(\boldsymbol{z}^{[n_k+1]};\boldsymbol{\omega}^{[n_k]}) &= \lim_{k\rightarrow\infty} 2\sum_{i=1}^{m} \omega^{[n_k]} z_i^{n_k+1} \\
&= \lim_{k\rightarrow\infty} \sum_{i=1}^{m} \frac{p z^{[n_k+1]}}{\left[\left(z_i^{[n_k]}\right)^2 + \epsilon^2\right]^{1-\frac{p}{2}}} \\
&= \nabla_{\boldsymbol{z}} f_p(\bar{\boldsymbol{z}};\epsilon).
\end{aligned} \tag{B.2.5}$$

Therefore, taking $k \rightarrow \infty$ in (B.2.3), we have

$$0 \in \nabla_{\boldsymbol{z}} Q(\bar{\boldsymbol{z}};\bar{\boldsymbol{\omega}}) + \mathcal{N}_{\mathcal{C}}(\bar{\boldsymbol{z}}), \tag{B.2.6}$$

which indicates that $\bar{z}$ is a KKT point of problem $\mathscr{P}_{\text{sm}}(\epsilon)$. We thus complete the proof.

2) As $f_p(\boldsymbol{z};\epsilon)$ is continuous and $\mathcal{C}$ is compact, we have the fact that the limit of the sequence $f_p(\boldsymbol{z}^{[n]};\epsilon)$ is finite. Furthermore, we have $f_p(\boldsymbol{z}^{[n+1]};\epsilon) \leq f_p(\boldsymbol{z}^{[n]};\epsilon)$ according to

(3.4.18). Based on the results in 1), we complete the proof. Note that a similar result was presented in [91] by leveraging the results in the EM algorithm theory.

Appendix C

Proofs in Chapter 4

C.1 Proof of Lemma 1

For simplicity, we denote $c_{k,1}(\mathbf{v}) \triangleq c_{k,1}(\mathbf{v}_{-k}, \mathbf{h}_k)$, $c_{k,2}(\mathbf{v}) \triangleq c_{k,2}(\mathbf{v}_k, \mathbf{h}_k)$ and $d_k(\mathbf{v}) \triangleq d_k(\mathbf{v}, \mathbf{h}_k)$. For any $\mathbf{v} \in \mathcal{V}$, $\forall k$, $d_k(\mathbf{v}) = c_{k,1}(\mathbf{v}) - c_{k,2}(\mathbf{v})$ is a DC function on $\mathcal{V}$, as both $c_{k,1}(\mathbf{v})$ and $c_{k,2}(\mathbf{v})$ are convex functions of $\mathbf{v}$. For any $\nu > 0$, we first prove that the following function

$$\psi\left(\max_{1\le k\le K} d_k(\mathbf{v}), \nu\right) = \frac{1}{\nu}\left[\left(\nu + \max_{1\le k\le K} d_k(\mathbf{v})\right)^+ - \left(\max_{1\le k\le K} d_k(\mathbf{v})\right)^+\right], \tag{C.1.1}$$

is also a DC function. The function $d_k(\mathbf{v})$ can be rewritten as

$$d_k(\mathbf{v}) = c_{k,1}(\mathbf{v}) + \sum_{i\neq k} c_{i,2}(\mathbf{v}) - \sum_{i=1}^{K} c_{i,2}(\mathbf{v}). \tag{C.1.2}$$

Therefore, the following function

$$\max_{1\le k\le K} d_k(\mathbf{v}) = \underbrace{\max_{1\le k\le K}\left\{c_{k,1}(\mathbf{v}) + \sum_{i\neq k} c_{i,2}(\mathbf{v})\right\}}_{C_1(\mathbf{v},\mathbf{h})} - \underbrace{\sum_{i=1}^{K} c_{i,2}(\mathbf{v})}_{C_2(\mathbf{v},\mathbf{h})}, \tag{C.1.3}$$

is a DC function, as both the functions $C_1(\mathbf{v}, \mathbf{h})$ and $C_2(\mathbf{v}, \mathbf{h})$ are convex in $\mathbf{v}$. Furthermore, for any $z_1, z_2 \in \mathbb{R}$ and $z = z_1 - z_2$, we have $z^+ = \max\{z_1, z_2\} - z_2$. Therefore,

$$\psi\left(\max_{1\le k\le K} d_k(\mathbf{v}), \nu\right) = \frac{1}{\nu}[m(\mathbf{v}, \nu) - m(\mathbf{v}, 0)], \tag{C.1.4}$$

is a DC function of $\mathbf{v}$, as

$$m(\mathbf{v},\nu)=\max\{\nu+C_1(\mathbf{v},\mathbf{h}),C_2(\mathbf{v},\mathbf{h})\},\tag{C.1.5}$$

is a convex function of $\mathbf{v}$. According to [113, Proposition 2.1], $\hat{f}(\mathbf{v},\nu)=\mathbb{E}[\psi\left(\max_{1\leq k\leq K}d_k(\mathbf{v},\mathbf{h}_k),\nu\right)]$ is a DC function on $\mathcal{V}$. Therefore, the proof is completed.

C.2 Proof of Theorem 3

In order to prove Theorem 3, we need to prove the following equality:

$$\inf_{\nu>0}\hat{f}(\mathbf{v},\nu)=f(\mathbf{v}).\tag{C.2.1}$$

First, we need to prove the monotonicity of the function $\hat{f}(\mathbf{v},\nu)$ in the variable ν. According to (C.1.4) and (C.1.5), the function $\hat{f}(\mathbf{v},\nu)$ can be rewritten as

$$\hat{f}(\mathbf{v},\nu)=\mathbb{E}[\pi(\nu,C_1(\mathbf{v},\mathbf{h}),C_2(\mathbf{v},\mathbf{h}))],\tag{C.2.2}$$

where

$$\pi(\nu,z_1,z_2)\triangleq\frac{1}{\nu}\left[\max\{\nu+z_1,z_2\}-\max\{z_1,z_2\}\right],\tag{C.2.3}$$

for any $z_1,z_2\in\mathbb{R}$ and $\nu>0$. Therefore, we only need to prove the monotonicity of the function $\pi(\nu,z_1,z_2)$ in the variable ν.

Define $z\triangleq z_1-z_2$, then we have

$$\pi(\nu,z_1,z_2)=\left(1+\frac{1}{\nu}z\right)1_{(-\nu,0]}(z)+1_{(0,+\infty)}(z).\tag{C.2.4}$$

For any $\nu_1>\nu_2>0$ and any $z_1,z_2\in\mathbb{R}$, we have

$$\pi(\nu_1,z_1,z_2)-\pi(\nu_2,z_1,z_2)=\left(1+\frac{1}{\nu_1}z\right)1_{(-\nu_1,-\nu_2]}(z)+$$
$$z\left(\frac{1}{\nu_1}-\frac{1}{\nu_2}\right)1_{(-\nu_2,0)}(z)\geq 0.\tag{C.2.5}$$

Therefore, $\hat{f}(\mathbf{v}, \nu)$ is nondecreasing in ν for $\nu > 0$. Hence, we have

$$\inf_{\nu>0} \hat{f}(\mathbf{v}, \nu) = \lim_{\nu \searrow 0} \hat{f}(\mathbf{v}, \nu) = \lim_{\nu \searrow 0} \frac{1}{\nu}[u(\mathbf{v}, \nu) - u(\mathbf{v}, 0)], \tag{C.2.6}$$

where $\nu \searrow 0$ indicates that ν decreasingly goes to 0. Thus, based on (C.2.6), in order to prove (C.2.1), we only need to prove

$$\lim_{\nu \searrow 0} \frac{1}{\nu}[u(\mathbf{v}, \nu) - u(\mathbf{v}, 0)] = f(\mathbf{v}). \tag{C.2.7}$$

Furthermore, if the partial derivation of $u(\mathbf{v}, \nu)$ exists, we have

$$\lim_{\nu \searrow 0} \frac{1}{\nu}[u(\mathbf{v}, \nu) - u(\mathbf{v}, 0)] = \frac{\partial}{\partial \nu} u(\mathbf{v}, 0). \tag{C.2.8}$$

Therefore, we need to prove that $\frac{\partial}{\partial \nu} u(\mathbf{v}, \nu)$ exists and $\frac{\partial}{\partial \nu} u(\mathbf{v}, 0) = f(\mathbf{v})$.

According to (C.1.5), we have $u(\mathbf{v}, \nu) = \mathbb{E}[m(\mathbf{v}, \mathbf{h}, \nu)] = \mathbb{E}[\max\{\nu + C_1(\mathbf{v}, \mathbf{h}), C_2(\mathbf{v}, \mathbf{h})\}]$. As

$$\frac{\partial}{\partial \nu} \left(\max\{\nu + z_1, z_2\}\right) = 1_{(-\nu, +\infty)}(z), \tag{C.2.9}$$

for any $z \neq -\nu$, and $\Pr\left\{\max_{1 \le k \le K} d_k(\mathbf{v}, \mathbf{h}) = -\nu\right\} = 0$, where $\max_{1 \le k \le K} d_k(\mathbf{v}, \mathbf{h}) \triangleq C_1(\mathbf{v}, \mathbf{h}) - C_2(\mathbf{v}, \mathbf{h})$ (C.1.3), we conclude that $\frac{\partial}{\partial \nu} u(\mathbf{v}, \nu)$ exists.

Let $\mathcal{T} \triangleq (-T, T)$ with $T > 0$ being an open set such that the cumulative distribution function $F(\mathbf{v}, \nu) \triangleq \Pr\{\max_{1 \le k \le K} d_k(\mathbf{v}) \le \nu)\}$ of the random variable $(\max_{1 \le k \le K} d_k(\mathbf{v}))$ is continuously differentiable for any $\nu \in \mathcal{T}$. Next we will show that

$$\begin{aligned} \frac{\partial}{\partial \nu} u(\mathbf{v}, \nu) &= \lim_{\delta \to 0} \frac{1}{\delta} \mathbb{E}[m(\mathbf{v}, \mathbf{h}, \nu + \delta) - m(\mathbf{v}, \mathbf{h}, \nu)] \\ &= \Pr\left\{\max_{1 \le k \le K} d_k(\mathbf{v}) > -\nu\right\} = 1 - F(\mathbf{v}, -\nu). \end{aligned} \tag{C.2.10}$$

For any $\nu \in \mathcal{T}$ and $\mathbf{v} \in \mathcal{V}$, define the random variable $X(\delta) \triangleq [m(\mathbf{v}, \mathbf{h}, \nu + \delta) - m(\mathbf{v}, \mathbf{h}, \nu)]/\delta$, then we have the following two facts:

1. The limit of $X(\delta)$ exists and we have

$$\lim_{\delta \to 0} X(\delta) = 1_{(-\nu, +\infty)}\left(\max_{1 \le k \le K} d_k(\mathbf{v}, \mathbf{h})\right), \tag{C.2.11}$$

with probability one.

2. $X(\delta)$ is dominated by a constant $C > 0$, i.e., $|X(\delta)| \leq C$, where $0 < C < \infty$. This can be justified by

$$\begin{aligned} |X(\delta)| &= \frac{1}{h}|m(\mathbf{v}, \mathbf{h}, \nu + \delta) - m(\mathbf{v}, \mathbf{h}, \nu)| \\ &= \frac{1}{\delta}|[\delta + Q(\mathbf{v}, \mathbf{h}, \nu)]^+ - [Q(\mathbf{v}, \mathbf{h}, \nu)]^+| \leq 1, \end{aligned}$$

where $Q(\mathbf{v}, \mathbf{h}, \nu) \triangleq \nu + \max_{1 \leq k \leq K} d_k(\mathbf{v}, \mathbf{h})$ and the last inequality is based on the fact $|[x]^+ - [y]^+| \leq |x - y|$.

From the above two facts on the random variable $X(\delta)$, by the dominated convergence theorem to interchange an expectation and the limit as $\delta \to 0$, and together with [151, Proposition 1], we have

$$\begin{aligned} \frac{\partial}{\partial \nu} u(\mathbf{v}, \nu) &= \lim_{\delta \to 0} \mathbb{E}[X(\delta)] = \mathbb{E}[\lim_{\delta \to 0} X(\delta)] \\ &= \mathbb{E}[1_{(-\nu, +\infty)}(\max_{1 \leq k \leq K} d_k(\mathbf{v}, \mathbf{h}))] = 1 - F(\mathbf{v}, -\nu). \end{aligned} \tag{C.2.12}$$

Therefore, we complete the proof by

$$\begin{aligned} \inf_{\nu > 0} \hat{f}(\mathbf{v}, \nu) &= \lim_{\nu \searrow 0} \frac{1}{\nu}[u(\mathbf{v}, \nu) - u(\mathbf{v}, 0)] \\ &= \frac{\partial}{\partial \nu} u(\mathbf{v}, 0) = 1 - F(\mathbf{v}, 0) = f(\mathbf{v}). \end{aligned} \tag{C.2.13}$$

C.3 Proof of Lemma 2

It is well known that non-constant real-valued functions of complex variables are not holomorphic (or $\mathbb{C}$-differentiable) [152]. Thus, the real-valued functions $d_k(\mathbf{v}, \mathbf{h}_k)$ in (4.3.1) are not differentiable in the complex domain $\mathbb{C}^{NK}$ (i.e., with respect to the complex vector $\mathbf{v}$). Define a real-valued function $m(\mathbf{v}, \mathbf{h}, \nu) \triangleq \max_{1 \leq k \leq K+1} s_k(\mathbf{v}, \mathbf{h}, \nu)$, which is convex in $\mathbf{v}$. Although this function is not holomorphic in $\mathbf{v}$, it can be viewed as a function of both $\mathbf{v}$ and its complex conjugate $\mathbf{v}^*$, i.e., $m(\mathbf{v}, \mathbf{v}^*, \mathbf{h}, \nu)$. It is easy to verify that the function $m(\mathbf{v}, \mathbf{v}^*, \mathbf{h}, \nu)$ is holomorphic in $\mathbf{v}$ for a fixed $\mathbf{v}^*$ and is also holomorphic in $\mathbf{v}^*$ for a fixed $\mathbf{v}$. Proving Lemma 2 is equivalent to proving that the gradient of $\mathbb{E}[m(\mathbf{v}, \mathbf{h}, \nu)]$ with respect to $\mathbf{v}^*$ exists and equals

$$\nabla_{\mathbf{v}^*}\mathbb{E}[m(\mathbf{v},\mathbf{h},\nu)] = \mathbb{E}[\nabla_{\mathbf{v}^*}m(\mathbf{v},\mathbf{h},\nu)]. \tag{C.3.1}$$

Based on the chain rule [152], the complex gradient of the function $m(\mathbf{v},\mathbf{h},\nu)$ with respect to $\mathbf{v}^*$ exists and is given by

$$\nabla_{\mathbf{v}^*}m(\mathbf{v},\mathbf{h},\nu) \triangleq \frac{\partial m(\mathbf{v},\mathbf{h},\nu)}{\partial \mathbf{v}^*} = \frac{\partial s_{k^\star}(\mathbf{v},\mathbf{h},\nu)}{\partial \mathbf{v}^*}, \tag{C.3.2}$$

with probability one, where $k^\star = \arg \max\limits_{1\le k\le K+1} s_k(\mathbf{v},\mathbf{h},\nu)$. It is a vector operator and gives the direction of the steepest ascent of a real scalar-valued function.

Denote $\frac{\partial m(\mathbf{v},\mathbf{h},\nu)}{\partial \mathbf{v}^*} \triangleq [\frac{\partial m}{\partial v_i^*}]_{1\le i\le NK}$ and $\frac{\partial s_{k^\star}(\mathbf{v},\mathbf{h}_k,\nu)}{\partial \mathbf{v}^*} \triangleq [\frac{\partial s_{k^\star}}{\partial v_i^*}]_{1\le i\le NK}$, where $\mathbf{v} = [v_1, v_2, \ldots, v_{NK}]$, and define the following complex random variable

$$Y(\Delta v_i^*) \triangleq \frac{1}{\Delta v_i^*}[m(\mathbf{v}_{-i}, v_i^* + \Delta v_i^*) - m(\mathbf{v}_{-i}, v_i^*)], \tag{C.3.3}$$

where $\mathbf{v}_{-i} \triangleq [v_k]_{k\neq i}$, $\Delta v_i^* \in \mathbb{C}$ and $m(\mathbf{v}) \triangleq m(\mathbf{v},\mathbf{h},\nu)$ for simplicity, then we have the following two facts on the random variable $Y(\Delta v_i^*)$:

1. The limit of $Y(\Delta v_i^*)$ exists and equals

$$\lim_{\Delta v_i^* \to 0} Y(\Delta v_i^*) = \frac{\partial s_{k^\star}}{\partial v_i^*}, \tag{C.3.4}$$

with probability one.

2. The random variable is dominated by a random variable Z with $\mathbb{E}[Z] \le +\infty$, i.e.,

$$|Y(\Delta v_i^*)| \le Z, \forall i, \tag{C.3.5}$$

which can be verified by the following lemma.

Lemma 3. *For any $\mathbf{x}, \mathbf{y} \in \mathcal{V}$, there exists a random variable Z with $\mathbb{E}[Z] \le \infty$ such that*

$$|m(\mathbf{x},\mathbf{h},\nu) - m(\mathbf{y},\mathbf{h},\nu)| \le Z\|\mathbf{x}-\mathbf{y}\|. \tag{C.3.6}$$

Proof. As $m(\mathbf{v})$ is convex in $\mathbf{v}$, we have

$$m(\mathbf{x}) \geq m(\mathbf{y}) + 2\langle \nabla_{\mathbf{v}^*} m(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle, \tag{C.3.7}$$

$$m(\mathbf{y}) \geq m(\mathbf{x}) + 2\langle \nabla_{\mathbf{v}^*} m(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle. \tag{C.3.8}$$

Based on the above two inequalities and by the Cauchy-Schwarz inequality, we have

$$|m(\mathbf{x}) - m(\mathbf{y})| \leq 2 \left(\max_{\mathbf{v}=\mathbf{x},\mathbf{y}} \|\nabla_{\mathbf{v}^*} m(\mathbf{v})\| \right) \|\mathbf{x} - \mathbf{y}\|. \tag{C.3.9}$$

Furthermore, for $1 \leq k \leq K$, we have

$$\begin{aligned} \|\nabla_{\mathbf{v}^*} s_k(\mathbf{v})\| &= \left(\sum_{i \neq k} \left\| \left(\mathbf{h}_k \mathbf{h}_k^{\mathsf{H}} + \frac{1}{\gamma_k^2} \mathbf{h}_i \mathbf{h}_i^{\mathsf{H}} \right) \mathbf{v}_i \right\|^2 \right)^{1/2} \\ &\leq \max_{i \neq k} \|\mathbf{v}_i\| \left(\sum_{i \neq k} \left\| \left(\mathbf{h}_k \mathbf{h}_k^{\mathsf{H}} + \frac{1}{\gamma_k^2} \mathbf{h}_i \mathbf{h}_i^{\mathsf{H}} \right) \right\|^2 \right)^{1/2} \\ &= Z_1, \end{aligned} \tag{C.3.10}$$

where Z_1 is a random variable with $\mathbb{E}[Z_1] \leq +\infty$, and for $k = K + 1$, we have

$$\begin{aligned} \|\nabla_{\mathbf{v}^*} s_{K+1}(\mathbf{v})\| &= \left(\sum_{i=1}^{K} \left\| \frac{1}{\gamma_i^2} \mathbf{h}_i \mathbf{h}_i^{\mathsf{H}} \mathbf{v}_i \right\|^2 \right)^{1/2} \\ &\leq \max_{1 \leq i \leq K} \|\mathbf{v}_i\| \left(\sum_{i=1}^{K} \left\| \frac{1}{\gamma_i^2} \mathbf{h}_i \mathbf{h}_i^{\mathsf{H}} \right\|^2 \right)^{1/2} \\ &= Z_2, \end{aligned} \tag{C.3.11}$$

where Z_2 is a random variable with $\mathbb{E}[Z_2] \leq +\infty$. Therefore, letting $Z \triangleq \max\{Z_1, Z_2\}$ with $\mathbb{E}[Z] < +\infty$, we have

$$\nabla_{\mathbf{v}^*} m(\mathbf{v}) = \frac{\partial s_{k^\star}(\mathbf{v})}{\partial \mathbf{v}^*} \leq \max\{Z_1, Z_2\} = Z. \tag{C.3.12}$$

According to (C.3.9) and (C.3.12), we have the inequality (C.3.6). □

Based on the above two facts (C.3.4) and (C.3.5) on the random variable $Y(\Delta v_i^*)$, and by the dominated convergence theorem to interchange an expectation and the limit as $\Delta v_i^* \to 0$

and [151, Proposition 1], we have

$$\lim_{\Delta v_i^* \to 0} \mathbb{E}[Y(\Delta v_i^*)] = \mathbb{E}\left[\lim_{\Delta v_i^* \to 0} Y(\Delta v_i^*)\right] = \mathbb{E}\left[\frac{\partial s_{k^\star}}{\partial v_i^*}\right]. \tag{C.3.13}$$

Based on the fact

$$\nabla_{\mathbf{v}^*} \mathbb{E}[m(\mathbf{v}, \mathbf{h}, \nu)] = \left[\lim_{\Delta v_i^* \to 0} \mathbb{E}[Y(\Delta v_i^*)]\right]_{1 \le i \le NK}, \tag{C.3.14}$$

we get (C.3.1) and thus complete the proof.

C.4 Proof of Theorem 5

For simplicity, we only consider the case with real variables and functions. The extension to complex variables is straightforward. Specifically, define $\mathcal{D}_0 = \{\mathbf{v} \in \mathcal{V} : f(\mathbf{v}) \le \epsilon\}$ as the feasible set of the SCB problem $\mathscr{P}_{\text{SCB}}$. To ensure the existence of the KKT paris for the SCB problem $\mathscr{P}_{\text{SCB}}$, we assume the following constraint qualification [150, Corollary 6.15] for program $\mathscr{P}_{\text{SCB}}$, i.e., for any feasible point $\mathbf{v} \in \mathcal{D}_0$, $\lambda = 0$ is the only value that satisfies the following linear system:

$$-\lambda \nabla_{\mathbf{v}} f(\mathbf{v}) \in \mathcal{N}_{\mathcal{V}}(\mathbf{v}), \lambda[f(\mathbf{v}) - \epsilon] = 0 \tag{C.4.1}$$

where $\lambda \ge 0$, and $\mathcal{N}_{\mathcal{V}}(\mathbf{v})$ is the normal cone to the convex set $\mathcal{V}$ at $\mathbf{v}$, i.e.,

$$\mathcal{N}_{\mathcal{V}}(\mathbf{v}) = \{\mathbf{x} | \langle \mathbf{x}, \mathbf{y} - \mathbf{v} \rangle \le 0, \forall \mathbf{y} \in \mathcal{V}\}. \tag{C.4.2}$$

With this constraint qualification, we have the KKT pairs $(\mathbf{v}^\star, \lambda^\star)$ [150, Corollary 6.15] for the SCB problem as

$$\Omega_0 : \begin{cases} -[\nabla_{\mathbf{v}} f_0(\mathbf{v}^\star) + \lambda^\star \nabla_{\mathbf{v}} f(\mathbf{v}^\star)] \in \mathcal{N}_{\mathcal{V}}(\mathbf{v}^\star) \\ \lambda^\star [f(\mathbf{v}^\star) - \epsilon] = 0 \\ \lambda^\star \ge 0, \mathbf{v}^\star \in \mathcal{V}, \end{cases} \tag{C.4.3}$$

where $f_0(\mathbf{v}) = \|\mathbf{v}\|^2$ is the objective function of $\mathscr{P}_{\text{SCB}}$.

Similarly, let $(\mathbf{v}^\star, \kappa^\star, \lambda^\star)$ be a KKT pair of the joint approximation program $\tilde{\mathscr{P}}_{\text{DC}}$ as follows

$$\Omega : \begin{cases} -\left\{\nabla_{\mathbf{v}} f_0(\mathbf{v}^\star) + \lambda^\star \nabla_{\mathbf{v}} [u(\mathbf{v}^\star, \kappa^\star) - \kappa^\star \epsilon - u(\mathbf{v}^\star, 0)]\right\} \\ \in \mathcal{N}_{\mathcal{V}}(\mathbf{v}^\star), \\ -\left\{\lambda^\star \nabla_\kappa [u(\mathbf{v}^\star, \kappa^\star) - \kappa^\star \epsilon - u(\mathbf{v}^\star, 0)]\right\} \in \mathcal{N}_{(0,+\infty)}(\kappa^\star), \\ \lambda^\star [u(\mathbf{v}^\star, \kappa^\star) - \kappa^\star \epsilon - u(\mathbf{v}^\star, 0)] = 0 \\ \lambda^\star \geq 0, \mathbf{v}^\star \in \mathcal{V}, \kappa > 0. \end{cases}$$

In order to prove Theorem 5, we first prove the following lemma illustrating the relationship between Ω_0 and Ω.

Lemma 4. *Suppose that there exists* $(\mathbf{v}^{[j]}, \kappa^{[j]}, \lambda^{[j]}) \in \Omega$*, such that* $(\mathbf{v}^{[j]}, \kappa^{[j]}, \lambda^{[j]}) \to (\hat{\mathbf{v}}, 0, \hat{\lambda})$*, then we have that* $(\hat{\mathbf{v}}, \hat{\lambda}) \in \Omega_0$*.*

Proof. We only need to consider two cases in terms of $\lambda^{[j]}$ being zeros or not.

Case one: suppose there exists a subsequence $\{\lambda^{[k_i]}\}$ of $\{\lambda^{[j]}\}$ such that $\lambda^{[k_i]} = 0, i = 0, 1, 2, \ldots$. As $\lambda^{[k_i]}$'s belong to Ω, we have $-\nabla_{\mathbf{v}} f_0(\mathbf{v}^{[k_i]}) \in \mathcal{N}_{\mathcal{V}}(\mathbf{v}^{[k_i]})$, which implies that $-\nabla_{\mathbf{v}} f_0(\hat{\mathbf{v}}) \in \mathcal{N}_{\mathcal{V}}(\hat{\mathbf{v}})$, as $i \to \infty$. This indicates $(\hat{\mathbf{v}}, 0) \in \Omega_0$.

Case two: suppose that $\lambda^{[n]} \neq 0$, for sufficiently large n. In this case, we have $\nabla_\kappa [u(\mathbf{v}^{[n]}, \kappa^{[n]}) - \kappa^{[n]} \epsilon] = 0$, as $\kappa^{[n]} > 0$ and $\mathcal{N}_{(0,+\infty)}(\kappa^{[n]}) = 0$. Based on (C.2.13), let $n \to \infty$ such that $\kappa^{[n]} \to 0$, we have

$$f(\hat{\mathbf{v}}) - \epsilon = 0. \tag{C.4.4}$$

Furthermore, as $\kappa^{[n]} \neq 0$, based on the KKT pairs in Ω, we have

$$-\nabla_{\mathbf{v}} f_0(\mathbf{v}^{[n]}) - \lambda^{[n]} \kappa^{[n]} \left\{ \frac{\nabla_{\mathbf{v}} [u(\mathbf{v}^{[n]}, \kappa^{[n]}) - u(\mathbf{v}^{[n]}, 0)]}{\kappa^{[n]}} \right\} \in \mathcal{N}_{\mathcal{V}}(\mathbf{v}^{[n]}), \tag{C.4.5}$$

and

$$\lambda^{[n]} \kappa^{[n]} \left\{ \nabla_\kappa [u(\mathbf{v}^{[n]}, \kappa^{[n]}) - \kappa^{[n]} \epsilon] \right\} = 0. \tag{C.4.6}$$

According to (C.2.12), we have

$$\frac{\partial}{\partial \kappa} \nabla_{\mathbf{v}} u(\mathbf{v}^{[n]}, \kappa^{[n]}) = \nabla_{\mathbf{v}} \left(\frac{\partial}{\partial \kappa} u(\mathbf{v}^{[n]}, \kappa^{[n]}) \right) = -\nabla_{\mathbf{v}} F(\mathbf{v}^{[k]}, -\kappa^{[n]}). \tag{C.4.7}$$

Therefore, we have

$$\lim_{n\to+\infty}\frac{\nabla_{\mathbf{v}}u(\mathbf{v}^{[n]},\kappa^{[n]})-\nabla_{\mathbf{v}}u(\mathbf{v}^{[k]},0)}{\kappa^{[n]}}=-\lim_{n\to+\infty}\nabla_{\mathbf{v}}F(\mathbf{v}^{[n]},-\bar{\kappa}^{[n]})$$
$$=-\nabla_{\mathbf{v}}F(\hat{\mathbf{v}},0)=\nabla_{\mathbf{v}}f(\hat{\mathbf{v}}),\qquad \text{(C.4.8)}$$

where $\bar{\kappa}^{[n]}\in(0,\kappa^{[n]}),\forall n$, due to the mean-value theorem.

Dividing both sides of equations (C.4.5) and (C.4.6) by $\lambda^{[n]}\kappa^{[n]}$, respectively, let $n\to\infty$ and suppose that $\lambda^{[n]}\kappa^{[n]}\to+\infty$, based on (C.4.4) and (C.4.8), we have

$$-\nabla_{\mathbf{v}}f(\hat{\mathbf{v}})\in\mathcal{N}_{\mathcal{V}}(\hat{\mathbf{v}}),f(\hat{\mathbf{v}})-\epsilon=0.\qquad \text{(C.4.9)}$$

However, this contradicts the constraint qualification (C.4.1). Therefore, we conclude that $\lambda^{[n]}\kappa^{[n]}\nrightarrow+\infty$. We thus assume that $\lambda^{[n]}\kappa^{[n]}\to\hat{\lambda}$ with $0\le\hat{\lambda}<+\infty$. Let $n\to\infty$, based on (C.4.4), (C.4.5), (C.4.6) and (C.4.8), we obtain

$$-\left\{\nabla_{\mathbf{v}}f_0(\hat{\mathbf{v}})+\hat{\lambda}\nabla_{\mathbf{v}}f(\hat{\mathbf{v}})\right\}\in\mathcal{N}_{\mathcal{V}}(\hat{\mathbf{v}}),\hat{\lambda}[f(\hat{\mathbf{v}})-\epsilon]=0.\qquad \text{(C.4.10)}$$

This indicates that $(\hat{\mathbf{v}},\hat{\lambda})\in\Omega_0$. We thus complete the proof. □

Based on Lemma 4, we further investigate whether κ converges to zero. The answer is positive in most scenarios except two special cases. Suppose that $(\hat{\mathbf{v}},\hat{\kappa})$ is a KKT point of the problem $\tilde{\mathscr{P}}_{\text{DC}}$. We consider two particular cases in terms of whether the SCB program $\mathscr{P}_{\text{SCB}}$ attaining its optimal value at the interior point or not.

Case one: When the SCB program $\mathscr{P}_{\text{SCB}}$ attains the optimal value at the interior point of its feasible region, then program $\tilde{\mathscr{P}}_{\text{SCB}}$ also attains its optimal value at the interior point of its feasible region based on Theorem 4. In this scenario, the DC constraint in $\tilde{\mathscr{P}}_{\text{SCB}}$ does not need to be tight. Thus, $\hat{\kappa}$ is not necessary to be zero and it has multiple choices, while $(\hat{\mathbf{v}},0)$ still belongs to Ω_0.

Case two: When all the optimal solutions of the SCB program $\mathscr{P}_{\text{SCB}}$ make the probability constraint tight. In this scenario, we have $[u(\hat{\mathbf{v}},\hat{\kappa})-\hat{\kappa}\epsilon]-u(\hat{\mathbf{v}},0)=0$. This reveals that $\kappa=0$ is a minimizer of the function $[u(\hat{\mathbf{v}},\kappa)-\kappa\epsilon]$ with respect to κ, i.e.,

$$\Pr\left\{\max_{1\le k\le K}d_k(\hat{\mathbf{v}},\mathbf{h})>0\right\}=\epsilon,\qquad \text{(C.4.11)}$$

where the calculation is based on (C.2.10). On the other hand, as $\hat{\kappa}$ satisfies the KKT conditions of program $\tilde{\mathscr{P}}_{\text{DC}}$, we have

$$\nabla_{\kappa}[u(\hat{\mathbf{v}}, \hat{\kappa}) - \hat{\kappa}\epsilon] = 0. \tag{C.4.12}$$

According to [153, Theorem 10] and [154, Appendix A4], the minimizer (i.e., $\hat{\kappa} \neq 0$ in (C.4.12)) of the function $[u(\hat{\mathbf{v}}, \kappa) - \kappa\epsilon]$ with respect to κ satisfies

$$\Pr\left\{\max_{1\leq k\leq K} d_k(\mathbf{v}, \mathbf{h}) > -\hat{\kappa}\right\} \leq \epsilon. \tag{C.4.13}$$

Combining (C.4.11) and (C.4.13), we conclude that $\Pr\left\{\max_{1\leq k\leq K} d_k(\hat{\mathbf{v}}, \mathbf{h}) \in (-\hat{\kappa}, 0]\right\} = 0$. This implies that the optimization variable κ in $\tilde{\mathscr{P}}_{\text{DC}}$ converges to zero, if for any $c > 0$, we have

$$\Pr\left\{\max_{1\leq k\leq K} d_k(\hat{\mathbf{v}}, \mathbf{h}) \in [-c, 0]\right\} \neq 0. \tag{C.4.14}$$

From numerical examples in Section 4.4, we will demonstrate that variable κ will indeed converge to zero.

Finally, based on Lemma 4, we only need to prove that the sequence generated by the stochastic DC programming algorithm converges to a KKT point of the program $\tilde{\mathscr{P}}_{\text{DC}}$. This directly follows [104, Property 3]. We thus complete the proof.

C.5 Proof of Theorem 6

By [81, Theorem 7.50] and [104, Theorem 6], we have that the SAA estimate $\bar{l}(\mathbf{v}, \kappa; \mathbf{v}^{[j]}, \kappa^{[j]})$ (4.3.22) converges to $l(\mathbf{v}, \kappa; \mathbf{v}^{[j]}, \kappa^{[j]})$ uniformly on the convex compact set $\mathcal{V}$ with probability one as $M \to +\infty$, i.e.,

$$\sup_{\mathbf{v}\in\mathcal{V}} |\bar{l}(\mathbf{v}, \kappa; \mathbf{v}^{[j]}, \kappa^{[j]}) - l(\mathbf{v}, \kappa; \mathbf{v}^{[j]}, \kappa^{[j]})| \to 0, M{\to}+\infty, \tag{C.5.1}$$

with probability one. Furthermore, by [81, Theorem 5.5], we have $V_M^{\star}(\mathbf{v}^{[j]}, \kappa^{[j]}) \to V^{\star}(\mathbf{v}^{[j]}, \kappa^{[j]})$ and $\mathbb{D}(\mathcal{P}_M^{\star}(\mathbf{v}^{[j]}, \kappa^{[j]}), \mathcal{P}^{\star}(\mathbf{v}^{[j]}, \kappa^{[j]})) \to 0$ with probability one as $M \to +\infty$. Therefore, we complete the proof.

Appendix D

Derivations in Chapter 5

D.1 Conic Formulation for Convex Programs

We shall present a systematic way to transform the original problem to the standard convex cone programming form. We first take the real-field problem $\mathscr{P}$ with the objective function $f(\mathbf{x}) = \|\mathbf{v}\|_2$ as an example. At the end of this subsection, we will show how to extend it to the complex-field.

According to the principle of the disciplined convex programming [131], the original problem $\mathscr{P}$ can be rewritten as the following disciplined convex programming form [131]

$$\begin{aligned} \mathscr{P}_{\text{cvx}} : \text{minimize} \quad & \|\mathbf{v}\|_2 \\ \text{subject to} \quad & \|\mathbf{D}_l \mathbf{v}\|_2 \le \sqrt{P_l}, l = 1, \dots, L && \text{(D.1.1)} \\ & \|\mathbf{C}_k \mathbf{v} + \mathbf{g}_k\|_2 \le \beta_k \mathbf{r}_k^T \mathbf{v}, k = 1, \dots, K, && \text{(D.1.2)} \end{aligned}$$

where $\mathbf{D}_l = \text{blkdiag}\{\mathbf{D}_l^1, \dots, \mathbf{D}_l^K\} \in \mathbb{R}^{N_l K \times NK}$ with $\mathbf{D}_l^k = \left[\mathbf{0}_{N_l \times \sum_{i=1}^{l-1} N_i}, \mathbf{I}_{N_l \times N_l}, \mathbf{0}_{N_l \times \sum_{i=l+1}^{L} N_i}\right] \in \mathbb{R}^{N_l \times N}$, $\beta_k = \sqrt{1 + 1/\gamma_k}$, $\mathbf{r}_k = \left[\mathbf{0}_{(k-1)N}^T, \mathbf{h}_k^T, \mathbf{0}_{(K-k)N}^T\right]^T \in \mathbb{R}^{NK}$, $\mathbf{g}_k = [\mathbf{0}_K^T, \sigma_k]^T \in \mathbb{R}^{K+1}$, and $\mathbf{C}_k = [\tilde{\mathbf{C}}_k, \mathbf{0}_{NK}]^T \in \mathbb{R}^{(K+1) \times NK}$ with $\tilde{\mathbf{C}}_k = \text{blkdiag}\{\mathbf{h}_k, \dots, \mathbf{h}_k\} \in \mathbb{R}^{NK \times K}$. It is thus easy to check the convexity of problem $\mathscr{P}_{\text{cvx}}$, following the disciplined convex programming ruleset [131].

D.1.1 Smith Form Reformulation

To arrive at the standard convex cone program $\mathscr{P}_{\text{cone}}$, we rewrite problem $\mathscr{P}_{\text{cvx}}$ as the following Smith form [130] by introducing a new variable for each subexpression in $\mathscr{P}_{\text{cvx}}$,

$$\begin{aligned} \text{minimize} \quad & x_0 \\ \text{subject to} \quad & \|\mathbf{x}_1\| = x_0, \mathbf{x}_1 = \mathbf{v} \\ & \mathcal{G}_1(l), \mathcal{G}_2(k), \forall k, l, \end{aligned} \tag{D.1.3}$$

where $\mathcal{G}_1(l)$ is the Smith form reformulation for the transmit power constraint for RRH l (D.1.1) as follows

$$\mathcal{G}_1(l) : \begin{cases} (y_0^l, \mathbf{y}_1^l) \in \mathcal{Q}^{KN_l+1} \\ y_0^l = \sqrt{P_l} \in \mathbb{R} \\ \mathbf{y}_1^l = \mathbf{D}_l \mathbf{v} \in \mathbb{R}^{KN_l}, \end{cases} \tag{D.1.4}$$

and $\mathcal{G}_2(k)$ is the Smith form reformulation for the QoS constraint for MU k (D.1.2) as follows

$$\mathcal{G}_2(k) : \begin{cases} (t_0^k, \mathbf{t}_1^k) \in \mathcal{Q}^{K+1} \\ t_0^k = \beta_k \mathbf{r}_k^T \mathbf{v} \in \mathbb{R} \\ \mathbf{t}_1^k = \mathbf{t}_2^k + \mathbf{t}_3^k \in \mathbb{R}^{K+1} \\ \mathbf{t}_2^k = \mathbf{C}_k \mathbf{v} \in \mathbb{R}^{K+1} \\ \mathbf{t}_3^k = \mathbf{g}_k \in \mathbb{R}^{K+1}. \end{cases} \tag{D.1.5}$$

Nevertheless, the Smith form reformulation (D.1.3) is not convex due to the non-convex constraint $\|\mathbf{x}_1\| = x_0$. We thus relax the non-convex constraint as $\|\mathbf{x}_1\| \leq x_0$, yielding the following relaxed Smith form

$$\begin{aligned} \text{minimize} \quad & x_0 \\ \text{subject to} \quad & \mathcal{G}_0, \mathcal{G}_1(l), \mathcal{G}_2(k), \forall k, l, \end{aligned} \tag{D.1.6}$$

where

$$\mathcal{G}_0 : \begin{cases} (x_0, \mathbf{x}_1) \in \mathcal{Q}^{NK+1} \\ \mathbf{x}_1 = \mathbf{v} \in \mathbb{R}^{NK}. \end{cases} \tag{D.1.7}$$

It can be easily proved that the constraint $\|\mathbf{x}_1\| \leq x_0$ has to be active at the optimal solution; otherwise, we can always scale down x_0 such that the cost function can be further minimized while still satisfying the constraints. Therefore, we conclude that the relaxed Smith form (D.1.6) is equivalent to the original problem $\mathscr{P}_{\text{cvx}}$.

D.1.2 Conic Reformulation

Now, the relaxed Smith form reformulation (D.1.6) is readily to be reformulated as the standard cone programming form $\mathscr{P}_{\text{cone}}$. Specifically, define the optimization variables $[x_0; \mathbf{v}]$ with the same order of equations as in $\mathcal{G}_0$, then $\mathcal{G}_0$ can be rewritten as

$$\mathbf{M}[x_0; \mathbf{v}] + \boldsymbol{\mu}_0 = \mathbf{m}, \tag{D.1.8}$$

where the slack variables belong to the following convex set

$$\boldsymbol{\mu}_0 \in \mathcal{Q}^{NK+1}, \tag{D.1.9}$$

and $\mathbf{M} \in \mathbb{R}^{(NK+1)\times(NK+1)}$ and $\mathbf{m} \in \mathbb{R}^{NK+1}$ are given as follows

$$\mathbf{M} = \left[\begin{array}{c|c} -1 & \\ \hline & -\mathbf{I}_{NK} \end{array}\right], \mathbf{m} = \left[\begin{array}{c} 0 \\ \hline \mathbf{0}_{NK} \end{array}\right], \tag{D.1.10}$$

respectively. Define the optimization variables $[y_0^l; \mathbf{v}]$ with the same order of equations as in $\mathcal{G}_1(l)$, then $\mathcal{G}_1(l)$ can be rewritten as

$$\mathbf{P}^l[y_0^l; \mathbf{v}] + \boldsymbol{\mu}_1^l = \mathbf{p}^l, \tag{D.1.11}$$

where the slack variables $\boldsymbol{\mu}_1^l \in \mathbb{R}^{KN_l+2}$ belongs to the following convex set formed by the Cartesian product of two convex sets

$$\boldsymbol{\mu}_1^l \in \mathcal{Q}^1 \times \mathcal{Q}^{KN_l+1}, \tag{D.1.12}$$

and $\mathbf{P}^l \in \mathbb{R}^{(KN_l+2)\times(NK+1)}$ and $\mathbf{p}^l \in \mathbb{R}^{KN_l+2}$ are given as follows

$$\mathbf{P}^l = \left[\begin{array}{c|c} 1 & \\ \hline -1 & \\ & -\mathbf{D}_l \end{array}\right], \mathbf{p}^l = \left[\begin{array}{c} \sqrt{P_l} \\ \hline 0 \\ \mathbf{0}_{KN_l} \end{array}\right], \tag{D.1.13}$$

respectively. Define the optimization variables $[t_0^k; \mathbf{v}]$ with the same order of equations as in $\mathcal{G}_2(k)$, then $\mathcal{G}_2(k)$ can be rewritten as

$$\mathbf{Q}^k[t_0^k; \mathbf{v}] + \boldsymbol{\mu}_2^k = \mathbf{q}^k, \tag{D.1.14}$$

where the slack variables $\boldsymbol{\mu}_2^k \in \mathbb{R}^{K+3}$ belong to the following convex set formed by the Cartesian product of two convex sets

$$\boldsymbol{\mu}_2^k \in \mathcal{Q}^1 \times \mathcal{Q}^{K+2}, \tag{D.1.15}$$

and $\mathbf{Q}^k \in \mathbb{R}^{(K+3)\times(NK+1)}$ and $\mathbf{q}^k \in \mathbb{R}^{K+3}$ are given as follows

$$\mathbf{Q}^k = \left[\begin{array}{c|c} 1 & -\beta_k \mathbf{r}_k^T \\ \hline -1 & \\ & -\mathbf{C}_k \end{array}\right], \mathbf{q}^k = \left[\begin{array}{c} 0 \\ \hline 0 \\ \mathbf{g}_k \end{array}\right], \tag{D.1.16}$$

respectively.

Therefore, we arrive at the standard form $\mathscr{P}_{\text{cone}}$ by writing the optimization variables $\boldsymbol{\nu} \in \mathbb{R}^n$ as follows

$$\boldsymbol{\nu} = [x_0; y_0^1; \ldots; y_0^L; t_0^1; \ldots, t_0^K; \mathbf{v}], \tag{D.1.17}$$

and $\mathbf{c} = [1; \mathbf{0}_{n-1}]$. The structure of the standard cone programming $\mathscr{P}_{\text{cone}}$ is characterized by the following data

$$
\begin{aligned}
n &= 1 + L + K + NK, && \text{(D.1.18)}\\
m &= (L+K) + (NK+1) + \sum_{l=1}^{L}(KN_l + 1) + K(K+2), && \text{(D.1.19)}\\
\mathcal{K} &= \underbrace{\mathcal{Q}^1 \times \cdots \times \mathcal{Q}^1}_{L+K} \times \mathcal{Q}^{NK+1} \times \underbrace{\mathcal{Q}^{KN_1+1} \times \cdots \times \mathcal{Q}^{KN_L+1}}_{L} \times \\
&\quad \underbrace{\mathcal{Q}^{K+2} \times \cdots \times \mathcal{Q}^{K+2}}_{K}, && \text{(D.1.20)}
\end{aligned}
$$

where $\mathcal{K}$ is the Cartesian product of $2(L+K)+1$ second-order ones, and $\mathbf{A}$ and $\mathbf{b}$ are given as follows:

$$
\mathbf{A} = \left[\begin{array}{c|ccc|ccc|c}
 & 1 & & & & & & \\
 & & \ddots & & & & & \\
 & & & 1 & & & & \\
\hline
 & & & & 1 & & & -\beta_1\mathbf{r}_1^T \\
 & & & & & \ddots & & \vdots \\
 & & & & & & 1 & -\beta_K\mathbf{r}_K^T \\
\hline
-1 & & & & & & & \\
 & & & & & & & -\mathbf{I}_{NK} \\
\hline
 & -1 & & & & & & \\
 & & & & & & & -\mathbf{D}_1 \\
\hline
 & & \vdots & & & \vdots & & \vdots \\
\hline
 & & & -1 & & & & \\
 & & & & & & & -\mathbf{D}_L \\
\hline
 & & & & -1 & & & \\
 & & & & & & & -\mathbf{C}_1 \\
\hline
 & & & & & \vdots & & \vdots \\
\hline
 & & & & & & -1 & \\
 & & & & & & & -\mathbf{C}_K
\end{array}\right], \mathbf{b} = \left[\begin{array}{c}
\sqrt{P_1} \\
\vdots \\
\sqrt{P_L} \\
\hline
0 \\
\vdots \\
0 \\
\hline
0 \\
\mathbf{0}_{NK} \\
\hline
0 \\
\mathbf{0}_{KN_1} \\
\hline
\vdots \\
\hline
0 \\
\mathbf{0}_{KN_1} \\
\hline
0 \\
\mathbf{g}_1 \\
\hline
\vdots \\
\hline
0 \\
\mathbf{g}_K
\end{array}\right], \qquad \text{(D.1.21)}
$$

respectively.

D.1.3 Matrix Stuffing

Given a problem instance $\mathscr{P}$, to arrive at the standard cone program form, we only need to copy the parameters of the maximum transmit power P_l's to the data of the standard form, i.e., $\sqrt{P_l}$'s in $\mathbf{b}$, copy the parameters of the SINR thresholds γ to the data of the standard form, i.e., β_k's in $\mathbf{A}$, and copy the parameters of the channel realizations $\mathbf{h}_k$'s to the data of the standard form, i.e., $\mathbf{r}_k$'s and $\mathbf{C}_k$'s in $\mathbf{A}$. As we only need to perform copying the memory for the transformation, this procedure can be very efficient compared to the state-of-the-art numerical based modeling frameworks like CVX.

D.1.4 Extension to the Complex Case

For $\mathbf{h}_k \in \mathbb{C}^N, \mathbf{v}_i \in \mathbb{C}^N$, we have

$$\mathbf{h}_k^{\mathsf{H}}\mathbf{v}_i \Longrightarrow \underbrace{\begin{bmatrix} \Re(\mathbf{h}_k) & -\Im(\mathbf{h}_k) \\ \Im(\mathbf{h}_k) & \Re(\mathbf{h}_k) \end{bmatrix}^T}_{\tilde{\mathbf{h}}_k} \underbrace{\begin{bmatrix} \Re(\mathbf{v}_i) \\ \Im(\mathbf{v}_i) \end{bmatrix}}_{\tilde{\mathbf{v}}_i}, \tag{D.1.22}$$

where $\tilde{\mathbf{h}}_k \in \mathbb{R}^{2N\times 2}$ and $\tilde{\mathbf{v}}_i \in \mathbb{R}^{2N}$. Therefore, the complex-field problem can be changed into the real-field problem by the transformations: $\mathbf{h}_k \Rightarrow \tilde{\mathbf{h}}_k$ and $\mathbf{v}_i \Rightarrow \tilde{\mathbf{v}}_i$.

References

[1] J. Andrews, S. Buzzi, W. Choi, S. Hanly, A. Lozano, A. Soong, and J. Zhang, "What will 5G be?" *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1065–1082, Jun. 2014.

[2] G. Fettweis, "The tactile internet: Applications and challenges," *IEEE Veh. Technol. Mag.*, vol. 9, no. 1, pp. 64–70, Mar. 2014.

[3] F. Rusek, D. Persson, B. K. Lau, E. Larsson, T. Marzetta, O. Edfors, and F. Tufvesson, "Scaling up MIMO: Opportunities and challenges with very large arrays," *IEEE Signal Process. Mag.*, vol. 30, no. 1, pp. 40–60, Jan. 2013.

[4] I. Hwang, B. Song, and S. Soliman, "A holistic view on hyper-dense heterogeneous and small cell networks," *IEEE Commun. Mag.*, vol. 51, no. 6, pp. 20–27, Jun. 2013.

[5] T. Marzetta, "Noncooperative cellular wireless with unlimited numbers of base station antennas," *IEEE Trans. Wireless Commun.*, vol. 9, no. 11, pp. 3590–3600, Nov. 2010.

[6] J. Andrews, "Seven ways that hetnets are a cellular paradigm shift," *IEEE Commun. Mag.*, vol. 51, no. 3, pp. 136–144, Mar. 2013.

[7] D. Gesbert, S. Hanly, H. Huang, S. Shamai Shitz, O. Simeone, and W. Yu, "Multi-cell MIMO cooperative networks: A new look at interference," *IEEE J. Sel. Areas Commun.*, vol. 28, no. 9, pp. 1380–1408, Sep. 2010.

[8] A. Lozano, R. Heath, and J. Andrews, "Fundamental limits of cooperation," *IEEE Trans. Inf. Theory*, vol. 59, no. 9, pp. 5213–5226, Sep. 2013.

[9] H. Huh, A. Tulino, and G. Caire, "Network MIMO with linear zero-forcing beamforming: Large system analysis, impact of channel estimation, and reduced-complexity scheduling," *IEEE Trans. Inf. Theory*, vol. 58, no. 5, pp. 2911–2934, May 2012.

[10] China Mobile, “C-RAN: the road towards green RAN,” *White Paper, ver. 3.0*, Dec. 2013.

[11] Y. Shi, J. Zhang, K. Letaief, B. Bai, and W. Chen, “Large-scale convex optimization for ultra-dense cloud-ran,” *IEEE Wireless Commun. Mag.*, vol. 22, no. 3, pp. 84–91, Jun. 2015.

[12] D. Wubben, P. Rost, J. Bartelt, M. Lalam, V. Savin, M. Gorgoglione, A. Dekorsy, and G. Fettweis, “Benefits and impact of cloud computing on 5G signal processing: Flexible centralization through Cloud-RAN,” *IEEE Signal Process. Mag.*, vol. 31, no. 6, pp. 35–44, Nov. 2014.

[13] D. Bojic and N. Europe, “Advanced wireless and optical technologies for small-cell mobile backhaul with dynamic software-defined management,” *IEEE Commun. Mag.*, vol. 51, no. 9, pp. 86–93, Sep. 2013.

[14] S. Park, O. Simeone, O. Sahin, and S. Shamai Shitz, “Fronthaul compression for cloud radio access networks: Signal processing advances inspired by network information theory,” *IEEE Signal Process. Mag.*, vol. 31, no. 6, pp. 69–79, Nov. 2014.

[15] D. P. Palomar and Y. C. Eldar, *Convex optimization in signal processing and communications*. Cambridge University Press, 2010.

[16] Z.-Q. Luo and W. Yu, “An introduction to convex optimization for communications and signal processing,” *IEEE J. Sel. Areas Commun.*, vol. 24, no. 8, pp. 1426–1438, Aug 2006.

[17] A. B. Gershman, N. D. Sidiropoulos, S. Shahbazpanahi, M. Bengtsson, and B. Ottersten, “Convex optimization-based beamforming: From receive to transmit and network designs,” *IEEE Signal Process. Mag*, vol. 27, no. 3, pp. 62–75, 2010.

[18] H. Dahrouj and W. Yu, “Coordinated beamforming for the multicell multi-antenna wireless system,” *IEEE Trans. Wireless Commun.*, vol. 9, no. 5, pp. 1748–1759, Sep. 2010.

[19] S. Kandukuri and S. Boyd, “Optimal power control in interference-limited fading wireless channels with outage-probability specifications,” *IEEE Trans. Wireless Commun.*, vol. 1, no. 1, pp. 46–55, 2002.

[20] E. Matskani, N. D. Sidiropoulos, Z. Q. Luo, and L. Tassiulas, "Convex approximation techniques for joint multiuser downlink beamforming and admission control," *IEEE Trans. Wireless Commun.*, vol. 7, no. 7, pp. 2682–2693, Jul. 2008.

[21] W.-C. Liao, M. Hong, H. Farmanbar, X. Li, Z.-Q. Luo, and H. Zhang, "Min flow rate maximization for software defined radio access networks," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1282–1294, Jun. 2014.

[22] S. P. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge University Press, 2004.

[23] R. Schaefer and H. Boche, "Physical layer service integration in wireless networks: Signal processing challenges," *IEEE Signal Process. Mag.*, vol. 31, no. 3, pp. 147–156, May 2014.

[24] S. Jafar, "Topological interference management through index coding," *IEEE Trans. Inf. Theory*, vol. 60, no. 1, pp. 529–568, Jan. 2014.

[25] Y. Shi, J. Zhang, and K. B. Letaief, "Low-rank matrix completion via Riemannian pursuit for topological interference management," in *Proc. IEEE Int. Symp. Inform. Theory (ISIT)*, Hong Kong, 2015.

[26] M. Razaviyayn, H. Baligh, A. Callard, and Z.-Q. Luo, "Joint user grouping and transceiver design in a mimo interfering broadcast channel," *Signal Processing, IEEE Transactions on*, vol. 62, no. 1, pp. 85–94, 2014.

[27] W. Yu and T. Lan, "Transmitter optimization for the multi-antenna downlink with per-antenna power constraints," *IEEE Trans. Signal Process.*, vol. 55, no. 6, pp. 2646–2660, 2007.

[28] Y. Shi, J. Zhang, and K. Letaief, "Group sparse beamforming for green cloud radio access networks," in *Proc. IEEE Global Communications Conf. (GLOBECOM)*. Atlanta, GA, USA, Dec. 2013, pp. 4635–4640.

[29] Y. Shi, J. Zhang, and K. B. Letaief, "Group sparse beamforming for green Cloud-RAN," *IEEE Trans. Wireless Commun.*, vol. 13, no. 5, pp. 2809–2823, May 2014.

[30] J. Hoydis, M. Kobayashi, and M. Debbah, “Green small-cell networks,” *IEEE Veh. Technol. Mag.*, vol. 6, no. 1, pp. 37–43, Mar. 2011.

[31] J. Wu, “Green wireless communications: from concept to reality [industry perspectives],” *IEEE Wireless Commun.*, vol. 19, no. 4, pp. 4–5, Aug. 2012.

[32] C. Li, J. Zhang, and K. Letaief, “Energy efficiency analysis of small cell networks,” in *Proc. IEEE Int. Conf. on Commun. (ICC).* Budapest, Hungary, Jun. 2013, pp. 4404–4408.

[33] S. Tombaz, P. Monti, K. Wang, A. Vastberg, M. Forzati, and J. Zander, “Impact of backhauling power consumption on the deployment of heterogeneous mobile networks,” in *Proc. IEEE Global Communications Conf. (GLOBECOM).* Houston, Texas, USA, Dec. 2011, pp. 1–5.

[34] J. Rao and A. Fapojuwo, “On the tradeoff between spectral efficiency and energy efficiency of homogeneous cellular networks with outage constraint,” *IEEE Trans. Veh. Technol.*, vol. 62, no. 4, pp. 1801–1814, May 2013.

[35] D. Donoho, “Compressed sensing,” *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[36] E. Candes and T. Tao, “Near-optimal signal recovery from random projections: Universal encoding strategies?” *IEEE Trans. Inf. Theory*, vol. 52, no. 12, pp. 5406–5425, Dec. 2006.

[37] C. Berger, Z. Wang, J. Huang, and S. Zhou, “Application of compressive sensing to sparse channel estimation,” *IEEE Commun. Mag.*, vol. 48, no. 11, pp. 164–174, Nov. 2010.

[38] F. Bach, R. Jenatton, J. Mairal, and G. Obozinski, “Optimization with sparsity-inducing penalties,” *Foundations Trends Mach. Learning*, vol. 4, no. 1, pp. 1–106, Jan. 2012.

[39] M. Yuan and Y. Lin, “Model selection and estimation in regression with grouped variables,” *J. R. Statist. Soc. B*, vol. 68, no. 1, pp. 49–67, 2006.

[40] S. Negahban and M. Wainwright, "Simultaneous support recovery in high dimensions: Benefits and perils of block ℓ_1/ℓ_∞-regularization," *IEEE Trans. Inf. Theory*, vol. 57, no. 6, pp. 3841–3863, Jun. 2011.

[41] M. Hong, R. Sun, H. Baligh, and Z.-Q. Luo, "Joint base station clustering and beamformer design for partial coordinated transmission in heterogeneous networks," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 2, pp. 226–240, Feb. 2013.

[42] J. Zhao, T. Q. Quek, and Z. Lei, "Coordinated multipoint transmission with limited backhaul data transfer," *IEEE Trans. Wireless Commun.*, vol. 12, no. 6, pp. 2762–2775, Jun. 2013.

[43] O. Mehanna, N. Sidiropoulos, and G. Giannakis, "Joint multicast beamforming and antenna selection," *IEEE Trans. Signal Process.*, vol. 61, no. 10, pp. 2660–2674, May 2013.

[44] S.-H. Park, O. Simeone, O. Sahin, and S. Shamai, "Robust and efficient distributed compression for cloud radio access networks," *IEEE Trans. Veh. Technol.*, vol. 62, no. 2, pp. 692–703, Feb. 2013.

[45] ——, "Joint precoding and multivariate backhaul compression for the downlink of cloud radio access networks," *IEEE Trans. Signal Process.*, vol. 61, no. 22, pp. 5646–5658, Nov. 2013.

[46] V. Cadambe and S. Jafar, "Interference alignment and degrees of freedom of the K-user interference channel," *IEEE Trans. Inf. Theory*, vol. 54, no. 8, pp. 3425–3441, Aug. 2008.

[47] G. Auer, V. Giannini, C. Desset, I. Godor, P. Skillermark, M. Olsson, M. Imran, D. Sabella, M. Gonzalez, O. Blume, and A. Fehske, "How much energy is needed to run a wireless network?" *IEEE Wireless Commun.*, vol. 18, no. 5, pp. 40–49, Oct. 2011.

[48] J. Kani and H. Nakamura, "Recent progress and continuing challenges in optical access network technologies," in *IEEE 3rd International Conference on Photonics (ICP)*, 2012, pp. 66–70.

[49] A. Dhaini, P.-H. Ho, G. Shen, and B. Shihada, "Energy efficiency in TDMA-based next-generation passive optical access networks," *IEEE/ACM Trans. Netw.*, vol. PP, no. 99, pp. 1–1, 2013.

[50] A. Wiesel, Y. Eldar, and S. Shamai, "Linear precoding via conic optimization for fixed MIMO receivers," *IEEE Trans. Signal Process.*, vol. 54, no. 1, pp. 161–176, Jan. 2006.

[51] T. Baran, D. Wei, and A. Oppenheim, "Linear programming algorithms for sparse filter design," *IEEE Trans. Signal Process.*, vol. 58, no. 3, pp. 1605–1617, Mar. 2010.

[52] S. Leyffer, *Mixed integer nonlinear programming*. Springer, 2012, vol. 154.

[53] Y. Cheng, M. Pesavento, and A. Philipp, "Joint network optimization and downlink beamforming for CoMP transmissions using mixed integer conic programming," *IEEE Trans. Signal Process.*, vol. 61, no. 16, pp. 3972–3987, Aug. 2013.

[54] G. Obozinski and F. Bach, "Convex relaxation for combinatorial penalties," *arXiv preprint arXiv:1205.1240*, 2012. [Online]. Available: http://arxiv.org/abs/1205.1240

[55] S. Vishwanath, N. Jindal, and A. Goldsmith, "Duality, achievable rates, and sum-rate capacity of gaussian MIMO broadcast channels," *IEEE Trans. Inf. Theory*, vol. 49, no. 10, pp. 2658–2668, Oct. 2003.

[56] D. R. Hunter and K. Lange, "A tutorial on MM algorithms," *Amer. Statistician*, vol. 58, no. 1, pp. 30–37, 2004.

[57] E. J. Candes, M. B. Wakin, and S. P. Boyd, "Enhancing sparsity by reweighted ℓ_1 minimization," *J. Fourier Anal. Appl.*, vol. 14, no. 5-6, pp. 877–905, Dec. 2008.

[58] B. K. Sriperumbudur, D. A. Torres, and G. R. Lanckriet, "A majorization-minimization approach to the sparse generalized eigenvalue problem," *Machine learning*, vol. 85, no. 1-2, pp. 3–39, Oct. 2011.

[59] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends in Mach. Learn.*, vol. 3, no. 1, pp. 1–122, Jul. 2011.

[60] J. Cheng, Y. Shi, B. Bai, W. Chen, J. Zhang, and K. Letaief, “Group sparse beamforming for multicast green Cloud-RAN via parallel semidefinite programming,” *IEEE Int. Conf. on Commun. (ICC), London, UK*, 2015.

[61] Y. Shi, J. Zhang, and K. Letaief, “Robust group sparse beamforming for multicast green cloud-ran with imperfect csi,” *IEEE Trans. Signal Process.*, vol. 63, no. 17, pp. 4647–4659, Sept 2015.

[62] Y. Shi, J. Cheng, J. Zhang, B. Bai, W. Chen, and K. B. Letaief, “Smoothed l_p-minimization for green cloud-ran with user admission control,” *IEEE J. Sel. Areas Commun.*, submitted, 2015.

[63] S. Rangan, T. Rappaport, and E. Erkip, “Millimeter-wave cellular wireless networks: potentials and challenges,” *Proc. IEEE*, vol. 102, no. 3, pp. 366–385, Mar. 2014.

[64] M. Peng, Y. Li, J. Jiang, J. Li, and C. Wang, “Heterogeneous cloud radio access networks: a new perspective for enhancing spectral and energy efficiencies,” *IEEE Wireless Commun. Mag.*, vol. 21, no. 6, pp. 126–135, Dec. 2014.

[65] J. Zhang, R. Chen, J. G. Andrews, A. Ghosh, and R. W. Heath, “Networked MIMO with clustered linear precoding,” *IEEE Trans. Wireless Commun.*, vol. 8, no. 4, pp. 1910–1921, Apr. 2009.

[66] S. Luo, R. Zhang, and T. J. Lim, “Downlink and uplink energy minimization through user association and beamforming in C-RAN,” *IEEE Trans. Wireless Commun.*, vol. 14, no. 1, pp. 494–508, Jan. 2015.

[67] Y. Shi, J. Zhang, and K. Letaief, “Scalable coordinated beamforming for dense wireless cooperative networks,” in *Proc. IEEE Global Communications Conf. (GLOBECOM), Austin, TX, USA*, Dec. 2014, pp. 3603–3608.

[68] B. Dai and W. Yu, “Sparse beamforming and user-centric clustering for downlink cloud radio access network,” *IEEE Access*, vol. 2, pp. 1326–1339, Nov. 2014.

[69] D. J. Love, R. W. Heath, V. K. Lau, D. Gesbert, B. D. Rao, and M. Andrews, “An overview of limited feedback in wireless communication systems,” *IEEE J. Sel. Areas Commun.*, vol. 26, no. 8, pp. 1341–1365, Oct. 2008.

[70] N. Jindal and A. Lozano, “A unified treatment of optimum pilot overhead in multipath fading channels,” *IEEE Trans. Commun.*, vol. 58, no. 10, pp. 2939–2948, Oct. 2010.

[71] Y. Shi, J. Zhang, and K. Letaief, “CSI overhead reduction with stochastic beamforming for cloud radio access networks,” in *Proc. of IEEE Int. Conf. on Commun. (ICC), Sydney, Australia*, Jun. 2014.

[72] ——, “Optimal stochastic coordinated beamforming for wireless cooperative networks with CSI uncertainty,” *IEEE Trans. Signal Process.*, vol. 63, no. 4, pp. 960–973, Feb. 2015.

[73] M. A. Maddah-Ali and D. Tse, “Completely stale transmitter channel state information is still very useful,” *IEEE Trans. Inf. Theory*, vol. 58, no. 7, pp. 4418–4431, Jul. 2012.

[74] J. Zhang, R. W. Heath, M. Kountouris, and J. G. Andrews, “Mode switching for the multi-antenna broadcast channel based on delay and channel quantization,” *EURASIP J. Adv. Signal Process. (Special Issue Multiuser Lim. Feedback)*, vol. 2009, Article ID 802548, 15 pages, 2009.

[75] H. Weingarten, Y. Steinberg, and S. Shamai, “The capacity region of the gaussian multiple-input multiple-output broadcast channel,” *IEEE Trans. Inf. Theory*, vol. 52, no. 9, pp. 3936–3964, Sep. 2006.

[76] E. Karipidis, N. D. Sidiropoulos, and Z.-Q. Luo, “Quality of service and max-min fair transmit beamforming to multiple cochannel multicast groups,” *IEEE Trans. Signal Process.*, vol. 56, no. 3, pp. 1268–1279, Mar. 2008.

[77] N. Golrezaei, A. Molisch, A. Dimakis, and G. Caire, “Femtocaching and device-to-device collaboration: A new architecture for wireless video distribution,” *IEEE Commun. Mag.*, vol. 51, no. 4, pp. 142–149, Apr. 2013.

[78] D. Bertsimas, D. B. Brown, and C. Caramanis, “Theory and applications of robust optimization,” *SIAM review*, vol. 53, no. 3, pp. 464–501, Aug. 2011.

[79] H. Weingarten, T. Liu, S. Shamai, Y. Steinberg, and P. Viswanath, “The capacity region of the degraded multiple-input multiple-output compound broadcast channel,” *IEEE Trans. Inf. Theory*, vol. 55, no. 11, pp. 5011–5023, Nov. 2009.

[80] R. Mochaourab and E. Jorswieck, "Optimal beamforming in interference networks with perfect local channel information," *IEEE Trans. Signal Process.*, vol. 59, no. 3, pp. 1128–1141, Mar. 2011.

[81] A. Shapiro, D. Dentcheva, and A. P. Ruszczyński, *Lectures on stochastic programming: modeling and theory.* SIAM, 2009, vol. 9.

[82] A. Ben-Tal, L. El Ghaoui, and A. Nemirovski, *Robust optimization.* Princeton University Press, 2009.

[83] C. Shen, T.-H. Chang, K.-Y. Wang, Z. Qiu, and C.-Y. Chi, "Distributed robust multicell coordinated beamforming with imperfect CSI: an ADMM approach," *IEEE Trans. Signal Process.*, vol. 60, no. 6, pp. 2988–3003, Jun. 2012.

[84] Z.-Q. Luo, W.-K. Ma, A.-C. So, Y. Ye, and S. Zhang, "Semidefinite relaxation of quadratic optimization problems," *IEEE Signal Process. Mag.*, vol. 27, no. 3, pp. 20–34, May 2010.

[85] E. Song, Q. Shi, M. Sanjabi, R.-Y. Sun, and Z.-Q. Luo, "Robust SINR-constrained MISO downlink beamforming: When is semidefinite programming relaxation tight?" *EURASIP J. Wireless Communun. Netw.*, vol. 2012, no. 1, pp. 1–11, 2012.

[86] E. Björnson, G. Zheng, M. Bengtsson, and B. Ottersten, "Robust monotonic optimization framework for multicell MISO systems," *IEEE Trans. Signal Process.*, vol. 60, no. 5, pp. 2508–2523, May 2012.

[87] E. Björnson and E. Jorswieck, "Optimal resource allocation in coordinated multi-cell systems," *Found. Trends Commun. Inf. Theory*, vol. 9, no. 2-3, pp. 113–381, Jan. 2013.

[88] E. J. Candes and X. Li, "Solving quadratic equations via phaselift when there are about as many equations as unknowns," *Found. Comput. Math.*, pp. 1–10, Jun. 2012.

[89] D. Ge, X. Jiang, and Y. Ye, "A note on the complexity of l_p minimization," *Mathematical Programming*, vol. 129, no. 2, pp. 285–299, 2011.

[90] K. Lange and J. S. Sinsheimer, "Normal/independent distributions and their applications in robust regression," *Journal of Computational and Graphical Statistics*, vol. 2, no. 2, pp. 175–198, 1993.

[91] D. Ba, B. Babadi, P. L. Purdon, and E. N. Brown, "Convergence and stability of iteratively re-weighted least squares algorithms," *IEEE Trans. Signal Process.*, vol. 62, no. 1, pp. 183–195, Jan. 2014.

[92] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the royal statistical society. Series B (methodological)*, pp. 1–38, 1977.

[93] I. Daubechies, R. DeVore, M. Fornasier, and C. S. Güntürk, "Iteratively reweighted least squares minimization for sparse recovery," *Communications on Pure and Applied Mathematics*, vol. 63, no. 1, pp. 1–38, 2010.

[94] R. Chartrand and W. Yin, "Iteratively reweighted algorithms for compressive sensing," in *Proc. Int. Conf. Accoust., Speech, Signal Process. (ICASSP)*, 2008, pp. 3869–3872.

[95] E. J. Candes, Y. C. Eldar, T. Strohmer, and V. Voroninski, "Phase retrieval via matrix completion," *SIAM Journal on Imaging Sciences*, vol. 6, no. 1, pp. 199–225, 2013.

[96] B. O'Donoghue, E. Chu, N. Parikh, and S. Boyd, "Conic optimization via operator splitting and homogeneous self-dual embedding," *arXiv preprint arXiv:1312.3039*, 2013. [Online]. Available: http://arxiv.org/abs/1312.3039

[97] Y. Shi, J. Zhang, B. O'Donoghue, and K. Letaief, "Large-scale convex optimization for dense wireless cooperative networks," *IEEE Trans. Signal Process.*, vol. 63, no. 18, pp. 4729–4743, Sept 2015.

[98] M. K. Karakayali, G. J. Foschini, and R. A. Valenzuela, "Network coordination for spectrally efficient communications in cellular systems," *IEEE Wireless Commun.*, vol. 13, no. 4, pp. 56–61, Aug. 2006.

[99] W.-L. Li, Y. J. Zhang, A.-C. So, and M. Z. Win, "Slow adaptive OFDMA systems through chance constrained programming," *IEEE Trans. Signal Process.*, vol. 58, no. 7, pp. 3858–3869, Jul. 2010.

[100] A. M.-C. So and Y. J. A. Zhang, "Distributionally robust slow adaptive OFDMA with soft QoS via linear programming," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 5, pp. 947–958, May 2013.

[101] V. Lau, F. Zhang, and Y. Cui, "Low complexity delay-constrained beamforming for multi-user MIMO systems with imperfect CSIT," *IEEE Trans. Signal Process.*, vol. 61, no. 16, pp. 4090–4099, Aug. 2013.

[102] W. Xu, A. Tajer, X. Wang, and S. Alshomrani, "Power allocation in MISO interference channels with stochastic CSIT," *IEEE Trans. Wireless Commun.*, vol. 13, no. 3, pp. 1716–1727, Mar. 2014.

[103] K.-Y. Wang, T.-H. Chang, W.-K. Ma, A.-C. So, and C.-Y. Chi, "Probabilistic SINR constrained robust transmit beamforming: A Bernstein-type inequality based conservative approach," in *Proc. of IEEE Int. Conf. Speech Signal Process. (ICASSP)*, May 2011, pp. 3080–3083.

[104] L. J. Hong, Y. Yang, and L. Zhang, "Sequential convex approximations to joint chance constrained programs: A Monte Carlo approach," *Oper. Res.*, vol. 59, no. 3, pp. 617–630, May-Jun. 2011.

[105] M. C. Campi and S. Garatti, "The exact feasibility of randomized solutions of uncertain convex programs," *SIAM J. Optim.*, vol. 19, no. 3, pp. 1211–1230, 2008.

[106] M. Razaviyayn, M. Hong, and Z.-Q. Luo, "A unified convergence analysis of block successive minimization methods for nonsmooth optimization," *SIAM J. Optim.*, vol. 23, no. 2, pp. 1126–1153, 2013.

[107] M. Razaviyayn, M. Sanjabi, and Z.-Q. Luo, "A stochastic successive minimization method for nonsmooth nonconvex optimization with applications to transceiver design in wireless communication networks," *CoRR*, vol. abs/1307.4457, 2013.

[108] A. Nemirovski and A. Shapiro, "Convex approximations of chance constrained programs," *SIAM J. Optim.*, vol. 17, no. 4, pp. 969–996, 2006.

[109] R. T. Rockafellar and S. Uryasev, "Optimization of conditional value-at-risk," *Journal of risk*, vol. 2, pp. 21–42, 2000.

[110] J. A. Bondy and U. S. R. Murty, *Graph theory with applications*. Macmillan London, 1976, vol. 290.

[111] N. Jindal, "MIMO broadcast channels with finite-rate feedback," *IEEE Trans. Inf. Theory*, vol. 52, no. 11, pp. 5045–5060, Nov. 2006.

[112] A. Adhikary, J. Nam, J.-Y. Ahn, and G. Caire, "Joint spatial division and multiplexing: the large-scale array regime," *IEEE Trans. Inf. Theory*, vol. 59, no. 10, pp. 6441–6463, Oct. 2013.

[113] R. Horst and N. V. Thoai, "DC programming: overview," *J. Optimiz. Theory Appl.*, vol. 103, no. 1, pp. 1–43, Oct. 1999.

[114] Y. Nesterov, A. Nemirovskii, and Y. Ye, *Interior-point polynomial algorithms in convex programming*. SIAM, 1994, vol. 13.

[115] B. Nosrat-Makouei, J. G. Andrews, and R. W. Heath, "MIMO interference alignment over correlated channels with imperfect CSI," *IEEE Trans. Signal Process.*, vol. 59, no. 6, pp. 2783–2794, 2011.

[116] E. Delage and Y. Ye, "Distributionally robust optimization under moment uncertainty with application to data-driven problems," *Oper. Res.*, vol. 58, no. 3, pp. 595–612, May-Jun. 2010.

[117] P.-A. Absil, R. Mahony, and R. Sepulchre, *Optimization algorithms on matrix manifolds*. Princeton University Press, 2009.

[118] S. Zhou, M. Zhao, X. Xu, J. Wang, and Y. Yao, "Distributed wireless communication system: a new architecture for future public wireless access," *IEEE Commun. Mag.*, vol. 41, no. 3, pp. 108–113, 2003.

[119] R. Zakhour and S. V. Hanly, "Base station cooperation on the downlink: Large system analysis," *IEEE Trans. Inf. Theory*, vol. 58, no. 4, pp. 2079–2106, Apr. 2012.

[120] J. F. Sturm, "Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones," *Optim. Methods Softw.*, vol. 11, no. 1-4, pp. 625–653, 1999.

[121] K.-C. Toh, M. J. Todd, and R. H. Tütüncü, "SDPT3—a MATLAB software package for semidefinite programming, version 1.3," *Optim. Methods Softw.*, vol. 11, no. 1-4, pp. 545–581, 1999.

[122] E. D. Andersen and K. D. Andersen, “The mosek interior point optimizer for linear programming: an implementation of the homogeneous algorithm,” in *High performance optimization*. Springer, 2000, pp. 197–232.

[123] CVX Research, Inc., “CVX: Matlab software for disciplined convex programming, version 2.0 (beta),” 2013. [Online]. Available: http://cvxr.com/cvx/

[124] J. Lofberg, “YALMIP: A toolbox for modeling and optimization in MATLAB,” in *IEEE Int. Symp. Computer-Aided Control Systems Design*, Taipei, Taiwan, R.O.C., Sep. 2004, pp. 284–289.

[125] E. Chu, N. Parikh, A. Domahidi, and S. Boyd, “Code generation for embedded second-order cone programming,” in *Control Conference (ECC), 2013 European*, Jul. 2013, pp. 1547–1552.

[126] E. Bjornson, M. Bengtsson, and B. Ottersten, “Optimal multiuser transmit beamforming: A difficult problem with a simple solution structure [lecture notes],” *IEEE Signal Process. Mag.*, vol. 31, no. 4, pp. 142–148, Jul. 2014.

[127] W.-C. Liao, M. Hong, Y.-F. Liu, and Z.-Q. Luo, “Base station activation and linear transceiver design for optimal resource management in heterogeneous networks,” *IEEE Trans. Signal Process.*, vol. 62, no. 15, pp. 3939–3952, Aug. 2014. [Online]. Available: http://arxiv.org/abs/1309.4138

[128] S. K. Joshi, M. Codreanu, and M. Latva-aho, “Distributed resource allocation for MISO downlink systems via the alternating direction method of multipliers,” *EURASIP J. Wireless Communun. Netw.*, vol. 2014, no. 1, pp. 1–19, Jan. 2014.

[129] T. Yoo and A. Goldsmith, “On the optimality of multiantenna broadcast scheduling using zero-forcing beamforming,” *IEEE J. Sel. Areas Commun.*, vol. 24, no. 3, pp. 528–541, Mar. 2006.

[130] E. Smith, “On the optimal design of continuous processes,” Ph.D. dissertation, Imperial College London (University of London), 1996.

[131] M. C. Grant and S. P. Boyd, “Graph implementations for nonsmooth convex programs,” in *Recent advances in learning and control*. Springer, 2008, pp. 95–110.

[132] Y. Ye, M. J. Todd, and S. Mizuno, "An $\mathcal{O}(\sqrt{n}L)$-iteration homogeneous and self-dual linear programming algorithm," *Math. Oper. Res.*, vol. 19, no. 1, pp. 53–67, 1994.

[133] Y.-F. Liu, Y.-H. Dai, and Z.-Q. Luo, "Coordinated beamforming for MISO interference channel: Complexity analysis and efficient algorithms," *IEEE Trans. Signal Process.*, vol. 59, no. 3, pp. 1142–1157, Mar. 2011.

[134] H. Tuy, "Monotonic optimization: Problems and solution approaches," *SIAM Journal on Optimization*, vol. 11, no. 2, pp. 464–494, 2000.

[135] E. A. Jorswieck, E. G. Larsson, and D. Danev, "Complete characterization of the pareto boundary for the MISO interference channel," *IEEE Trans. Signal Process.*, vol. 56, no. 10, pp. 5292–5296, Oct. 2008.

[136] R. Zhang and S. Cui, "Cooperative interference management with MISO beamforming," *IEEE Trans. Signal Process.*, vol. 58, no. 10, pp. 5450–5458, Oct. 2010.

[137] J. Mattingley and S. Boyd, "CVXGEN: a code generator for embedded convex optimization," *Optimization and Engineering*, vol. 13, no. 1, pp. 1–27, 2012.

[138] T. A. Davis, *Direct Methods for Sparse Linear Systems.* Society for Industrial and Applied Mathematics, 2006. [Online]. Available: http://epubs.siam.org/doi/abs/10.1137/1.9780898718881

[139] E. Chu, B. O'Donoghue, N. Parikh, and S. Boyd, "A primal-dual operator splitting method for conic optimization," *http://www.stanford.edu/ boyd/papers/pdos.html*, 2013. [Online]. Available: http://www.stanford.edu/ boyd/papers/pdos.html

[140] T. Goldstein, B. O'Donoghue, S. Setzer, and R. Baraniuk, "Fast alternating direction optimization methods," *SIAM Journal on Imaging Sciences*, vol. 7, no. 3, pp. 1588–1623, 2014. [Online]. Available: http://dx.doi.org/10.1137/120896219

[141] R. Vanderbei, "Symmetric quasidefinite matrices," *SIAM J. Optim.*, vol. 5, no. 1, pp. 100–113, 1995. [Online]. Available: http://dx.doi.org/10.1137/0805005

[142] N. Parikh and S. Boyd, "Proximal algorithms," *Foundations and trends in optimization*, vol. 1, no. 3, Jan. 2014. [Online]. Available: http://www.stanford.edu/ boyd/papers/

[143] J. Poulson, B. Marker, R. A. van de Geijn, J. R. Hammond, and N. A. Romero, "Elemental: A new framework for distributed memory dense matrix computations," *ACM Trans. Math. Softw.*, vol. 39, no. 2, p. 13, Feb. 2013.

[144] J. Mattingley and S. Boyd, "Real-time convex optimization in signal processing," *IEEE Signal Process. Mag*, vol. 27, no. 3, pp. 50–61, May 2010.

[145] A. Domahidi, E. Chu, and S. Boyd, "ECOS: An SOCP solver for embedded systems," in *Control Conference (ECC), 2013 European*. IEEE, 2013, pp. 3071–3076.

[146] S. Diamond, E. Chu, and S. Boyd, "CVXPY: A Python-embedded modeling language for convex optimization, version 0.2," May 2014.

[147] J. Zhang and J. G. Andrews, "Adaptive spatial intercell interference cancellation in multicell wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 28, no. 9, pp. 1455–1468, Dec. 2010.

[148] C. B. Peel, B. M. Hochwald, and A. L. Swindlehurst, "A vector-perturbation technique for near-capacity multiantenna multiuser communication-part I: channel inversion and regularization," *IEEE Trans. Commun.*, vol. 53, no. 1, pp. 195–202, Jan. 2005.

[149] R. T. Rockafellar, *Convex analysis*. Princeton university press, 1997, vol. 28.

[150] R. T. Rockafellar and R. J.-B. Wets, *Variational analysis*. Springer, 1998, vol. 317.

[151] M. Broadie and P. Glasserman, "Estimating security price derivatives using simulation," *Management Sci.*, vol. 42, no. 2, pp. 269–285, Feb. 1996.

[152] A. Hjørungnes, *Complex-valued matrix derivatives: with applications in signal processing and communications*. Cambridge University Press, 2011.

[153] R. T. Rockafellar and S. Uryasev, "Conditional value-at-risk for general loss distributions," *Journal of Banking and Finance*, vol. 26, no. 7, pp. 1443–1471, 2002.

[154] Z. Hu, L. J. Hong, and L. Zhang, "A smooth monte carlo approach to joint chance-constrained programs," *IIE Transactions*, vol. 45, no. 7, pp. 716–735, 2013.