MIMO Wireless Networks: Modeling and Optimization

Jia Liu

Dissertation submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Electrical Engineering

Y. Thomas Hou, Chairman
R. Michael Buehrer
Jeffrey H. Reed
Hanif D. Sherali
Yi Shi
Yaling Yang

February 4, 2010
Blacksburg, Virgnia

Keywords: MIMO, wireless networks, modeling, optimization, algorithm, cross-layer design

MIMO Wireless Networks: Modeling and Optimization

Jia Liu

ABSTRACT

A critical factor affecting the future prospects of wireless networks for wide-scale deployment is *network capacity*: the end users wish to have their communication experience over wireless networks to be comparable or similar to that for wireline networks. An effective approach to increase network capacity is to increase spectrum efficiency. Such an approach can be achieved by the use of multiple antenna systems (also known as multiple-input multiple-output (MIMO) technology). The benefits of substantial improvements in capacity at no cost of additional spectrum and power have positioned MIMO as one of the breakthrough technologies in modern wireless communications. As expected, research activities on applying MIMO to a variety of wireless networks have soared in recent years.

However, compared with the simple point-to-point MIMO channel, which is relatively well-understood nowadays, network design and performance optimization for MIMO-based wireless networks is considerably more challenging. Many fundamental problems remain unsolved. Due to the complex characteristics of MIMO physical layer technology, it is not only desirable but also necessary to consider models and constraints at multiple layers (e.g., physical, link, and network) jointly. The formulations of these cross-layer problems for MIMO wireless networks, however, are usually mathematically challenging. In this dissertation, we aim to develop some novel algorithmic design and optimization techniques that provide optimal or near-optimal solutions.

Based on network structure, this dissertation is organized into two parts. In the first part, we focus on single-hop MIMO wireless networks, while in the second part, we focus on multi-hop MIMO networks. The main results and contributions of this dissertation are summarized as follows.

Single-hop MIMO Networks. In the first part of this dissertation, we study

three different optimization problems for single-hop MIMO networks. The first problem addresses weighted proportional fair (WPF) scheduling associated with MIMO broadcast channels (Chapter 2). For the WPF scheduling problem in MIMO broadcast channels, we develop two algorithms that can efficiently determine the optimal dirty paper encoding order and power allocation to achieve an optimal WPF performance. To our knowledge, our work is the first that provides solutions to the WPF scheduling problem in MIMO broadcast channels.

Our next problem concerns single-hop MIMO ad hoc networks (Chapter 3), which are quite different from the MIMO broadcast channels studied in the previous chapter. Single-hop MIMO ad hoc networks can be simply described as "multiple one-to-one," as compared with MIMO broadcast channels, which are "one-to-many." Performance optimization for such networks is known to be challenging due to the non-convex mathematical structure. Indeed, these networks can be viewed as the general case of interference channels in network information theory context, for which the capacity region remains unknown even under the two-user case. In this chapter, we treat the co-channel interference in the network as noise. We consider the maximum weighted sum rate problem under the single-carrier setting. We propose a global optimization approach that combines branch-and-bound (BB) and the reformulation-linearization technique (RLT). This technique is guaranteed to find a global optimal solution.

Multi-hop MIMO Networks. In addition to managing resources such as power and scheduling in single-hop networks, routing and end-to-end session rate control need to be considered in multi-hop MIMO networks. Thus, performance optimization problems in multi-hop MIMO networks are more interesting and yet challenging. In Chapter 4, we first consider the problem of jointly optimizing power and bandwidth allocation at each node and multihop/multipath routing in a multi-hop MIMO network that employs orthogonal channels. We show that this problem has some special structure that admits a decomposition into a set of subproblems in its dual domain. Based on this finding, we propose both centralized and distributed optimization algorithms to solve this problem optimally.

In Chapter 5, we relax the orthogonal channel assumption. More specifically, we exploit the advantage of "dirty paper coding" (DPC) to allow multiple links originated from the same node to share the same channel media simultaneously. However, the formulation of cross-layer optimization problem with DPC has a non-convex structure and an exponentially large search space inherent in enumerating DPC's encoding orders. To address these difficulties, we propose an approach to reformulate and convexify the original problem. Based on the reformulated problem, we design an efficient solution procedure by exploiting decomposable dual structure.

One thing in common in Chapters 4 and 5 is that we adopt the classical matrix-based MIMO channel models at the physical layer. Although this approach has its merit, the complex matrix operations in the classical MIMO models may pose a barrier for researchers in networking research community to gain fundamental understanding on MIMO networks. To bridge this gap between communications and networking communities, in Chapter 6, we propose a simple, accurate, and tractable model to enable the networking community to carry out cross-layer research for multi-hop MIMO networks. At the physical layer, we develop an accurate and simple model for MIMO channel capacity computation that captures the essence of spatial multiplexing and transmit power limit without involving complex matrix operations and the water-filling algorithm. At the link layer, we devise a space-time scheduling scheme called order-based interference cancellation (OBIC) that significantly advances the existing zero-forcing beamforming (ZFBF) to handle interference in a multi-hop network setting. The proposed OBIC scheme employs simple algebraic computation on matrix dimensions to simplify ZFBF in a multi-hop network. Finally, we apply both the new physical and link layer models to study a cross-layer optimization problem for a multi-hop MIMO network.

To my father.

Acknowledgements

A Ph.D. degree is not a solitary endeavor. Since the day when I arrived on the beautiful campus of Virginia Tech more than six years ago, I have received so much help from people around me.

First and foremost, I would like to thank Prof. Tom Hou, my advisor, for his constant support, guidance, and encouragement during my Ph.D. study. I would never forget the countless hours he spent with me, teaching me how to conduct pioneering research, polishing my paper-writing skills, and training me to deliver high quality research talks. I thank him for guiding me into this field of MIMO network research, which has led to a number of significant publications and the existence of this dissertation.

I would also like to thank my committee member, Prof. Hanif D. Sherali, who has contributed on optimization theory that is directly related to my Ph.D. work. Throughout the years, we not only establish an excellent collaboration, but also a great friendship. Prof. Sherali is also the best teacher I have ever met. His classes intrigued me so much that I wound up taking all optimization courses that he offers at Virginia Tech. He is truly a role model to me in all aspects.

The thanks also go to the rest of my thesis committee. I want to thank Dr. Yi Shi for his help and friendship. Our numerous lengthy discussions have deepened my understanding of the research problems. Prof. Mike Buehrer has offered me so many valuable insights on wireless communications and MIMO technologies. Prof. Jeff H. Reed also gave me genuine supports and encouragements during my Ph.D. studies. Prof. Yaling Yang shared with me many of her thoughts and experiences on research as well as life in academia.

I also want to thank my current and former colleagues: Sastry Kompella, Tong Liu, Xiaojun Wang, Cunhao Gao, Canming Jiang, Sushant Sharma, and Liguang Xie, for their friendship and support. Without these fellow officemates, life in Blacksburg

would have been less interesting.

Last but not least, I want to thank my family. I am so lucky to be born and raised in an intellectual family. It was my father, a math professor at Zhongshan University, who first discovered my talents in math in my early age and encouraged me to pursue a career in science and engineering. Although he passed away and was not able to see this dissertation, his words keep inspiring me. My mother has made huge sacrifice for me and never complained her loneliness to me all these years while I have been away to pursue my academic career. My sister never hesitated to lend me her helping hands when I needed the most. Finally, I would like to thank my wife, Qian Chen. Her love and support to me are beyond what words can describe. Her companionship makes my life full of joy.

Jia Liu
February 4, 2010

Contents

1 Introduction **1**

1.1 Background and Scope 1

1.2 State of the Art 4

1.3 Summary of Contributions 9

1.3.1 Thrust I: Single-hop MIMO Networks 9

1.3.2 Thrust II: Multi-hop MIMO Networks 11

1.4 Dissertation Outline 15

1.5 General Notation 16

I Single-hop MIMO Wireless Networks **17**

2 Fair Scheduling for MIMO Broadcast Channels **18**

2.1 Introduction 18

2.1.1 Summary of Main Results 20

2.1.2 Chapter Organization 21

2.2 System Model and Problem Formulation 22

2.3 Problem Reformulation and Optimal Corner Point 25

2.3.1 Uplink-Downlink Duality and Problem Reformulation 25

2.3.2 Dual Capacity Region and Optimal Corner Point 28

2.4 Corner Point Optimality Conditions 32

2.4.1 Two-User Case 33

2.4.2 K-User Case 36
2.5 Determining Optimal Decoding Order 39
2.5.1 Iterative Gradient Sorting 39
2.5.2 Convergence and Complexity Analysis 41
2.5.3 Numerical Results 45
2.5.4 Further Discussion 46
2.6 Power Allocation 47
2.6.1 An Efficient Algorithm 48
2.6.2 An Upper Bound of the D-WPF Problem 52
2.6.3 Numerical Results 54
2.7 Chapter Summary 55

3 Capacity of Single-hop MIMO Ad Hoc Networks 57
3.1 Introduction 57
3.2 Network Model and Problem Formulation 59
3.3 Solution Procedure 61
3.3.1 Overview of BB/RLT Method 61
3.3.2 Factorization and Linearization 62
3.3.3 Linear Relaxation to Nonlinear Logarithmic Functions 64
3.3.4 Linearizing the Determinants 64
3.3.5 RLT-Based Relaxation 67
3.3.6 Partitioning Variables and Their Upper and Lower Bounds . . 68
3.3.7 Convergence Speedup Techniques 68
3.4 Numerical Results 69
3.5 Chapter Summary 74

II Multi-hop MIMO Wireless Networks 75

4 Multi-hop MIMO Networks with Orthogonal Channels 76
4.1 Introduction 76

4.2 Network Model 78
4.2.1 MIMO Power Allocation 78
4.2.2 Link Capacity and Bandwidth Allocation 79
4.2.3 Routing 80
4.2.4 Problem Formulation 83
4.3 Problem Decomposition 84
4.4 The Link-Physical Layer Subproblem 86
4.4.1 Computing the Gradients 87
4.4.2 A Polynomial-Time Algorithm for Performing Projection . . . 89
4.5 Solving the Lagrangian Dual Problem 96
4.5.1 A Cutting-Plane Method Based on Outer-Linearization 96
4.5.2 Subgradient-Based Mechanism 98
4.5.3 Recovering Primal Optimal Solution 99
4.6 Distributed Implementation 101
4.7 Numerical Results 104
4.8 Chapter Summary 108

5 Multi-hop MIMO Ad Hoc Networks with Dirty Paper Coding 111
5.1 Introduction 111
5.1.1 Summary of Main Results and Contributions 113
5.1.2 Chapter Organization 114
5.2 Network Model and Problem Formulation 114
5.2.1 DPC-Based Link Layer 116
5.2.2 Network Layer 118
5.2.3 Problem Formulation 119
5.3 Problem Reformulation 120
5.4 A Solution to the Reformulated Problem 124
5.4.1 Dual Decomposition 125
5.4.2 Solving the Master-Dual Problem 127
5.4.3 Solving the Link Layer Subproblem $\Theta_{\text{link}}(\mathbf{u})$ 129

5.5 Numerical Results 138
5.6 Chapter Summary 142

6 A Tractable and Accurate Cross-Layer Model for Multi-hop MIMO Networks 146
6.1 Introduction 146
6.2 A Model for Physical Layer Capacity Computation 149
6.2.1 Why Existing Physical Model for MIMO is Difficult to Use? . 149
6.2.2 A Simple and Accurate Model for MIMO Channel Capacity . 153
6.3 Link Layer Modeling for Multi-hop MIMO Networks 157
6.3.1 Zero-Forcing Beamforming: Benefits and Challenges 157
6.3.2 OBIC: Basic Idea 159
6.3.3 OBIC: A Mathematical Model 164
6.3.4 Performance Comparison between OBIC and the H-S Model . 173
6.4 Application in Multi-hop Ad Hoc Networks 176
6.5 Chapter Summary 185

7 Summary and Future Work 187
7.1 Summary 187
7.2 Future Research Direction 190

A Some Relevant Results for MIMO Networks: A Background 193
A.1 Point-to-Point MIMO Channel 194
A.2 MIMO Broadcast Channel and MIMO Multiple Access Channel . . . 198
A.2.1 MIMO Broadcast Channel 198
A.2.2 MIMO Multiple Access Channel 202
A.2.3 Uplink-Downlink Duality 203

Bibliography 206

List of Tables

2.1 IGS for a 15-user MIMO-BC example. 47

3.1 SNR- and INR-values for a heavily interfered 5-link network (in dB). 70
3.2 SNR- and INR-values for a less interfered 5-link network. 71
3.3 SNR- and INR-values for a 10-link network. 72

4.1 Bandwidth allocation of the 15-node network (×20MHz). 106
4.2 Power allocation in the 15-node ad hoc network (mW). 107

5.1 Power allocation in the 15-node ad hoc network (mW). 140
5.2 Power allocation, encoding order, or bandwidth allocation at N5. . . . 141

6.1 Normalized gap versus the number of antennas. 156
6.2 Enumerating the transitivity relationship. 165
6.3 Optimal node ordering in each time slot of a frame for the six-node network example in Fig. 6.8. 181
6.4 Links' capacities for the six-node network in Fig. 6.8 (in Mbps). . . . 183
6.5 Optimal node ordering in each time slot of a frame. 184

List of Figures

1.1 MIMO networking. 5

2.1 Dirty paper coding for a MIMO-BC with K receiving nodes. 22
2.2 A MIMO-BC and its dual MIMO-MAC. 26
2.3 An example of the capacity region of a two-user dual MIMO-MAC: $\mathbf{H}_1 = [1\ 0.5]$, $\mathbf{H}_2 = [0.5\ 1]$, and $P = 10$. 29
2.4 An example of the capacity region of a three-user dual MIMO-MAC: $\mathbf{H}_1 = [1\ 0.4\ 0.5]$, $\mathbf{H}_2 = [0.5\ 1\ 0.4]$, $\mathbf{H}_3 = [0.4\ 0.5\ 1]$, and $P = 10$. . . . 30
2.5 An example of a two-user dual MIMO-MAC capacity region illustrating the corner point optimality conditions. 33
2.6 An example for iterative gradient sorting. 40
2.7 Network topology of a 15-user MIMO-BC. 46
2.8 Convergence behavior of the CGP algorithm. 55
2.9 Data rates converge to proportional fair status. 56

3.1 Polyhedral outer-approximation $y = \ln x$. 65
3.2 Relationship among BB variables in MSMI. 69
3.3 A 5-link heavily interfered network example. 70
3.4 A 5-link less interfered network example. 71
3.5 A 10-link network example. 73

4.1 Network topology of a 15-node ad hoc network. 109
4.2 Routing and flow rates of session 1 (in Mbps). 109

4.3 Routing and flow rates of session 2 (in Mbps). 109
4.4 Routing and flow rates of session 3 (in Mbps). 109
4.5 Convergence behavior of gradient projection algorithms for link-physical layer subproblem (for $\mathbf{u} = \mathbf{1}$). 110
4.6 Convergence behavior of cutting-plane (centralized) and subgradient (distributed) algorithms for the 15-node ad hoc network example. . . 110

5.1 Visualizing a MIMO ad hoc network as MIMO-BC subsystems. . . . 115
5.2 The link layer model for a MIMO-BC with DPC. 116
5.3 A MIMO-BC and its dual MIMO-MAC. 121
5.4 Network topology of a 15-node MIMO-based ad hoc network with 3 sessions. 144
5.5 Routing directions and flow rates of session 1 (in Mbps). 144
5.6 Routing directions and flow rates of session 2 (in Mbps). 144
5.7 Routing directions and flow rates of session 3 (in Mbps). 144
5.8 Convergence behavior of the centralized algorithm. 145
5.9 Convergence behavior of the distributed algorithm. 145

6.1 A MIMO channel. 150
6.2 The equivalent parallel scalar channels after transformation. 151
6.3 A two-link example. 160
6.4 DoF region of two-link example. 160
6.5 A three-link example. 161
6.6 The flow chart of OBIC scheduling scheme. 162
6.7 Achievable DoF region comparison between OBIC and CiM for the example in Fig. 6.3. 163
6.8 A six-node MIMO ad hoc network. 179
6.9 The optimal end-to-end session rate (in Mbps), routing, and scheduling solutions for session N2 $\rightarrow$ N4. 180

6.10 The optimal end-to-end session rate (in Mbps), routing, and scheduling solutions for session N5 → N6. 181
6.11 The optimal power allocation solution for session N2 → N4. 182
6.12 The optimal power allocation solution for session N5 → N6. 183
6.13 A 50-node 5-session MIMO-based ad hoc network. 185
6.14 Scheduling result on each link. 186

A.1 A matrix representation of spatial channels for a MIMO link. 195
A.2 A MIMO Gaussian broadcast channel. 199
A.3 A three-user broadcast channel example. 200
A.4 A MIMO Gaussian multiple access channel. 203
A.5 Uplink-Downlink Duality 204

Chapter 1

Introduction

1.1 Background and Scope

Fueled by the demand of new civil and military applications, wireless networking research has attracted growing interest in recent years. A critical factor affecting the future prospects of wireless networks for wide-scale deployment and applications is *network capacity*: The end users wish to have their communication experience over wireless networks to be comparable or similar to that for their wireline networks. One approach to increase network capacity is to utilize more communication spectrum. Examples under this approach include classical broadband communications techniques (e.g., OFDM [5, 14, 25, 101, 106, 107, 127, 155], spread spectrum modulation [26,46,120–123,137], and UWB [11,105,126,145,156]), multi-channel multi-radio (MC-MR) wireless networks (e.g., [2, 7, 34, 70, 78, 82, 112, 124]) and software defined radio (SDR) (or cognitive radio (CR))-based networks [59, 65, 103, 125, 135]. However, communication spectrum is a scared resource whose usage involves complicated regulatory issues (i.e., the FCC).

Another important approach to increase network capacity is to increase spectrum efficiency, which is measured by the amount of bit rate that can be carried in each

channel use (i.e., b/s/Hz). The pioneer works by Foschini and Gans [40], and Teletar [146] have shown that much higher spectrum efficiency and capacity gain can be achieved by the use of multiple antenna systems, now known as multiple-input multiple-output (MIMO) technology. The benefits of substantial improvements in capacity at no cost of additional spectrum and power have quickly positioned MIMO as one of the breakthrough technologies in modern wireless communications. As expected, research activities on MIMO have soared in the past decade, particularly at the physical layer.

The fundamental idea in MIMO is the so-called *space-time processing* technique [13,45], which is inherent in the use of multiple spatially-distributed antennas. Under space-time processing, the time dimension is coupled with the spatial dimension in order to achieve spatial multiplexing gain (to boost the capacity of the system) or to achieve spatial diversity gain (to combat multipath fading).

Spatial diversity gain is attained by space-time coding. Two most well-known space-time coding schemes are the space-time block code (STBC) and the space-time trellis code (STTC). As their names suggest, one is block-based and the other is trellis-based. STBC, first discovered by Alamouti [4], was designed originally for 2×2 MIMO. It was later generalized to systems with arbitrary number of antennas by Tarokh [142]. The discovery of STTC was also due to Tarokh [143]. Since their introduction, there has been extensive work aiming at improving the performance of STTC (e.g., [8,22,38,54,90,110,158]) and STBC (e.g., [3,16,41,43,68,69,91,138,142]). So far, the block-based approach appears to be the more dominant one in space-time coding literature due to its lower complexity.

In essence, diversity gain improves transmission reliability by sending coded replica

of the same message. Diversity gain contributes to a gain in SNR, which in turn leads to a capacity gain that scales logarithmically, i.e., the increase of diversity gain diminishes as the number of antennas gets large.

By contrast, multiplexing gain provided by spatial multiplexing schemes directly contributes to a *linear* capacity increase with respect to the number of antennas, a feature that is largely responsible for MIMO's success. Among others, V-BLAST is perhaps the earliest and most well-known spatial multiplexing scheme [39]. Variants of V-BLAST with enhanced decoding techniques can also be found in the literature (e.g., sphere decoding [30] and nulling and cancellation [47,111], just to name a few).

Multiplexing gain and diversity gain are intrinsically connected. In [166], Zheng and Tse showed that there is a fundamental trade-off between these two types of gains. On the other hand, it is also possible to design systems that exploit this trade-off. Methods that maximize the average data rate and provide diversity order guarantee have been reported in the literature [31, 56, 129, 142].

There are other subareas in MIMO physical layer study that have received substantial attention, including antenna selection (e.g., [49,60,104]), limited channel state information (CSI) feedback (e.g., [72, 92–97, 130, 131]), MIMO channel measurement and modeling (e.g., [18, 35, 44, 80, 109, 149]), etc.

However, compared to the research on MIMO physical layer, for which many results are available (see [45, 48] and references therein), research on system design and performance optimization for MIMO-based wireless networks is still in its infancy. Many fundamental problems, particularly for multi-hop ad hoc networks, remain unsolved. Employing MIMO in networking environment is far from trivial. As discussed by Winters [157], a MIMO-based network with each node equipped with M antennas

does not necessarily mean that the network capacity is also increased by M-fold. The potential network capacity gain with the use of MIMO depends on the coordinated mechanisms at the physical, link, and network layers. An improperly designed algorithm could diminish any potential capacity gain from MIMO. As a result, joint optimization across multiple layers is not only desirable, but also necessary.

However, optimization problems with cross-layer formulations for MIMO networks usually involve nonlinear and non-convex structures and are mathematically challenging. As a result, algorithmic designs and theoretical results for these problems are difficult. Previous work in the literature are mostly heuristics without any performance guarantee. In this dissertation, we will develop several efficient algorithms to provide optimal or near-optimal solutions for both single- and multi-hop MIMO networks. These results are important not only for theoretical understanding, but also for providing guidelines to design distributed algorithms and protocols and measure their performances.

1.2 State of the Art

The field of MIMO networking is vast and research topics in this area are rich. In this section, we will offer a scan of the state-of-the-art in this area. We organize major research topics in MIMO networking in Fig. 1.1. In this figure, we highlight those subareas that we have made contributions using shaded boxes.

As shown in Fig. 1.1, every MIMO wireless network can be categorized into two large classes: single-hop and multi-hop. Single-hop MIMO networks can be further classified into three types: MIMO multiple-access channel (MIMO-MAC), MIMO

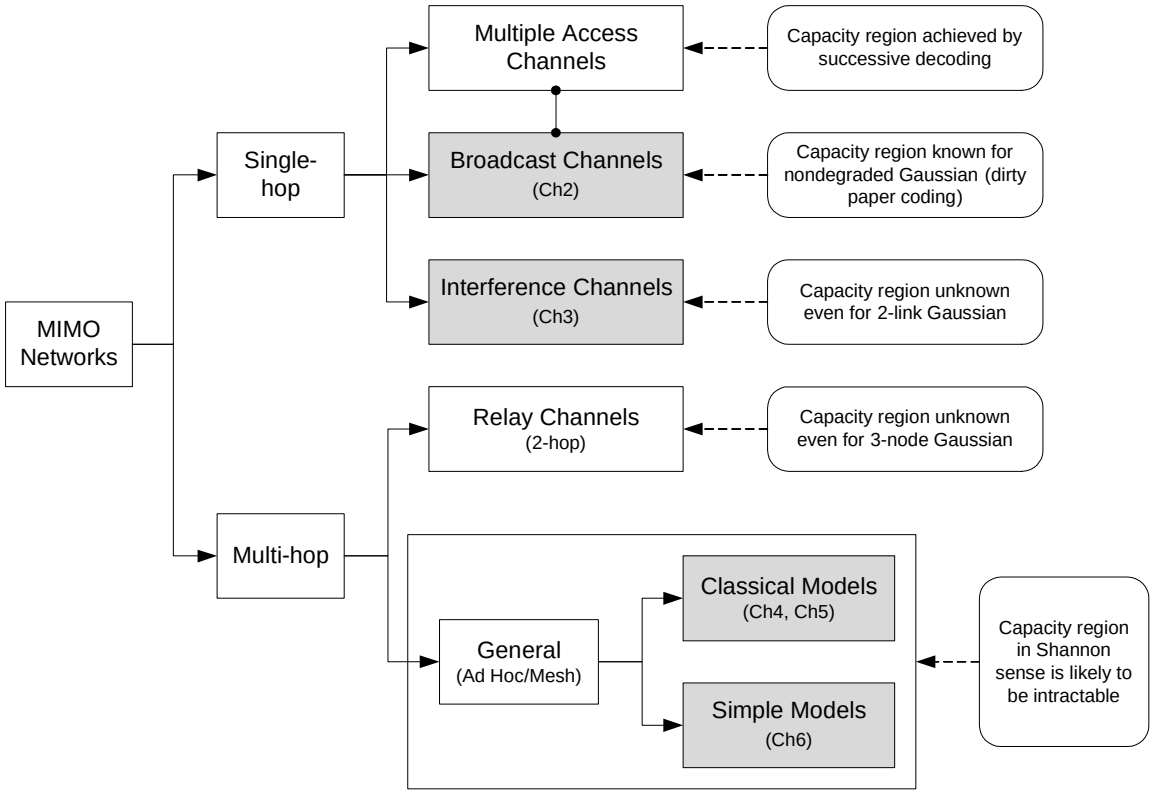


Figure 1.1: MIMO networking.

broadcast channel (MIMO-BC), and MIMO interference channel (MIMO-IC).[1] Simply speaking, MAC, BC, and IC correspond to the following cases: "many-to-one," "one-to-many," and "multiple one-to-one," respectively. Examples of MIMO-MAC and MIMO-BC include the uplink (or reversed link) and the downlink (or the forward link) in multi-antenna cellular networks. Examples of MIMO-IC include MIMO-based single-hop ad hoc networks.

Among these, MIMO-MAC is the most well-understood. The capacity region of MIMO-MAC has been known for many years and the capacity region of MIMO-MAC

[1]For ease of exposition, in this section, the names of different types of networks follow the standard in network information theory [29]. We notice that these networks could be referred to under other names in different literature.

is achieved by successive decoding [29] (also known as successive interference cancellation in communication theory [150]).[2] Moreover, optimization problems over the MIMO-MAC capacity region can exploit the polymatroid structure of the successive decoding rate region [147], which leads to a nice convex property. Although MIMO-MAC is not a main topic in this dissertation, it proves to be a powerful tool for solving other MIMO networking problems.

MIMO-BC's structure is similar to that of MIMO-MAC (with the communication direction being reversed). Surprisingly, the capacity-related problems in MIMO-BC turn out to be much harder than those in MIMO-MAC. The main reason is that MIMO-BC falls into the category of "nondegraded broadcast channels,"[3] for which the capacity region remains unknown even for two-user case. In recent years, a major breakthrough for MIMO-BC is that a scheme called dirty paper coding (DPC) [27] has been shown to be capacity-achieving for Gaussian MIMO-BC [154]. However, unlike MIMO-MAC, the DPC rate region is non-convex, making most optimization problems over the DPC rate region very difficult to solve. Chapter 2 of this dissertation is devoted to the weighted proportional fair scheduling in MIMO-BC, an unsolved problem in the literature.

As seen in Fig. 1.1, the boxes of MAC and BC are connected. This reason is that these two types of networks are intrinsically related. It is known that they are the dual channels to each other. This duality is true not only for MIMO-MAC and

[2]In this dissertation, all capacity regions are in Shannon sense. That is, communication rates in the capacity region are reliable (error probabilities go to zero asymptotically) as the codebook size goes to infinity.

[3]Roughly speaking, a broadcast channel is degraded if the powers of all noises can be ranked. It has long been known that the capacity region of degraded BC is achieved by superposition coding [28]. For MIMO-BC, however, all noise powers are matrix-based, for which a natural order is not well defined.

MIMO-BC but also for all general BC and MAC. Such a duality is called "uplink-downlink duality" in the literature [151, 152, 161]. In this dissertation, this result will play an important role that turns MIMO-MAC into a powerful tool for solving MIMO-BC related problems.

Capacity-related problems in MIMO-IC are far more difficult compared to those in MIMO-MAC and MIMO-BC. Even the capacity region problem of two-user single-antenna IC remains unsolved. Over the years, little progress has been made. The best achievable rate region in IC to date, known as Han-Kobayashi inner bound [55], was discovered nearly thirty years ago. In recent years, major efforts to approach the IC capacity region problem are centered around deriving tighter outer bounds and developing simplified models. For examples, a recent outer bound derived by Etkin *et al.* [37] shows that the Han-Kobayashi inner bound achieves to within 1 bits/s/Hz of the true capacity region. This important result reveals that the Han-Kobayashi inner bound is tight under the high SNR regime. Another major result in recent years is a scheme called "interference alignment" (IA), which is due to Cadambe and Jafar [19]. It is shown that IA achieves the maximum sum of degrees-of-freedom (DoF) in IC (including MIMO case). It should be noted, though, that the DoF characterization for IC is a first-order approximation and is accurate only under the high SNR regime. In contrast to these information-theoretic studies, our focus in this dissertation is not on capacity limit of MIMO-IC. In Chapters 3 (single-carrier setting) and **??** (multiple-carrier setting), we treat MIMO-IC problems in a way that is more realistic and suitable for single-hop ad hoc networks: there is no cooperation among the nodes and each receiving node treats interference as noise.

In Fig. 1.1, we classify multi-hop MIMO networks into two main types. The first

one is two-hop MIMO networks. This type of networks falls into the category of relay channels [29] in network information theory context (or more popularly known as cooperative communications [83] in recent years). As in MIMO-IC, even the capacity region of single-antenna three-node relay channel is unknown.[4] As a result, the area of MIMO relay channels is underexplored and results are limited. Recent notable works in two-hop MIMO relay channels include [76, 108, 141], where the optimal amplify-and-forward strategy is pursued.

The other type of multi-hop networks in Fig. 1.1 is the general multi-hop MIMO ad hoc networks. Needless to say, the capacity region of such networks is the least understood. Indeed, fundamental capacity limits for single-antenna multi-hop network itself is an active research area and lots of problems have no satisfactory solutions, let alone of having more complex MIMO on each node. Due to the randomness of the network topology of multi-hop wireless networks, it is almost certain that the capacity region in Shannon sense is intractable. In this dissertation, we focus on developing optimization algorithms for finite-size multi-hop MIMO networks with considerations of routing, scheduling, and power allocation. My work in this area spans two different approaches: one adopting the classical MIMO channel and signaling models (Chapter 4 with orthogonal channel and Chapter 5 with dirty paper coding); and another simplifying the classical MIMO models to reduce problem complexity, while not losing too much accuracy (Chapter 6).

[4] An outer bound based on max-flow min-cut theorem is known for relay channels [29].

1.3 Summary of Contributions

As mentioned earlier, my work covers both MIMO network settings: single-hop MIMO networks and multi-hop MIMO networks. Accordingly, the main contributions of my research work will be organized into two thrusts.

1.3.1 Thrust I: Single-hop MIMO Networks

For single-hop MIMO networks, we mainly focus on MIMO-BC and single-hop MIMO ad hoc networks (i.e., MIMO-IC). In this part, we investigate the following three important problems.

- **Fair Scheduling for MIMO-BC.** Although there has been tremendous interest in exploring the capacity region of MIMO-BC, fairness, a very important performance measure of multi-user communications systems and networks, has not been addressed for MIMO-BC in the literature. In particular, designing scheduling schemes to achieve the optimal weighted proportional fairness (WPF) over the DPC rate region proves to be difficult and remains an open problem. The difficulty in optimizing the WPF performance over the DPC rate region lies in that it contains two difficult subproblems: 1) a complex combinatorial optimization problem to determine the optimal DPC encoding order (having an exponentially large search space) and 2) a non-convex optimization problem for computing the optimal power allocation to achieve optimal WPF performance. To circumvent the difficulty in the first subproblem, we derive a set of optimality conditions for the optimal decoding order. Based on these optimality conditions, we design an efficient algorithm called *iterative gradient sorting* (IGS) to determine the optimal decoding order. We prove

that the IGS method is guaranteed to find the optimal encoding order in polynomial time under some mild conditions. For the second subproblem, we extend the CGP method from our previous work [85, 86] to determine a local-optimal power allocation solution. By comparing with an upper bound of optimal WPF values, we show that the gap between the solutions obtained from CGP and the true optimal solutions is narrow.

- **Power Control in Single-hop MIMO Ad Hoc Networks.** In network information theory, the capacity region of multiuser Gaussian interference channels (even for single-antenna case) is a well-known open problem [29]. In this work, we consider the problem of maximizing the total mutual information of a multiuser MIMO Gaussian interference channel, where interference is treated as noise. This is a well-known challenging problem due to its non-convex mathematical structure. In this work, we investigate this problem from a global optimization perspective. We solve this problem by proposing a new and powerful global optimization method using a *branch-and-bound* (BB) framework, coupled with a novel *reformulation linearization technique* (RLT). Our main contributions include the mathematical developments of the BB/RLT solution procedure to solve the problem of finding the maximum sum of mutual information (MSMI) for multiuser MIMO systems, and its convergence speedup techniques. Specifically, we derive tight upper and lower bounds for each potential partitioning variable used for the problem. Each nonlinear term is relaxed into a set of linear constraints based on these upper and lower bounds. As a result, the original nonlinear problem is relaxed to a linear program, which can be solved efficiently. We also utilize a polyhedral outer approximation method to accurately approximate the logarithmic function. During

each iteration of the branch-and-bound procedure, we propose a variable selection policy based not only on the relaxation error, but also on the relative significance of the variables in our problem. We show that our proposed method guarantees the finding of a global optimal solution.

1.3.2 Thrust II: Multi-hop MIMO Networks

In our second thrust, we consider multi-hop MIMO networks, which not only involve power allocation/control at the physical layer, but also routing/flow control at the network layer. Specifically, we investigate the following three problems.

- **Multi-hop MIMO Networks with Orthogonal Channels.** In this work, we turn our attention to multi-hop MIMO networks. We consider the problem of jointly optimizing power and bandwidth allocation at each node and multi-hop/multipath routing in a MIMO-based mesh networks where links operate on orthogonal channels. The goal is to support a set of user communication sessions such that some network utility function is maximized. However, to achieve high capacity for such networks, many challenging problems must be addressed. One problem is how to determine optimal power allocation at each transmitting node, optimal bandwidth allocation for each transmission, and optimal flow routing for the network. This problem is considerably more challenging than that for conventional single antenna-based wireless ad hoc networks. This is because, compared to the simple scalar channels in the single antenna case, power allocations are now performed over complex matrix channels. Also, compared to single-user MIMO systems, power allocation for multiple outgoing links at a node has to be jointly considered.

For this problem, we show that it has some special structure that allows us to decompose the original problem into a set of subproblems in its dual domain. Specifically, our solution procedure first decouples the dual problem into a network layer subproblem and a link-physical layer subproblem. For the dual problem, we propose two strategies, i.e., a cutting-plane method based on outer-linearization and the subgradient-based scheme. Our proposed cutting-plane method allows an easy recovery of primal feasible and optimal solutions. Finally, based on the subgradient-based approach, we design a distributed algorithm that achieves the same optimal solution as that of the centralized algorithm. We show that the excess link capacity of each link can be used for message exchange in our distributed algorithm.

- **Multi-hop MIMO Networks with Dirty Paper Coding.** It is known that "dirty paper coding" (DPC) is the optimal transmission strategy for MIMO-BC in the sense that the DPC rate region coincides with the broadcast channel's capacity region. However, this fundamental information-theoretic result is still not adequately exposed to the networking community.

  However, applying DPC in MIMO networks is far from trivial because DPC brings unique changes and challenges in network design philosophy. First, the potential network capacity gain with the use of DPC and MIMO depends on coordinated mechanisms at the physical, link, and network layers. An improperly designed algorithm could diminish any potential capacity gain from DPC and MIMO. Second, due to the interferences induced by simultaneous transmissions to different users, the capacity region of a MIMO-BC with DPC is non-convex with respect to the transmit power variables. The non-convexity of DPC rate region imposes much difficulty on designing efficient algorithms. Third, for a node in the network

having, say, K users, there exist $K!$ possible encoding orders. Different encoding orders coupled with different power allocation will affect the shape and size of the capacity region. Even the optimization for a single MIMO-BC with K users is itself a challenging combinatorial problem, not to mention the cross-layer design in a networking environment with *multiple* MIMO-BC.

Our contributions in this work are the following. First, we show that the cross-layer optimization problem with DPC is a non-convex problem. To address non-convexity, we propose to reformulate and convexify the original problem by exploiting uplink-downlink duality. We show that the reformulated problem is convex with respect to the input covariance matrices under the dual MIMO multiple access channel. Second, for the reformulated problem, we propose an efficient solution procedure based on dual decomposition approach. We show that the combinatorial physical layer subproblem in the dual domain can be solved without enumerating all possible encoding orders, thus paving the way to efficiently solve the challenging physical layer subproblem. Finally, we conduct extensive numerical study to analyze the different physical layer behavior resulting from DPC affect the routing and flow control decision at the network layer. Our numerical results show that, by judiciously managing radio resources, DPC can achieve significant performance gain compared with conventional transmission schemes.

- **Tractable and Accurate Modeling of Multi-hop MIMO Networks.** Although there have been extensive studies on MIMO at the physical layer for point-to-point and cellular communications (see, e.g., [13] for an overview), fundamental understanding and results on MIMO in multi-hop networks remain limited, particularly from a cross-layer perspective. This stagnation is mainly due to the lack of

an accurate and more importantly, tractable model that is amenable for analysis by networking researchers. The goal of this work is to fill this void. We want to construct a model for MIMO that is both *tractable* and *accurate* for cross-layer optimization. Our main contributions are as follows.

- At the physical layer, we devise a simple model for computing MIMO channel capacity. This model captures the essence of both spatial multiplexing and transmit power constraint. More importantly, this model does not require complex matrices computation and complicated water-filling process (which does not admit a close-form solution). We show that the gap between our proposed model and the exact capacity model is negligible.
- At the link layer, we construct a model that takes into account of interference nulling/supression by exploiting zero-forcing beamforming (ZFBF). More specifically, we propose a space-time scheduling scheme called OBIC (abbreviation of order-based interference cancellation). The proposed OBIC employs simple algebraic computation on matrix dimensions to model ZFBF in a multi-hop network. Moreover, by carefully arranging the cancellation order among the nodes, OBIC does not waste unnecessary DoF resources on interference mitigation, thus offering superior throughput performance than those in [12, 53].
- As an application, we use the proposed new models to study a cross-layer utility maximization problem for multi-hop MIMO networks. We show that the resulting optimization problem no longer involves complex matrix variables and operations. Further, the formulated problem shares a lot of similarities

with those cross-layer optimization problems under single-antenna ad hoc networks, which have been actively studied in recent years. This suggests that new solutions to multi-hop MIMO networks may be developed by drawing upon the experiences gained for single-antenna ad hoc networks. Moreover, the scalar-based problem formulation makes it possible to employ many existing efficient optimization tools. For example, the problem we study in this work is solved by CPLEX integer programming solver.

1.4 Dissertation Outline

The remainder of this dissertation is organized as follows. We divide this dissertation into two parts, corresponding to single-hop and multi-hop MIMO networks, respectively. Part I includes Chapters 2 and 3; and Part II includes Chapters 4, 5, and 6. In Chapter 2, we discuss the weighted proportional fairness scheduling for MIMO Gaussian broadcast channels. Chapter 3 investigates the problem of maximizing the sum rate of single-hop MIMO ad hoc networks. In Chapter 4, we investigate the cross-layer optimization problem of routing, power allocation, and bandwidth allocation for multi-hop MIMO ad hoc networks under orthogonal channels. Chapter 5 addresses the cross-layer optimization problem of routing and power allocation for multi-hop MIMO ad hoc networks with DPC. In Chapter 6, we propose a simple, tractable, and accurate model for multi-hop MIMO ad hoc networks. Chapter 7 concludes this dissertation and outlines some future research directions. In order for this dissertation to be self-contained, Appendix A summarizes some key information-theoretic results that will be frequently used throughout this dissertation, including point-to-point MIMO channels, MIMO-BC, and MIMO-MAC.

1.5 General Notation

In this section, we first introduce the following general notation for matrices, vectors, and complex scalars in this dissertation. We use boldface to denote matrices and vectors. For a matrix $\mathbf{A}$, $\mathbf{A}^\dagger$ denotes the conjugate transpose. $\mathrm{Tr}\{\mathbf{A}\}$ denotes the trace of $\mathbf{A}$. $\mathrm{Diag}\{\mathbf{A}_1, \ldots, \mathbf{A}_n\}$ represents the block diagonal matrix with matrices $\mathbf{A}_1, \ldots, \mathbf{A}_n$ on its main diagonal. We let $\mathbf{I}$ denote the identity matrix with dimension determined from context. $\mathbf{A} \succeq 0$ represents that $\mathbf{A}$ is Hermitian and positive semidefinite (PSD). $\mathbf{1}$ and $\mathbf{0}$ denote vectors whose elements are all ones and zeros, respectively, and their dimensions are determined from context. $(\mathbf{v})_m$ represents the m^{th} entry of vector $\mathbf{v}$. For a real vector $\mathbf{v}$ and a real matrix $\mathbf{A}$, $\mathbf{v} \geq \mathbf{0}$ and $\mathbf{A} \geq \mathbf{0}$ mean that all entries in $\mathbf{v}$ and $\mathbf{A}$ are nonnegative, respectively. We let $\mathbf{e}_i$ be the unit column vector where the ith entry is 1 and all other entries are 0. The dimension of $\mathbf{e}_i$ is determined from context as well. The operator "$\langle \cdot, \cdot \rangle$" represents vector or matrix inner product operation.

Part I

Single-hop MIMO Wireless Networks

Chapter 2

Fair Scheduling for MIMO Broadcast Channels

2.1 Introduction

A major recent milestone in network information theory is the triumph in finding the capacity limit of Gaussian MIMO broadcast channels (MIMO-BC), a well-known difficult problem that had plagued researchers for more than a decade. Solving the MIMO-BC capacity region problem is important because broadcast is a fundamental nature of wireless communications and MIMO-BC is the most general information-theoretic formulation for modeling this key property [57]. For example, a MIMO-BC can be viewed as the downlink of all infrastructure-based wireless networks (e.g., cellular networks, wireless LANs, etc) or a portion of an ad hoc network where a node transmitting to its one-hop neighbors. By allowing co-channel interference under DPC, the broadcast nature can be used to simultaneously transmit *different* messages to K receiving nodes at high rates. This is in stark contrast to the conventional broadcasting where the same message is received at each receiving node, or multiplexing schemes where the transmission media is divided into orthogonal sub-channels (in frequency, time, or code domains) in order to carry multiple messages. Thus, the

success of DPC in MIMO-BC opens up a new paradigm and many opportunities for designing new scheduling algorithms at the link layer.

While DPC is known to achieve the entire capacity region of MIMO-BC, most scheduling optimization problems over the DPC rate region turn out to be very difficult. So far, the only solvable scheduling problem under DPC is the so-called maximum weighted sum rate (MWSR) problem [73, 85, 153, 160]. The MWSR problem naturally arises from designing a class of schedulers known as "maximum weight matching" (MWM), which has been shown to be throughput optimal (see, e.g., [113, 114, 144]). However, it is equally well-known that the throughput optimality of these MWM schedulers is achieved at the expense of fairness. The unfairness of the MWM schedulers usually results in high queuing delay, diminishing the otherwise achieved throughput gain. The deficiency of the MWM scheduling motivates us to study a more meaningful performance metric under DPC – *weighted proportional fairness* (WPF) [75], which strikes a balance between fairness and throughput. As we mentioned earlier, since MWSR is the only solvable problem under DPC, the WPF problem is, not surprisingly, an open problem.

The reason that the WPF problem remains unsolved comes directly from the challenges inherent in its problem structure. In the literature, the proportional fairness of a user i is defined as $\log R_i$ [75], where R_i represents user i's data rate. Thus, for a K-user MIMO-BC, the WPF objective function can be written as $\sum_{i=1}^{K} w_i \log R_i$, where $w_i > 0$ is some given weight for user i. Despite the simplicity of the WPF objective function, we have two main challenges in solving the WPF problem. First, unlike the MWSR problem, the *nonlinearity* of the WPF objective function makes it very difficult to determine the optimal encoding order – a unique problem for DPC.

Without any smart algorithm, one has to enumerate and compare all $K!$ encoding orders. Even worse, in some cases, the optimal solution is achieved by time-sharing between multiple ordering. In such cases, it is impossible to find the optimal solution even if one manages to enumerate and compare all encoding orders. Second, even after we have the knowledge of the optimal encoding order(or orders), the WPF problem remains an intractable non-convex structure. This is again due to the WPF objective function, which is in the form of sum of weighted logs.

In this chapter, we tackle the aforementioned challenges by designing WPF scheduling algorithms for MIMO-BC. To our knowledge, our work is the first one that provides a solution to this important problem.

2.1.1 Summary of Main Results

Our efforts include of the following two steps: 1) determine the optimal DPC encoding order; and 2) given the optimal encoding order, compute the optimal power allocation to maximize the WPF fairness. Our approaches and contributions are summarized as follows:

1. To determine the optimal encoding order, we first develop a set of necessary and sufficient *optimality conditions* by exploiting the uplink-downlink duality [151, 152, 161] and the geometrical structure of the dual MIMO-MAC capacity region. For the case where the optimal WPF solution is achieved by time-sharing between multiple orders, we derive a set of closed-form expressions that can be used to compute the optimal WPF rates.

2. Based on the optimality conditions, we design an efficient algorithm called iterative gradient sorting (IGS) to determine the optimal decoding order in the

dual MIMO-MAC by iteratively sorting the gradient entries and moving across the corner points. We prove that under some mild conditions, our proposed IGS method is guaranteed to find the optimal encoding order with at most K rounds of gradient sorting for a K-user MIMO-BC. Moreover, numerical results show that even for large MIMO-BC systems, the optimal encoding order can be determined in a few rounds of gradient sorting.

3. To address non-convexity in the power allocation problem, we design an efficient algorithm based on conjugate gradient projection (CGP). To evaluate the performance of our proposed CGP method, we further derive an upper bound for the optimal WPF values and compare the solution obtained from CGP with this upper bound. Our results show that the gap between the CGP solution and the upper bound is narrow, implying that the performance gap with respect to the true optimal solution is even smaller.

2.1.2 Chapter Organization

The remainder of this chapter is organized as follows. In Section 2.2, we present the network model and the WPF problem formulation under DPC. Section 2.3 reformulates the WPF problem by using uplink-downlink duality and introduce the notion of optimal corner point. In Section 2.4, we derive the optimality conditions for the optimal corner point and discuss its geometrical insights. Section 2.5 presents an efficient algorithm called IGS to determine the optimal decoding order in the dual MIMO-MAC based on the corner point optimality conditions we derive. In Section 2.6, we investigate how to compute optimal power allocation after the optimal encoding order is determined by the IGS algorithm. Section 2.7 concludes this chapter.

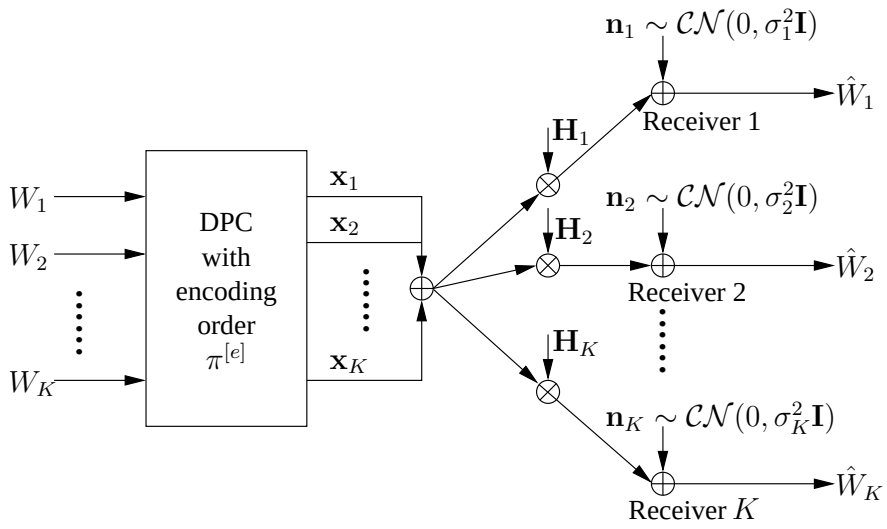


Figure 2.1: Dirty paper coding for a MIMO-BC with K receiving nodes.

2.2 System Model and Problem Formulation

We start by formally characterizing the DPC rate region in mathematical form. Then, we present the formulation of the WPF problem over the DPC rate region.

The DPC Rate Region: Suppose that a MIMO-BC has K receiving nodes, as shown in Fig. 2.1. The transmitting node has K independent messages $W_1, \ldots, W_K$ intended for these K receiving nodes. The transmitter picks a codeword $\mathbf{x}_i$ for each message W_i and broadcasts the sum of these codewords to all receiving nodes. Each receiving node decodes its own message independently and the decoding output is denoted as $\hat{W}_i$. An error occurs when $\hat{W}_i \neq W_i$. As is standard in information theory, a rate vector $\mathbf{R} = [R_1, \ldots, R_K]^T$ is said to be achievable if there exists a sequence of $(n, 2^{nR_1}, \ldots, 2^{nR_K})_{\text{BC}}$ codebooks for which the average probability of error $P_e \triangleq \Pr\{\hat{W}_i \neq W_i \text{ for some i}\} \to 0$ as the code length $n \to \infty$.

For ease of exposition, we assume that each user is equipped with n_r antennas,

and the transmitter has n_t antennas.[1] Denote the channel gain matrix for user i as $\mathbf{H}_i \in \mathbb{C}^{n_r \times n_t}$. For convenience, we denote $\mathbf{H} \triangleq \{\mathbf{H}_i : i = 1, \ldots, K\}$ the collection of all channel gain matrices. Assume that the noise $\mathbf{n}_i$ seen at each receiving node is circularly symmetry white Gaussian distributed, i.e., $\mathbf{n}_i \sim \mathcal{CN}(0, \sigma_i^2 \mathbf{I})$, where σ_i^2 represents the variance.

Let $\pi^{[e]}$ be some encoding order of the set $\{1, 2, \ldots, K\}$: $\pi^{[e]}(i) = j$ represents that the i-th position in $\pi^{[e]}$ is user j. Under DPC, since the transmitter encodes each receiver's message one by one, when encoding receiver $\pi^{[e]}(i)$, the interference from those codewords already encoded for receivers $\pi^{[e]}(1), \ldots, \pi^{[e]}(i-1)$ is known to the transmitter. Therefore, the transmitter can apply DPC encoding technique [27] to pre-subtract these known interference. As a result, receiver $\pi^{[e]}(i)$ sees no interference from users encoded before him. On the other hand, since the transmitter do not have the knowledge of interference caused by those codewords for subsequent encoded receivers, receiver $\pi^{[e]}(i)$ is interfered by all users encoded after him. Thus, the achievable rate of user $\pi^{[e]}(i)$ (assuming unit bandwidth) can be computed as [151]

$$R^{\mathrm{DPC}}_{\pi^{[e]}(i)}(\mathbf{\Gamma}) = \log \left| \mathbf{I} + \mathbf{H}_{\pi^{[e]}(i)} \mathbf{\Gamma}_{\pi^{[e]}(i)} \mathbf{H}^{\dagger}_{\pi^{[e]}(i)} \mathbf{Z}^{-1}_{\pi^{[e]}(i)} \right|, \tag{2.2.1}$$

where the matrix $\mathbf{Z}_{\pi^{[e]}(i)}$ is defined as

$$\mathbf{Z}_{\pi^{[e]}(i)} \triangleq \sigma^2_{\pi^{[e]}(i)} \mathbf{I} + \sum_{j=i+1}^{K} \mathbf{H}_{\pi^{[e]}(j)} \mathbf{\Gamma}_{\pi^{[e]}(j)} \mathbf{H}^{\dagger}_{\pi^{[e]}(j)}, \tag{2.2.2}$$

representing the power of aggregated interference plus noise at node $\pi^{[e]}(i)$. In (2.2.1) and (2.2.2), $\mathbf{\Gamma}_i \in \mathbb{C}^{n_t \times n_t}$, $i = 1, \ldots, K$, are the power allocation matrix variables, defined as $\mathbf{\Gamma}_i = \mathbb{E}\{\mathbf{x}_i \mathbf{x}_i^{\dagger}\}$, where $\mathbb{E}\{\cdot\}$ denotes expectation. For convenience, we

[1] We note that our analysis is general and all results in this chapter can be straightforwardly extended to cases where users have unequal number of antennas.

denote $\mathbf{\Gamma} \triangleq \{\mathbf{\Gamma}_i : i = 1, \ldots, K\}$ the collection of all power allocation matrix variables. We point out that these $\mathbf{W}_{\pi^{[e]}(\cdot)}$ terms make the achievable rate in (2.2.1) *neither convex nor concave* with respect to the power allocation variables $\mathbf{\Gamma}$.

Suppose that the maximum transmit power of this K-user MIMO is P. Due to this finite transmit power limit, we have the following power allocation constraint

$$\sum_{i=1}^{K} \mathrm{Tr}\{\mathbf{\Gamma}_i\} \leq P. \tag{2.2.3}$$

Let $\mathbf{R}^{\mathrm{DPC}}(\pi^{[e]}, \mathbf{\Gamma}) \triangleq [R^{\mathrm{DPC}}_{\pi^{[e]}(1)}(\mathbf{\Gamma}), \ldots, R^{\mathrm{DPC}}_{\pi^{[e]}(K)}(\mathbf{\Gamma})]^T$ be the achievable rate vector under an encoding order $\pi^{[e]}$ and some power allocation $\mathbf{\Gamma}$ at node n. Then, the DPC rate region $\mathcal{C}_{\mathrm{DPC}}(P, \mathbf{H})$ is defined as the convex hull of the union of all rates vectors over all $\mathbf{\Gamma}$ satisfying the power allocation constraint (2.2.3) and over all $K!$ possible encoding orders, i.e.,

$$\mathcal{C}_{\mathrm{DPC}}(P, \mathbf{H}) \triangleq Conv\left(\cup_{\pi^{[e]}} \left\{\mathbf{R}^{\mathrm{DPC}}(\pi^{[e]}, \mathbf{\Gamma}) : \sum_{i=1}^{K} \mathrm{Tr}\{\mathbf{\Gamma}_i\} \leq P,\ \mathbf{\Gamma}_i \succeq 0, \forall i\right\}\right), \tag{2.2.4}$$

where $Conv(\cdot)$ represents the convex hull operation. It has been shown in [154] that the achievable rate region in (2.2.4) coincides with the entire capacity region of the MIMO-BC, meaning that DPC is the optimal transmission scheme for MIMO-BC.

The WPF Problem over the DPC Rate Region: Recall that the WPF objective function is in the form of sum of logs. Thus, by putting all the rate expressions and constraints described earlier, we can write the WPF problem over the DPC rate region as follows:

$$\begin{array}{lll} \underline{\textbf{WPF:}} & \text{Maximize} & \sum_{i=1}^{K} w_i \log R^{\mathrm{DPC}}_{\pi^{[e]}(i)}(\mathbf{\Gamma}) \\ & \text{subject to} & \mathbf{R}^{\mathrm{DPC}}(\pi^{[e]}, \mathbf{\Gamma}) \in \mathcal{C}_{\mathrm{DPC}}(P, \mathbf{H}) \end{array} \tag{2.2.5}$$

where w_i is the weight of user i.

Two remarks for the WPF problem are in order. First, as we pointed out, the DPC rate functions $R^{\mathrm{DPC}}_{\pi^{[e]}(\cdot)}(\mathbf{\Gamma})$ in (2.2.1) are non-convex with respect to the power allocation matrix variables $\mathbf{\Gamma}$. This non-convexity creates a lot of difficulty in power allocation optimization. Second, the WPF problem is over all $K!$ possible encoding orders. This means that the search space of WPF grows *exponentially* as K gets large. Indeed, in the rest of this chapter, our efforts are centered around these two challenges. Our first step towards solving the WPF problem is to reformulate WPF so that the new formulation is more amenable for analysis. Our reformulation hinges on a result called uplink-downlink duality, which will be described in detail in the next section.

2.3 Problem Reformulation and Optimal Corner Point

In this section, we will first present how to reformulate the WPF problem by using uplink-downlink duality. After we re-cast the WPF problem in the dual domain, we will reveal some novel geometrical insights of the dual uplink capacity region and establish the notion of *optimal corner point*.

2.3.1 Uplink-Downlink Duality and Problem Reformulation

Simply speaking, uplink-downlink duality [151, 152, 161] says that for every MIMO-BC, there exists a so-called dual MIMO multiple access channel (MIMO-MAC) that achieves the same capacity region as that of the MIMO-BC. Fig. 2.2 shows how to construct a dual MIMO-MAC for a given K-user MIMO-BC. First, we change the receivers in the MIMO-BC into transmitters and change the transmitter into a receiver. Next, we conjugate transpose all the channel gain matrices. Finally, we let

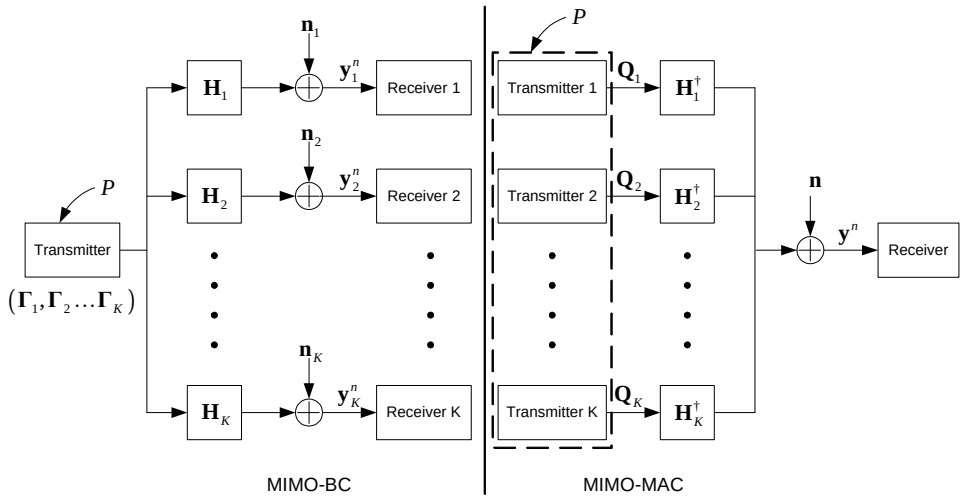


Figure 2.2: A MIMO-BC and its dual MIMO-MAC.

the sum of the maximum powers of the transmitters in the dual MIMO-MAC equal to the maximum transmit power P in the MIMO-BC. As shown in Fig. 2.2, we use $\mathbf{Q}$ to denote the covariance matrices (i.e., the power allocation variables) in the dual MIMO-MAC channels to differentiate from those covariance matrices $\mathbf{\Gamma}$ in the original MIMO-BC.

Denote the capacity region of a dual MIMO-MAC as $\mathcal{C}_{\text{MAC}}(P, \mathbf{H}^\dagger)$. We now restate the uplink-downlink duality in the following lemma and this result was proved in several different ways [151, 152, 161].

Lemma 2.3.1. *The DPC rate region of a MIMO-BC with maximum power constraint P is equal to the capacity region of the dual MIMO-MAC with a sum power constraint P, i.e., $\mathcal{C}_{\text{DPC}}(P, \mathbf{H}) = \mathcal{C}_{\text{MAC}}(P, \mathbf{H}^\dagger)$.*

Lemma 2.3.1 implies that, given a feasible $\mathbf{\Gamma}$, there exists a set of feasible uplink power allocation matrices in the dual MIMO-MAC, denoted by $\mathbf{Q}$, such that $R_i^{\text{MAC}}(\mathbf{Q}) = R_i^{\text{DPC}}(\mathbf{\Gamma})$. The reversed is also true: given a feasible $\mathbf{Q}$ in the dual MIMO-MAC, there exists a set of feasible downlink power allocation matrices in

the original MIMO-BC, denoted by $\mathbf{\Gamma}$, such that $R_i^{\mathrm{DPC}}(\mathbf{\Gamma}) = R_i^{\mathrm{MAC}}(\mathbf{Q})$. It is also shown in [151] that any rate vector in a MIMO-BC with a particular encoding order corresponds to a *reversed* successive decoding order in its dual MIMO-MAC.

Now, consider the following version of the WPF problem in the dual MIMO-MAC, denoted by D-WPF:

$$\begin{array}{rll} \underline{\textbf{D-WPF:}} & \text{Maximize} & \sum_{i=1}^{K} w_i \log R_{\pi^{[d]}(i)}^{\mathrm{MAC}}(\mathbf{Q}) \\ & \text{subject to} & \mathbf{R}_{\pi^{[d]}(i)}^{\mathrm{MAC}}(\mathbf{Q}) \in \mathcal{C}_{\mathrm{MAC}}(P, \mathbf{H}^{\dagger}) \end{array} \tag{2.3.1}$$

where $\pi^{[d]}$ represents a decoding order.

Since $\mathcal{C}_{\mathrm{DPC}}(P, \mathbf{H}) = \mathcal{C}_{\mathrm{MAC}}(P, \mathbf{H}^{\dagger})$, it is clear that D-WPF is equivalent to WPF in the sense that they achieve the same optimal objective value. For simplicity, we drop the superscript "MAC" and simply refer to user i's rate in the dual MIMO-MAC as R_i. Also, we drop the superscript "[d]" and simply refer to a decoding order in the dual MIMO-MAC as π.

The motivation of the reformulation in (2.3.1) is that, as seen in the next subsection, the rate expressions for the dual MIMO-MAC capacity region are *convex* and have the so-called polymatroid structure that we can exploit when solving the WPF problem. This means that, rather than dealing with the unstructured MIMO-BC rate expressions, we can instead analyze the D-WPF problem in the dual MIMO-MAC capacity region, which is much easier due to its convexity and its polymatroid structure. Further, from uplink-downlink duality, once we have the solution $\mathbf{Q}^*$ and the corresponding *successive decoding order* [29] in the dual MIMO-MAC, we can easily recover $\mathbf{\Gamma}^*$ and the optimal encoding order in the original MIMO-BC through the MAC-to-BC mapping [151]. Thus, in the rest of the chapter, we will focus on exploiting the special geometrical structure of the dual MIMO-MAC capacity region for solving WPF.

2.3.2 Dual Capacity Region and Optimal Corner Point

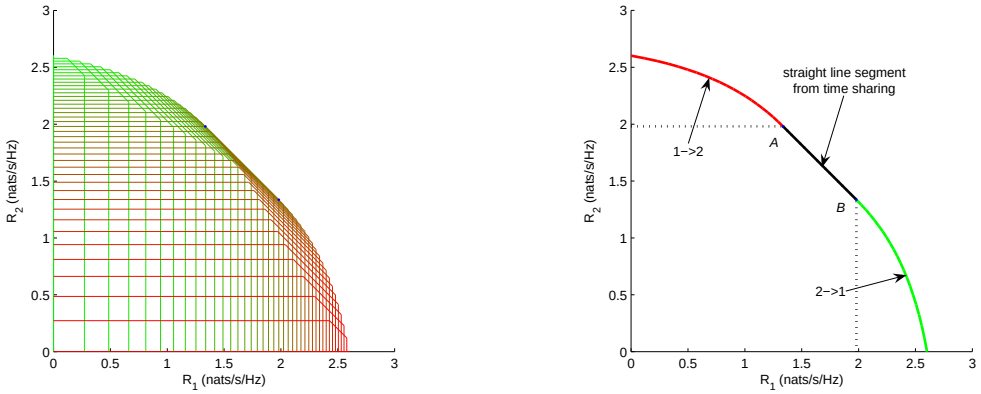
In this subsection, we will first reveal some important geometrical insights for the capacity region of the dual MIMO-MAC. Then, we will establish the notion of optimal corner point, which will play an important role in determining the optimal decoding order in the dual MIMO-MAC.

From [29, Theorem 14.3.5], we can derive the capacity region of a dual MIMO-MAC as in (2.3.2).

$$\mathcal{C}_{\mathrm{MAC}}(P, \mathbf{H}^{\dagger}) = \left\{ (R_1, \ldots, R_K) \;\middle|\; \begin{array}{l} \sum_{i\in\mathcal{S}} R_i \leq \log\left|\mathbf{I} + \sum_{i\in\mathcal{S}} \mathbf{H}_i^{\dagger}\mathbf{Q}_i\mathbf{H}_i\right|, \forall \mathcal{S} \subseteq \{1, \ldots, K\}, \\ \sum_{i=1}^{K} \mathrm{Tr}(\mathbf{Q}_i) \leq P, \mathbf{Q}_i \succeq 0, \ \forall i \end{array} \right\}. \tag{2.3.2}$$

It can be seen from (2.3.2) that, since there are $2^K - 1$ non-empty subsets $\mathcal{S}$ for $\{1, 2, \ldots, K\}$, the capacity region of a K-user dual MIMO-MAC has the same *polymatroid* structure as in conventional multiple access channels (i.e., with $2^K - 1$ sum rate constraints in total [147]). Geometrically, the region defined by (2.3.2) is a union of beveled boxes, each of which has $2^K - 1$ faces and corresponds to a feasible power allocation $\mathbf{Q}$ satisfying the condition in (2.3.2). For example, Fig. 2.3(a) and Fig. 2.4(a) show the capacity region of a two-user and a three-user dual MIMO-MAC, respectively. As indicated earlier, we can see that the regions in Fig. 2.3(a) and Fig. 2.4(a) are formed by a set of beveled boxes. For easier visualization, we plot the boundaries of the respective capacity regions in Fig. 2.3(b) and Fig. 2.4(b).

As shown in Fig. 2.3(b), the boundary of the two-user dual MIMO-MAC capacity region contains two curves that correspond to decoding orders $1 \rightarrow 2$ and $2 \rightarrow 1$, respectively, and a straight line segment representing *time sharing* (i.e., a convex combination $\mu_1 R_1 + \mu_2 R_2$ with $\mu_1 + \mu_2 = 1$) between corner points A and B, both



(a) Union of 2D beveled boxes in the two-user dual MIMO-MAC.

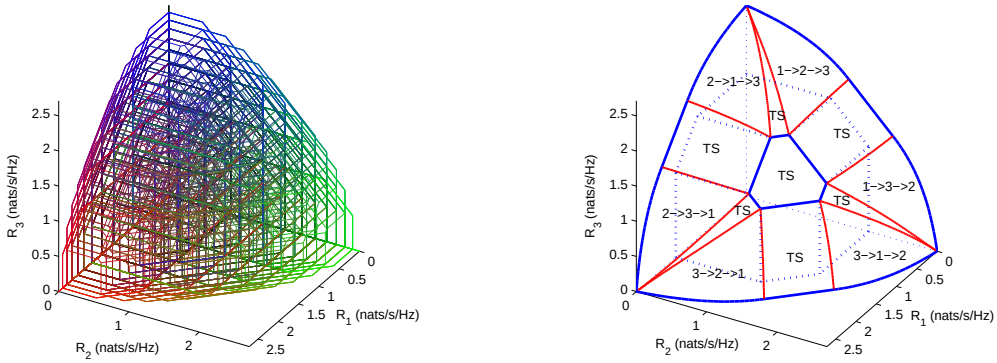
(b) The boundary of the two-user dual MIMO-MAC capacity region.

Figure 2.3: An example of the capacity region of a two-user dual MIMO-MAC: $\mathbf{H}_1 = [1\ 0.5]$, $\mathbf{H}_2 = [0.5\ 1]$, and $P = 10$.

achieving the maximum sum rate (with user 1 and user 2 being decoded first, respectively). Similarly, as shown in Fig. 2.4(b), there are six subregions on the boundary corresponding to $3! = 6$ decoding orders. Also, there exists a hexagon defined by six corner points and six other subregions on the boundary surface that are achieved by time sharing (labeled as "TS" in Fig. 2.4(b)). In Fig. 2.4, the polyhedron (with a polymatroid structure) in dotted line represents the power allocation that achieves the 3! corner point on the boundary.

For a general K-user dual MIMO-MAC, it is easy to see that there are $K!$ corner points in total, corresponding to $K!$ successive decoding orders in the dual MIMO-MAC. Also, we have the following lemma to count the number of time-sharing subregions on the boundary of a K-user dual MIMO-MAC.

Lemma 2.3.2. *For a K-user dual MIMO-MAC, the total number of time-sharing subregions on the boundary is given by $\sum_{i=2}^{K} \binom{K}{i}(K-i+1)!$.*



(a) Union of 3D beveled boxes in the three-user dual MIMO-MAC.

(b) The boundary of the three-user dual MIMO-MAC capacity region.

Figure 2.4: An example of the capacity region of a three-user dual MIMO-MAC: $\mathbf{H}_1 = [1\ 0.4\ 0.5]$, $\mathbf{H}_2 = [0.5\ 1\ 0.4]$, $\mathbf{H}_3 = [0.4\ 0.5\ 1]$, and $P = 10$.

Proof. For a time-sharing subregion involving i users in a decoding order, there are $\binom{K}{i}$ possible combinations in total. Treating those i users as a single element in an ordering, we have $(K-i+1)!$ permutations. Thus, the total number of time-sharing subregions is $\sum_{i=2}^{K}\binom{K}{i}(K-i+1)!$. □

Since $\sum_i w_i \log R_i$ is a monotonically increasing function with respect to R_i's, it is clear that the optimal value of problem (2.3.1) must be achieved on the boundary of the dual MIMO-MAC capacity region. Suppose that the optimal value of $\sum_i w_i \log R_i$ is attained on a subregion corresponding to one particular order, then we call this order as the optimal decoding order. Also, since each order corresponds to one particular corner point, we have the following definition.

Definition 2.3.1 (Optimal Corner Point)**.** If $\pi(\cdot)$ is the optimal decoding order of a dual MIMO-MAC, then the corner point corresponding to $\pi(\cdot)$ is called the optimal corner point.

In Fig. 2.3(b), for example, if the optimal solution is achieved at the red subregion between A and R_2 axis, then the optimal decoding order is $1 \rightarrow 2$ and A is the optimal corner point. It is also possible that the optimal WPF solution is not achieved by a single ordering in a dual MIMO-MAC. In Fig. 2.3(b), for example, if the optimal solution is achieved at the line segment AB, then neither $1 \rightarrow 2$ nor $2 \rightarrow 1$ is optimal. However, if there exists an optimal decoding order, then following lemma shows that this optimal decoding order must be unique.

Lemma 2.3.3 (Uniqueness of Optimal Decoding Order)**.** *If there exists an optimal decoding order in a dual MIMO-MAC, it must be unique.*

The uniqueness of the optimal corner point follows from the fact that the objective of (2.3.1) is strictly concave and the feasible region is convex with respect to R_i's and non-empty, which implies that a unique optimal solution to (2.3.1) exists.

We conclude the discussion on the geometrical structure of a dual MIMO-MAC's capacity region by introducing how to compute the rates at each corner point in the following lemma. This result will play an important role in the next section, where we will derive the corner point optimality condition.

Lemma 2.3.4. *Let $\pi(\cdot)$ be the order corresponding to a corner point A. Let $\{\mathbf{Q}_1^*, \ldots, \mathbf{Q}_K^*\}$ be the power allocation matrices that achieve the corner points. Then, the rates at A can be computed as*

$$R_{\pi(K)}^{(A)} = \log \left| \mathbf{I} + \mathbf{H}_{\pi(K)}^{\dagger} \mathbf{Q}_{\pi(K)}^{*} \mathbf{H}_{\pi(K)} \right| \tag{2.3.3}$$

and

$$R_{\pi(i)}^{(A)} = \log \left| \mathbf{I} + \sum_{j=i}^{K} \mathbf{H}_{\pi(j)}^{\dagger} \mathbf{Q}_{\pi(j)}^{*} \mathbf{H}_{\pi(j)} \right| - \log \left| \mathbf{I} + \sum_{j=i+1}^{K} \mathbf{H}_{\pi(j)}^{\dagger} \mathbf{Q}_{\pi(j)}^{*} \mathbf{H}_{\pi(j)} \right|, \tag{2.3.4}$$

for $i = 1, 2, \ldots, K-1$.

Proof. Let us first examine the last user $\pi(K)$. Since the last user sees no interference in successive decoding, the bound in (2.3.2) for $\pi(K)$ is tight. Thus, we can immediately have

$$R^{(A)}_{\pi(K)} = \log \left| \mathbf{I} + \mathbf{H}^{\dagger}_{\pi(K)} \mathbf{Q}^{*}_{\pi(K)} \mathbf{H}_{\pi(K)} \right|.$$

Next, for the last two users $\pi(K-1)$ and $\pi(K)$, since they see no interference from users decoded before them in successive decoding, the bound in (2.3.2) involving $\pi(K-1)$ and $\pi(K)$ is tight, i.e.,

$$R^{(A)}_{\pi(K-1)} + R^{(A)}_{\pi(K)} = \log \left| \mathbf{I} + \sum_{j=K-1}^{K} \mathbf{H}^{\dagger}_{\pi(j)} \mathbf{Q}^{*}_{\pi(j)} \mathbf{H}_{\pi(j)} \right|.$$

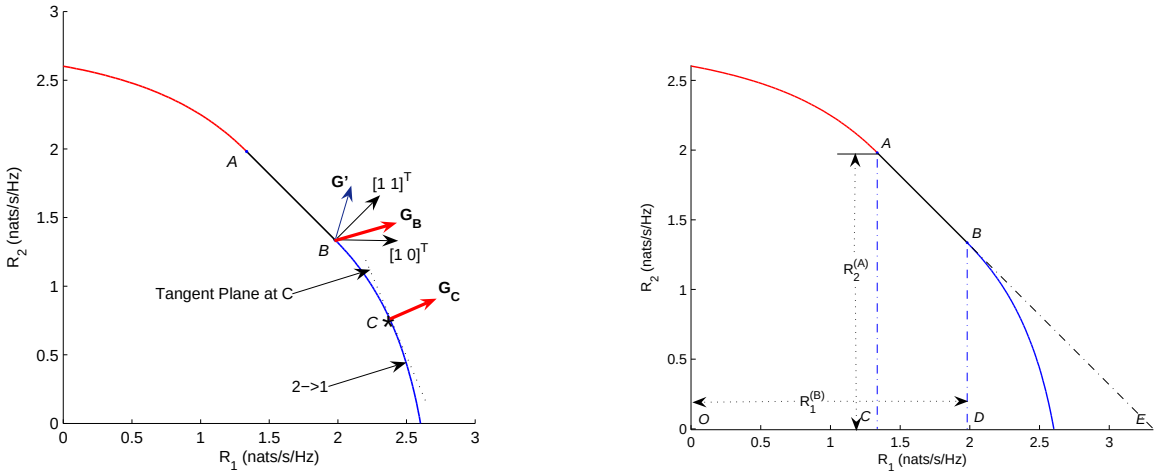
Note that $R^{(A)}_{\pi(K)}$ is already known, we have

$$R^{(A)}_{\pi(K-1)} = \log \left| \mathbf{I} + \sum_{j=K-1}^{K} \mathbf{H}^{\dagger}_{\pi(j)} \mathbf{Q}^{*}_{\pi(j)} \mathbf{H}_{\pi(j)} \right| - \log \left| \mathbf{I} + \mathbf{H}^{\dagger}_{\pi(K)} \mathbf{Q}^{*}_{\pi(K)} \mathbf{H}_{\pi(K)} \right|.$$

The rate computations for the remaining users from $\pi(K-2)$ to $\pi(1)$ follow exactly the same argument as above and it is easy to verify that the rate expressions in (2.3.4) hold. This completes the proof. □

2.4 Corner Point Optimality Conditions

In the previous section, we have shown that for a K-user dual MIMO-MAC, there are $K!$ corner points in its capacity region and one of them may correspond to the optimal decoding order. It is easy to see that as the number of users gets large, finding the optimal decoding order quickly becomes intractable. In order to avoid blindly enumerating and comparing all possible decoding orders, in this section, we will first establish the conditions that can be used to check the optimality for a given corner point. These conditions will serve as the basis for designing algorithms to



(a) Gradients and the cone.

(b) Conditions in (2.4.2) and (2.4.3) cannot be both satisfied at the same time.

Figure 2.5: An example of a two-user dual MIMO-MAC capacity region illustrating the corner point optimality conditions.

determine the optimal corner point. To reveal geometrical insights, we will start with a two-user dual MIMO-MAC. The results will subsequently be generalized to the case of K users.

2.4.1 Two-User Case

In this subsection, we consider the optimal corner point of a two-user dual MIMO-MAC. As indicated earlier, since the log function is monotonically increasing, we must have that the optimal WPF solution is achieved on the boundary. Fig. 2.5(a) shows the boundary of a two-user dual MIMO-MAC.

Note that the gradient of each corner point is in the form of $[\frac{w_1}{R_1}, \ \frac{w_2}{R_2}]$, where R_i's can be computed according to (2.3.3) and (2.3.4). Further, since all weights and rates are positive, all gradients must lie in the positive orthant.

Without loss of generality, suppose that the optimal WPF value is achieved at

point C (as shown in Fig. 2.5(a)) where user 2 is decoded first. This means that corner point B is the optimal corner point. Note that the gradient at each point always points to the increasing objective contours. Therefore, it is easy to see that the gradient at corner point B, denoted as $\mathbf{G}_B$, must be contained in the *cone* formed by vectors $\begin{bmatrix} 1 & 0 \end{bmatrix}^T$ and $\begin{bmatrix} 1 & 1 \end{bmatrix}^T$ as shown in Fig. 2.5(a). Otherwise, if the gradient is outside of the cone, as for $\mathbf{G}'$ in Fig. 2.5(a), then $\mathbf{G}'$ would have led the search of the optimal WPF value to move away from the subregion where user 2 is decoded first, a contradiction.

Now, let $R_1^{(B)}$ and $R_2^{(B)}$ denote the rates at corner point B. The above geometrical fact can be stated as the following linear combination:

$$\mathbf{G}_B = \begin{bmatrix} \frac{w_1}{R_1^{(B)}} \\ \frac{w_2}{R_2^{(B)}} \end{bmatrix} = \alpha_1 \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \alpha_2 \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \tag{2.4.1}$$

where $\alpha_1, \alpha_2 \geq 0$. Solving (2.4.1), we have $\alpha_1 = \frac{w_1}{R_1^{(B)}} - \frac{w_2}{R_2^{(B)}}$ and $\alpha_2 = \frac{w_2}{R_2^{(B)}}$. Since α_1 is non-negative, we must have $\frac{w_1}{R_1^{(B)}} - \frac{w_2}{R_2^{(B)}} \geq 0$. This can be further rewritten as

$$\frac{R_1^{(B)}}{R_2^{(B)}} \leq \frac{w_1}{w_2}. \tag{2.4.2}$$

Similarly, if corner point A is the optimal corner point (corresponding to decoding user 1 first), we have

$$\frac{R_2^{(A)}}{R_1^{(A)}} \leq \frac{w_2}{w_1}. \tag{2.4.3}$$

Now, we have two conditions (2.4.2) and (2.4.3) corresponding to decoding orders $2 \rightarrow 1$ and $1 \rightarrow 2$ being optimal, respectively. Next, we will show that (2.4.2) and (2.4.3) cannot be both satisfied at the same time.

Proposition 2.4.1. *For any two-user dual MIMO-BC, at most one of the corner point optimality conditions in (2.4.2) and (2.4.3) can be satisfied.*

Proof. To show this result, we can extend the line segment AB to cross the R_1 axis at point E, as shown in Fig. 2.5(b). Also, draw two line segments AC and BD, which are perpendicular to the R_1 axis. It is obvious that the lengths $|AC| = R_2^{(A)}$ and $|OD| = R_1^{(B)}$. Now, by contradiction, suppose that it is possible to simultaneously satisfy (2.4.2) and (2.4.3). Since (2.4.2) and (2.4.3) can be rewritten as

$$R_1^{(B)} \leq \frac{w_1}{w_1 + w_2} R_{\text{Sum}}, \quad \text{and} \quad R_2^{(A)} \leq \frac{w_2}{w_1 + w_2} R_{\text{Sum}},$$

where R_{Sum} denotes the maximum sum rate. It then follows that

$$R_1^{(B)} + R_2^{(A)} \leq R_{\text{Sum}}. \tag{2.4.4}$$

From Fig. 2.5(b), we see that $R_{\text{Sum}} = |OD| + |DE| = R_1^{(B)} + |DE|$. Also, note that $|DE| = |BD|$ and $|BD| < |AC|$. It follows that $R_{\text{Sum}} < R_1^{(B)} + |AC| = R_1^{(B)} + R_2^{(A)}$, a contradiction to (2.4.4). □

It should be noted that (2.4.2) and (2.4.3) can be both violated since the optimal WPF value can be achieved on the line segment AB. In this case, we can derive the following result.

Proposition 2.4.2. *If none of the conditions (2.4.2) and (2.4.3) is satisfied, then the following relationship between (R_1^*, R_2^*) and (w_1, w_2) holds:*

$$\begin{cases} R_1^* = \frac{w_1}{w_1 + w_2} R_{\text{Sum}} \\ R_2^* = \frac{w_2}{w_1 + w_2} R_{\text{Sum}}, \end{cases}$$

where (R_1^*, R_2^*) represents the optimal rate pair and R_{Sum} represents the maximum achievable sum rate.

Proof. Since the optimal WPF value is achieved on the line segment AB, the gradient direction at the optimal solution must be perpendicular to AB, i.e. $[\frac{w_1}{R_1^*} \ \frac{w_2}{R_2^*}]^T =$

$\alpha[1\ 1]^T$, where $\alpha > 0$ is some constant to be determined. Since every point on AB achieves the maximum sum rate, we have $R_1^* + R_2^* = R_{\text{Sum}}$. Then, the results stated in the proposition follows by combining these two facts and solving for α. □

2.4.2 K-User Case

We now extend the conditions in (2.4.2) and (2.4.3) to the general case with K users. We state the corner point optimality condition for the general K-user case in the following theorem.

Theorem 2.4.3 (Corner Point Optimality Condition)**.** *In a K-user dual MIMO-MAC channel, a corner point O is optimal if and only if its corresponding decoding order $\pi^*(\cdot)$ and the data rates at O satisfy the following condition:*

$$\frac{w_{\pi^*(i)}}{R^{(O)}_{\pi^*(i)}} \leq \frac{w_{\pi^*(i+1)}}{R^{(O)}_{\pi^*(i+1)}}, \quad \textit{for } i = 1, \ldots, K-1. \tag{2.4.5}$$

Proof. The basic idea of proving Theorem 2.4.3 is to exploit the convexity of the dual MIMO-MAC capacity region and apply Karush-Kuhn-Tucker (KKT) condition [10]. First, we note that if the optimal corner point exists, this optimal corner point must have the largest objective value among all corner points. Otherwise, there would have been an improving direction pointing away from the subregion corresponding to this optimal corner point, a contradiction to the fact that the optimal solution is achieved in this subregion. As a result, to find the optimal corner point, we can simply consider optimizing $\sum_i w_i \log R_i$ over the beveled box that corresponds to the $K!$ corner points (e.g., the dotted box in Fig. 2.4(b)) instead of over the entire capacity region.

Suppose that the optimal corner point is O and the optimal decoding order is $\pi^*(\cdot)$. Since the WPF objective function is convex and the Slater condition [10] holds (i.e., strictly feasible solutions exist), we have that Karush-Kuhn-Tucker (KKT) condition

is both necessary and sufficient for optimality [10]. According to Lemma 2.3.4, we have that the active rate constraints at O can be written as

$$\sum_{i=j}^{K} R^{(O)}_{\pi^*(i)} \;\leq\; \log\left|\mathbf{I}+\sum_{i=j}^{K}\mathbf{H}^{\dagger}_{\pi^*(i)}\mathbf{Q}^{*}_{\pi^*(i)}\mathbf{H}_{\pi^*(i)}\right|, \quad j=1,\ldots,K,$$

where $\mathbf{Q}^*_i, i=1,\ldots,K$, are the optimal input covariance matrices that achieves the maximum sum rate. Then, by KKT condition, we can derive that

$$\begin{bmatrix} \frac{w_{\pi^*(1)}}{R^{(O)}_{\pi^*(1)}} \\ \vdots \\ \frac{w_{\pi^*(K-1)}}{R^{(O)}_{\pi^*(K-1)}} \\ \frac{w_{\pi^*(K)}}{R^{(O)}_{\pi^*(K)}} \end{bmatrix} = u_1 \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} + u_2 \begin{bmatrix} 0 \\ \vdots \\ 1 \\ 1 \end{bmatrix} + \cdots + u_K \begin{bmatrix} 1 \\ \vdots \\ 1 \\ 1 \end{bmatrix}, \tag{2.4.6}$$

where $u_i \geq 0$ for all i. Solving for u_i in (2.4.6), we have

$$u_K = \frac{w_{\pi^*(1)}}{R^{(O)}_{\pi^*(1)}} \quad \text{and} \quad u_{K-i} = \frac{w_{\pi^*(i+1)}}{R^{(O)}_{\pi^*(i+1)}} - \frac{w_{\pi^*(i)}}{R^{(O)}_{\pi^*(i)}},$$

$i=1,\ldots,K-1$. Since $u_i \geq 0$, it then follows that

$$\frac{w_{\pi^*(i)}}{R^{(O)}_{\pi^*(i)}} \leq \frac{w_{\pi^*(i+1)}}{R^{(O)}_{\pi^*(i+1)}}, \quad \text{for } i=1,\ldots,K-1,$$

which is the result stated in the theorem. □

Since the condition in Theorem 2.4.3 is both necessary and sufficient, it implies that if no corner point satisfies (2.4.5), then the optimal decoding order does not exist. In this case, however, we have the following result.

Theorem 2.4.4. *If the optimal WPF solution is achieved by a mixed ordering, then the optimal WPF solution is located on one of the* $\sum_{i=2}^{K}\binom{K}{i}(K-i+1)!$ *time-sharing subregions on the boundary. Suppose that the optimal WPF solution is achieved at a time-sharing subregion that involves a subset of users* $\mathcal{S} \subseteq \{1,2,\ldots,K\}$. *Then we*

must have $R_j^* = \frac{w_j}{\sum_{k\in\mathcal{S}} w_k} R_{\text{Sum}}$, $j \in \mathcal{S}$, *where* R_{Sum} *is the sum rate achieved by the time-sharing among the user set* $\mathcal{S}$.

Proof. Without loss of generality, we index the users in subset $\mathcal{S}$ as $k = 1, 2, \ldots, |\mathcal{S}|$. The first part of the theorem follows immediately from Lemma 2.3.2. To show the second part, we note that the gradient direction at the optimal point must be orthogonal to the hyperplane $\sum_{k\in\mathcal{S}} R_k \leq \log\left|\mathbf{I} + \sum_{k=1}^{i} \mathbf{H}_k^\dagger \mathbf{Q}_k \mathbf{H}_k\right|$. This means that $[\frac{w_1}{R_1^*}, \ldots, \frac{w_i}{R_{|\mathcal{S}|}^*}]^T = \alpha \mathbf{1}$, for some $\alpha > 0$. Thus, we have

$$\frac{w_1}{R_1^*} = \frac{w_2}{R_2^*} = \cdots = \frac{w_{|\mathcal{S}|}}{R_{|\mathcal{S}|}^*}.$$

On the other hand, from the geometrical structure of the dual MIMO-MAC capacity region, we know that every point on the time-sharing subregion achieves the *same* sum rate for users $1, 2, \ldots, |\mathcal{S}|$, i.e., we have $\sum_{k\in\mathcal{S}} R_k^* = R_{\text{Sum}}$, where R_{Sum} represents the sum rate achieved by this time-sharing subregion.

By solving for an appropriate α, it then follows that $R_j^* = \frac{w_j}{\sum_{k\in\mathcal{S}} w_k} R_{\text{Sum}}$ and the proof is complete. □

The R_{Sum} in Theorem 2.4.4 can be computed as follows. Treat the whole time-sharing group of users $\mathcal{S}$ as a virtual user. Then, all the decoding orders involved in the time-sharing can be grouped in to a single decoding order: $\bar{\pi}(1), \bar{\pi}(2), \ldots, \bar{\pi}(j) = \mathcal{S}, \ldots, \bar{\pi}(K - |\mathcal{S}| + 1)$, where $\bar{\pi}(j) = \mathcal{S}$ means the j-th position in $\bar{\pi}$ is $\mathcal{S}$. Also, let $(\bar{\mathbf{Q}}_1^*, \ldots, \bar{\mathbf{Q}}_K^*)$ be the optimal power allocation that achieves the WPF solution. Then, R_{Sum} can be computed as

$$R_{\text{Sum}} = \log\left|\mathbf{I} + \sum_{k=j}^{K-|\mathcal{S}|+1} \mathbf{H}_{\bar{\pi}(k)}^\dagger \bar{\mathbf{Q}}_{\bar{\pi}(k)}^* \mathbf{H}_{\bar{\pi}(k)}\right| - \log\left|\mathbf{I} + \sum_{k=j+1}^{K-|\mathcal{S}|+1} \mathbf{H}_{\bar{\pi}(k)}^\dagger \bar{\mathbf{Q}}_{\bar{\pi}(k)}^* \mathbf{H}_{\bar{\pi}(k)}\right|. \quad (2.4.7)$$

The argument of deriving this expression is very similar to that of Lemma 2.3.4 and is omitted here to avoid repetition.

2.5 Determining Optimal Decoding Order

In the previous section, we have derived a necessary and sufficient condition for the optimal corner point. One can immediately see that the optimal corner point can be found by checking all corner points to see if they satisfy the condition in Theorem 2.4.3. However, since there are $K!$ corner points, such a brute force search is not feasible for large-sized networks. In this section, we will exploit the special geometrical structure of the capacity region and take advantage of the gradient information at each corner point to design an efficient algorithm called "iterative gradient sorting" (IGS). More specifically, in Section 2.5.1, we will first describe the IGS algorithm. Then, in Section 2.5.2, we will analyze the convergence property and complexity of the IGS algorithm. Numerical examples will be provided in Section 2.5.3 to show the efficacy of the proposed IGS algorithm.

2.5.1 Iterative Gradient Sorting

The basic idea of IGS is that, instead of blindly enumerating all the corner points, we exploit the gradient information at each corner point to proactively search for the optimal order. First, recall that the optimal corner point condition in (2.4.5) is written as

$$\frac{w_{\pi^*(i)}}{R^{(O)}_{\pi^*(i)}} \leq \frac{w_{\pi^*(i+1)}}{R^{(O)}_{\pi^*(i+1)}}, \quad \text{for } i = 1, \ldots, K-1. \tag{2.5.1}$$

This shows that the gradient entries at the optimal corner point follow a *non-decreasing* values in the order of π^*. This important observation prompts us to the following idea. Suppose we arbitrarily pick a corner point and use its gradient as an approximation of the gradient at the true optimal corner point. Now, we sort the entries in this approximate gradient to have an approximation of the true optimal decoding order.

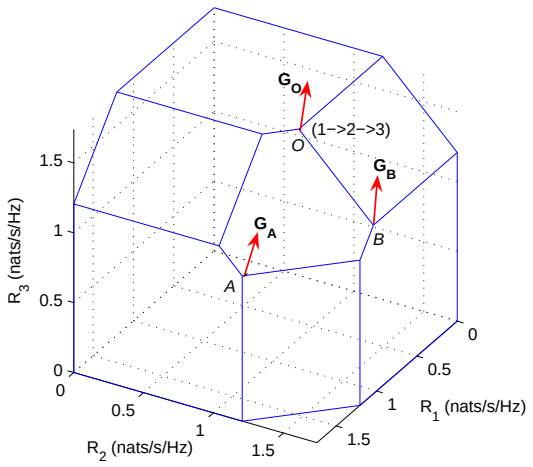


Figure 2.6: An example for iterative gradient sorting.

If the approximate gradient direction is "close" to the true gradient direction, we can expect that the resultant approximate decoding order is also close to the optimal decoding order.

We use a three-user dual MIMO-MAC as shown in Fig. 2.6 to further illustrate this idea geometrically. Fig. 2.6 shows the beveled box for which the hexagon contains exactly the same six corner points of the dual MIMO-MAC capacity region. Suppose that the optimal decoding order is achieved at corner point O with $\pi^* = 1 \rightarrow 2 \rightarrow 3$. Now, we arbitrarily start at some corner point, say A. We use the gradient at A, denoted by $\mathbf{G}_A$, as an approximation of $\mathbf{G}_O$. We then sort the entries of $\mathbf{G}_A$ in a non-decreasing order and denote the resultant ordering as π'. If $\mathbf{G}_A$ is close to $\mathbf{G}_O$ (i.e., $\|\mathbf{G}_A - \mathbf{G}_O\|^2$ is small), we can expect that π' is also close to π^*. Next, we can use the corner point optimality conditions to check if π' is indeed the optimal decoding order. If yes, then the optimal decoding order is found. Otherwise, we move to a new corner point that corresponds to π', say B as shown in Fig. 2.6. Again, we can

Algorithm 2.5.1 Searching for the Optimal Corner Point

1. Compute the set of input covariance matrices $\mathbf{Q}_1^*, \ldots, \mathbf{Q}_K^*$ that maximizes the sum rate capacity of the dual MIMO-MAC channel. Arbitrarily pick an order $\pi(\cdot)$ to start.
2. For the current order $\pi(\cdot)$, use (2.4.5) to test whether or not it is optimal. If yes, return $\pi(\cdot)$ and stop.
3. Compute the gradient at the corner point corresponding to $\pi(\cdot)$, denoted by $\mathbf{G}$. Sort the entries of $\mathbf{G}$ to have a new ordering π'. Let $\pi = \pi'$ and repeat Step 2.

treat B as an approximation of the optimal corner point and sort the entries of the gradient $\mathbf{G}_B$. This sorting and moving process continues iteratively until we find the optimal decoding order. The general IGS algorithm for K-user dual MIMO-MAC is shown in Algorithm 2.5.1.

2.5.2 Convergence and Complexity Analysis

In this subsection, we study the convergence and complexity property of the IGS method. Throughout this subsection, we denote the optimal order for a K-user dual MIMO-MAC as $\pi^*(\cdot)$ and let O be the optimal corner point. To show the convergence property of the IGS method, we first prove the following result.

Lemma 2.5.1. *Let the IGS method start from an arbitrary corner point A. Denote the order after sorting the entries in gradient $\mathbf{G}_A$ as $\pi'(\cdot)$. If $\pi'(\cdot)$ satisfies*

$$\frac{w_{\pi'(1)}}{R^{(A)}_{\pi'(1)}} \leq \frac{w_{\pi'(j)}}{\bar{R}_{\pi'(j)}},\ \forall j \neq 1, \tag{2.5.2}$$

where $\bar{R}_i \triangleq \log|\mathbf{I} + \mathbf{H}_i^\dagger \mathbf{Q}_i \mathbf{H}_i|$, then we have $\pi'(1) = \pi^*(1)$.

Proof. First, we note that $\bar{R}_i$ is the upper bound of user i's data rate R_i. This is because from (2.3.4), we can have that under some decoding order,

$$R_i = \log\left|\mathbf{I} + \hat{\mathbf{H}}_i^\dagger \mathbf{Q}_i^* \hat{\mathbf{H}}_i\right|,$$

where $\hat{\mathbf{H}}_i$ is the *effective channel gain matrix* under the combined interference and noise at user i, which can be computed as

$$\hat{\mathbf{H}}_i = \left(\mathbf{I} + \sum_{j \neq i, j \in \mathcal{S}} \mathbf{H}_j^\dagger \mathbf{Q}_j^* \mathbf{H}_j\right)^{-\frac{1}{2}} \mathbf{H}_i, \tag{2.5.3}$$

where $\mathcal{S}$ is some subset of $\{1, \ldots, K\}$. Note that if $\mathcal{S} = \varnothing$ (i.e., there is no interference at user i, which corresponds to user i being decoded last), then $\hat{\mathbf{H}}_i$ reduces to $\mathbf{H}_i$ and thus $R_i = \bar{R}_i$. Otherwise, $R_i < \bar{R}_i$ because $|\hat{\mathbf{H}}_i| < |\mathbf{H}_i|$. It follows from this fact that at A, $R_{\pi'(j)}^{(A)} \leq \bar{R}_{\pi'(j)}$. Combined with the condition given in the lemma, we further have

$$\frac{w_{\pi'(1)}}{R_{\pi'(1)}^{(A)}} \leq \frac{w_{\pi'(j)}}{\bar{R}_{\pi'(j)}} \leq \frac{w_{\pi'(j)}}{R_{\pi'(j)}^{(A)}}, \quad \forall j \neq 1. \tag{2.5.4}$$

Note that $\frac{w_{\pi'(j)}}{\bar{R}_{\pi'(j)}} = \frac{w_{\pi^*(k)}}{\bar{R}_{\pi^*(k)}}$ if $\pi'(j) = \pi^*(k)$. Then, we have

$$\frac{w_{\pi'(1)}}{R_{\pi'(1)}^{(A)}} \leq \frac{w_{\pi'(j)}}{\bar{R}_{\pi'(j)}} = \frac{w_{\pi^*(k)}}{\bar{R}_{\pi^*(k)}} \leq \frac{w_{\pi^*(k)}}{R_{\pi^*(k)}^{(O)}}, \tag{2.5.5}$$

for all k such that $\pi^*(k) = \pi'(j)$, $j \neq 1$. That is, $\frac{w_{\pi'(1)}}{R_{\pi'(1)}^{(A)}}$ is smaller than all but one entry in the gradient $\mathbf{G}_O$. Recall from the optimal corner point optimality condition that

$$\frac{w_{\pi^*(i)}}{R_{\pi^*(i)}^{(O)}} \leq \frac{w_{\pi^*(i+1)}}{R_{\pi^*(i+1)}^{(O)}}, \quad \text{for } i = 1, \ldots, K-1.$$

Thus, we must have that $\pi'(1) = \pi^*(1)$. □

Lemma 2.5.1 says that if the stated condition is satisfied, then the first element in the order achieved by sorting entries in $\mathbf{G}_A$ is guaranteed to be correct. In other words, IGS can determine at least one element correctly in each iteration.

Now, we establish the convergence property of the IGS method in the following theorem.

Theorem 2.5.2. *Suppose that a K-user dual MIMO-MAC has an optimal corner point O for which the decoding order is $\pi^*(\cdot)$. Denote $\pi^k(\cdot)$ the order achieved by sorting the corner point A during the k-th iteration of the IGS method. If $\pi^k(\cdot)$ satisfies the following condition,*

$$\frac{w_{\pi^k(k)}}{R^{(A)}_{\pi^k(k)}} \leq \frac{w_{\pi^k(j)}}{\bar{R}_{\pi^k(j)}}, \quad k < j \leq K, \tag{2.5.6}$$

then the IGS method can find the optimal order in at most K iterations.

Proof. When $k = 1$, the condition in (2.5.6) reduces to (2.5.2). Then, according to Lemma 2.5.1, we can determine $\pi^*(1)$ correctly after one iteration. Note that in $\pi^*(\cdot)$, user $\pi^*(1)$ is the first one to be decoded. Thus, it is subject to the maximum number of interferers. As a result, $R^{(O)}_{\pi^*(1)}$ is a lower bound of $R_{\pi^*(1)}$. Therefore, in all subsequent iterations, we have

$$\frac{w_{\pi^k(1)}}{R^{(A)}_{\pi^k(1)}} \leq \frac{w_{\pi^*(1)}}{R^{(O)}_{\pi^*(1)}},$$

which means that, once $\pi^*(1)$ is determined after the first iteration, all subsequent iterations must have $\pi^k(1) = \pi^*(1)$, $k = 2, \ldots, K$.

Thus, in the 2nd iteration, we can ignore $\pi^2(1)$ and only consider the correctness of $\pi^2(j)$, $j = 2, \ldots, K$. This is equivalent to finding the optimal order for a $(K-1)$-user dual MIMO-MAC, where user $\pi^*(1)$ is eliminated. It is easy to see that, for this $(K-1)$-user channel, the condition in (2.5.6) is exactly the same as that in (2.5.2). Therefore, according to Lemma 2.5.1, we can guarantee that $\pi^2(2)$ is determined correctly after the second iteration.

Since we correctly determine at least one more user in each iteration without affecting all users that were determined previously, it is easy to see that it takes at most K iterations to find the optimal ordering for a K-user MIMO-MAC. □

One remaining question for Theorem 2.5.2 is that how restrictive the condition in (2.5.6) is. In what follows, we argue that (2.5.6) is a rather mild condition for dual MIMO-MAC. A closer observation of Theorem 2.5.2 reveals that, in the k-th iteration, we are in fact using $\frac{w_i}{\bar{R}_i}$ as a lower bound for $\frac{w_i}{R_i^{(A)}}$ to guarantee that once we find $\pi^k(k)$ in the k-th iteration, it is indeed the k-th smallest entry in $\mathbf{G}_O$ and its position will not be changed in the subsequent iterations. Thus, the question boils down to how close the gap between $\frac{w_i}{\bar{R}_i}$ and $\frac{w_i}{R_i^{(A)}}$, i.e., the difference between $\bar{R}_i$ and $R_i^{(A)}$. The smaller the difference, the less restrictive the condition in (2.5.6) is. To this end, we let $\underline{R}_i$ be the lower bound of R_i (i.e., user i is interfered by all other users). From the rate expressions of the dual MIMO-MAC, we have $\underline{R}_i = \log|\mathbf{I} + \hat{\mathbf{H}}_i^\dagger \mathbf{Q}_i \hat{\mathbf{H}}_i|$, where $\hat{\mathbf{H}}_i = (\mathbf{I} + \sum_{j\neq i} \mathbf{H}_j^\dagger \mathbf{Q}_j \mathbf{H}_j)^{-\frac{1}{2}} \mathbf{H}_i$. Thus, we have that

$$\bar{R}_i - R_i^{(A)} \leq \bar{R}_i - \underline{R}_i = \log|\mathbf{I} + \mathbf{H}_i^\dagger \mathbf{Q}_i \mathbf{H}_i| - \log|\mathbf{I} + \hat{\mathbf{H}}_i^\dagger \mathbf{Q}_i \hat{\mathbf{H}}_i|. \tag{2.5.7}$$

It can be seen from (2.5.7) that as the number of interferers increases, the change from $\bar{R}_i$ to $\underline{R}_i$ scales logarithmically, meaning that with reasonably high SINR, the gap between $\bar{R}_i$ and $R_i^{(A)}$ is minor. As a result, the condition in (2.5.6) is not very restrictive.

Theorem 2.5.2 says that we can determine at least one position correctly after one round of gradient sorting. Thus, at most $K-1$ rounds of sorting are needed to determine the optimal decoding order. As a result, the complexity of IGS is $O((K-1)K\log K)$, meaning that IGS is a *polynomial-time* algorithm. In fact, this complexity estimation is too conservative since in one iteration, IGS can usually accurately determine far more than one position. Thus, the running time of IGS is usually much less than K iterations, as evidenced in the numerical results in the next subsection.

Finally, it is worth pointing out that the condition in Theorem 2.5.2 is only a sufficient condition. It provides us cases where the convergence analysis is more tractable and the polynomial time complexity can be easily shown. This does not mean that IGS will not converge if the condition in Theorem 2.5.2 is violated. However, due to the lack of a close form expression of the capacity region in R_i's, deriving the necessary and sufficient condition for IGS's convergence appear to be difficult.

2.5.3 Numerical Results

We use a 15-user MIMO-BC example to demonstrate the efficacy of IGS. It is easy to verify that there are $15! \approx 1.3077 \times 10^{12}$ corner points in total, which means it is not viable to use a brute force search. As shown in Fig. 2.7, the nodes are indexed from 1 to 15 and are randomly distributed in a square region. The transmitter is located at the center. The transmitter and all the receivers are equipped with four antennas. For user 1 to user 15, the weights are 1, 1, 1.2, 0.7, 0.33, 0.25, 0.35, 0.2, 0.9, 5, 0.65, 0.8, 0.4, 0.78, and 1, respectively. In this example, the optimal decoding order in the dual MIMO-MAC exists and is: $4 \rightarrow 12 \rightarrow 9 \rightarrow 15 \rightarrow 11 \rightarrow 3 \rightarrow 7 \rightarrow 5 \rightarrow 6 \rightarrow 8 \rightarrow 13 \rightarrow 14 \rightarrow 1 \rightarrow 10 \rightarrow 2$.

To test IGS's efficiency and robustness, we start from two completely reversed initial orders $1 \rightarrow 2 \rightarrow \ldots \rightarrow 15$ and $15 \rightarrow 14 \rightarrow \ldots \rightarrow 1$, such that their corresponding corner points are far away from each other on the capacity region. For convenience, we call them Case 1 and Case 2, respectively. For both cases, the result of IGS in each iteration is shown in Table 2.1. In both cases, IGS found the optimal decoding order in just *two* iterations. The running time is less than 1 second for both cases. This shows the efficiency of the IGS algorithm. We notice that after the first iteration, the first 5 position in Case 1 and the first 4 positions in Case 2 are already correct.

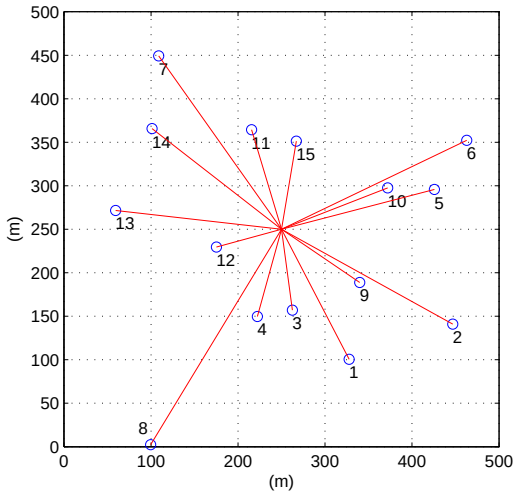


Figure 2.7: Network topology of a 15-user MIMO-BC.

This shows that in one iteration, IGS can accurately determine far more than one position, which is the reason that makes the IGS method so efficient.

2.5.4 Further Discussion

In the development of the IGS algorithm, we have implicitly assumed that the optimal WPF solution is achieved by a single decoding order. However, when the optimal WPF solution is achieved on a time-sharing subregion on the capacity region boundary (i.e., a mixed ordering), identifying the proper time-sharing subregion becomes even more complex. In such cases, we cannot expect that the IGS method terminates in finite number of steps. Instead, we observe that when K is not large (e.g., less than or equal to 4), the IGS will oscillate between a subset of corner points and the subregion corresponds to these corner points often turns out to be the correct time-sharing subregion. However, it is unclear (and unlikely) that IGS can still give us the correct time-sharing subregion as K becomes large. So far, finding the correct

Table 2.1: IGS for a 15-user MIMO-BC example.

	Case 1			Case 2		
No.	Initial	Iter. 1	Iter. 2	Initial	Iter. 1	Iter. 2
1	1	4	4	15	4	4
2	2	12	12	14	12	12
3	3	9	9	13	9	9
4	4	15	15	12	15	15
5	5	11	11	11	3	11
6	6	13	3	10	7	3
7	7	14	7	9	5	7
8	8	6	5	8	6	5
9	9	5	6	7	1	6
10	10	7	8	6	8	8
11	11	10	13	5	11	13
12	12	8	14	4	2	14
13	13	3	1	3	14	1
14	14	1	10	2	10	10
15	15	2	2	1	13	2

time-sharing subregion remains an open problem and will be an interesting topic in our future investigation. Since the number of time-sharing subregions grows quickly as K gets large, we expect that significant amount of effort will be needed to address this problem.

2.6 Power Allocation

So far, we have studied how to determine the optimal decoding order $\pi^*(\cdot)$ in a dual MIMO-MAC. In this section, we will focus on computing the optimal power allocation variables under the knowledge of $\pi^*(\cdot)$ such that the objective value of the D-WPF problem in (2.3.1) is maximized.

Unfortunately, even with the knowledge of $\pi^*(\cdot)$, D-WPF remains a difficult non-convex optimization problem with respect to $\mathbf{Q}$. To see why D-WPF is non-convex, let us further examine the problem structure in (2.3.1). For convenience, we let $\mathbf{M}_{\pi(i)} \triangleq \mathbf{I} + \sum_{j=i}^{K} \mathbf{H}_{\pi(j)}^{\dagger} \mathbf{Q}_{\pi(j)} \mathbf{H}_{\pi(j)}$ for some order $\pi(\cdot)$. Substituting the rate expressions in

Lemma 2.3.4 into D-WPF, we can further write D-WPF as

$$\begin{aligned}
&\underline{\textbf{D-WPF:}}\\
&\text{Maximize } w_{\pi^*(K)} \log(\log|\mathbf{M}_{\pi^*(K)}|)+\\
&\qquad\qquad \textstyle\sum_{i=1}^{K-1} w_{\pi^*(i)} \log(\log|\mathbf{M}_{\pi^*(i)}| - \log|\mathbf{M}_{\pi^*(i+1)}|)\\
&\text{subject to } \textstyle\sum_i \mathrm{Tr}\{\mathbf{Q}_i\} \le P,\ \mathbf{Q}_i \succeq 0,\ i = 1, 2, \ldots, K.
\end{aligned}$$

Note that $\mathbf{M}_{\pi^*(i)}$ is a positive definite matrix for all i. It then follows that $\log|\mathbf{M}_{\pi^*(i)}|$ is a concave function with respect to $\mathbf{M}_{\pi^*(i)}$ [29]. However, since the convexity of the difference of two concave $\log|\cdot|$ functions is indefinite [64], we have that each term in the form of $\log(\log|\cdot| - \log|\cdot|)$ in the objective function of D-WPF is non-convex, thus rendering the entire D-WPF problem non-convex. Due to this non-convexity, solving D-WPF to global optimality is difficult. In what follows, we will propose an efficient algorithm based on conjugate gradient projection (CGP) to determine a local optimal solution. Then, we will evaluate the quality of this local optimal solution by comparing it with an upper bound.

2.6.1 An Efficient Algorithm

The basic idea of our proposed method is to use conjugate gradient projection (CGP) to search for a local optimal solution. Our method utilizes an important concept called Hessian-conjugate direction to deflect the conventional gradient direction appropriately. The purpose of this deflection is to achieve an asymptotic K-step superlinear convergence rate [10], similar to that of the quasi-Newton methods (e.g., BFGS method). We first state the framework of CGP in Algorithm 5.4.2.

There are two key elements in Algorithm 5.4.2: 1) computing the gradients in Step 2; and 2) performing the gradient projection in Step 4. In what follows, we will introduce these components in detail.

Algorithm 2.6.1 Determine the Optimal Power Allocation.

1. Choose $\mathbf{Q}_{\pi^*}^{(0)} = [\mathbf{Q}_{\pi^*(1)}^{(0)}, \mathbf{Q}_{\pi^*(2)}^{(0)}, \ldots, \mathbf{Q}_{\pi^*(K)}^{(0)}]^T$. Let $k = 0$.
2. Calculate the conjugate gradients $\mathbf{G}_{\pi(i)}^{(k)}$, $i = 1, 2, \ldots, K$.
3. Choose an appropriate step size s_k. Let $\mathbf{Q}_{\pi^*(i)}^{'(k)} = \mathbf{Q}_{\pi^*(i)}^{(k)} + s_k \mathbf{G}_{\pi^*(i)}^{(k)}$, for $i = 1, 2, \ldots, K$.
4. Let $\bar{\mathbf{Q}}_{\pi^*}^{(k)}$ be the projection of $\mathbf{Q}_{\pi^*}^{'(k)}$ onto $\Omega_+(n)$.
5. Choose an appropriate step size α_k. Let $\mathbf{Q}_{\pi^*(i)}^{(k+1)} = \mathbf{Q}_{\pi^*(i)}^{(k)} + \alpha_k(\bar{\mathbf{Q}}_{\pi^*(i)}^{(k)} - \mathbf{Q}_{\pi^*(i)}^{(k)})$, $i = 1, 2, \ldots, K$.
6. Let $k = k + 1$. If the maximum absolute value of the elements in $\mathbf{Q}_{\pi^*(i)}^{(k)} - \mathbf{Q}_{\pi^*(i)}^{(k-1)} < \epsilon$, for $i = 1, \ldots, K$, then stop; else go to Step 2.

Computing the gradients: For simplicity, we denote the objective function of D-WPF as $F(\mathbf{Q}_{\pi^*})$. The gradient $\bar{\mathbf{G}}_{\pi^*(j)} \triangleq \nabla_{\mathbf{Q}_{\pi^*(j)}} F(\mathbf{Q}_{\pi^*})$ depends on the partial derivatives of $F(\mathbf{Q}_{\pi^*})$ with respect to $\mathbf{Q}_{\pi^*(j)}$. By using the formula $\frac{\partial \ln|\mathbf{A}+\mathbf{BXC}|}{\partial \mathbf{X}} = [\mathbf{C}(\mathbf{A}+\mathbf{BXC})^{-1}\mathbf{B}]^T$ from matrix differential calculus [98, 159], we can compute the partial derivative of the i-th term in the summation of $F(\mathbf{Q}_{\pi^*})$ with respect to $\mathbf{Q}_{\pi^*(j)}$, $j \geq i$, as follows:

$$\begin{aligned}\frac{\partial F^{(i)}}{\partial \mathbf{Q}_{\pi^*(j)}} &\triangleq \frac{\partial \left(w_{\pi^*(i)} \log \left(\log \left| \mathbf{M}_{\pi^*(i)} \right| - \log \left| \mathbf{M}_{\pi^*(i+1)} \right| \right) \right)}{\partial \mathbf{Q}_{\pi^*(j)}} \\ &= w_{\pi^*(i)} \frac{\left(\mathbf{H}_{\pi^*(j)} \left[\mathbf{M}_{\pi^*(i)}^{-1} - \mathbf{M}_{\pi^*(i+1)}^{-1} \right] \mathbf{H}_{\pi^*(j)}^{\dagger} \right)^T}{\log \left| \mathbf{M}_{\pi^*(i)} \right| - \log \left| \mathbf{M}_{\pi^*(i+1)} \right|}.\end{aligned}$$

To compute the gradient of $F(\mathbf{Q}_{\pi^*})$ with respect to $\mathbf{Q}_{\pi^*(j)}$, we note that only the first j terms in $F(\mathbf{Q}_{\pi^*})$ involve $\mathbf{Q}_{\pi^*(j)}$. From the definition $\nabla_z f(z) = 2(\partial f(z)/\partial z)^*$ [58], we can derive the following result.

Proposition 2.6.1. *In the D-WPF problem, the gradient of user $\pi^*(i)$ can be computed as*

$$\begin{aligned}\bar{\mathbf{G}}_{\pi^*(j)} =& \sum_{i=1}^{j} 2 \left(\frac{\partial F^{(i)}}{\partial \mathbf{Q}_{\pi^*(j)}} \right)^* = \frac{2 w_{\pi^*(j)} \mathbf{H}_{\pi^*(j)} \mathbf{M}_{\pi^*(j)}^{-1} \mathbf{H}_{\pi^*(j)}^{\dagger}}{\log \left| \mathbf{M}_{\pi^*(j)} \right| - \log \left| \mathbf{M}_{\pi^*(j+1)} \right|} \\ &+ \sum_{i=1}^{j-1} 2 w_{\pi^*(i)} \frac{\mathbf{H}_{\pi^*(j)} \left(\mathbf{M}_{\pi^*(i)}^{-1} - \mathbf{M}_{\pi^*(i+1)}^{-1} \right) \mathbf{H}_{\pi^*(j)}^{\dagger}}{\log \left| \mathbf{M}_{\pi^*(i)} \right| - \log \left| \mathbf{M}_{\pi^*(i+1)} \right|}. \end{aligned} \tag{2.6.1}$$

The conjugate gradient direction in the m-th iteration can be computed as $\mathbf{G}_{\pi^*(j)}^{(m)} = \bar{\mathbf{G}}_{\pi^*(j)}^{(m)} + \kappa_m \mathbf{G}_{\pi^*(j)}^{(m-1)}$. In this chapter, we adopt the Fletcher and Reeves' choice of deflection [10], which can be computed as $\kappa_m = \|\bar{\mathbf{G}}_{\pi^*(j)}^{(m)}\|^2 / \|\bar{\mathbf{G}}_{\pi^*(j)}^{(m-1)}\|^2$. As mentioned earlier, the purpose of deflecting the gradient is to find the Hessian-conjugate direction that tend to reduce the "zigzagging" phenomenon encountered in the conventional gradient projection method and achieve an asymptotic K-step superlinear convergence rate without actually storing a large Hessian approximation matrix [10].

Projecting the gradients: In Algorithm 5.4.2, we note from (2.6.1) that $\mathbf{G}_{\pi^*(j)}$ is Hermitian. It follows that $\mathbf{Q}_{\pi^*(j)}^{\prime(k)} = \mathbf{Q}_{\pi^*(j)}^{(k)} + s_k \mathbf{G}_{\pi^*(j)}^{(k)}$ is also a Hermitian matrix. Thus, the task of the projection problem is to simultaneously project K Hermitian matrices onto the set

$$\Omega_+(n) \triangleq \left\{ \mathbf{Q}_i \;\middle|\; \begin{array}{l} \sum_i \mathrm{Tr}\{\mathbf{Q}_i\} \le P, \\ \mathbf{Q}_i \succeq 0,\ i = 1, 2, \ldots, K \end{array} \right\}.$$

For this purpose, we construct a block diagonal matrix $\mathbf{D} = \mathrm{Diag}\{\mathbf{Q}_{\pi^*(1)} \ldots \mathbf{Q}_{\pi^*(K)}\} \in \mathbb{C}^{(K \cdot n_r) \times (K \cdot n_r)}$. It is easy to recognize that $\mathbf{Q}_{\pi^*(j)} \in \Omega_+(n)$, $j = 1, \ldots, K$, if and only if $\mathrm{Tr}(\mathbf{D}) = \sum_{j=1}^{K} \mathrm{Tr}\left(\mathbf{Q}_{\pi^*(j)}\right) \le P$ and $\mathbf{D} \succeq 0$. Thus, the projection problem can be translated into the following optimization problem:

$$\begin{array}{ll} \text{Minimize} & \frac{1}{2}\|\tilde{\mathbf{D}} - \mathbf{D}\|_F^2 \\ \text{subject to} & \mathrm{Tr}(\tilde{\mathbf{D}}) \le P,\ \tilde{\mathbf{D}} \succeq 0, \end{array} \tag{2.6.2}$$

where $\|\cdot\|_F$ denotes the Frobenius norm. Problem (5.4.16) means that given a block diagonal matrix $\mathbf{D}$, we wish to find a matrix $\tilde{\mathbf{D}} \in \Omega_+(n)$ such that $\tilde{\mathbf{D}}$ minimizes $\|\tilde{\mathbf{D}} - \mathbf{D}\|_F$. Problem (5.4.16) falls within the category of "matrix nearness problem" in the literature [17,99], which is not easy to solve in general. However, by exploiting its quadratic structure, we are able to design a polynomial-time algorithm to solve

it. The reason that we can solve (5.4.16) in polynomial time is based on the result stated in the following proposition.

Proposition 2.6.2. *The projection problem in (5.4.16) is equivalent to the following optimization problem*

$$\begin{array}{ll} \text{Maximize} & -\frac{1}{2}\sum_{j=1}^{K\cdot n_r}\left(\max\left\{0,\lambda_j-\mu\right\}\right)^2-\mu P \\ \text{subject to} & \mu\geq 0, \end{array} \tag{2.6.3}$$

where λ_j represents the eigenvalues of the block diagonal matrix $\mathbf{D}$. Let μ^* be the optimal solution upon solving (5.4.19). Let the eigenvalue decomposition of $\mathbf{D}$ be $\mathbf{D}=\mathbf{U}\mathbf{\Lambda}\mathbf{U}^\dagger$. Then, the solution of (5.4.16) can be recovered by $\tilde{\mathbf{D}}=\mathbf{U}(\mathbf{\Lambda}-\mu^*\mathbf{I})_+\mathbf{U}^\dagger$.

Proof. It can be verified that (5.4.16) is a convex problem since its objective is convex and the constraints are affine. So we can equivalently solve it through its dual problem (assuming a suitable constraint qualification [10]). For this purpose, we associate a Hermitian matrix $\mathbf{\Pi}$ to the constraint $\tilde{\mathbf{D}}\succeq 0$ and μ to the constraint $\mathrm{Tr}(\tilde{\mathbf{D}})\leq P$. Then, we can write the Lagrangian of (5.4.16) as $g(\mathbf{\Pi},\mu)=\min_{\tilde{\mathbf{D}}}\{(1/2)\|\tilde{\mathbf{D}}-\mathbf{D}\|_F^2-\mathrm{Tr}(\mathbf{\Pi}^\dagger\tilde{\mathbf{D}})+\mu(\mathrm{Tr}(\tilde{\mathbf{D}})-P)\}$. After some simplifications (see [84] for details), the Lagrangian dual problem can be written as

$$\begin{array}{ll} \text{Maximize} & -\frac{1}{2}\|\mathbf{D}-\mu\mathbf{I}+\mathbf{\Pi}\|_F^2-\mu P+\frac{1}{2}\|\mathbf{D}\|_F^2 \\ \text{subject to} & \mathbf{\Pi}\succeq 0,\mu\geq 0. \end{array} \tag{2.6.4}$$

After solving (5.4.18), we have that the optimal solution to (5.4.16) can be recovered as $\tilde{\mathbf{D}}^*=\mathbf{D}-\mu^*\mathbf{I}+\mathbf{\Pi}^*$, where μ^* and $\mathbf{\Pi}^*$ are the optimal dual solutions to problem (5.4.18). According to Moreau Decomposition [61], we can derive that $\min_{\mathbf{\Pi}}\|\mathbf{D}-\mu\mathbf{I}+\mathbf{\Pi}\|_F=(\mathbf{D}-\mu\mathbf{I})_+$, where the operation $(\mathbf{A})_+$ represents projecting a matrix $\mathbf{A}$ onto the positive semidefinite cone. This can be done by 1) apply eigenvalue decomposition on matrix $\mathbf{A}$; 2) keep the eigenvector matrix unchanged; 3)

set all non-positive eigenvalues to zero; and 4) multiply back. Thus, we can remove the matrix variable $\mathbf{\Pi}$ in (5.4.18) and rewrite the problem as

$$\begin{array}{l} \text{Maximize } \psi(\mu) \triangleq -\frac{1}{2}\left\|(\mathbf{D}-\mu\mathbf{I})_+\right\|_F^2 - \mu P \\ \text{subject to } \mu \geq 0. \end{array} \tag{2.6.5}$$

Let the eigenvalue decomposition of $\mathbf{D}$ be $\mathbf{D} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^\dagger$. Thus, $(\mathbf{D}-\mu\mathbf{I})_+$ can be written as $\mathbf{U}(\mathbf{\Lambda}-\mu\mathbf{I})_+\mathbf{U}^\dagger$. Note that $\mathbf{U}$ is unitary, we further have $\left\|(\mathbf{D}-\mu\mathbf{I})_+\right\|_F^2 = \left\|(\mathbf{\Lambda}-\mu\mathbf{I})_+\right\|_F^2$. Let λ_i, $i = 1, \ldots, K \times n_r$ represent the eigenvalue in $\mathbf{\Lambda}$. Without loss of generality, we assume that the eigenvalues are sorted in non-increasing order: $\mathbf{\Lambda} = \text{Diag}\{\lambda_1\ \lambda_2 \ldots\ \lambda_{K\cdot n_r}\}$, where $\lambda_1 \geq \ldots \geq \lambda_{K\cdot n_r}$. It then follows that $\left\|(\mathbf{\Lambda}-\mu\mathbf{I})_+\right\|_F^2 = \sum_{j=1}^{K\cdot n_r}\left(\max\{0, \lambda_j - \mu\}\right)^2$, As a result, the objective function in (2.6.5) can be further simplified to

$$\psi(\mu) = -\frac{1}{2}\sum_{j=1}^{K\cdot n_r}\left(\max\{0, \lambda_j - \mu\}\right)^2 - \mu P,$$

which is in exactly the same form as in Proposition 2.6.2. This completes the proof.

□

For the equivalent problem in (5.4.19), we develop a polynomial time algorithm to solve it. We summarize the algorithm in Algorithm 5.4.3. Due to space limitation, we refer readers to [84] for the detailed complexity analysis.

2.6.2 An Upper Bound of the D-WPF Problem

Since the CGP method converges to a local optimal solution, it is necessary to evaluate the gap between this local optimal solution and the global optimal solution. In this chapter, we move one step further: we compare the local optimal solution with an upper bound of the true optimum. If this gap is small, we can conclude that the gap between the local optimum and the global optimum is even smaller. For this

Algorithm 2.6.2 Projecting the Gradients

Initiation:

1. Construct a block diagonal matrix $\mathbf{D}$. Perform eigenvalue decomposition $\mathbf{D} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^{\dagger}$, sort the eigenvalues in non-increasing order.
2. Introduce $\lambda_0 = \infty$ and $\lambda_{K \cdot n_t + 1} = -\infty$. Let $\hat{I} = 0$. Let the endpoint objective value $\psi_{\hat{I}}(\lambda_0) = 0$, $\phi^* = \psi_{\hat{I}}(\lambda_0)$, and $\mu^* = \lambda_0$.

Main Loop:

1. If $\hat{I} > K \cdot n_r$, go to the final step; else let $\mu^*_{\hat{I}} = (\sum_{j=1}^{\hat{I}} \lambda_j - P)/\hat{I}$.
2. If $\mu^*_{\hat{I}} \in [\lambda_{\hat{I}+1}, \lambda_{\hat{I}}] \cap \mathbb{R}_+$, then let $\mu^* = \mu^*_{\hat{I}}$ and go to the final step.
3. Compute $\psi_{\hat{I}}(\lambda_{\hat{I}+1})$. If $\psi_{\hat{I}}(\lambda_{\hat{I}+1}) < \phi^*$, then go to the final step; else let $\mu^* = \lambda_{\hat{I}+1}$, $\phi^* = \psi_{\hat{I}}(\lambda_{\hat{I}+1})$, $\hat{I} = \hat{I} + 1$ and continue.

Final Step: Compute $\tilde{\mathbf{D}}$ as $\tilde{\mathbf{D}} = \mathbf{U}\left(\mathbf{\Lambda} - \mu^* \mathbf{I}\right)_+ \mathbf{U}^{\dagger}$.

purpose, we now derive an upper bound for the D-WPF problem, which is stated in the following theorem.

Theorem 2.6.3. *The D-WPF problem is upper bounded by the optimal objective value of the following convex optimization problem*

$$\begin{array}{rl} \underline{\textbf{D-WPF-UB:}} & \\ \text{Maximize} & \log\left(\sum_{i=1}^{K}(w_{\pi^*(i)} - w_{\pi^*(i-1)})_+ \log|\mathbf{M}_{\pi^*(i)}|\right) \\ \text{subject to} & \sum_i \mathrm{Tr}\{\mathbf{Q}_i\} \leq P,\ \mathbf{Q}_i \succeq 0,\ i = 1, 2, \ldots, K, \end{array}$$

where the operation $(\cdot)_+$ *is defined as* $\max(\cdot, 0)$.

Proof. Since log function is concave, from Jensen's inequality, we have the following chain of inequalities for the objective function of the D-WPF problem:

$$\begin{array}{rl} F(\mathbf{Q}_{\pi^*}) = & w_{\pi^*(K)} \log(\log|\mathbf{M}_{\pi^*(K)}|) + \\ & \sum_{i=1}^{K-1} w_{\pi^*(i)} \log(\log|\mathbf{M}_{\pi^*(i)}| - \log|\mathbf{M}_{\pi^*(i+1)}|) \\ \leq & \log\big(w_{\pi^*(K)} \log|\mathbf{M}_{\pi^*(K)}| + \sum_{i=1}^{K-1} w_{\pi^*(i)} \times \\ & \qquad \big(\log|\mathbf{M}_{\pi^*(i)}| - \log|\mathbf{M}_{\pi^*(i+1)}|\big)\big) \\ \leq & \log\big(\sum_{i=1}^{K}(w_{\pi^*(i)} - w_{\pi^*(i-1)}) \log|\mathbf{M}_{\pi^*(i)}|\big) \\ \leq & \log\big(\sum_{i=1}^{K}(w_{\pi^*(i)} - w_{\pi^*(i-1)})_+ \log|\mathbf{M}_{\pi^*(i)}|\big), \end{array}$$

where the last inequality holds because each log term associated with negative $(w_{\pi^*(i)} - w_{\pi^*(i-1)})$ is set to zero. Further, since $\log|\mathbf{M}_{\pi^*(i)}|$ is concave for all i and the constraints are linear, it is clear that the D-WPF-UB problem is a convex program and

the proof is complete. □

The upper bound in Theorem 2.6.3 can be computed efficiently since D-WPF-UB is a convex program. In fact, due to the similar structure between D-WPF and D-WPF-UB, we can again use CGP to compute the optimal value of D-WPF-UB. The only difference is that a new gradient expression for D-WPF-UB is needed. Due to space limitation, we state the gradient expression in the following proposition and refer readers to [84] for the details of its proof.

Proposition 2.6.4. *For the D-WPF-UB problem, the gradient for user $\pi^*(i)$ can be computed as*

$$\tilde{\mathbf{G}}_{\pi^*(i)} = \frac{2\sum_{j=1}^{i}(w_{\pi^*(j)} - w_{\pi^*(j-1)})_+ \mathbf{H}_{\pi^*(j)}\mathbf{M}_{\pi^*(j)}^{-1}\mathbf{H}_{\pi^*(j)}^{\dagger}}{\sum_{j=1}^{K}(w_{\pi^*(j)} - w_{\pi^*(j-1)})_+ \log|\mathbf{M}_{\pi^*(j)}|}.$$

Since D-WPF-UB is convex, the solution obtained from CGP is a global optimal solution. As mentioned earlier, the optimal value of D-WPF-UB will provide an estimate of the gap between true optimal value of the D-WPF problem and the local optimal solution determined by CGP.

2.6.3 Numerical Results

We first study the convergence performance of the CGP method. For the 15-user MIMO-BC example in Fig. 2.7, we plot the convergence process of the CGP algorithm in Fig. 2.8. In this example, we start at the optimal corner point that corresponds to the optimal decoding order. In Fig. 2.8, we can see that CGP takes only 35 iterations to converge. The convergence processes of all 15 users' data rates are plotted in Fig. 2.9, from which we can see that the data rates converge to a proportional fair status.

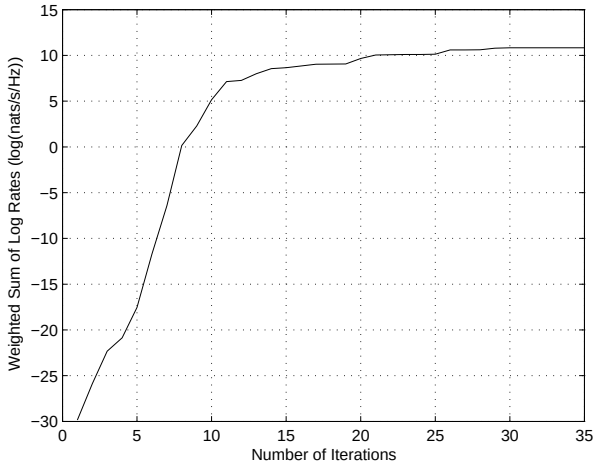


Figure 2.8: Convergence behavior of the CGP algorithm.

2.7 Chapter Summary

In this chapter, we studied the weighted proportional fairness scheduling problem in MIMO broadcast channels (MIMO-BC) with dirty paper coding (DPC). This problem is difficult due to its non-convex property and an exponentially large search space inherent in enumerating DPC's encoding orders. To solve this problem, we first derived a set of optimality conditions that the optimal DPC encoding order must satisfy. Then, based on the optimality conditions, we designed an efficient algorithm called iterative gradient sorting (IGS) to determine the optimal decoding order. We showed that under some mild condition, the IGS method is guaranteed to find the optimal encoding order in polynomial time. Third, we proposed an efficient algorithm based on conjugate gradient projection (CGP) for computing the optimal power allocation. By comparing with an upper bound of the optimal WPF solutions, we showed that the gap between the solutions obtained from CGP and the true optimal solutions is small. Collectively, these results fill an important gap in dealing with fairness issues

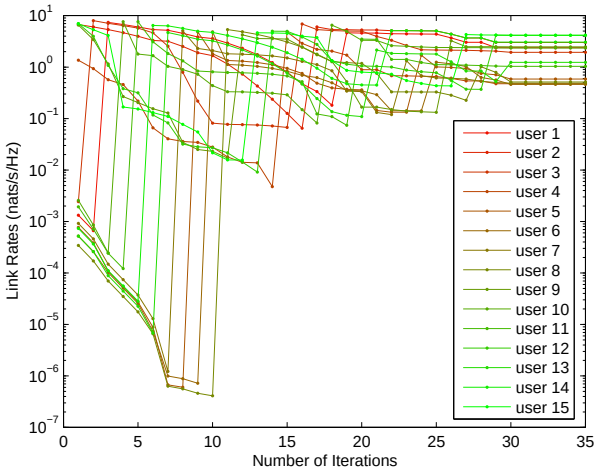


Figure 2.9: Data rates converge to proportional fair status.

in MIMO-BC with DPC.

Chapter 3

Capacity of Single-hop MIMO Ad Hoc Networks

3.1 Introduction

In this chapter, we focus on single-hop MIMO-based ad hoc networks, where there exist K concurrent MIMO transmission pairs. We refer to each active transmission pair as a link. Each link is interfered by all other links in the network. In fact, such a network can be viewed as the general case of vector Gaussian interference channel in network information theory, for which the capacity region is still unknown [29]. We assume in this chapter that all receiving nodes in the network can only perform single-user detection (i.e., treating all interferences as noise). We are interested in finding the maximum sum of mutual information (i.e., sum rate capacity) in such a network setting.

It is well-known that finding the sum rate capacity of such MIMO-based single-hop ad hoc networks under single-user detection is a hard problem due to non-convexity of the problem structure. In [15], Blum showed that the sum of mutual information of a multiuser MIMO system is neither a convex nor a concave function. Also, a change in the covariance matrix of one user will result in a change of the mutual information

of all users. It is thus very difficult to solve the problem analytically. Currently, most researchers resort to iterative local optimization methods for solving this problem.

In [74], Jorswieck and Boche analyzed the worst-case performance of a multiuser MIMO system with interference. In [32, 33], Demirkol and Ingram introduced an iterative method based on stream control. This algorithm is based on a trial-and-error scheme and only considers simple network configurations, e.g., rectangular or hexagonal network topology. An interesting work was reported in [21], where Chen and Gans analyzed the network spectral efficiency of a MIMO ad hoc network with L simultaneous transmission pairs. They showed that, in the absence of channel state information (CSI) at the transmitters, the network's asymptotic spectral efficiency is limited by r nats/s/Hz as $L \to \infty$, and at least $t + r + 2\sqrt{tr}$ nats/s/Hz when CSI is available at the transmitters. In contrast to such scaling law analysis, which shows trend for very large networks, in this chapter, we are interested in designing an algorithm that can compute the exact maximum capacity for a given (finite size) network topology.

Perhaps the most relevant work to this chapter is [159], where Ye and Blum proposed using a gradient projection (GP) method, which is an extension of the well-known steepest descent method coupled with gradient projection. However, GP can be classified as local optimization techniques, which can quickly find a local optimal solution, but cannot guarantee global optimality for non-convex optimization problems.

In this chapter, we investigate the problem from a global optimization perspective. We propose a solution procedure based on *branch-and-bound* (BB) framework coupled with the *reformulation linearization technique* (BB/RLT). The main contributions of

this work include the mathematical developments of the solution procedure to solve the problem based on BB/RLT technique, and its convergence speedup techniques. Specifically, we derive tight upper and lower bounds for each potential partitioning variable used for the problem. Each nonlinear term is relaxed with a set of linear constraints based on the bounds we develop to generate a higher dimensional upper-bounding problem. We also utilize a polyhedral outer approximation method to accurately approximate the logarithmic function. During each iteration of the branch-and-bound procedure, we propose a variable selection policy based not only on the relaxation error, but also on the relative significance of the variables in our problem. Our proposed method guarantees the finding of a global optimal solution for the sum rate capacity of such networks.

The remainder of this chapter is organized as follows. Section 3.2 presents network model and problem formulation. Section 3.3 introduces the BB/RLT framework and key problem-specific components, including factorization, linearization, and convergence speedup techniques. Simulation results are presented in Section 3.4. Section 3.5 concludes this chapter.

3.2 Network Model and Problem Formulation

We consider a network consisting of L interfering concurrent MIMO transmission pairs (links), which are indexed by $1, 2, \ldots, L$. In this chapter, it is assumed that the transmitters have full CSI. Let the matrix $\mathbf{H}_{jl} \in \mathbb{C}^{n_r \times n_t}$ represent the wireless channel gain matrix from the transmitting node of link j to the receiving node of link l, where n_t and n_r are the numbers of transmitting and receiving antenna elements of each node, respectively. Denote ρ_{jl} the signal-to-noise ratio per unit transmit power

if $j = l$, or the interference-to-noise ratio per unit transmit power if $j \neq l$. Denote matrix $\mathbf{Q}_l$ the covariance matrix of a zero-mean Gaussian input symbol vector $\mathbf{x}_l$ at link l, i.e., $\mathbf{Q}_l = \mathbb{E}\left\{\mathbf{x}_l \cdot \mathbf{x}_l^\dagger\right\}$. Assume, also, that all nodes in the network are subject to the same maximum transmitting power constraint, i.e., $\mathrm{Tr}\left\{\mathbf{Q}_l\right\} \leq P_{\max}$, where $P_{\max}$ is the maximum transmission power. Let $\mathbf{R}_l$ represent the covariance matrix of interference plus noise. Define $\mathcal{I}_l$ as the set of links that can produce interference on link l. The interference-plus-noise is Gaussian distributed and its covariance matrix can be computed as

$$\mathbf{R}_l = \sum_{j \in \mathcal{I}_l} \rho_{jl} \mathbf{H}_{jl} \mathbf{Q}_j \mathbf{H}_{jl}^\dagger + \mathbf{I}. \tag{3.2.1}$$

Hence, the mutual information of a MIMO link l with co-channel interference can be computed as $I_l = \log_2 \det\left(\rho_{ll}\mathbf{H}_{ll}\mathbf{Q}_l\mathbf{H}_{ll}^\dagger + \mathbf{R}_l\right) - \log_2 \det \mathbf{R}_l$. Our goal is to maximize the sum of mutual information (MSMI) of this L-link MIMO interference system. Summarizing the previous discussion, this optimization problem can be mathematically formulated as follows:

$$\begin{aligned} \max \quad & \textstyle\sum_{l=1}^{L} I_l \\ \text{s.t.} \quad & I_l = \log_2 \det\left(\rho_{ll}\mathbf{H}_{ll}\mathbf{Q}_l\mathbf{H}_{ll}^\dagger + \mathbf{R}_l\right) - \log_2 \det \mathbf{R}_l \\ & \mathbf{R}_l = \textstyle\sum_{j \in \mathcal{I}_l} \rho_{jl}\mathbf{H}_{jl}\mathbf{Q}_j\mathbf{H}_{jl}^\dagger + \mathbf{I} \\ & \mathrm{Tr}\{\mathbf{Q}_l\} \leq P_{\max}, \mathbf{Q}_l \succeq 0, 1 \leq l \leq L. \end{aligned}$$

In this chapter, we consider a network where each antenna element in a transmitting node employs equal power allocation. It is necessary to point out that, by saying "equal power allocation", we mean the total power at the same source node is equally allocated to its antenna elements, while different source nodes may have different total transmitting power. The reason behind this approach is that an optimal power allocation, wherein different antenna elements at the same source node have different transmitting power level, puts a high demand of linearity in transmit power amplifiers,

which is extremely costly from a practical standpoint [71]. Thus, for low cost hardware implementation, an equal power allocation scheme is more attractive.

Under the equal power allocation approach, the MSMI problem is translated into an optimal power control problem. That is, we are interested in finding an L-dimension power vector $\mathbf{p} = (p_1, p_2, \ldots, p_L)^t$, where $0 < p_l \leq P_{\max}$, $l = 1, 2, \ldots, L$, such that this power vector $\mathbf{p}$ maximizes the sum of mutual information of the links in the network. Mathematically, with equal power allocation to each transmitter at the same node, the input covariance matrix $\mathbf{Q}_l$ becomes an n_t-dimension scaled identity matrix, i.e., $\mathbf{Q}_l = \frac{p_l}{n_t}\mathbf{I}$. Hence, the MSMI problem formulation can be further re-written as follows:

$$\begin{array}{rl} \max & \sum_{l=1}^{L} I_l \\ \text{s.t.} & I_l = \log_2 \det \left(\frac{\rho_{ll} p_l}{n_t} \left(\mathbf{H}_{ll} \mathbf{H}_{ll}^{\dagger} \right) + \mathbf{R}_l \right) - \log_2 \det \mathbf{R}_l \\ & \mathbf{R}_l = \mathbf{I} + \sum_{j \in \mathcal{I}_l} \frac{\rho_{jl} p_j}{n_t} \left(\mathbf{H}_{jl} \mathbf{H}_{jl}^{\dagger} \right) \end{array} \tag{3.2.2}$$

where $0 < p_l \leq P_{\max}, 1 \leq l \leq L$.

3.3 Solution Procedure

3.3.1 Overview of BB/RLT Method

For a non-convex optimization problem, conventional nonlinear programming methods (e.g., [159]) can at best yield local optimal solutions. On the other hand, the proposed BB/RLT procedure in this chapter can find a provably global optimal solution [132–134]. The basic idea of BB/RLT is that, by using the RLT technique, we can construct a linear programming (LP) relaxation for the original nonlinear programming (NLP) problem, which can be used to efficiently compute a global upper bound, UB, for the original NLP problem. This relaxation solution is either a feasible solution to the original NLP problem or, if not feasible, can be used as a

starting point for a local search algorithm to find a feasible solution to the original NLP problem. This feasible solution will then serve to provide a global lower bound, LB, and an incumbent solution to the original NLP problem. However, it is worth pointing out that local search is not necessary in this chapter since the LP relaxation for our problem is still subject to the same power constraint $0 \leq p_l \leq P_{\max}$ of the original problem. As a result, solving the LP relaxation of our problem still gives us a feasible solution to the original NLP problem. This will become clearer after we introduce the LP relaxation in Section 3.3.5. The branch-and-bound process will proceed by tightening UB and LB, and terminates when $LB \geq (1-\epsilon)UB$ is satisfied, where ϵ is the desired approximation error. There is a formal proof that BB/RLT converges to a global optimal solution as long as the partitioning intervals are compact (see [132–134] for further details). The general framework of BB/RLT is shown in Algorithm 3.3.1.

In the remainder of this section, we develop the key components in the BB/RLT framework, which are problem-specific.

3.3.2 Factorization and Linearization

Observe that in the MSMI problem formulation, the link mutual information expressions in (3.2.2) are nonlinear. To linearize these nonlinear constraints, we first introduce four new variables X_l, Y_l, V_l, and W_l as follows.

$$\begin{cases} X_l = \det\left(\frac{\rho_{ll} p_l}{n_t}\left(\mathbf{H}_{ll}\mathbf{H}_{ll}^{\dagger}\right) + \mathbf{R}_l\right) \triangleq \det \mathbf{D}_l, \\ V_l = \det \mathbf{R}_l, \quad Y_l = \ln X_l, \quad W_l = \ln V_l. \end{cases} \tag{3.3.1}$$

The link mutual information constraint in (3.2.2) can then be translated into $I_l = \frac{1}{\ln 2}\left(Y_l - W_l\right)$, which is a linear constraint. Also, four groups of new constraints in (3.3.1) are added to the problem formulation.

Algorithm 3.3.1 BB/RLT Solution Procedure

Initialization:

1. Let optimal solution $\psi^* = \emptyset$. The initial lower bound $LB = -\infty$.
2. Determine partitioning variables (variables associated with nonlinear terms) and derive their initial bounding intervals.
3. Let the initial problem list contain only the original problem, denoted by P_1.
4. Introduce one new variable for each nonlinear term. Add linear constraints for these variables to build a linear relaxation. Denote the solution to linear relaxation as $\hat{\psi}_1$ and its objective value as the upper bound UB_1.

Main Loop:

1. Select problem P_z that has the largest upper bound among all problems in the problem list.
2. Find, if necessary, a feasible solution ψ_z via a local search algorithm for Problem P_z. Denote the objective value of ψ_z by LB_z.
3. If $LB_z > LB$ then let $\psi^* = \psi_z$ and $LB = LB_z$. If $LB \geq (1-\epsilon)UB$ then stop with the $(1-\epsilon)$-optimal solution ψ^*; else, remove all problems $P_{z'}$ having $(1-\epsilon)UB_{z'} \leq LB$ from the problem list.
4. Compute relaxation error for each nonlinear term.
5. Select a partitioning variable having the maximum relaxation error and divide its bounding interval into two new intervals by partitioning at its value in $\hat{\psi}_z$.
6. Remove the selected problem P_z from the problem list, construct two new problems P_{z1} and P_{z2} based on the two partitioned intervals.
7. Compute two new upper bounds UB_{z1} and UB_{z2} by solving the linear relaxations of P_{z1} and P_{z2}, respectively.
8. If $LB < (1-\epsilon)UB_{z1}$ then add problem P_{z1} to the problem list. If $LB < (1-\epsilon)UB_{z2}$ then add problem P_{z2} to the problem list.
9. If the problem list is empty, stop with the $(1-\epsilon)$-optimal solution ψ^*. Otherwise, go to the Main Loop again.

3.3.3 Linear Relaxation to Nonlinear Logarithmic Functions

Next, we propose using a polyhedral outer approximation for the curve of logarithmic function. As shown in Fig. 3.1, the function $y = \ln x$, over an interval defined by suitable upper and lower bounds on x, can be upper-bounded by three tangential segments I, II, and III, which are constructed at $(x_L, \ln x_L)$, $(x_\beta, \ln x_\beta)$, and $(x_U, \ln x_U)$, where x_β is computed as follows:

$$x_\beta = \frac{x_L x_U (\ln x_U - \ln x_L)}{x_U - x_L}. \tag{3.3.2}$$

Here x_β is the x-value for the point at the intersection of the extended tangent segments I and III. Segment IV is the chord that joins $(x_L, \ln x_L)$ and $(x_U, \ln x_U)$. The convex region defined by the four segments can be described by the following four linear constraints:

$$\begin{gathered} x_L \cdot y - x \leq x_L(\ln x_L - 1), \\ x_\beta \cdot y - x \leq x_\beta(\ln x_\beta - 1), \\ x_U \cdot y - x \leq x_U(\ln x_U - 1), \\ (x_U - x_L)y + (\ln x_L - \ln x_U)x \geq x_U \cdot \ln x_L - x_L \cdot \ln x_U. \end{gathered}$$

3.3.4 Linearizing the Determinants

Substituting the expression for $\mathbf{R}_l$ into that for $\mathbf{D}_l$, and observing the similarity between the expressions for $\mathbf{D}_l$ and $\mathbf{R}_l$, we can write $\mathbf{D}_l$ and $\mathbf{R}_l$ in a more compact form by introducing a notion called the "super interference set" of link l, denoted by $\hat{\mathcal{I}}_l$, with $\hat{\mathcal{I}}_l = \mathcal{I}_l \cup \{l\}$, as follows:

$$\begin{aligned} \mathbf{D}_l &= \textstyle\sum_{j \in \hat{\mathcal{I}}_l} \frac{\rho_{jl}}{n_t} \left(\mathbf{H}_{jl} \mathbf{H}_{jl}^\dagger \right) p_j + \mathbf{I}, \\ \mathbf{R}_l &= \textstyle\sum_{j \in \mathcal{I}_l} \frac{\rho_{jl}}{n_t} \left(\mathbf{H}_{jl} \mathbf{H}_{jl}^\dagger \right) p_j + \mathbf{I}. \end{aligned} \tag{3.3.3}$$

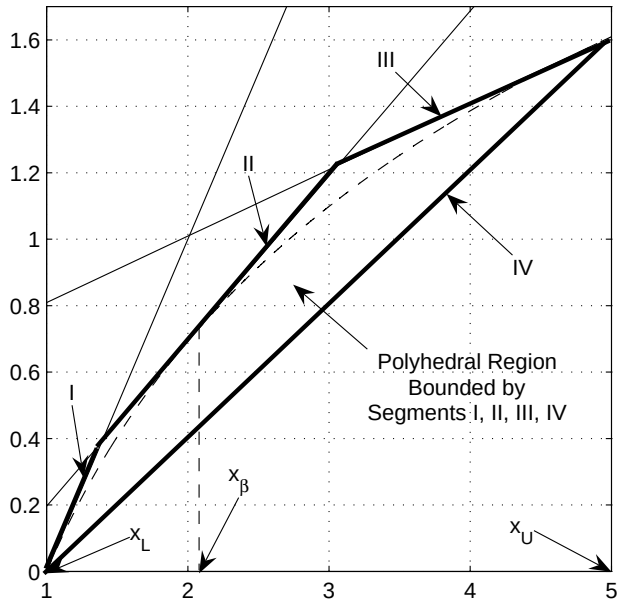


Figure 3.1: Polyhedral outer-approximation $y = \ln x$.

It is evident from (3.3.3) that the determinants of $\mathbf{D}_l$ and $\mathbf{R}_l$ are in essence n_r-order polynomials of the variables $p_1, p_2, \ldots, p_L$. To illustrate how to linearize determinants, let us consider a multiuser MIMO network where every link has two receiving antennas. This means that $\mathbf{D}_l$ and $\mathbf{R}_l$ are 2×2 square matrices. The determinant X_l and V_l are shown in (3.3.4) and (3.3.5), respectively.

$$
\begin{aligned}
X_l = 1 &+ \sum_{j\in\hat{\mathcal{I}}_l} \frac{\rho_{jl}}{n_r}\left[\left(\mathbf{H}_{jl}\mathbf{H}_{jl}^{\dagger}\right)_{(1,1)} + \left(\mathbf{H}_{jl}\mathbf{H}_{jl}^{\dagger}\right)_{(2,2)}\right] p_j \\
&+ \sum_{j\in\hat{\mathcal{I}}_l}\sum_{k\in\hat{\mathcal{I}}_l} \frac{\rho_{jl}\rho_{kl}}{n_r^2}\left[\left(\mathbf{H}_{jl}\mathbf{H}_{jl}^{\dagger}\right)_{(1,1)}\left(\mathbf{H}_{jl}\mathbf{H}_{jl}^{\dagger}\right)_{(2,2)}\right. \\
&-\mathfrak{Re}\left(\left(\mathbf{H}_{jl}\mathbf{H}_{jl}^{\dagger}\right)_{(2,1)}\right)\mathfrak{Re}\left(\left(\mathbf{H}_{kl}\mathbf{H}_{kl}^{\dagger}\right)_{(2,1)}\right) \\
&\left.-\mathfrak{Im}\left(\left(\mathbf{H}_{jl}\mathbf{H}_{jl}^{\dagger}\right)_{(2,1)}\right)\mathfrak{Im}\left(\left(\mathbf{H}_{kl}\mathbf{H}_{kl}^{\dagger}\right)_{(2,1)}\right)\right] p_j p_k. \qquad (3.3.4)
\end{aligned}
$$

$$
\begin{aligned}
V_l = 1 &+ \sum_{j\in\mathcal{I}_l} \frac{\rho_{jl}}{n_r}\left[\left(\mathbf{H}_{jl}\mathbf{H}_{jl}^{\dagger}\right)_{(1,1)} + \left(\mathbf{H}_{jl}\mathbf{H}_{jl}^{\dagger}\right)_{(2,2)}\right] p_j \\
&+ \sum_{j\in\mathcal{I}_l}\sum_{k\in\mathcal{I}_l} \frac{\rho_{jl}\rho_{kl}}{n_r^2}\left[\left(\mathbf{H}_{jl}\mathbf{H}_{jl}^{\dagger}\right)_{(1,1)}\left(\mathbf{H}_{jl}\mathbf{H}_{jl}^{\dagger}\right)_{(2,2)}\right. \\
&\quad -\mathfrak{Re}\left(\left(\mathbf{H}_{jl}\mathbf{H}_{jl}^{\dagger}\right)_{(2,1)}\right)\mathfrak{Re}\left(\left(\mathbf{H}_{kl}\mathbf{H}_{kl}^{\dagger}\right)_{(2,1)}\right) \\
&\left.-\mathfrak{Im}\left(\left(\mathbf{H}_{jl}\mathbf{H}_{jl}^{\dagger}\right)_{(2,1)}\right)\mathfrak{Im}\left(\left(\mathbf{H}_{kl}\mathbf{H}_{kl}^{\dagger}\right)_{(2,1)}\right)\right] p_j p_k. \qquad (3.3.5)
\end{aligned}
$$

We see that the product terms $p_j p_k$ are the only nonlinear terms, which need to be linearized. To show how linearization works, let us consider a general second-order polynomial term $p_j p_k$, for which we have the following bounding constraints:

$$
\begin{aligned}
&p_j - (p_j)_L \geq 0, \quad (p_j)_U - p_j \geq 0, \\
&p_k - (p_k)_L \geq 0, \quad (p_k)_U - p_k \geq 0,
\end{aligned} \qquad (3.3.6)
$$

where $(p_j)_L$ and $(p_k)_L$ denote the lower bounds of p_j and p_k, respectively, and $(p_j)_U$ and $(p_k)_U$ denote the upper bounds of p_j and p_k, respectively. Adopting RLT [132],

we can derive the following four so-called bounding-factor constraints:

$$p_j p_k - (p_k)_L p_j - (p_j)_L p_k \geq -(p_j)_L (p_k)_L,$$
$$p_j p_k - (p_k)_U p_j - (p_j)_L p_k \leq -(p_j)_L (p_k)_U,$$
$$p_j p_k - (p_k)_L p_j - (p_j)_U p_k \leq -(p_j)_U (p_k)_L,$$
$$p_j p_k - (p_k)_U p_j - (p_j)_U p_k \geq -(p_j)_U (p_k)_U.$$

In particular, if $j = k$, $p_j p_k$ is given by a general square term p_j^2. Using the following bounding constraints:

$$p_j - (p_j)_L \geq 0 \quad \text{and} \quad (p_j)_U - p_j \geq 0, \tag{3.3.7}$$

we can derive the following three bounding-factor constraints:

$$p_j^2 - 2(p_j)_L p_j \geq -(p_j)_L^2, \quad p_j^2 - 2(p_j)_U p_j \geq -(p_j)_U^2,$$
$$p_j^2 - ((p_j)_L + (p_j)_U) p_j \leq -(p_j)_U (p_j)_L.$$

We now introduce new variables P_{jk} to replace the product terms $p_j p_k$, and P_{jj} to replace the square term p_j^2, respectively. By doing so, the determinant expressions of X_l and V_l become linear constraints. Also, the equality relation $P_{jk} = p_j p_k$ will be relaxed by the above bounding-factor constraint relaxations. All these newly introduced bounding-factor constraints will be appended to the original problem, thus achieving a LP relaxation for the constraints in the original NLP problem.

3.3.5 RLT-Based Relaxation

For convenience, we define the right hand sides of (3.3.4) and (3.3.5) as $\Phi_{X_l}(\mathbf{p})$ and $\Phi_{V_l}(\mathbf{p})$, respectively. From the discussions in the previous sections, we have the final

R-MSMI formulation for a multiuser MIMO system as follows:

R-MSMI

$$
\begin{aligned}
\max \quad & \textstyle\sum_{l=1}^{L} I_l \\
\text{s.t.} \quad & I_l - \tfrac{1}{\ln 2}\left(Y_l - W_l\right) = 0, \quad \forall l. \\
& \text{Three tangential supports for } (Y_l, X_l), \quad \forall l. \\
& \text{Three tangential supports for } (W_l, V_l), \quad \forall l. \\
& X_l - \Phi_{X_l}(\mathbf{p}) = 1, V_l - \Phi_{V_l}(\mathbf{p}) = 1, \quad \forall l. \\
& \text{Bounding constraints for } P_{jk} \text{ and } P_{jj}. \\
& \text{Same power constraints for power variables } p_l, \quad \forall l.
\end{aligned}
$$

3.3.6 Partitioning Variables and Their Upper and Lower Bounds

The partitioning variables in the branch-and-bound process are those that are involved in nonlinear terms, for which we have therefore defined new variables, and whose bounding intervals will need to be partitioned during the RLT-based branch-and-bound algorithm [132–134]. In R-MSMI, these partitioning variables include X_l, V_l, and p_l, $l = 1, 2, \ldots, L$. The upper and lower bounds for p_l are $(p_l)_L = 0$, $(p_l)_U = P_{\max}$, for $l = 1, \ldots, L$. From these expressions, we see that the upper bounds for X_l and V_l can, respectively, be computed as $\Phi_{X_l}(P_{\max} \times \mathbf{1})$ and $\Phi_{V_l}(P_{\max} \times \mathbf{1})$, with $\mathbf{1}, \mathbf{0} \in \mathbb{R}^L$. The lower bounds for X_l and V_l are: $(X_l)_L = 1$, and $(V_l)_L = 1$.

3.3.7 Convergence Speedup Techniques

In the worst case, BB/RLT has exponential complexity. However, it is possible to exploit certain special structure of the underlying problem to speedup convergence time. For our problem, since the decrease of the global upper bound plays the most critical role in the convergence process, partitioning variables that are able to tighten the upper bound (i.e., X_l and V_l as shown in Fig. 3.2) should be selected first. Hence,

we adopt the following convergence speedup technique, as shown in Algorithm 3.3.2. In our numerical results, the threshold value ϵ_1 for $\ln(Z_l^*)_U - \ln(Z_l^*)_L$ are set to 1 since we find that threshold value less than 1 does not further improve the accuracy of the final solution. The relaxation error refers to the difference between the newly introduced variable and its corresponding product terms (e.g., $P_{ij} - p_i p_j$). The threshold value ϵ_2 for testing the relaxation errors are 10^{-3} in this chapter.

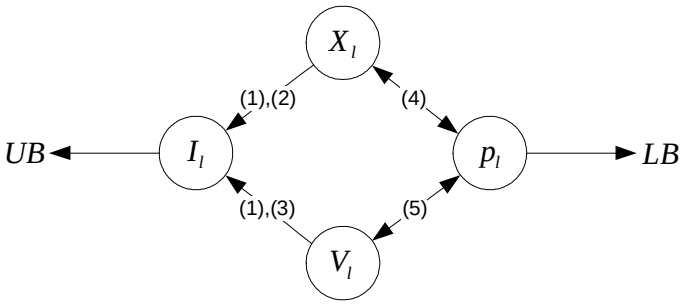


Figure 3.2: Relationship among BB variables in MSMI.

Algorithm 3.3.2 Modified BB Variable Selection Strategy

1. Among all X_l and V_l, choose the one having the largest relaxation error and denoted it as Z_l^*.
2. If $(\ln(Z_l^*)_U - \ln(Z_l^*)_L \leq \epsilon_1)$ then
   a) Among all p_l, choose one, say p_l^*, with the largest relaxation error.
   Denoted this relaxation error as E_p;
   b) If $E_p \leq \epsilon_2$, then remove this subproblem; else return p_l^*;
   else return Z_l^*.

3.4 Numerical Results

We first describe our simulation settings. L links are uniformly distributed within a square region. Each node in the network is equipped with two antennas. The maximum transmit power for each node is set to $P_{\max} = 10$ dBm. The network operates in 2.4 GHz ISM band. The channel bandwidth is $W = 30$ MHz. The pathloss index is chosen to be $\alpha = 2$.

We use two 5-link network examples to demonstrate the convergence properties of BB/RLT. The first one is an example of a heavily interfered network where each link is interfered by every other link. The desired error bound is chosen to be $\epsilon = 0.01$. That is, the iterative process stops once $LB \geq (1-\epsilon)UB$. The network's SNR- and INR-values are shown in Table 3.1. For example, a cell intersected by row i and column j contains the INR (in dB) from link i to link j. The convergence process is depicted in Fig. 3.3.

Table 3.1: SNR- and INR-values for a heavily interfered 5-link network (in dB).

	SNR (in dB)	INR (in dB)					
			L0 Rx	L1 Rx	L2 Rx	L3 Rx	L4 Rx
L0	20.98	L0 Tx	–	13.57	3.79	9.13	2.23
L1	27.04	L1 Tx	18.90	–	6.33	12.38	4.35
L2	20.67	L2 Tx	4.31	6.61	–	7.53	13.39
L3	21.03	L3 Tx	7.39	9.48	9.29	–	4.26
L4	22.57	L4 Tx	4.10	6.19	11.61	5.58	–

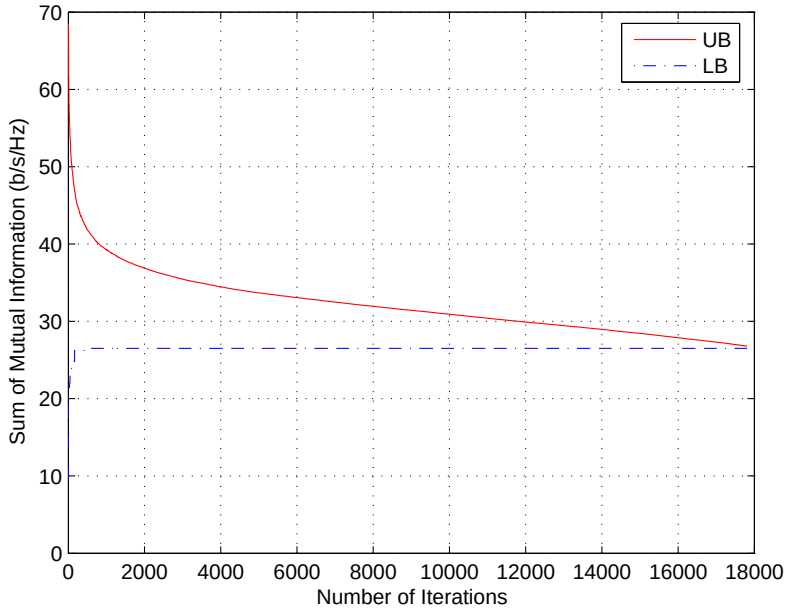


Figure 3.3: A 5-link heavily interfered network example.

Table 3.2: SNR- and INR-values for a less interfered 5-link network.

	SNR (in dB)	INR (in dB)					
			L0 Rx	L1 Rx	L2 Rx	L3 Rx	L4 Rx
L0	20.98	L0 Tx	–	10.98	1.75	0.56	3.48
L1	27.04	L1 Tx	14.84	–	4.02	3.03	7.34
L2	20.67	L2 Tx	1.85	3.41	–	9.02	4.19
L3	21.03	L3 Tx	0.74	2.89	7.93	–	7.74
L4	22.18	L4 Tx	3.17	6.29	4.41	7.51	–

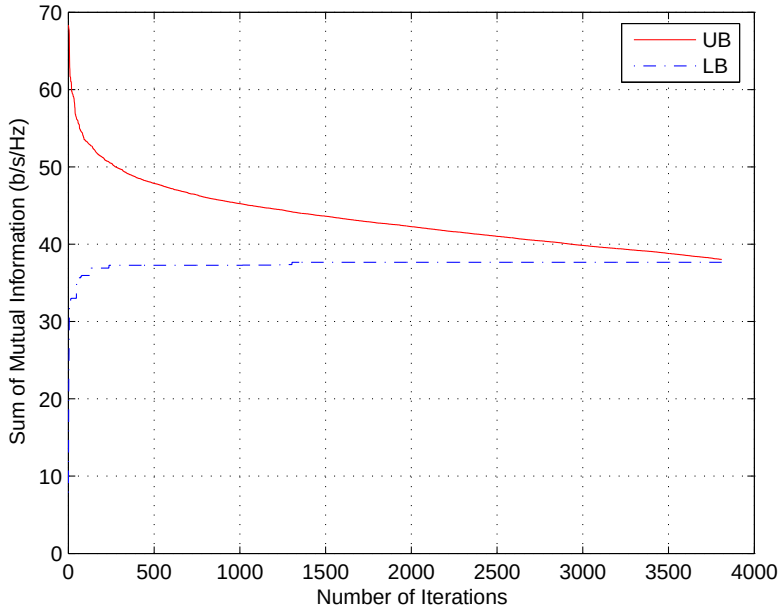


Figure 3.4: A 5-link less interfered network example.

Fig. 3.3 illustrates the UB and LB in terms of the sum of mutual information (b/s/Hz) at each iteration. In this heavily interfered example, after 17000 iterations, the UB and LB values are both driven to 26.51 b/s/Hz, meaning that the global optimum for the MSMI is 26.51 b/s/Hz. In this example, the optimal power vector is $\mathbf{p} = \begin{bmatrix} p_0 & p_1 & p_2 & p_3 & p_4 \end{bmatrix}^t = \begin{bmatrix} 8.904 & 0.071 & 2.83 & 1.096 & 2.83 \end{bmatrix}^t$ (in mW).

To see how significantly the system's performance could be degraded by interference, we compute the sum of total mutual information for the same network as if

Table 3.3: SNR- and INR-values for a 10-link network.

	SNR (in dB)	INR (in dB)										
			L0 Rx	L1 Rx	L2 Rx	L3 Rx	L4 Rx	L5 Rx	L6 Rx	L7 Rx	L8 Rx	L9 Rx
L0	23.19	L0 Tx	–	-0.12	2.29	-2.34	8.01	-1.47	4.73	2.59	-0.28	-3.37
L1	23.99	L1 Tx	0.43	–	8.67	7.54	2.12	0.79	7.98	4.76	-1.53	1.85
L2	20.44	L2 Tx	2.75	8.69	–	2.79	3.58	-0.52	12.74	4.19	-1.77	-0.62
L3	22.41	L3 Tx	-2.20	7.61	1.44	–	-0.22	2.86	2.09	3.32	-1.21	8.34
L4	21.63	L4 Tx	9.11	1.21	1.76	-0.63	–	2.15	5.74	7.88	4.04	-1.08
L5	23.82	L5 Tx	-0.90	1.83	-0.89	3.00	2.18	–	1.31	6.76	5.46	6.86
L6	26.59	L6 Tx	4.99	7.51	9.15	2.81	7.64	1.39	–	8.47	0.27	0.28
L7	23.58	L7 Tx	3.35	4.02	2.27	2.60	9.00	6.76	6.47	–	5.08	2.43
L8	25.23	L8 Tx	0.79	-1.25	-2.18	-1.51	3.90	5.88	0.22	4.44	–	-0.07
L9	25.56	L9 Tx	-3.33	1.97	-1.64	6.03	-1.24	5.56	-0.53	2.09	-0.09	–

there is no interference. In this particular example, the total mutual information for the case of no interference is obtained as 71.2 b/s/Hz. Thus, it can be seen that, even after carefully choosing the optimal power vector $\mathbf{p}$, the spectral efficiency only accounts for approximately 37% of that of the no interference case.

It can also be seen from Fig. 3.3 that the rate of decrease in UB plays the major role in determining how fast the overall BB/RLT process converges. In this example, UB starts out from 68.47 b/s/Hz, and changes by 41.96 in magnitude by the time the algorithm terminates. Comparatively, LB starts out from 8.37 b/s/Hz and it only changes 18.14 in magnitude by the end of the convergence process. Moreover, the rate of decline of UB becomes slower as it approaches the global optimum.

Next, for comparison, we study another 5-link network having less interference as compared to the previous example. The network's SNR- and INR-values are shown in Table 3.2. The convergence process is depicted in Fig. 3.4. In this example, BB/RLT converges to the global optimal rather quickly (in about 3810 iterations), yielding an optimum value of 37.65 b/s/Hz. This convergence speed is comparable to many fast local search algorithm, such as GGP [159] and IWF [163]. For this particular example, we also compute the MSMI assuming there is no interference,

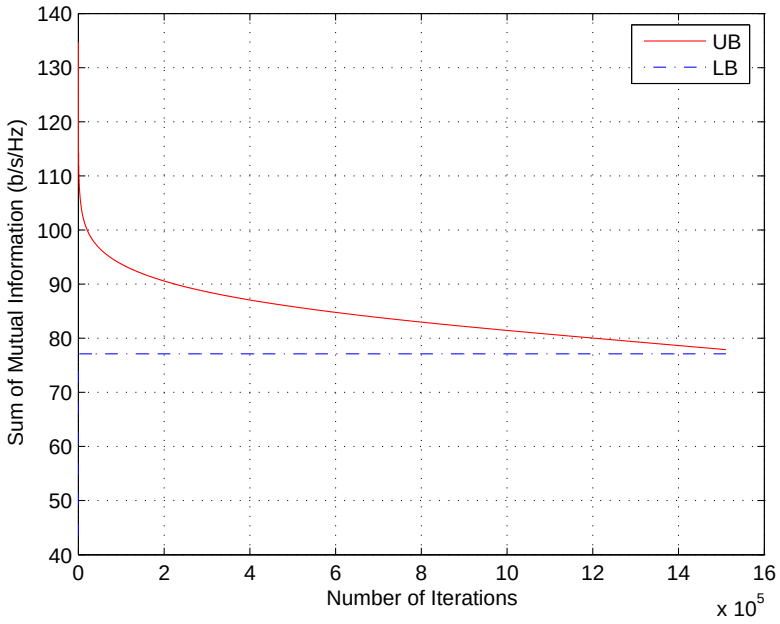


Figure 3.5: A 10-link network example.

which gives a value of 74.59 b/s/Hz. That is, in this less interfered network, the spectral efficiency is approximately 50% of that of the no interference case. For this less interfered example, the optimal power vector is $\mathbf{p} = \begin{bmatrix} p_0 & p_1 & p_2 & p_3 & p_4 \end{bmatrix}^t = \begin{bmatrix} 1.815 & 0.435 & 4.371 & 2.973 & 5.942 \end{bmatrix}^t$ (in mW). The reason why the convergence time is much smaller than the previous heavily interfered network example is because the initial upper bounds of $(X_l)_U$ and $(V_l)_U$ are much smaller. This means that we have relatively a small interval for X_l and V_l variables to partition, which contributes to a faster convergence speed. Also, in a less interfered network, the initial gap between UB and LB is smaller. In this particular example, LB reaches 36.65 b/s/Hz only after 49 iterations, which is already very close to the final optimal value of 37.65 b/s/Hz. At the 49th iteration, UB is at 56.15 b/s/Hz. This gap between LB and UB is about 19.5, which is much smaller than that in the first example. For these

reasons, we can see that the less interference in the network, the faster BB/RLT will converge to find the global optimal solution.

To shed light on the huge effect of using the modified partitioning variable selection strategy, we consider the following 10-link network example, whose SNR- and INR-values are shown in Table 3.3. The convergence process is depicted in Fig. 3.5. The global optimal value for this 10-link network is 77.11 b/s/Hz. It takes approximately 1.5×10^6 iterations to converge with the modified partitioning variable selection strategy (Algorithm 3.3.2). On the other hand, if we solve the same 10-link example using BB/RLT without using this strategy, the convergence time will be much slower. Via rough estimate, we find that our proposed speedup technique can accelerate the convergence by at least 1000 times for this particular example.

3.5 Chapter Summary

In this chapter, we studied the maximum sum of mutual information problem for multiuser MIMO network. We proposed a powerful global optimization method using a branch-and-bound framework coupled with the reformulation-linearization technique (BB/RLT). We also proposed a modified branch-and-bound variable selection strategy to accelerate the convergence process. Numerical examples are given to demonstrate the efficacy of the proposed solution.

Part II

Multi-hop MIMO Wireless Networks

Chapter 4

Multi-hop MIMO Networks with Orthogonal Channels

4.1 Introduction

Wireless ad hoc network has been a focal research area in the research community for some years. However, compared to the research on single-user MIMO, for which many results are available (see [45, 48] and references therein), research on multiuser MIMO systems is still in its infancy and many fundamental problems, particularly for multi-hop ad hoc networks, remain unsolved. Employing MIMO in an ad hoc network is far from trivial. As discussed by Winters [157], a MIMO-based ad hoc network with each node equipped with M antennas does not necessarily mean that the network capacity is also increased by M-fold. The potential network capacity gain with the use of MIMO depends on the coordinated mechanisms at the physical, link, and network layers. An improperly designed algorithm could diminish any potential capacity gain from MIMO. As a result, joint optimization across multiple layers is not only desirable, but also necessary.

In this chapter, we investigate cross-layer optimization for MIMO-based ad hoc

networks under orthogonal channels. The goal is to support a set of user communication sessions such that some network utility is maximized. However, to achieve high capacity for such networks, many challenging problems must be addressed. We are interested in determining the optimal power allocation at each transmitting node, the optimal bandwidth allocation for each transmission, and the optimal flow routing for the network. This problem is considerably more challenging than that in conventional single antenna-based wireless ad hoc networks. This is because, compared to the simple scalar channels in the single antenna case, power allocations are now performed over complex matrix channels. Also, compared to single-user MIMO systems, power allocation for multiple outgoing links at a node has to be jointly considered.

For this challenging cross-layer optimization problem, we develop a solution procedure that can solve this problem to optimality. We show that it has some special structure which allows us to decompose the original problem into a set of subproblems in its dual domain. Specifically, our solution procedure first decouples the dual problem into a network layer subproblem and a link-physical layer subproblem. For the link-physical layer subproblem (corresponding to multi-antenna power allocation and bandwidth allocation), which is the most difficult part in the dual problem, we employ techniques in matrix differential calculus and develop an efficient algorithm using gradient projection. By exploiting the piece-wise quadratic structure of the projection subproblem, our proposed method for projection enjoys polynomial-time complexity. Then, for the dual problem, we propose two strategies, i.e., a cutting-plane method based on outer-linearization and the subgradient-based scheme. The cutting-plane method is a centralized algorithm and is more efficient and robust compared with the subgradient method. Also, our proposed cutting-plane method allows

an easy recovery of primal feasible and optimal solutions. However, the major benefit of the subgradient method is that it can be implemented in a distributed fashion. Also, the distributed algorithm can achieve the same optimal solution as that of the cutting-plane method. We show that the excess link capacity of each link can serve as the exchanging message for distributed cross-layer optimization.

The remainder of this chapter is organized as follows. In Section 4.2, we present the network model and problem formulation. Section 4.3 presents our decomposition framework and key subproblems to be solved at each layer. In Section 4.4, we focus on the challenging physical-link layer subproblem and the design of gradient projection method. In Section 4.5, we propose the cutting-plane and the subgradient methods for solving the dual problem, respectively. Section 4.6 discusses the design of a distributed algorithm. Section 4.7 presents numerical results. Section 4.8 concludes this chapter.

4.2 Network Model

4.2.1 MIMO Power Allocation

Let the matrix $\mathbf{H}_l \in \mathbb{C}^{n_r \times n_t}$ represent the wireless channel gain matrix from the transmitting node to the receiving node of link l, where n_t and n_r are the numbers of transmitting and receiving antenna elements at each node, respectively. Suppose that $\mathbf{H}_l$ is known at the transmitting node of link l. Although wireless channels in reality are time-varying, we consider a "constant" channel model in this chapter, i.e., $\mathbf{H}_l$'s coherence time is larger than the transmission period we consider. This simplification is still of much interest for it enables to find the ergodic capacity for block-wise fading channels [48]. The received complex base-band signal vector for MIMO link l with n_t

transmitting antennas and n_r receiving antennas in a Gaussian channel is given by

$$\mathbf{r}_l = \sqrt{\rho_l}\mathbf{H}_l\mathbf{t}_l + \mathbf{n}_l, \tag{4.2.1}$$

where $\mathbf{r}_l$ and $\mathbf{t}_l$ represent the received and transmitted signal vectors, $\mathbf{n}_l$ is the normalized additive white Gaussian noise (AWGN) vector, ρ_j captures path-loss effect.

Let matrix $\mathbf{Q}_l$ represent the covariance matrix of a zero-mean Gaussian input symbol vector $\mathbf{t}_l$ at link l, i.e., $\mathbf{Q}_l = \mathbb{E}\left\{\mathbf{t}_l \cdot \mathbf{t}_l^\dagger\right\}$. The definition of $\mathbf{Q}_l$ implies that it is Hermitian and PSD. Physically, $\mathbf{Q}_l$ represents the power allocation in different antennas in link l's transmitting node. In this chapter, we use matrix $\mathbf{Q} \triangleq \left[\begin{array}{cccc} \mathbf{Q}_1 & \mathbf{Q}_2 & \ldots & \mathbf{Q}_L \end{array}\right] \in \mathbb{C}^{n_t \times (n_t \cdot L)}$ to denote the collection of all input covariance matrices in the network.

We define $\mathcal{O}(n)$ and $\mathcal{I}(n)$ as the sets of links that are outgoing from and incoming to node n, respectively. At the physical layer, since the total transmit power of each node is subject to a maximum power constraint, we have that $\sum_{l\in\mathcal{O}(n)} \mathrm{Tr}\{\mathbf{Q}_l\} \leq P_{\max}^{(n)}$, $1 \leq n \leq N$, where $P_{\max}^{(n)}$ represents the maximum transmit power of node n.

4.2.2 Link Capacity and Bandwidth Allocation

The link capacity of a MIMO link l in a AWGN channel can be computed as

$$\Phi_l(W_l, \mathbf{Q}_l) \triangleq W_l \log_2 \det\left(\mathbf{I} + \rho_l \mathbf{H}_l \mathbf{Q}_l \mathbf{H}_l^\dagger\right), \tag{4.2.2}$$

where W_l represents the communication bandwidth of link l. It can be readily verified that $\Phi_l(W_l, \mathbf{Q}_l)$ is a monotone increasing concave function for $W_l > 0$ and $\mathbf{Q}_l \succeq 0$. It can be seen that the values of W_l and $\mathbf{Q}_l$, i.e., the allocation of bandwidth and power on link l, directly affect the capacity of link l. As a result, bandwidth allocation variables W_l together with power allocation variables $\mathbf{Q}_l$ play an important role in our cross-layer optimization problem.

Since the total bandwidth of all outgoing links at a node, say node n, cannot exceed its assigned bandwidth, denoted by B_n, we have $\sum_{l \in \mathcal{O}(n)} W_l \leq B_n$, $1 \leq n \leq N$. We denote matrix $\mathbf{W} = \begin{bmatrix} W_1 & W_2 & \dots & W_L \end{bmatrix}^T \in \mathbb{R}^{L \times 1}$ the collection of all bandwidth variables.

4.2.3 Routing

The topology of a MIMO-based wireless ad hoc network is represented by a directed graph, denoted by $\mathcal{G} = \{\mathcal{N}, \mathcal{L}\}$, where $\mathcal{N}$ and $\mathcal{L}$ are the set of nodes and all possible MIMO links, respectively. We assume that $\mathcal{G}$ is always connected. Suppose that the cardinalities of the sets $\mathcal{N}$ and $\mathcal{L}$ are $|\mathcal{N}| = N$ and $|\mathcal{L}| = L$, respectively. In our model, a link (the line segment defined by a pair of nodes) exists if the link is shorter than or equal to the maximum transmission range R_T, i.e., $\mathcal{L} = \{(i,j) : D_{ij} \leq R_T,\, i,j \in \mathcal{N}, i \neq j\}$, where D_{ij} represents the distance between node i and node j. R_T can be determined by a node's maximum transmission power.

It is worth pointing out that link does not physically exist in wireless networks and any node pair may be treated as a possible link. However, in practice, some nodes are separated so far away from each other that the channel gain is very low. In such a case, even if the transmitting node allocates all its power to this weak link, the received SNR remains low enough such that the link capacity is practically zero (i.e., impossible to decode due to the low received SNR). As a result, such node pairs can be safely removed from the set of possible links without affecting the accuracy of the final optimal solution. This is because allocating power to "bad" links leads to virtually no capacity gain in these links while significantly penalizing the capacities of other "good" links, which obviously could not be optimal.

The network topology of $\mathcal{G}$ can be represented by a *node-arc incidence matrix*

(NAIM) [9] $\mathbf{A} \in \mathbb{R}^{N \times L}$, whose entry a_{nl} associating with node n and are (link) l is defined as

$$a_{nl} = \begin{cases} 1 & \text{if } n \text{ is the transmitting node of arc } l \\ -1 & \text{if } n \text{ is the receiving node of arc } l \\ 0 & \text{otherwise.} \end{cases} \tag{4.2.3}$$

We use a multi-commodity flow model for the routing of data packets across a wireless ad hoc network. That is, in a wireless ad hoc network, some source nodes send different data to their intended destination nodes, possibly through *multi-path* and *multi-hop* routing. We assume that the flow conservation law at each node is satisfied, i.e., the network is a flow-balanced system. Suppose that there is a total of F sessions in the network, representing F different commodities. We denote the source and destination nodes of session f, $1 \leq f \leq F$ as $\text{src}(f)$ and $\text{dst}(f)$, respectively. For session f, we denote $\mathbf{s}_f \in \mathbb{R}^N$ the *source-sink vector*, whose entries, other than at the positions of $\text{src}(f)$ and $\text{dst}(f)$, are all zeros. In addition, from the flow conservation law, we must have $(\mathbf{s}_f)_{\text{src}(f)} = -(\mathbf{s}_f)_{\text{dst}(f)}$. Without loss of generality, we let $(\mathbf{s}_f)_{\text{src}(f)} \geq 0$ and simply denote it as a scalar s_f. Therefore, we can further write the source-sink vector of session f in the form of

$$\mathbf{s}_f = s_f \begin{bmatrix} \cdots & 1 & \cdots & -1 & \cdots \end{bmatrix}^T, \tag{4.2.4}$$

where the dots represent zeros, and 1 and -1 are in the positions of $\text{src}(f)$ and $\text{dst}(f)$, respectively. Note that for the source-sink vector of a session f, 1 does not necessarily appear before -1 as in (4.2.4), which is only for illustrative purpose. Using the notation "$=_{x,y}$" to represent the component-wise equality of a vector except at the x^{th} and the y^{th} entries, we have $\mathbf{s}_f =_{\text{src}(f),\text{dst}(f)} \mathbf{0}$. In addition, denoting matrix

$\mathbf{S} \triangleq \begin{bmatrix} \mathbf{s}_1 & \mathbf{s}_2 & \dots & \mathbf{s}_F \end{bmatrix} \in \mathbb{R}^{N \times F}$ the collection of all source-sink vectors $\mathbf{s}_f$, we have

$$\mathbf{S}\mathbf{e}_f =_{\mathrm{src}(f),\mathrm{dst}(f)} \mathbf{0}, \quad 1 \le f \le F,$$

$$\langle \mathbf{1}, \mathbf{S}\mathbf{e}_f \rangle = 0, \qquad 1 \le f \le F,$$

$$(\mathbf{S}\mathbf{e}_f)_{\mathrm{src}(f)} = s_f, \qquad 1 \le f \le F,$$

where $\mathbf{e}_f$ is the f^{th} unit column vector.

For link l, we let $x_l^{(f)} \ge 0$ be the amount of flow of session f on link l. We define $\mathbf{x}^{(f)} \in \mathbb{R}^L$ as the *flow vector* for session f. At each node n, components of the flow vector and source-sink vector for the same session satisfy the following flow conservation law:

$$\sum_{l \in \mathcal{O}(n)} x_l^{(f)} - \sum_{l \in \mathcal{I}(n)} x_l^{(f)} = (\mathbf{s}_f)_n, \quad 1 \le n \le N,\ 1 \le f \le F.$$

With NAIM, the flow conservation law for the entire network can be written as $\mathbf{A}\mathbf{x}^{(f)} = \mathbf{s}_f, \quad 1 \le f \le F$. Denote matrix $\mathbf{X} \triangleq \begin{bmatrix} \mathbf{x}^{(1)} & \mathbf{x}^{(2)} & \dots & \mathbf{x}^{(F)} \end{bmatrix} \in \mathbb{R}^{L \times F}$ the collection of all flow vectors $\mathbf{x}^{(f)}$. Then, the flow conservation law can be written as

$$\mathbf{A}\mathbf{X} = \mathbf{S}.$$

Since the network flow traversing a link cannot exceed the link's capacity limit, we have

$$\sum_{f=1}^{F} x_l^{(f)} \le \Phi_l(W_l, \mathbf{Q}_l), \quad 1 \le l \le L,$$

Using matrix-vector notation, it can be compactly written as

$$\langle \mathbf{1}, \mathbf{X}^T \mathbf{e}_l \rangle \le \Phi_l(W_l, \mathbf{Q}_l), \quad 1 \le l \le L.$$

Note that in a wireless network, link capacity $\Phi_l(W_l, \mathbf{Q}_l)$ is not fixed and can be varied as the power allocation and bandwidth allocation change.

In summary, the multicommodity network flow model imposes the following group of constraints:

$$\begin{cases} \mathbf{AX} = \mathbf{S}, \\ \mathbf{X} \geq \mathbf{0}, \\ \langle \mathbf{1}, \mathbf{X}^T \mathbf{e}_l \rangle \leq \Phi_l(W_l, \mathbf{Q}_l), \quad \forall l, \end{cases}$$

where $\mathbf{S}$ satisfies $\mathbf{Se}_f =_{\mathrm{src}(f),\mathrm{dst}(f)} \mathbf{0}$, $\langle \mathbf{1}, \mathbf{Se}_f \rangle = 0$, and $(\mathbf{Se}_f)_{\mathrm{src}(f)} = s_f$, for $f = 1, 2, \ldots, F$.

4.2.4 Problem Formulation

We consider an FDMA MIMO-based ad hoc network, where each node has been assigned non-overlapping (possibly reused) frequency bands for its incoming and outgoing links so that nodes can simultaneously transmit and receive, and cause no interference among each other. There is a vast amount of literature on how to perform channel assignments and its discussion is beyond the scope of this chapter. In this chapter, we focus on how to jointly optimize routing at the network layer, bandwidth allocation at the link layer, and power allocation at the physical layer. We adopt the proportional fairness utility function, i.e., $\ln(s_f)$ for session f [75]. The objective is to maximize the sum of utilities of all sessions. Putting together the physical layer constraints in Subsection 4.2.1, the link layer constraints in Subsection 4.2.2, and the network layer constraints in Subsection 4.2.3, we have the problem formulation of

cross-layer routing and power/bandwidth allocation (CRPBA) as follows.

$$
\begin{aligned}
\textbf{CRPBA}: \text{Maximize} \quad & \textstyle\sum_{f=1}^{F} \ln(s_f) & \\
\text{subject to} \quad & \mathbf{AX} = \mathbf{S} & \\
& \mathbf{X} \geq \mathbf{0} & \\
& \mathbf{S}\mathbf{e}_f =_{\mathrm{src}(f),\mathrm{dst}(f)} \mathbf{0} & \forall\, f \\
& \langle \mathbf{1}, \mathbf{S}\mathbf{e}_f \rangle = 0 & \forall\, f \\
& (\mathbf{S}\mathbf{e}_f)_{\mathrm{src}(f)} = s_f & \forall\, f \\
& \langle \mathbf{1}, \mathbf{X}^T \mathbf{e}_l \rangle \leq \Phi_l(W_l, \mathbf{Q}_l) & \forall\, l \\
& \textstyle\sum_{l \in \mathcal{O}(n)} \mathrm{Tr}\{\mathbf{Q}_l\} \leq P_{\max}^{(n)} & \forall\, n \\
& \textstyle\sum_{l \in \mathcal{O}(n)} W_l \leq B_n & \forall\, n \\
& \mathbf{Q}_l \succeq 0,\ W_l \geq 0 & \forall\, l \\
\text{Variables}: \quad & \mathbf{S},\ \mathbf{X},\ \mathbf{Q}, \mathbf{W} &
\end{aligned}
$$

4.3 Problem Decomposition

We exploit the following special structure in CRPBA: The network layer variables and the link layer variables are coupled through the link capacity constraints $\langle \mathbf{1}, \mathbf{X}^T \mathbf{e}_l \rangle \leq \Phi_l(W_l, \mathbf{Q}_l)$. Thus, we can employ Lagrangian dual decomposition to solve CRPBA efficiently. Generally, given a nonlinear programming problem, several different Lagrangian dual problems can be constructed depending on which constraints are associated with Lagrangian dual variables [10]. For CRPBA, we associate a Lagrangian multiplier u_l to link l's coupling constraint $\langle \mathbf{1}, \mathbf{X}^T \mathbf{e}_l \rangle \leq \Phi_l(W_l, \mathbf{Q}_l)$. Let vector $\mathbf{u} \triangleq \begin{bmatrix} u_1 & u_2 & \dots & u_L \end{bmatrix}$ represent the collection of all dual variables. Hence, the Lagrangian can be written as

$$
\Theta(\mathbf{u}) = \sup_{\mathbf{S},\mathbf{X},\mathbf{Q},\mathbf{W}} \{ L(\mathbf{S}, \mathbf{X}, \mathbf{Q}, \mathbf{W}, \mathbf{u}) | (\mathbf{S}, \mathbf{X}, \mathbf{Q}, \mathbf{W}) \in \Gamma \},
$$

where

$$L(\mathbf{S},\mathbf{X},\mathbf{Q},\mathbf{W},\mathbf{u}) = \sum_f \ln(s_f) + \sum_l u_l \left(\Phi_l(W_l,\mathbf{Q}_l) - \langle \mathbf{1}, \mathbf{X}^T\mathbf{e}_l\rangle\right) \quad (4.3.1)$$

and Γ is defined as

$$\Gamma \triangleq \left\{ (\mathbf{S},\mathbf{X},\mathbf{Q},\mathbf{W}) \;\middle|\; \begin{array}{ll} \mathbf{AX}=\mathbf{S} & \\ \mathbf{X}\geq\mathbf{0} & \\ \mathbf{Se}_f =_{\mathrm{src}(f),\mathrm{dst}(f)} \mathbf{0} & \forall f \\ \langle \mathbf{1}, \mathbf{Se}_f\rangle = 0 & \forall f \\ (\mathbf{Se}_f)_{\mathrm{src}(f)} = s_f & \forall f \\ \sum_{l\in\mathcal{O}(n)} \mathrm{Tr}\{\mathbf{Q}_l\} \leq P_{\max}^{(n)} & \forall n \\ \mathbf{Q}_l \succeq 0, W_l \geq 0 & \forall l \\ \sum_{l\in\mathcal{O}(n)} W_l \leq B_n & \forall n \end{array} \right\}$$

The Lagrangian dual problem of CRPBA can be written as:

$$\begin{array}{rll} \mathbf{D}^{\mathrm{CRPBA}}: & \mathsf{Minimize} & \Theta(\mathbf{u}) \\ & \mathsf{subject\ to} & \mathbf{u}\geq\mathbf{0}. \end{array}$$

It is easy to recognize that, for any given Lagrangian multiplier $\mathbf{u}$, the Lagrangian can be separated into two terms:

$$\Theta(\mathbf{u}) = \Theta_{\mathrm{net}}(\mathbf{u}) + \Theta_{\mathrm{link}}(\mathbf{u}),$$

where Θ_{net} and Θ_{link} are two subproblems respectively corresponding to the network layer and the link-physical layer:

$$\begin{array}{rll} \Theta_{\mathrm{net}}(\mathbf{u}) \triangleq & \mathsf{Maximize} & \sum_f \ln(s_f) \\ & & -\sum_l u_l \langle \mathbf{1}, \mathbf{X}^T\mathbf{e}_l\rangle \\ & \mathsf{subject\ to} & \mathbf{AX}=\mathbf{S} \\ & & \mathbf{X}\geq\mathbf{0} \\ & & \mathbf{Se}_f =_{\mathrm{src}(f),\mathrm{dst}(f)} \mathbf{0} \quad \forall f \\ & & \langle \mathbf{1}, \mathbf{Se}_f\rangle = 0 \quad \forall f \\ & & (\mathbf{Se}_f)_{\mathrm{src}(f)} = s_f \quad \forall f \\ & \mathsf{Variables}: & \mathbf{S},\ \mathbf{X} \end{array}$$

$$
\begin{array}{rll}
\Theta_{\text{link}}(\mathbf{u}) \triangleq & \text{Maximize} \sum_l u_l \Phi_l(W_l, \mathbf{Q}_l) & \\
\text{subject to} & \sum_{l \in \mathcal{O}(n)} \text{Tr}\{\mathbf{Q}_l\} \leq P_{\max}^{(n)} & \forall\, n \\
& \sum_{l \in \mathcal{O}(n)} W_l \leq B_n & \forall\, n \\
& \mathbf{Q}_l \succeq 0, W_l \geq 0 & \forall\, l \\
\text{Variables}: & \mathbf{Q},\, \mathbf{W} &
\end{array}
$$

Then, the CRPBA Lagrangian dual problem can be transformed into the following master dual problem:

$$
\begin{array}{rll}
\mathbf{MD}^{\text{CRPBA}}: & \text{Minimize} & \Theta_{\text{net}}(\mathbf{u}) + \Theta_{\text{link}}(\mathbf{u}) \\
& \text{subject to} & \mathbf{u} \geq \mathbf{0}
\end{array}
$$

Now, the task of solving the decomposed Lagrangian dual problem boils down to evaluating the subproblems $\Theta_{\text{net}}(\mathbf{u})$ and $\Theta_{\text{link}}(\mathbf{u})$, and solving the master problem $\mathbf{MD}^{\text{CRPBA}}$. Note that in the network layer subproblem $\Theta_{\text{net}}(\mathbf{u})$, the objective function is concave and all constraints are affine. Therefore, $\Theta_{\text{net}}(\mathbf{u})$ can be readily solved by many polynomial-time convex programming methods. However, solving $\Theta_{\text{link}}(\mathbf{u})$ is not trivial because the objective function and constraints involve many complex matrices variables. Even though $\Theta_{\text{link}}(\mathbf{u})$ is a convex problem in nature, standard convex optimization methods without exploiting the special structure of $\Theta_{\text{link}}(\mathbf{u})$ are not very efficient. In the following section, we propose a custom-designed method based on gradient projection to solve $\Theta_{\text{link}}(\mathbf{u})$.

4.4 The Link-Physical Layer Subproblem

In this chapter, we propose to use conjugate gradient projection-based method (GP) to solve the link-physical layer subproblem due to its simplicity. The framework of our proposed CGP method is shown in Algorithm 4.4.1.

Algorithm 4.4.1 Conjugate Gradient Projection Method

Initialization:

Choose the initial conditions $\mathbf{Q}^{(0)} = [\mathbf{Q}_1^{(0)}, \mathbf{Q}_2^{(0)}, \ldots, \mathbf{Q}_K^{(0)}]^T$. Let $k = 0$.

Main Loop:

1. Calculate the conjugate gradients $\mathbf{G}_i^{(k)}$, $i = 1, 2, \ldots, K$.
2. Choose an appropriate step size s_k. Let $\mathbf{Q}_i^{'(k)} = \mathbf{Q}_i^{(k)} + s_k \mathbf{G}_i^{(k)}$, for $i = 1, 2, \ldots, K$.
3. Let $\bar{\mathbf{Q}}^{(k)}$ be the projection of $\mathbf{Q}^{'(k)}$ onto $\Omega_+(n)$, where $\Omega_+(n) \triangleq \{\mathbf{Q}_i,\ i = 1, \ldots, K | \mathbf{Q}_i \succeq 0, \sum_{i=1}^K \mathrm{Tr}\{\mathbf{Q}_i\} \leq P\}$.
4. Choose appropriate step size α_k. Let $\mathbf{Q}_i^{(k+1)} = \mathbf{Q}_i^{(k)} + \alpha_k(\bar{\mathbf{Q}}_i^{(k)} - \mathbf{Q}_i^{(k)})$, $i = 1, 2, \ldots, K$.
5. $k = k + 1$. If the maximum absolute value of the elements in $\mathbf{Q}_i^{(k)} - \mathbf{Q}_i^{(k-1)} < \epsilon$, for $i = 1, 2, \ldots, L$, then stop;
   else go to step 1.

In order to make the CGP algorithm work, we will have to fill in two problem specific components, i.e., during the k^{th} duration, how to compute the gradients and how to project $\mathbf{Q}_l^{(k)'}$ and $W_l^{(k)'}$ onto $\Omega_+(n) \triangleq \{(W_l, \mathbf{Q}_l) : l \in \mathcal{O}(n), W_l \geq 0, \mathbf{Q}_l \succeq 0, \sum_{l \in \mathcal{O}(n)} W_l \leq B_n, \sum_{l \in \mathcal{O}(n)} \mathrm{Tr}\{\mathbf{Q}_l\} \leq P_{\max}^{(n)}\}$.

4.4.1 Computing the Gradients

For the gradient with respect to W_l, it is not difficult to see that $G_{W_l} \triangleq \nabla_{W_l} \Theta_{\text{link}} = u_l \log_2 \det(\mathbf{I} + \rho_l \mathbf{H}_l \mathbf{Q}_l \mathbf{H}_l^\dagger)$. The gradient $\mathbf{G}_{\mathbf{Q}_l} \triangleq \nabla_{\mathbf{Q}_l} \Theta_{\text{link}}(\mathbf{Q})$ depends on the partial derivative of $\Theta_{\text{link}}(\mathbf{Q})$ with respect to $\mathbf{Q}_l$. Before computing the partial derivative of $\Theta_{\text{link}}(\mathbf{Q})$, we need the following lemma [98].

Lemma 4.4.1. *For matrices* $\mathbf{B} \in \mathbb{C}^{p \times m}$, $\mathbf{X} \in \mathbb{C}^{m \times n}$, *and* $\mathbf{C} \in \mathbb{C}^{n \times p}$, *if* $(\mathbf{BXC})$ *is invertible, then we have* $\frac{\partial \det(\mathbf{BXC})}{\partial \mathbf{X}} = \det(\mathbf{BXC}) \left[\mathbf{C}(\mathbf{BXC})^{-1}\mathbf{B}\right]^T$.

With Lemma 5.4.3, we have the following corollary.

Corollary 4.4.2. *For matrices* $\mathbf{A} \in \mathbb{C}^{p \times p}$, $\mathbf{B} \in \mathbb{C}^{p \times m}$, $rank(\mathbf{B}) = p$, $\mathbf{X} \in \mathbb{C}^{m \times n}$, $\mathbf{C} \in \mathbb{C}^{n \times p}$, *and* $rank(\mathbf{C}) = p$, *if* $(\mathbf{A} + \mathbf{BXC})$ *is invertible, then we have* $\frac{\partial}{\partial \mathbf{X}} \ln \det(\mathbf{A} + \mathbf{BXC}) = \left[\mathbf{C}(\mathbf{A} + \mathbf{BXC})^{-1}\mathbf{B}\right]^T$.

Proof. By chain rule, we have

$$\begin{aligned}\frac{\partial}{\partial \mathbf{X}} \ln\det(\mathbf{BXC}) &= \frac{\partial \ln\det(\mathbf{BXC})}{\partial \det(\mathbf{BXC})} \cdot \frac{\partial \det(\mathbf{BXC})}{\partial \mathbf{X}} \\ &= \frac{1}{\det(\mathbf{BXC})} \cdot \det(\mathbf{BXC}) \left[\mathbf{C}(\mathbf{BXC})^{-1}\mathbf{B}\right]^T \\ &= \left[\mathbf{C}(\mathbf{BXC})^{-1}\mathbf{B}\right]^T. \end{aligned} \tag{4.4.1}$$

Since $rank(\mathbf{B}) = p$ and $rank(\mathbf{C}) = p$, we have that $\mathbf{B}$ and $\mathbf{C}$ have right and left inverses, respectively. Let $\mathbf{Y}$ be such that $\mathbf{BYC} = \mathbf{A} + \mathbf{BXC}$, i.e., $\mathbf{Y} = \mathbf{X} + \mathbf{B}_R\mathbf{AC}_L$, where $\mathbf{B}_R$ and $\mathbf{C}_L$ are the right and left inverses of $\mathbf{B}$ and $\mathbf{C}$, respectively. It then follows that $\partial\mathbf{Y}/\partial\mathbf{X} = \mathbf{I}$. Thus, we have

$$\begin{aligned}&\frac{\partial \ln\det(\mathbf{A} + \mathbf{BXC})}{\partial \mathbf{X}} = \frac{\partial \ln\det(\mathbf{BYC})}{\partial \mathbf{Y}} \cdot \frac{\partial \mathbf{Y}}{\partial \mathbf{X}} = \\ &\left[\mathbf{C}(\mathbf{BYC})^{-1}\mathbf{B}\right]^T \cdot \frac{\partial \mathbf{Y}}{\partial \mathbf{X}} = \left[\mathbf{C}(\mathbf{A} + \mathbf{BXC})^{-1}\mathbf{B}\right]^T \cdot \frac{\partial \mathbf{Y}}{\partial \mathbf{X}} \\ &= \left[\mathbf{C}(\mathbf{A} + \mathbf{BXC})^{-1}\mathbf{B}\right]^T.\end{aligned}$$

□

We can now compute the partial derivative of $\Theta_{\text{link}}(\mathbf{Q})$ with respect to $\mathbf{Q}_l$, which is given by

$$\frac{\partial \Theta_{\text{link}}(\mathbf{Q})}{\partial \mathbf{Q}_l} \;=\; W_l \frac{\partial}{\partial \mathbf{Q}_l}\left[\log_2\det\left(\mathbf{I} + \rho_j \mathbf{H}_j \mathbf{Q}_j \mathbf{H}_j^\dagger\right)\right].$$

Assuming that the channel gain matrices $\mathbf{H}$ are of full row rank (if not, we can always delete the linearly dependent rows). Applying Corollary 4.4.2 by letting $\mathbf{A} = \mathbf{I}$, $\mathbf{B} = \rho_l \mathbf{H}_l$, $\mathbf{X} = \mathbf{Q}_l$, and $\mathbf{C} = \mathbf{H}_l^\dagger$, we have

$$\frac{\partial \Theta_{\text{link}}(\mathbf{Q})}{\partial \mathbf{Q}_l} = \frac{W_l u_l \rho_l}{\ln 2}\left[\mathbf{H}_l^\dagger\left(\mathbf{I} + \rho_l \mathbf{H}_l \mathbf{Q}_l \mathbf{H}_l^\dagger\right)^{-1}\mathbf{H}_l\right]^T,$$

where we have used the fact that $\mathbf{R}_l$ does not depend on $\mathbf{Q}_l$.

Recall that, for a function $f(z)$, where $z = x + jy$ is a complex variable, its derivative is defined as $\frac{\partial f(z)}{\partial z} = \frac{1}{2}\left(\frac{\partial f(z)}{\partial x} - j\frac{\partial f(z)}{\partial y}\right)$, and the gradient is defined as $\nabla_z f(z) = \frac{\partial f(z)}{\partial x} + j\frac{\partial f(z)}{\partial y}$ [58, 159]. Hence, we have $\nabla_z f(z) = 2\left(\frac{\partial f(z)}{\partial z}\right)^*$. Therefore,

$$\mathbf{G}_{\mathbf{Q}_l} = \frac{2W_l u_l \rho_l}{\ln 2}\mathbf{H}_l^\dagger \left(\mathbf{I} + \rho_l \mathbf{H}_l \mathbf{Q}_l \mathbf{H}_l^\dagger\right)^{-1} \mathbf{H}_l. \tag{4.4.2}$$

Note that in (4.4.2), since $\left(\mathbf{I} + \rho_j \mathbf{H}_l \mathbf{Q}_l \mathbf{H}_l^\dagger\right)^{-1}$ is a Hermitian matrix, $\mathbf{G}_{\mathbf{Q}_l}$ is also Hermitian.

4.4.2 A Polynomial-Time Algorithm for Performing Projection

Since $\mathbf{G}_{\mathbf{Q}_l}$ is Hermitian, it then follows that $\mathbf{Q}_l'(k) = \mathbf{Q}_l(k) + s_k \mathbf{G}_{\mathbf{Q}_l}(k)$ is Hermitian. On the other hand, since $\Omega_+(n)$ contains a constraint on power sum for each node n having $|\mathcal{O}(n)|$ outgoing links, we need to project the $|\mathcal{O}(n)|$ W-scalars and $|\mathcal{O}(n)|$ $\mathbf{Q}$-covariance matrices onto $\Omega_+(n)$ simultaneously.

Toward this end, we construct a block diagonal matrix $\mathbf{D}_n$ as follows:

$$\mathbf{D}_n = \left[\begin{array}{c|c} \mathbf{W}_n & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{Q}_n \end{array}\right] \in \mathbb{C}^{|\mathcal{O}(n)|(n_t+1)\times|\mathcal{O}(n)|(n_t+1)},$$

where $\mathbf{W}_n$ is defined as $\mathbf{W}_n \triangleq \mathrm{Diag}\{W_l : l \in \mathcal{O}(n)\} \in \mathbb{C}^{|\mathcal{O}(n)|\times|\mathcal{O}(n)|}$, and $\mathbf{Q}_n$ is defined as $\mathbf{Q}_n \triangleq \mathrm{Diag}\{\mathbf{Q}_l : l \in \mathcal{O}(n)\} \in \mathbb{C}^{|\mathcal{O}(n)|n_t\times|\mathcal{O}(n)|n_t}$. We introduce two more matrices $\mathbf{E}_1^{(n)}$ and $\mathbf{E}_2^{(n)}$ as follows:

$$\mathbf{E}_1^{(n)} = \left[\begin{array}{c|c} \mathbf{I}_{|\mathcal{O}(n)|} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} \end{array}\right] \in \mathbb{C}^{|\mathcal{O}(n)|(n_t+1)\times|\mathcal{O}(n)|(n_t+1)},$$

$$\mathbf{E}_2^{(n)} = \left[\begin{array}{c|c} \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{I}_{|\mathcal{O}(n)|n_t} \end{array}\right] \in \mathbb{C}^{|\mathcal{O}(n)|(n_t+1)\times|\mathcal{O}(n)|(n_t+1)}.$$

It is easy to recognize that if $\mathbf{D}_n \in \Omega_+(n)$, we have $\mathrm{Tr}(\mathbf{E}_1^{(n)}\mathbf{D}_n) = \sum_{l\in\mathcal{O}(n)} W_l \leq B_n$, $\mathrm{Tr}(\mathbf{E}_2^{(n)}\mathbf{D}_n) = \sum_{l\in\mathcal{O}(n)} \mathrm{Tr}\left(\mathbf{Q}_l\right) \leq P_{\max}^{(n)}$, and $\mathbf{D}_n \succeq 0$. In this chapter, Frobenius norm, which is the counterpart of the Euclidean norm in the vector space, is used as the matrix distance metric. By the definition of Frobenius norm, the distance between two matrices $\mathbf{A}$ and $\mathbf{B}$ is $\|\mathbf{A}-\mathbf{B}\|_F = \left(\mathrm{Tr}\left[(\mathbf{A}-\mathbf{B})^\dagger(\mathbf{A}-\mathbf{B})\right]\right)^{\frac{1}{2}}$. Thus, given the block diagonal matrix $\mathbf{D}_n$, we wish to find a matrix $\tilde{\mathbf{D}}_n \in \Omega_+(n)$ such that $\tilde{\mathbf{D}}_n$ minimizes $\|\tilde{\mathbf{D}}_n - \mathbf{D}_n\|_F$, i.e.,

$$\begin{aligned} \textsf{Minimize} \quad & \|\tilde{\mathbf{D}}_n - \mathbf{D}_n\|_F \\ \textsf{subject to} \quad & \mathrm{Tr}(\mathbf{E}_1^{(n)}\tilde{\mathbf{D}}_n) \leq B_n \\ & \mathrm{Tr}(\mathbf{E}_2^{(n)}\tilde{\mathbf{D}}_n) \leq P_{\max}^{(n)} \\ & \tilde{\mathbf{D}}_n \succeq 0 \end{aligned} \tag{4.4.3}$$

For more convenient algebraic manipulations, we instead consider the following equivalent optimization problem:

$$\begin{aligned} \textsf{Minimize} \quad & \tfrac{1}{2}\left\|\tilde{\mathbf{D}}_n - \mathbf{D}_n\right\|_F^2 \\ \textsf{subject to} \quad & \mathrm{Tr}(\mathbf{E}_1^{(n)}\tilde{\mathbf{D}}_n) \leq B_n \\ & \mathrm{Tr}(\mathbf{E}_2^{(n)}\tilde{\mathbf{D}}_n) \leq P_{\max}^{(n)} \\ & \tilde{\mathbf{D}}_n \succeq 0. \end{aligned} \tag{4.4.4}$$

Note that the objective function of this minimization problem is convex in $\tilde{\mathbf{D}}_n$, the constraint $\tilde{\mathbf{D}}_n \succeq 0$ represents the convex cone of positive semidefinite matrices, and the constraints $\mathrm{Tr}(\mathbf{E}_1^{(n)}\tilde{\mathbf{D}}_n) \leq B_n$ and $\mathrm{Tr}(\mathbf{E}_2^{(n)}\tilde{\mathbf{D}}_n) \leq P_{\max}^{(n)}$ are linear constraints. Therefore, this problem is a convex optimization problem, and we can solve this minimization problem by solving its Lagrangian dual problem.

Associating Hermitian matrix $\mathbf{\Pi}$ to the constraint $\tilde{\mathbf{D}}_n \succeq 0$, ν to the constraint $\mathrm{Tr}(\mathbf{E}_1^{(n)}\tilde{\mathbf{D}}_n) \leq B_n$, and μ to the constraint $\mathrm{Tr}(\mathbf{E}_2^{(n)}\tilde{\mathbf{D}}_n) \leq P_{\max}^{(n)}$, we can write the

Lagrangian as

$$\begin{aligned} g(\mathbf{\Pi},\nu,\mu) &= \min_{\tilde{\mathbf{D}}_n}\left\{\frac{1}{2}\left\|\tilde{\mathbf{D}}_n-\mathbf{D}_n\right\|_F^2-\mathrm{Tr}(\mathbf{\Pi}^\dagger\tilde{\mathbf{D}}_n)\right. \\ &\left.+\nu\left(\mathrm{Tr}[\mathbf{E}_1^{(n)}\tilde{\mathbf{D}}_n]-B_n\right)+\mu\left(\mathrm{Tr}[\mathbf{E}_2^{(n)}\tilde{\mathbf{D}}_n]-P_{\max}^{(n)}\right)\right\}. \end{aligned} \tag{4.4.5}$$

Since $g(\mathbf{\Pi},\nu,\mu)$ is a convex quadratic function in $\tilde{\mathbf{D}}_n$ and $\tilde{\mathbf{D}}_n$ becomes unconstrained after moving the positive semidefinite constraint to the objective function, we can compute the minimizer of (4.4.5) by simply setting the derivative of (4.4.5) to zero, i.e.,

$$(\tilde{\mathbf{D}}_n-\mathbf{D}_n)-\mathbf{\Pi}^\dagger+\nu\mathbf{E}_1^{(n)}+\mu\mathbf{E}_2^{(n)}=0.$$

Noting that $\mathbf{\Pi}^\dagger=\mathbf{\Pi}$, we have

$$\tilde{\mathbf{D}}_n=\mathbf{D}_n+\mathbf{\Pi}-\nu\mathbf{E}_1^{(n)}-\mu\mathbf{E}_2^{(n)}. \tag{4.4.6}$$

Substituting $\tilde{\mathbf{D}}_n$ into (4.4.5), we have

$$\begin{aligned} g(\mathbf{\Pi},\nu,\mu) &= \frac{1}{2}\left\|\mathbf{\Pi}-\nu\mathbf{E}_1^{(n)}-\mu\mathbf{E}_2^{(n)}\right\|_F^2-\nu B_n-\mu P_{\max}^{(n)} \\ &+\mathrm{Tr}\left[\left(\nu\mathbf{E}_1^{(n)}+\mu\mathbf{E}_2^{(n)}-\mathbf{\Pi}\right)\left(\mathbf{D}_n+\mathbf{\Pi}-\nu\mathbf{E}_1^{(n)}-\mu\mathbf{E}_2^{(n)}\right)\right] \\ &=-\frac{1}{2}\left\|\mathbf{D}_n-\nu\mathbf{E}_1^{(n)}-\mu\mathbf{E}_2^{(n)}+\mathbf{\Pi}\right\|_F^2-\nu B_n-\mu P_{\max}^{(n)} \\ &+\frac{1}{2}\|\mathbf{D}_n\|_F^2. \end{aligned}$$

Therefore, the Lagrangian dual problem can be written as

$$\begin{aligned} \mathsf{Maximize}\quad & -\tfrac{1}{2}\left\|\mathbf{D}_n-\nu\mathbf{E}_1^{(n)}-\mu\mathbf{E}_2^{(n)}+\mathbf{\Pi}\right\|_F^2 \\ & -\nu B_n-\mu P_{\max}^{(n)}+\tfrac{1}{2}\|\mathbf{D}_n\|_F^2 \\ \mathsf{subject\ to}\quad & \mathbf{\Pi}\succeq 0,\nu\geq 0,\mu\geq 0 \end{aligned} \tag{4.4.7}$$

After solving (4.4.7) and by saddle-point optimality condition, we have the optimal solution to the primal problem as:

$$\tilde{\mathbf{D}}_n^*=\mathbf{D}_n-\nu^*\mathbf{E}_1^{(n)}-\mu^*\mathbf{E}_2^{(n)}+\mathbf{\Pi}^*, \tag{4.4.8}$$

where ν^*, μ^* and $\mathbf{\Pi}^*$ are the optimal solutions to Lagrangian dual problem in (4.4.7).

From matrix theory, we know that the eigenvalues of a Hermitian matrix $\mathbf{A} \in \mathbb{C}^{n\times n}$ are real. Suppose that we sort these eigenvalues of $\mathbf{A}$, denoted by λ_i, $i = 1, \ldots, p$, in non-increasing order, i.e., $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$, and perform eigenvalue decomposition on $\mathbf{A}$ yielding $\mathbf{A} = \mathbf{U_A}\text{Diag}\{\lambda_i : i = 1, \ldots, p\}\mathbf{U}_\mathbf{A}^\dagger$. In this decomposition, $\mathbf{U_A}$ is the unitary matrix formed by the eigenvectors corresponding to the non-increasing eigenvalues. Then, we have the positive semidefinite and negative semidefinite projections of $\mathbf{A}$ as follows:

$$\mathbf{A}_+ = \mathbf{U_A}\text{Diag}\{\max\{\lambda_1, 0\}, \ldots, \max\{\lambda_p, 0\}\}\mathbf{U}_\mathbf{A}^\dagger, \tag{4.4.9}$$

$$\mathbf{A}_- = \mathbf{U_A}\text{Diag}\{\min\{\lambda_1, 0\}, \ldots, \min\{\lambda_p, 0\}\}\mathbf{U}_\mathbf{A}^\dagger. \tag{4.4.10}$$

The proof of the results in (4.4.9) and (4.4.10) is a straightforward application of Moreau decomposition [61] by noting that $\mathbf{A}_+ \succeq 0$, $\mathbf{A}_- \preceq 0$, $\langle \mathbf{A}_+, \mathbf{A}_- \rangle = 0$, $\mathbf{A}_+ + \mathbf{A}_- = \mathbf{A}$, and the positive semidefinite cone and negative semidefinite cone are polar cones to each other.

Now we consider the term $\mathbf{D}_n - \nu\mathbf{E}_1^{(n)} - \mu\mathbf{E}_2^{(n)} + \mathbf{\Pi}$, which is the only term involving $\mathbf{\Pi}$ in the dual objective function. We can rewrite it as $\mathbf{D}_n - \nu\mathbf{E}_1^{(n)} - \mu\mathbf{E}_2^{(n)} - (-\mathbf{\Pi})$, where we note that $-\mathbf{\Pi} \preceq 0$. Finding a negative semidefinite matrix $-\mathbf{\Pi}$ such that $\|\mathbf{D}_n - \nu\mathbf{E}_1^{(n)} - \mu\mathbf{E}_2^{(n)} - (-\mathbf{\Pi})\|_F$ is minimized is equivalent to finding the projection of $\mathbf{D}_n - \nu\mathbf{E}_1^{(n)} - \mu\mathbf{E}_2^{(n)}$ onto the negative semidefinite cone. From our previous discussion, we have

$$-\mathbf{\Pi} = \left(\mathbf{D}_n - \nu\mathbf{E}_1^{(n)} - \mu\mathbf{E}_2^{(n)}\right)_- . \tag{4.4.11}$$

Substituting (4.4.11) back to the Lagrangian dual objective function, we have

$$\min_{\mathbf{\Pi}} \left\|\mathbf{D}_n - \nu\mathbf{E}_1^{(n)} - \mu\mathbf{E}_2^{(n)} + \mathbf{\Pi}\right\|_F = \left(\mathbf{D}_n - \nu\mathbf{E}_1^{(n)} - \mu\mathbf{E}_2^{(n)}\right)_+ .$$

Thus, the matrix variable $\mathbf{\Pi}$ in the Lagrangian dual problem can be analytically solved and the Lagrangian dual problem (4.4.7) can be simplified to

$$\begin{array}{ll} \mathsf{Maximize} & \psi(\nu,\mu) \triangleq -\frac{1}{2}\|(\mathbf{D}_n - \nu\mathbf{E}_1^{(n)} - \mu\mathbf{E}_2^{(n)})_+\|_F^2 \\ & -\nu B_n - \mu P_{\max}^{(n)} + \frac{1}{2}\|\mathbf{D}_n\|_F^2 \\ \mathsf{subject\ to} & \nu \geq 0, \mu \geq 0 \end{array}$$

Suppose that after performing eigenvalue decomposition on $\mathbf{D}_n$, we have $\mathbf{D}_n = \mathbf{U}_n\mathbf{\Lambda}_n\mathbf{U}_n^\dagger$, where $\mathbf{\Lambda}_n$ is the diagonal matrix formed by the eigenvalues of $\mathbf{D}_n$, $\mathbf{U}_n$ is the unitary matrix formed by the corresponding eigenvectors. From the fact that $\mathbf{E}_1^{(n)} = \mathbf{U}_n\mathbf{E}_1^{(n)}\mathbf{U}_n^\dagger$ and $\mathbf{E}_2^{(n)} = \mathbf{U}_n\mathbf{E}_2^{(n)}\mathbf{U}_n^\dagger$, we have $(\mathbf{D}_n - \nu\mathbf{E}_1^{(n)} - \mu\mathbf{E}_2^{(n)})_+ = \mathbf{U}_n(\mathbf{\Lambda}_n - \nu\mathbf{E}_1^{(n)} - \mu\mathbf{E}_2^{(n)})_+\mathbf{U}_n^\dagger$. Since $\mathbf{U}_n$ is unitary, we have $\|(\mathbf{D}_n - \nu\mathbf{E}_1^{(n)} - \mu\mathbf{E}_2^{(n)})_+\|_F^2 = \|(\mathbf{\Lambda}_n - \nu\mathbf{E}_1^{(n)} - \mu\mathbf{E}_2^{(n)})_+\|_F^2$. In particular, we denote the eigenvalues in $\mathbf{\Lambda}_n$ corresponding to $\mathbf{W}_n$ and $\mathbf{Q}_n$ by $\lambda_i^{(\mathbf{W}_n)}$ and $\lambda_j^{(\mathbf{Q}_n)}$, respectively, and sort the eigenvalues in these two groups in non-increasing order as follows:

$$\mathbf{\Lambda}_n = \mathrm{Diag}\{\lambda_1^{(\mathbf{W}_n)}, \ldots, \lambda_{|\mathcal{O}(n)|}^{(\mathbf{W}_n)}, \lambda_1^{(\mathbf{Q}_n)}, \ldots, \lambda_{|\mathcal{O}(n)|\times n_t}^{(\mathbf{Q}_n)}\},$$

where $\lambda_1^{(\mathbf{W}_n)} \geq \ldots \geq \lambda_{|\mathcal{O}(n)|}^{(\mathbf{W}_n)}$ and $\lambda_1^{(\mathbf{Q}_n)} \geq \ldots \geq \lambda_{|\mathcal{O}(n)|\times n_t}^{(\mathbf{Q}_n)}$. It then follows that

$$\begin{aligned} & \|(\mathbf{\Lambda}_n - \nu\mathbf{E}_1^{(n)} - \mu\mathbf{E}_2^{(n)})_+\|_F^2 \\ = & \sum_{i=1}^{|\mathcal{O}(n)|}\left(\max\left\{0, \lambda_i^{(\mathbf{W}_n)} - \nu\right\}\right)^2 + \sum_{j=1}^{|\mathcal{O}(n)|n_t}\left(\max\left\{0, \lambda_j^{(\mathbf{Q}_n)} - \mu\right\}\right)^2. \quad (4.4.12) \end{aligned}$$

From (4.4.12), we have

$$\begin{aligned} \psi(\nu,\mu) = & -\frac{1}{2}\sum_{i=1}^{|\mathcal{O}(n)|}\left(\max\left\{0, \lambda_i^{(\mathbf{W}_n)} - \nu\right\}\right)^2 - \nu B_n - \frac{1}{2}\sum_{j=1}^{|\mathcal{O}(n)|n_t}\left(\max\left\{0, \lambda_j^{(\mathbf{Q}_n)} - \mu\right\}\right)^2 \\ & -\mu P_{\max}^{(n)} + \frac{1}{2}\|\mathbf{D}_n\|_F^2 = \psi(\nu) + \psi(\mu) + \frac{1}{2}\|\mathbf{D}_n\|_F^2, \quad (4.4.13) \end{aligned}$$

where $\psi(\nu) \triangleq -\frac{1}{2}\sum_{i=1}^{|\mathcal{O}(n)|}(\max\{0, \lambda_i^{(\mathbf{W}_n)} - \nu\})^2 - \nu B_n$ and $\psi(\mu) \triangleq -\frac{1}{2}\sum_{j=1}^{|\mathcal{O}(n)|n_t}(\max\{0, \lambda_j^{(\mathbf{Q}_n)} - \mu\})^2 - \mu P_{\max}^{(n)}$, i.e., we separate $\psi(\nu,\mu)$ into two parts. It can be readily verified that

$\psi(\nu,\mu)$ is continuous and piece-wise concave in ν and μ. Generally, piece-wise concave maximization problems can be solved by subgradient method. In this problem, it is easy to derive the subgradients with respect to ν and μ as follows:

$$\frac{\partial\psi}{\partial\nu} = \sum_{i=1}^{|\mathcal{O}(n)|} \max\left\{0, \lambda_i^{(\mathbf{W}_n)} - \nu\right\} - B_n,$$
$$\frac{\partial\psi}{\partial\mu} = \sum_{j=1}^{|\mathcal{O}(n)|\times n_t} \max\left\{0, \lambda_j^{(\mathbf{Q}_n)} - \mu\right\} - P_{\max}^{(n)}.$$

However, due to the heuristic nature of its step size selection strategy, subgradient algorithm usually does not perform well. In fact, since $\psi(\nu,\mu)$ is piece-wise quadratic and separable, we can solve $\psi(\nu,\mu)$ by exploiting this special structure.

For example, we can start searching the optimal value of ν as follows. Let the pieces of $\psi(\nu)$ indexed by $\hat{I}$, $\hat{I} = 0, 1, \ldots, |\mathcal{O}(n)|$. Initially we set $\hat{I} = 0$ and increase $\hat{I}$ subsequently. Also, we introduce $\lambda_0^{(\mathbf{W}_n)} = \infty$ and $\lambda_{|\mathcal{O}(n)|+1}^{(\mathbf{W}_n)} = -\infty$. If $\hat{I} = 0$, we let endpoint objective value $\psi_{\hat{I}}\left(\lambda_0^{(\mathbf{W}_n)}\right) = 0$, $\phi^* = \psi_{\hat{I}}\left(\lambda_0^{(\mathbf{W}_n)}\right)$, and let $\nu^* = \lambda_0^{(\mathbf{W}_n)}$. If $\hat{I} > |\mathcal{O}(n)|$, the search stops. For a particular index $\hat{I}$, suppose that $\nu \in \left[\lambda_{\hat{I}+1}^{(\mathbf{W}_n)}, \lambda_{\hat{I}}^{(\mathbf{W}_n)}\right] \cap \mathbb{R}_+$, where $\mathbb{R}_+$ denotes the set of non-negative real numbers. Solve for $\nu_{\hat{I}}^*$ by setting

$$\frac{\partial}{\partial\nu}\psi_{\hat{I}}(\nu) \triangleq \frac{\partial}{\partial\nu}\left(-\frac{1}{2}\sum_{i=1}^{\hat{I}}\left(\lambda_i^{(\mathbf{W}_n)} - \nu\right)^2 - \nu B_n\right) = 0,$$

for which we have

$$\nu_{\hat{I}}^* = \frac{\sum_{i=1}^{\hat{I}} \lambda_i^{(\mathbf{W}_n)} - B_n}{\hat{I}}. \tag{4.4.14}$$

Now we consider the following two cases:

1. If $\nu_{\hat{I}}^* \in [\lambda_{\hat{I}+1}^{(\mathbf{W}_n)}, \lambda_{\hat{I}}^{(\mathbf{W}_n)}] \cap \mathbb{R}_+$, then we have already found the optimal solution for

ν because $\psi(\nu, \mu)$ is continuous concave quadratic in ν, and the point with zero-value first derivative, if exists, must be the unique global maximum solution. Thus, we can let $\nu^* = \nu^*_{\hat{I}}$ and the search is done.

2. If $\nu^*_{\hat{I}} \notin [\lambda^{(\mathbf{W}_n)}_{\hat{I}+1}, \lambda^{(\mathbf{W}_n)}_{\hat{I}}] \cap \mathbb{R}_+$, we must have that the local maximum in the interval $[\lambda^{(\mathbf{W}_n)}_{\hat{I}+1}, \lambda^{(\mathbf{W}_n)}_{\hat{I}}] \cap \mathbb{R}_+$ is achieved at one of the two end points. We note that the objective value $\psi_{\hat{I}}\left(\lambda^{(\mathbf{W}_n)}_{\hat{I}}\right)$ has been computed in the previous iteration. This is because from the continuity of the objective function, we have $\psi_{\hat{I}}\left(\lambda^{(\mathbf{W}_n)}_{\hat{I}}\right) = \psi_{\hat{I}-1}\left(\lambda^{(\mathbf{W}_n)}_{\hat{I}}\right)$. Thus, we only need to compute the objective value $\psi_{\hat{I}}\left(\lambda^{(\mathbf{W}_n)}_{\hat{I}+1}\right)$ of another endpoint. If $\psi_{\hat{I}}\left(\lambda^{(\mathbf{W}_n)}_{\hat{I}+1}\right) < \psi_{\hat{I}}\left(\lambda^{(\mathbf{W}_n)}_{\hat{I}}\right) = \phi^*$, then we know ν^* is the optimal solution; else let $\nu^* = \lambda^{(\mathbf{W}_n)}_{\hat{I}+1}$, $\phi^* = \psi_{\hat{I}}\left(\lambda^{(\mathbf{W}_n)}_{\hat{I}+1}\right)$, let $\hat{I} \leftarrow \hat{I} + 1$ and continue.

Since there are $|\mathcal{O}(n)| + 1$ intervals, the search process takes at most $|\mathcal{O}(n)| + 1$ steps to find the optimal solution ν^*. Likewise, the search process for μ can be done in a similar fashion.

After finding ν^* and μ^*, we have

$$\tilde{\mathbf{D}}^*_n = \left(\mathbf{D}_n - \nu^*\mathbf{E}^{(n)}_1 - \mu^*\mathbf{E}^{(n)}_2\right)_+ = \mathbf{U}_n\left(\mathbf{\Lambda}_n - \nu^*\mathbf{E}^{(n)}_1 - \mu^*\mathbf{E}^{(n)}_2\right)_+ \mathbf{U}^\dagger_n, \quad (4.4.15)$$

That is, the projection $\tilde{\mathbf{D}}_n$ can be computed by adjusting the eigenvalues of $\mathbf{D}_n$ using ν^* and μ^* and keeping the eigenvectors unchanged.

The projection of $\mathbf{D}_n$ onto $\Omega_+(n)$ is summarized in Algorithm 4.4.2 and Algorithm 4.4.3.

Algorithm 4.4.2 Projection onto $\Omega_+(n)$

1. Construct a block diagonal matrix $\mathbf{D}_n$. Perform eigenvalue decomposition $\mathbf{D}_n = \mathbf{U}_n\mathbf{\Lambda}_n\mathbf{U}_n^\dagger$; separate the eigenvalues in two groups corresponding to $\mathbf{W}_n$ and $\mathbf{Q}_n$; sort them in non-increasing order within each group, respectively.
2. For each group, use Algorithm 4.4.3 to find the optimal dual variables ν^* and μ^*.
3. Compute $\tilde{\mathbf{D}}_n = \mathbf{U}_n(\mathbf{\Lambda}_n - \nu^*\mathbf{E}_1^{(n)} - \mu^*\mathbf{E}_2^{(n)})_+\mathbf{U}_n^\dagger$.

Algorithm 4.4.3 Search the Optimal Dual Variable ν^* and μ^*

Initiation:

Introduce $\lambda_0 = \infty$ and $\lambda_K = -\infty$. Let $\hat{I} = 0$. Let endpoint objective $\psi_{\hat{I}}(\lambda_0) = 0$, $\phi^* = \psi_{\hat{I}}(\lambda_0)$, and $\mu^* = \lambda_0$.

Main Loop:

1. If $\hat{I} > K$, return μ^*; else let $\mu^*_{\hat{I}} = (\sum_{j=1}^{\hat{I}} \lambda_j - P)/\hat{I}$.
2. If $\mu^*_{\hat{I}} \in [\lambda_{\hat{I}+1}, \lambda_{\hat{I}}] \cap \mathbb{R}_+$, then let $\mu^* = \mu^*_{\hat{I}}$ and return μ^*.
3. Compute $\psi_{\hat{I}}(\lambda_{\hat{I}+1})$. If $\psi_{\hat{I}}(\lambda_{\hat{I}+1}) < \phi^*$, then return μ^*; else let $\mu^* = \lambda_{\hat{I}+1}$, $\phi^* = \psi_{\hat{I}}(\lambda_{\hat{I}+1})$, $\hat{I} \leftarrow \hat{I} + 1$ and go to Step 1.

4.5 Solving the Lagrangian Dual Problem

4.5.1 A Cutting-Plane Method Based on Outer-Linearization

Compared to the subgradient-based approaches for solving Lagrangian dual problems, which we will discuss in the next subsection, the attractive feature of the cutting-plane method is its robustness, efficiency, and simplicity in recovering optimal primal feasible solutions.

We briefly review the basic idea of cutting-plane method as follows. Let $z = \Theta(\mathbf{u})$. The inequality $z \geq \sum_f \ln(s_f) + \sum_l u_l \left(\Phi_l(W_l, \mathbf{Q}_l) - \langle \mathbf{1}, \mathbf{X}^T\mathbf{e}_l\rangle\right)$ must hold for all $(\mathbf{S}, \mathbf{X}, \mathbf{Q}, \mathbf{W}) \in \Gamma$. Thus, the dual problem is equivalent to

$$\begin{aligned} \text{Minimize} \quad & z \\ \text{subject to} \quad & z \geq \textstyle\sum_f \ln(s_f) + \\ & \textstyle\sum_l u_l \left(\Phi_l(W_l, \mathbf{Q}_l) - \langle \mathbf{1}, \mathbf{X}^T\mathbf{e}_l\rangle\right) \\ & \mathbf{u} \geq 0, \end{aligned} \tag{4.5.1}$$

where $(\mathbf{S}, \mathbf{X}, \mathbf{Q}, \mathbf{W}) \in \Gamma$. However, (4.5.1) is a linear program with infinite constraints

not known explicitly. Instead, we can consider the following *approximating problem*:

$$\begin{array}{rl} \textsf{Minimize} & z \\ \textsf{subject to} & z \geq \sum_f \ln\left(s_f^{(j)}\right) + \\ & \sum_l u_l \left(\Phi_l(W_l^{(j)}, \mathbf{Q}_l^{(j)}) - \langle \mathbf{1}, \mathbf{X}^{(j)T}\mathbf{e}_l\rangle\right) \\ & \mathbf{u} \geq 0, \end{array} \tag{4.5.2}$$

where the points $(\mathbf{S}^{(j)}, \mathbf{X}^{(j)}, \mathbf{Q}^{(j)}, \mathbf{W}^{(j)}) \in \Gamma$, for $j = 1, \ldots, k-1$. The problem in (4.5.2) is a linear program with a finite number of constraints and can be solved efficiently. Let $(z^{(k)}, \mathbf{u}^{(k)})$ be an optimal solution to the approximating problem, which we refer to as the *master program*. If the solution is feasible to (4.5.1), then it is an optimal solution to the Lagrangian dual problem. To check feasibility, we consider the following *subproblem*:

$$\begin{array}{rl} \textsf{Maximize} & \sum_f \ln(s_f) + \\ & \sum_l u_l^{(k)} \left(\Phi_l(W_l, \mathbf{Q}_l) - \langle \mathbf{1}, \mathbf{X}^T\mathbf{e}_l\rangle\right) \\ \textsf{subject to} & (\mathbf{S}, \mathbf{X}, \mathbf{Q}, \mathbf{W}) \in \Gamma \end{array} \tag{4.5.3}$$

Suppose that $(\mathbf{S}^{(k)}, \mathbf{X}^{(k)}, \mathbf{Q}^{(k)}, \mathbf{W}^{(k)})$ is an optimal solution to the subproblem (4.5.3) and $\Theta^*(\mathbf{u}^{(k)})$ is the corresponding optimal objective value. If $z_k \geq \Theta^*(\mathbf{u}^{(k)})$, then $\mathbf{u}^{(k)}$ is an optimal solution to the Lagrangian dual problem. Otherwise, for $\mathbf{u} = \mathbf{u}^{(k)}$, the inequality constraint in (4.5.1) is not satisfied for $(\mathbf{S}^{(k)}, \mathbf{X}^{(k)}, \mathbf{Q}^{(k)}, \mathbf{W}^{(k)})$. Thus, we can add the constraint

$$\begin{aligned} z \geq & \sum_f \ln\left(s_f^{(k)}\right) \\ & + \sum_l u_l \left(\Phi_l(W_l^{(k)}, \mathbf{Q}_l^{(k)}) - \langle \mathbf{1}, \mathbf{X}^{(k)T}\mathbf{e}_l\rangle\right) \end{aligned} \tag{4.5.4}$$

to (4.5.2), and solve the master linear program again. Obviously, $(z^{(k)}, \mathbf{u}^{(k)})$ violates (4.5.4) and will be cut off by (4.5.4). The cutting plane algorithm is summarized in Algorithm 4.5.1.

Algorithm 4.5.1 Cutting Plane Algorithm for Solving $\mathbf{D}^{\text{CRPBA}}$

Initialization:
Find a point $(\mathbf{S}^{(0)}, \mathbf{X}^{(0)}, \mathbf{Q}^{(0)}, \mathbf{W}^{(0)}) \in \Gamma$. Let $k = 1$.
Main Loop:
1. Solve the master program in (4.5.2). Let $(z^{(k)}, \mathbf{u}^{(k)})$ be an optimal solution.
2. Solve the subproblem in (4.5.3). Let $(\mathbf{S}^{(k)}, \mathbf{X}^{(k)}, \mathbf{Q}^{(k)}, \mathbf{W}^{(k)})$ be an optimal point, and let $\Theta^*(\mathbf{u}^{(k)})$ be the corresponding optimal objective value.
3. If $z^{(k)} \geq \Theta(\mathbf{u}^{(k)})$, then stop with $\mathbf{u}^{(k)}$ as the optimal dual solution. Otherwise, add the constraint (4.5.4) to the master program, replace k by $k+1$, and go to Step 1.

4.5.2 Subgradient-Based Mechanism

Since the dual objective function is piece-wise differentiable [10], subgradient method can be used to solve the master dual problem. Subgradient algorithm for minimization problems is a generalization of steepest descent algorithm in which the negative gradient direction is replaced by a suitable negative subgradient direction. For $\Theta(\mathbf{u})$, starting with an initial $\mathbf{u}^{(0)}$ and after evaluating subproblems $\Theta_{\text{net}}(\mathbf{u})$ and $\Theta_{\text{link}}(\mathbf{u})$ for $\mathbf{u}^{(k)}$ in the k^{th} iteration, we update the dual variables by

$$\mathbf{u}^{(k+1)} = \left[\mathbf{u}^k - s_k \mathbf{d}^{(k)}\right]_+ . \tag{4.5.5}$$

In (4.5.5), the operator $[\cdot]_+$ projects a vector onto the nonnegative orthant, and s_k denotes a positive scalar step size. $\mathbf{d}^{(k)}$ is a subgradient of the Lagrangian at point $\mathbf{u}^{(k)}$. Although it has been shown in [10] that the subgradient algorithm converges if the step size s_k satisfies $s_k \to 0$ as $k \to \infty$, $\sum_{k=0}^{\infty} s_k = \infty$, and $\sum_{k=0}^{\infty} s_k^2 < \infty$, one has to carefully select the step size to avoid stalling and accelerate the convergence. It is shown in [10] that the best choice of the step size s_k is

$$s_k = \frac{\beta_k[\Theta(\mathbf{u}^{(k)}) - \hat{\Theta}]}{\|\mathbf{d}_k\|}, \tag{4.5.6}$$

where $\beta_k > 0$ and $\hat{\Theta}$ is an estimate of the optimal value of Θ. However, this optimal step size selection strategy requires global information and the estimation of Θ, which

is difficult in implementation. Another possible step size selection strategy is the divergent harmonic series

$$\beta \sum_{k=1}^{\infty} \frac{1}{k} = \infty. \tag{4.5.7}$$

For the dual master problem, the subgradient for the Lagrangian dual problem is

$$d_l^{(k)} = \Phi_l(\mathbf{Q}^*(\mathbf{u})) - \langle \mathbf{1}, \mathbf{X}^*(\mathbf{u})^T \mathbf{e}_l \rangle, \quad l = 1, 2, \ldots, L. \tag{4.5.8}$$

It is worth pointing out that the dual variables $\mathbf{u}^{(k)}$ can be economically interpreted as "prices" of the links during the k^{th} iteration. The subgradients $\mathbf{d}^{(k)}$ indicates the excess capacities of the links during the k^{th} iteration. The dual updating scheme of the subgradient algorithm can be viewed as a pricing scheme. When a link, say link k, is under-utilized, then $d_l^{(k)} > 0$. From (4.5.5), we can see that the price of link k will be reduced. On the other hand, when link k is over-utilized, then $d_l^{(k)} < 0$. Again, from (4.5.5), it can be seen that the price of that link k will be increased. The subgradient algorithm is summarized in Algorithm 4.5.2.

Algorithm 4.5.2 A Subgradient Algorithm for Solving MWSR

Initialization:

Choose the initial starting points $\mathbf{u}^{(0)}$. Let $k = 0$.

Main Loop:

1. Compute $(\mathbf{S}^{(k)}, \mathbf{X}^{(k)}, \mathbf{Q}^{(k)}, \mathbf{W}^{(k)})$ by solving the network layer and link layer subproblems.
2. Choose an appropriate step size s_k. Compute the subgradient $\mathbf{d}^{(k)}$ using (4.5.8) with $(\mathbf{S}^{(k)}, \mathbf{X}^{(k)}, \mathbf{Q}^{(k)}, \mathbf{W}^{(k)})$.
3. Update dual variables $\mathbf{u}^{(k)}$ using (4.5.5) with $\mathbf{d}^{(k)}$.
4. If $\|\mathbf{u}^{(k+1)} - \mathbf{u}^{(k)}\| < \epsilon$, then return $(\mathbf{S}^{(k)}, \mathbf{X}^{(k)}, \mathbf{Q}^{(k)}, \mathbf{W}^{(k)})$ as the final optimal solution and stop. Otherwise, let $k \leftarrow k + 1$ and go to Step 1.

4.5.3 Recovering Primal Optimal Solution

Thus far, we have investigated the procedures for solving the Lagrangian dual problem. During the course of solving the dual problem, the following problem, which is

used to evaluate $\Theta(\mathbf{u})$ at $\mathbf{u}$, is needed to solve for many times:

$$\begin{array}{ll} \text{Maximize} & \sum_f \ln(s_f) + \\ & \sum_l u_l \left(\Phi_l(W_l, \mathbf{Q}_l) - \langle \mathbf{1}, \mathbf{X}^T \mathbf{e}_l \rangle\right) \\ \text{subject to} & (\mathbf{S}, \mathbf{X}, \mathbf{Q}, \mathbf{W}) \in \Gamma. \end{array} \tag{4.5.9}$$

Suppose that $(\mathbf{S}^*, \mathbf{X}^*, \mathbf{Q}^*, \mathbf{W}^*)$ is an optimal solution to (4.5.9). If $(\mathbf{S}^*, \mathbf{X}^*, \mathbf{Q}^*, \mathbf{W}^*)$ is also feasible to the primal problem and $(\mathbf{u}^*)^T \left(\Phi_l(W_l, \mathbf{Q}_l) - \langle \mathbf{1}, \mathbf{X}^T \mathbf{e}_l \rangle\right) = 0$, then it is clear that $(\mathbf{S}^*, \mathbf{X}^*, \mathbf{Q}^*, \mathbf{W}^*, \mathbf{u}^*)$ is a saddle point and $(\mathbf{S}^*, \mathbf{X}^*, \mathbf{Q}^*, \mathbf{W}^*)$ solves the primal problem. However, $(\mathbf{S}^*, \mathbf{X}^*, \mathbf{Q}^*, \mathbf{W}^*)$ may not be feasible to the primal problem in general since it only solves (4.5.9), a problem related but different to the primal problem. Therefore, extra effort has to be taken to recover a primal optimal solution when the saddle point conditions does not hold.

To this end, suppose that $(\mathbf{S}^{(j)}, \mathbf{X}^{(j)}, \mathbf{Q}^{(j)}, \mathbf{W}^{(j)})$, for $j = 1, 2, \ldots, k$ are the optimal solution of (4.5.9) for $\mathbf{u} = \mathbf{u}^{(j)}$. Now, let us consider the following linear programming problem:

$$\begin{array}{ll} \text{Maximize} & \sum_{j=0}^{k} \tau_j \sum_{f=1}^{F} \log(s_f^{(j)}) \\ \text{subject to} & \sum_{j=0}^{k} \tau_j \left(\langle \mathbf{1}, \mathbf{X}^{(j)T} \mathbf{e}_l \rangle - \Phi_l(W_l^{(j)}, \mathbf{Q}_l^{(j)})\right) \leq 0, \forall l \\ & \sum_{j=0}^{k} \tau_j (\mathbf{S}^{(j)}, \mathbf{X}^{(j)}, \mathbf{Q}^{(j)}, \mathbf{W}^{(j)}) \in \Gamma \\ & \sum_{j=0}^{k} \tau_j = 1 \\ & \tau_j \geq 0, \forall j. \end{array} \tag{4.5.10}$$

We have the following theorem for recovering a primal feasible near-optimal solution.

Theorem 4.5.1. *Let τ_j^*, for $j = 1, \ldots, k$ be an optimal solution to (4.5.10). Then $(\bar{\mathbf{S}}, \bar{\mathbf{X}}, \bar{\mathbf{Q}}, \bar{\mathbf{W}}) = \sum_{j=1}^{k} \tau_j (\mathbf{S}^{(j)}, \mathbf{X}^{(j)}, \mathbf{Q}^{(j)}, \mathbf{W}^{(j)})$ is a feasible solution to the primal problem. Furthermore, let $\bar{V}$ be the objective value corresponding to $(\bar{\mathbf{S}}, \bar{\mathbf{X}}, \bar{\mathbf{Q}}, \bar{\mathbf{W}})$, $V_k = \sum_{j=0}^{k} \tau_j^* \sum_{f=1}^{F} \log(s_f^{(j)})$, and V^* be the true primal optimal objective value. If $\Theta(\mathbf{u}) - \bar{V} \leq \epsilon$ for some $\mathbf{u} \geq \mathbf{0}$, then $\bar{V} \geq V^* - \epsilon$.*

Proof. Since the function $\langle \mathbf{1}, \mathbf{X}^T \mathbf{e}_l \rangle - \Phi_l(W_l, \mathbf{Q}_l)$ is convex, we have that $\langle \mathbf{1}, \bar{\mathbf{X}}^T \mathbf{e}_l \rangle - \Phi_l(\bar{W}_l, \bar{\mathbf{Q}}_l) \leq \sum_{j=0}^{k} \tau_j^* (\langle \mathbf{1}, \mathbf{X}^{(j)T} \mathbf{e}_l \rangle - \Phi_l(W_l^{(j)}, \mathbf{Q}_l^{(j)})) \leq 0$. Thus, $(\bar{\mathbf{S}}, \bar{\mathbf{X}}, \bar{\mathbf{Q}}, \bar{\mathbf{W}})$ is feasible to the primal problem. Noting the concavity of $\sum \log(\cdot)$, we have

$$\bar{V} \geq \sum_{j=0}^{k} \tau_j^* \sum_{f=1}^{F} \log(s_{f,j}) = V_k \geq \Theta(\mathbf{u}) - \epsilon \geq V^* - \epsilon.$$

This completes the proof. □

It is worth pointing out that we *need not* to solve (4.5.10) separately when we use the cutting-plane method to solve the Lagrangian dual problem. This is because (4.5.10) is precisely the linear programming dual of (4.5.2). As a result, the values of τ_j are immediately available after we solve (4.5.2).

4.6 Distributed Implementation

Algorithm 4.6.1 Distributed Implementation

Initialization:

Initialize the iteration index $k = 0$, and choose initial values for $u_l^{(0)}$, for all l.

Main Loop:

1. Each node independently solves the decomposed link layer subproblem in (4.6.1). After that, each node independently updates dual variables u_l for all its outgoing links using (4.5.5) (where the step size and subgradient computation follow (4.5.7) and (4.5.8)) and broadcast them to its next hop neighbor.
2. Upon receiving some dual information from other neighbors, each node relays it to its next hop neighbor excluding the node where this information comes from.
3. Upon receiving all links' dual information $\mathbf{u}$, each source node solves the decomposed network layer subproblem in (4.6.2). After that, each source node f performs source routing and store these routing information $x_l^{(f)}$, $\forall l$ in the header.
4. Upon receiving the source routing information, each intermediate node routes the packets according to the routing information $x_l^{(f)}$ in the header.
5. Based on current values of dual variables $u_l^{(k)}$ and the iteration number k, compute $u_l^{(k+1)}$. If $u_l^{(k+1)} - u_l^{(k)} < \epsilon$, or k has reached a predefined number of iterations, the algorithm stops. Otherwise, let $k \leftarrow k + 1$ and go to Step 1.

As mentioned earlier, the Lagrangian dual problem is solvable by subgradient algorithm. Although the original motivation in applying subgradient algorithm is the

non-differentiability of Lagrangian dual objective function (piece-wise concave/convex function in general), we find that the subgradient method has the additional advantage of being amenable to distributed implementation. Specifically, the subgradient method has the following properties.

- Subgradient computation only requires local traffic information $\langle \mathbf{1}, \mathbf{X}^T \mathbf{e}_l \rangle$ and the available link capacity information $\Phi_l(W_l, \mathbf{Q}_l)$ at each link l: The subgradient can be computed as $\partial\Theta(\mathbf{u})/\partial u_l = \Phi_l(W_l, \mathbf{Q}_l) - \langle \mathbf{1}, \mathbf{X}^T \mathbf{e}_l \rangle$. In this case, subgradient only involves local variables W_l, $\mathbf{Q}_l$, and $\mathbf{X}^T \mathbf{e}_l$ at each link and thus can be computed locally.

- The choice of step sizes can be chosen as $\lambda_k = \beta/k$, $k = 1, 2, \ldots$, where $0 < \beta \leq 1$ is a predefined constant. This step size choice obviously satisfies the convergence conditions. This choice of step size depends only on the iteration index k (can be defined as some function of elapsed time), and does not require any other global knowledge. In conjunction with the first property, the dual variable, in the iterative form of $u_l^{(k+1)} = u_l^{(k)} + \lambda_k(\partial\Theta_l(\mathbf{u})/\partial u_l)$, can also be computed locally.

- The objective functions Θ_{link} and Θ_{net} can be decomposed such that each node in the network can perform the computation locally. Recall that the link layer subproblem is: Max $\Theta_{\text{link}} \triangleq \sum_{l=1}^{L} u_l C_l(\mathbf{Q}_l)$ s.t. $\sum_{l \in \mathcal{O}(n)} \text{Tr}\{\mathbf{Q}_l\} \leq P_{\max}, \mathbf{Q}_l \succeq 0$, $\forall l$, where $\mathcal{O}(n)$ denotes the set of outgoing links from node n. For this subproblem, it is not hard to see that it can be decomposed into a set of new

subproblems at each node n of the following form:

$$\begin{array}{ll} \text{Maximize} & \Theta_{\text{link}}^{(n)} \triangleq u_l C_l(\mathbf{Q}_l) \\ \text{subject to} & \sum_{l\in\mathcal{O}(n)} \text{Tr}\{\mathbf{Q}_l\} \leq P_{\max} \\ & \mathbf{Q}_l \succeq 0, l \in \mathcal{O}\left(n\right). \end{array} \tag{4.6.1}$$

The dual original link layer subproblem can then be transformed to $\Theta_{\text{link}} = \sum_{n=1}^{N} \Theta_{\text{link}}^{(n)}$. This suggests that the optimization of each problem $\Theta_{\text{link}}^{(n)}$ in (4.6.1) only requires local information of channel gains (all outgoing links from node n) and the locally-computed dual variable u_l. Thus, the link layer subproblem can be solved distributively. Likewise, for the network layer subproblem (i.e., Max $\Theta_{\text{net}} \triangleq \sum_{f=1}^{F} \ln(s_f) - \sum_l u_l \langle \mathbf{1}, \mathbf{X}^T \mathbf{e}_l \rangle$, s.t. flow balance constraints for all flows), we may decompose it into a set of subproblems based on the source node of each session f:

$$\begin{array}{ll} \text{Maximize} & \Theta_{\text{net}}^{(f)} \triangleq \ln(s_f) - \sum_l u_l \langle \mathbf{1}, \mathbf{X}^T \mathbf{e}_l \rangle \\ \text{subject to} & \text{flow balance constraints for flow } f. \end{array} \tag{4.6.2}$$

The original network layer subproblem can then be simply transformed to $\Theta_{\text{net}} = \sum_{f=1}^{F} \Theta_{\text{net}}^{(f)}$. Again, this suggests that the optimization of each problem $\Theta_{\text{net}}^{(f)}$ in (4.6.2) only requires the locally-computed dual variable u_l for the links. In each iteration, all links can send the locally-computed dual information back to the source node of each session. As a result, the network layer subproblem can be solved in a distributed fashion.

The key steps in this distributed algorithm include:

1. Initialize the iteration index $k = 0$, and choose initial values for $u_l^{(0)}$, for all l.

2. At node n, based on the current dual information u_l, where $l \in \mathcal{O}(n)$, node n can solve the decomposed link layer subproblem in (4.6.1). After that, node

n updates dual variables u_l for all its outgoing links using (4.5.5) (where the step size and subgradient computation follow (4.5.7) and (4.5.8)) and broadcast them to its next hop neighbor. In the meantime, upon receiving some dual information from other neighbors, node n relays it to its next hop neighbor excluding the node where this information comes from. Since the network is assumed to be connected, these dual information will eventually reach each source node.

3. At source node f, upon receiving all links' dual information $\mathbf{u}$, node s solves the decomposed network layer subproblem in (4.6.2). After that, source node f has an updated flow rate s_f and updated routing information $x_l^{(f)}$, $\forall l$. Then, source n shall be able to do source routing and store these routing information $x_l^{(f)}$, $\forall l$ in the packet headers.

4. Upon receiving the source routing information, each intermediate node n will route the packets according to the routing information $x_l^{(f)}$ in the packet header.

5. Based on current values of dual variables $u_l^{(k)}$ and the iteration number k, compute $u_l^{(k+1)}$. If $u_l^{(k+1)} - u_l^{(k)} < \epsilon$, or k has reached a predefined number of iterations, the algorithm stops. Otherwise, let $k = k + 1$ and continue.

The distributed implementation of subgradient algorithm is summarized in Algorithm 4.6.1.

4.7 Numerical Results

In this section, we present some pertinent numerical results for our solution procedure. We first describe our simulation settings. As shown in Fig. 4.1, we have 15 nodes

uniformly distributed in a square region of 1200m $\times$ 1200m. Each node in the network is equipped with two antennas. The maximum transmit power for each node is set to $P_{\max} = 20$ dBm (100 mW). The path loss index is 3. The total bandwidth at each node is 20 MHz. There are three sessions in the network: node 14 to node 1, node 6 to node 10, and node 5 to node 4.

After executing our solution procedure, we find that the optimal rates for these three sessions are $s_1 = 125$ Mbps, $s_2 = 190.6$ Mbps, and $s_3 = 258$ Mbps. The routings and flow rates of sessions 1, 2, and 3 are shown in Fig. 4.2, Fig. 4.3, and Fig. 4.4, respectively. These figures show that flow routings for sessions 1, 2, and 3 are all multi-path and multi-hop. It can easily be verified that the flow rates in Figures 4.2, 4.3, and 4.4 satisfy flow conservation.

Denote $W_{(i,j)}$ and $\mathbf{Q}_{(i,j)}$ the bandwidth allocation and power allocation for the transmission from node i to node j. Table 4.1 shows the optimal bandwidth allocation of the network. Table 4.2 shows the optimal power allocation of the network. In Table 4.1, the value in each cell represents the fraction of total bandwidth of the link's transmitting node. For example, $W_{(11,7)} = 0.23$ means that 0.23 of N11's total bandwidth is allocated to the transmission from N11 to N7. In Table 4.2, each cell with four entries corresponds to a 2×2 $\mathbf{Q}$ matrix, which represents a power allocation. Take the transmission from N11 to N7 for example, $\mathbf{Q}_{(11,7)}$ in Table 4.2 says that the power allocations to the two antennas at N11 are 11.83 mW and 12.10 mW. Also, the signals sent through these two antennas, denoted by x_1 and x_2, should also be correlated with power $\mathbb{E}[x_1 x_2^\dagger] = (-0.25 + 0.01i)$ mW and $\mathbb{E}[x_2 x_1^\dagger] = (-0.25 - 0.01i)$ mW.

It can be observed from Table 4.1 and Table 4.2 that not every node allocates

its full power and total bandwidth for its outgoing links. For example, N14 has only one outgoing link $(14, 7)$. We can see that $W_{(14,7)} = 0.71$ and $\mathrm{Tr}(\mathbf{Q}_{(14,7)}) = 79.47$ mW, which shows N14 does not transmit at full power and utilize all of its assigned bandwidth. This is due to the existence of bottleneck nodes in the network. Even if the non-bottleneck nodes increase the total transmit power and bandwidth, it will not help increase its session rate because the end-to-end session rate is bounded by the minimum bottleneck link along the path. In this network, it can be verified that N3, N7, N11, and N12 are bottleneck nodes. For example, at N11, $W_{(11,3)} + W_{(11,5)} + W_{(11,7)} + W_{(11,12)} = 1$ and $\mathrm{Tr}(Q_{(11,3)}) + \mathrm{Tr}(\mathbf{Q}_{(11,5)}) + \mathrm{Tr}(\mathbf{Q}_{(11,7)}) + \mathrm{Tr}(\mathbf{Q}_{(11,12)}) = 100$ mW. This means that the power and bandwidth at N11 has been fully utilized and cannot be further increased.

Table 4.1: Bandwidth allocation of the 15-node network ($\times$20MHz).

$W_{(1,4)}$	0.71	$W_{(4,1)}$	0.44	$W_{(2,9)}$	0.24	$W_{(9,2)}$	0.35	$W_{(2,12)}$	0.41	$W_{(12,2)}$	0.18	$W_{(2,15)}$	0.24
$W_{(15,2)}$	0.40	$W_{(3,4)}$	0.17	$W_{4,3}$	0.24	$W_{(3,8)}$	0.14	$W_{(8,3)}$	0.24	$W_{(3,11)}$	0.23	$W_{(11,3)}$	0.22
$W_{(3,12)}$	0.20	$W_{(12,3)}$	0.24	$W_{(3,13)}$	0.25	$W_{(13,3)}$	0.21	$W_{(4,12)}$	0.24	$W_{(12,4)}$	0.18	$W_{(5,7)}$	0.38
$W_{(7,5)}$	0.20	$W_{(5,11)}$	0.53	$W_{(11,5)}$	0.21	$W_{(6,9)}$	0.37	$W_{(9,6)}$	0.29	$W_{(6,15)}$	0.45	$W_{(15,6)}$	0.24
$W_{(7,10)}$	0.30	$W_{(10,7)}$	0.24	$W_{(7,11)}$	0.22	$W_{(11,7)}$	0.23	$W_{(7,13)}$	0.15	$W_{(13,7)}$	0.18	$W_{(7,14)}$	0.14
$W_{(14,7)}$	0.71	$W_{(8,10)}$	0.27	$W_{(10,8)}$	0.24	$W_{(8,13)}$	0.26	$W_{(13,8)}$	0.18	$W_{(9,15)}$	0.34	$W_{(15,9)}$	0.26
$W_{(10,13)}$	0.24	$W_{(13,10)}$	0.33	$W_{(11,12)}$	0.34	$W_{(12,11)}$	0.39						

We plot the convergence behavior of gradient projection algorithm for the link-physical layer subproblem (with dual variable $\mathbf{u} = \mathbf{1}$) in Fig. 4.5. It can be seen that GP takes 28 iterations to converge to the maximum objective value of $\Theta_{\mathrm{link}}(\mathbf{u})$ with $\mathbf{u} = \mathbf{1}$.

The convergence process of the Lagrangian dual problem is illustrated in Fig. 4.6. The step size selection strategy for the distributed subgradient is $\lambda_k = \frac{0.1}{k}$. In this figure, "Dual UB" denotes the current objective value of the Lagrangian dual function, which can be thought of as an upper bound of the primal objective value. "Primal

Table 4.2: Power allocation in the 15-node ad hoc network (mW).

$\mathbf{Q}_{(1,4)}$	35.46	$0.00-0.00i$	$\mathbf{Q}_{(4,1)}$	21.84	$-0.09+0.02i$	$\mathbf{Q}_{(2,9)}$	11.82	$0.00-0.00i$
	$0.00+0.00i$	35.46		$-0.09-0.02i$	21.87		$0.00+0.00i$	11.82
$\mathbf{Q}_{(9,2)}$	17.23	$-0.09+0.10i$	$\mathbf{Q}_{(2,12)}$	24.51	$-0.08-0.08i$	$\mathbf{Q}_{(12,2)}$	8.87	$0.00-0.00i$
	$-0.09-0.10i$	17.47		$-0.08+0.08i$	23.98		$0.00+0.00i$	8.87
$\mathbf{Q}_{(2,15)}$	11.82	$0.00-0.00i$	$\mathbf{Q}_{(15,2)}$	20.58	$-0.01+0.16i$	$\mathbf{Q}_{(3,4)}$	8.90	$0.01+0.00i$
	$0.00+0.00i$	11.82		$-0.01-0.16i$	20.50		$0.01-0.00i$	8.91
$\mathbf{Q}_{(4,3)}$	11.82	$0.000-0.000i$	$\mathbf{Q}_{(3,8)}$	7.36	$0.10+0.01i$	$\mathbf{Q}_{(8,3)}$	12.18	$0.00-0.00i$
	$0.00+0.00i$	11.82		$0.10-0.01i$	7.23		$0.00+0.00i$	12.15
$\mathbf{Q}_{(3,11)}$	11.37	$0.00+0.33i$	$\mathbf{Q}_{(11,3)}$	10.76	$0.00+0.00i$	$\mathbf{Q}_{(3,12)}$	9.60	$0.13-0.18i$
	$0.00-0.33i$	12.04		$0.00-0.00i$	10.70		$0.13+0.18i$	10.27
$\mathbf{Q}_{(12,3)}$	11.75	$0.00-0.00i$	$\mathbf{Q}_{(3,13)}$	12.22	$0.35+0.69i$	$\mathbf{Q}_{(13,3)}$	11.09	$-0.03-0.05i$
	$0.00+0.00i$	11.73		$0.35-0.69i$	12.09		$-0.03+0.05i$	11.08
$\mathbf{Q}_{(4,12)}$	11.82	$0.00-0.00i$	$\mathbf{Q}_{(12,4)}$	9.22	$0.04-0.09i$	$\mathbf{Q}_{(5,7)}$	19.62	$-0.01-0.08i$
	$0.00+0.00i$	11.82		$0.04+0.09i$	9.25		$-0.01+0.08i$	19.68
$\mathbf{Q}_{(7,5)}$	9.54	$0.00-0.04i$	$\mathbf{Q}_{(5,11)}$	26.56	$-0.10-0.14i$	$\mathbf{Q}_{(11,5)}$	10.42	$0.01-0.12i$
	$0.00+0.04i$	9.51		$-0.10+0.14i$	26.90		$0.01+0.12i$	10.39
$\mathbf{Q}_{(6,9)}$	21.22	$0.00-0.00i$	$\mathbf{Q}_{(9,6)}$	14.78	$-0.10-0.08i$	$\mathbf{Q}_{(6,15)}$	22.84	$0.21+0.14i$
	$0.00+0.00i$	21.22		$-0.10+0.08i$	14.55		$0.21-0.14i$	23.24
$\mathbf{Q}_{(15,6)}$	11.82	$0.00-0.00i$	$\mathbf{Q}_{(7,10)}$	14.19	$-0.05-0.09i$	$\mathbf{Q}_{(10,7)}$	11.82	$0.00-0.00i$
	$0.00+0.00i$	11.82		$-0.05+0.09i$	13.82		$0.00+0.00i$	11.82
$\mathbf{Q}_{(7,11)}$	12.48	$1.15-0.43i$	$\mathbf{Q}_{(11,7)}$	11.83	$-0.25-0.01i$	$\mathbf{Q}_{(7,13)}$	7.54	$0.12-0.02i$
	$1.15+0.43i$	11.21		$-0.25+0.01i$	12.10		$0.12+0.02i$	7.51
$\mathbf{Q}_{(13,7)}$	8.87	$0.00-0.00i$	$\mathbf{Q}_{(7,14)}$	7.10	$0.00-0.00i$	$\mathbf{Q}_{(14,7)}$	39.76	$0.19+0.04i$
	$0.00+0.00i$	8.87		$0.00+0.00i$	7.10		$0.19-0.04i$	39.71
$\mathbf{Q}_{(8,10)}$	15.00	$0.05-0.05i$	$\mathbf{Q}_{(10,8)}$	11.82	$0.0-0.00i$	$\mathbf{Q}_{(8,13)}$	12.80	$0.03-0.04i$
	$0.05+0.05i$	14.95		$0.016+0.00i$	11.82		$0.03+0.04i$	12.82
$\mathbf{Q}_{(13,8)}$	10.00	$0.54-0.59i$	$\mathbf{Q}_{(9,15)}$	17.21	$0.32+0.28i$	$\mathbf{Q}_{(15,9)}$	13.09	$0.10-0.08i$
	$0.54+0.59i$	9.69		$0.32-0.28i$	16.77		$0.10+0.08i$	12.95
$\mathbf{Q}_{(10,13)}$	11.82	$0.00-0.00i$	$\mathbf{Q}_{(13,10)}$	15.70	$0.15+0.01i$	$\mathbf{Q}_{(11,12)}$	16.76	$-0.04+0.02i$
	$0.00+0.00i$	11.82		$0.15-0.01i$	15.71		$-0.04-0.02i$	17.04
$\mathbf{Q}_{(12,11)}$	20.85	$0.20+0.11i$						
	$0.20-0.11i$	19.47						

Feasible Solution" denotes the current primal feasible solution recovered from the Lagrangian dual, which can be thought of as a lower bound of the optimal primal objective value. During each iteration, the cutting-plane and subgradient methods each solve the Lagrangian dual problem. The upper bounds of the optimal objective value are non-increasing with iterations. Meanwhile, the primal feasible objective values keep increasing with iterations. As expected, the upper bounds and the lower bound converge and give the optimal solution, as shown in this figure. We find that for this 15-node ad hoc network, centralized cutting-plane algorithm and distributed

subgradient algorithm converge in approximately 115 iterations and 160 iterations, respectively. The optimal value of the network utility function is 6.64 (in log(b/s/Hz)).

For performance gain comparison, we compare the heuristic strategy discussed in [157], which uses equal power allocation and equal bandwidth allocation to all the outgoing links at each node, which yields an objective value of 3.01. In comparison, the performance gap to our optimal value (6.64) is significant.

4.8 Chapter Summary

In this chapter, we investigated the problem of cross-layer optimization of routing, power allocation, and bandwidth allocation for MIMO-based ad hoc networks. We developed a mathematical solution procedure, which combines Lagrangian decomposition, gradient projection, cutting-plane, and subgradient methods. We showed the decomposable structure of the Lagrangian dual problem and the high efficiency of our proposed algorithms make it an attractive method for optimizing the performance of MIMO-based ad hoc networks across multiple layers. We also presented distributed implementation for the subgradient approach. Our results showed that the performance gain from our optimal cross-layer design is significant.

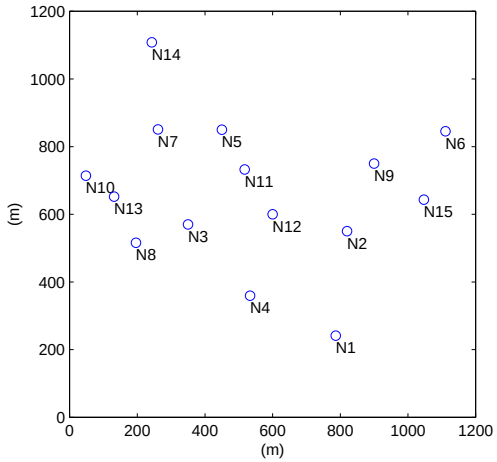


Figure 4.1: Network topology of a 15-node ad hoc network.

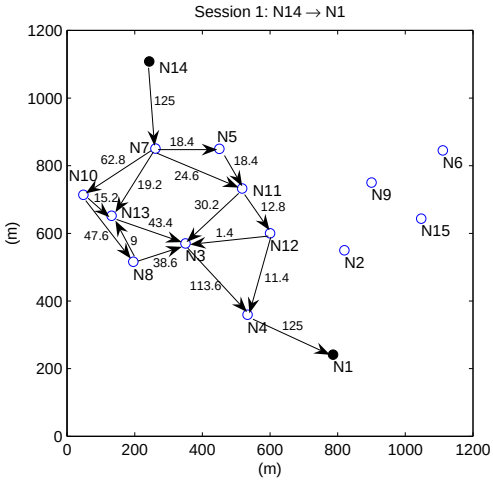


Figure 4.2: Routing and flow rates of session 1 (in Mbps).

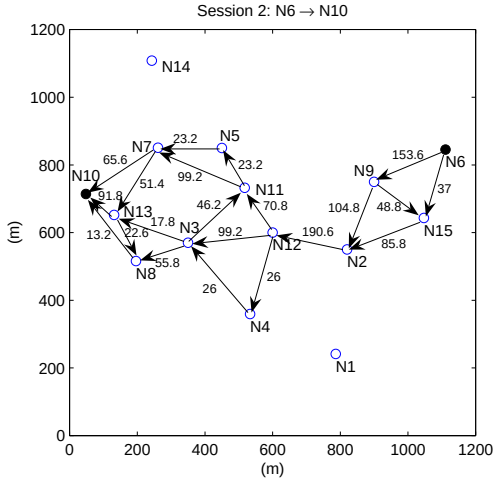


Figure 4.3: Routing and flow rates of session 2 (in Mbps).

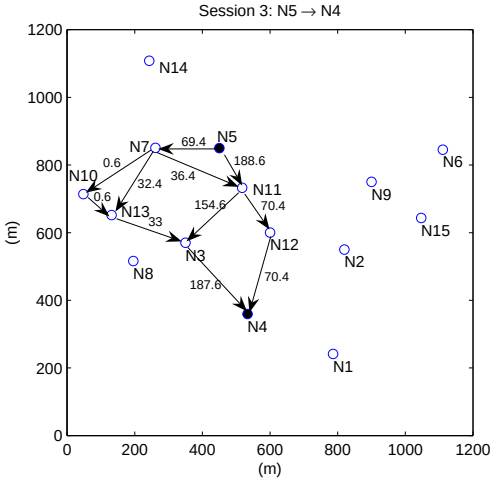


Figure 4.4: Routing and flow rates of session 3 (in Mbps).

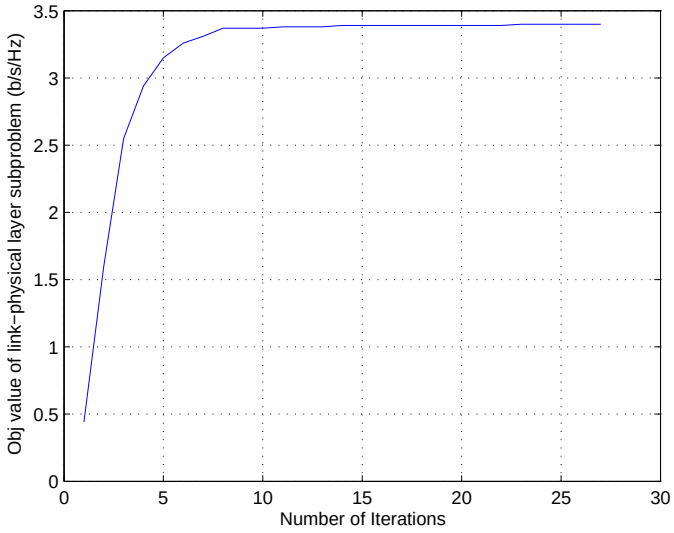


Figure 4.5: Convergence behavior of gradient projection algorithms for link-physical layer subproblem (for $\mathbf{u} = \mathbf{1}$).

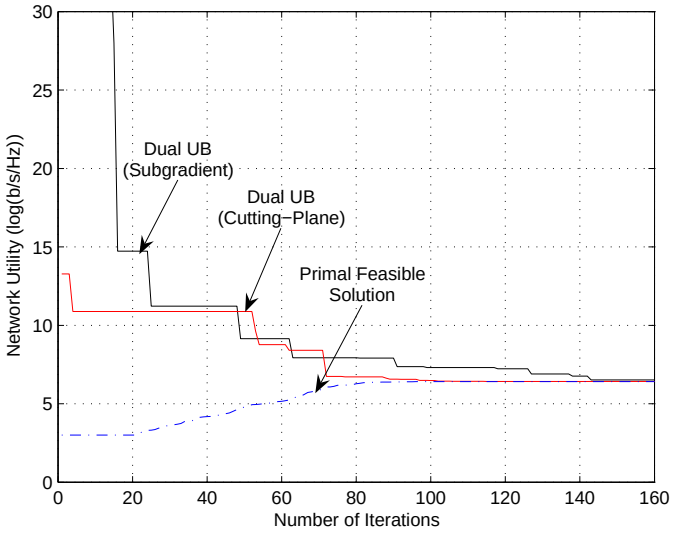


Figure 4.6: Convergence behavior of cutting-plane (centralized) and subgradient (distributed) algorithms for the 15-node ad hoc network example.

Chapter 5

Multi-hop MIMO Ad Hoc Networks with Dirty Paper Coding

5.1 Introduction

MIMO has found its ways into many infrastructure-based wireless networking standards, such as wireless LAN (802.11n), WiMAX access networks (802.16), and 4G cellular networks (LTE). Meanwhile, MIMO's tremendous capacity potential has been also receiving increasing attention from the ad hoc network research community, as evidenced by some interesting recent works on MIMO ad hoc network optimization (see, e.g., [24, 53, 77, 87]).

These early efforts on MIMO ad hoc networks optimization, however, are unaware of a new and yet fundamental information-theoretic result called *dirty paper coding* (DPC), which has a huge ramification on fully exploiting the capacity gain of multi-user MIMO channels. In their ground breaking work [154], Weigarten *et al.* showed that DPC is the optimal transmission scheme for multi-user MIMO broadcast channels (MIMO-BC) in the sense that the DPC rate region $\mathcal{C}_{\text{DPC}}$ coincides with MIMO-BC's capacity region $\mathcal{C}_{\text{BC}}$, i.e., $\mathcal{C}_{\text{BC}} = \mathcal{C}_{\text{DPC}}$. More importantly, this result reveals that, in multi-user communication channels, those traditional transmission schemes based on

interference avoidance (e.g., time-slotted or frequency division multiplexing systems) are *sub-optimal* since they can only achieve a fraction of the entire capacity region. In addition to theoretical study, initial attempts on practical implementations of DPC have also been reported [119, 164, 165]. Since capacity is a critical factor affecting ad hoc networks, an in-depth investigation on how to exploit DPC's potential in MIMO ad hoc networks is necessary. So far, this question remains open and our main objective in this chapter is to fill this gap.

However, due to the stark contrast between DPC and the traditional transmission schemes based on interference avoidance, there are some unique challenges to optimize the performance of DPC-based MIMO ad hoc networks.

1) **Complex interactions between protocol layers:** The network capacity gain obtained by using DPC in MIMO ad hoc networks depends on the coordinated mechanisms among power allocation at the physical layer, DPC encoding ordering at the link layer, and multi-hop routing at the network layer. In fact, an improperly designed system could significantly diminish the potential capacity gain from DPC and MIMO. As a result, joint optimization across multiple layers is not only desirable, but absolutely necessary.

2) **Non-convex problem structure under co-channel interference:** Due to the co-channel interference induced by simultaneous transmissions to different users in DPC, the achievable rate region under DPC is non-convex with respect to the transmit power variables. This non-convexity imposes challenges on designing efficient algorithms for cross-layer optimization.

3) **Exponentially large search space inherent in DPC:** Consider a node having, say, K neighboring nodes such that they form a MIMO-BC. There exists a

total of $K!$ possible DPC encoding orders. Different encoding orders coupled with different power allocation will directly affect the achievable DPC rate region. Thus, even the DPC optimization for a single MIMO-BC with K receiving nodes has an exponentially large search space, not to mention in an ad hoc network environment that may contain multiple MIMO-BCs.

In this chapter, we strive to answer the following questions:

1) How to incorporate DPC into multi-hop MIMO ad hoc network modeling and reduce the associated complexity in the mathematical formulation?

2) How to obtain an efficient algorithm to optimize MIMO ad hoc networks with DPC?

3) How much network performance gain can be achieved by using DPC compared to the conventional transmission schemes based on interference avoidance?

5.1.1 Summary of Main Results and Contributions

Corresponding to the questions posed above, the main results of this chapter can be stated as follows. First, we show that, after incorporating DPC into MIMO ad hoc network modeling, the mathematical formulation has a challenging non-convex structure. To reduce this complexity, we reformulate the original problem into an equivalent convex problem. This reformulation is important because it is now possible to find an optimal solution to the original non-convex problem.

We note that, although being convex, the reformulated problem still has an exponentially large search space that arises from enumerating all possible DPC encoding orders. Further, MIMO channels' complex matrix structure renders the search space to even higher dimensions. To address these difficulties, we show that, through dual

decomposition and further simplifications, the exponential DPC link layer subproblem can be solved *without* enumerating all possible encoding orders. This important finding, combined with powerful techniques from matrix differential calculus, lead us to design a polynomial-time algorithm for the reformulated problem.

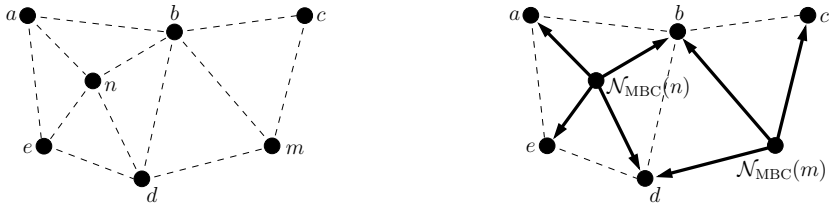
Finally, we conduct an extensive numerical study to show that, by judiciously managing radio resources, DPC indeed achieves a significant network performance gain compared to conventional transmission schemes based on interference avoidance.

5.1.2 Chapter Organization

The remainder of this chapter is organized as follows. In Section 5.2, we discuss the network model and problem formulation. In Section 5.3, we demonstrate how to reformulate the original problem to yield an equivalent convex optimization problem. In Section 5.4, we introduce the key components for solving the reformulated problem. Numerical results are presented in Section 5.5 to illustrate the efficacy of our proposed solution procedure and to study the network performance gain by using DPC. Section 5.6 concludes this chapter.

5.2 Network Model and Problem Formulation

In this chapter, we study how to exploit DPC's potential in cross-layer optimization for multi-hop MIMO ad hoc networks. The topology of a multi-hop MIMO wireless ad hoc network is represented by a directed graph, denoted by $\mathcal{G} = \{\mathcal{N}, \mathcal{L}\}$, where $\mathcal{N}$ and $\mathcal{L}$ are the sets of nodes and all possible MIMO links, respectively. By saying "possible" we mean that the distance between a pair of nodes is less than or equal to the maximum transmission range $D_{\max}$, i.e., $\mathcal{L} = \{(i,j) : D_{ij} \leq D_{\max}, i,j \in \mathcal{N}, i \neq j\}$,



(a) A MIMO ad hoc network. (b) MIMO-BC subsystems.

Figure 5.1: Visualizing a MIMO ad hoc network as MIMO-BC subsystems.

where D_{ij} represents the distance between node i and node j. $D_{\max}$ can be determined by a node's maximum transmission power. Suppose that the cardinalities of the sets $\mathcal{N}$ and $\mathcal{L}$ are $|\mathcal{N}| = N$ and $|\mathcal{L}| = L$, respectively. For convenience, we index the links numerically (i.e., $1, 2, \ldots, L$).

In this chapter, DPC is incorporated into multi-hop MIMO ad hoc networks as follows. First, we note that a multi-hop MIMO ad hoc networks can be visualized as a collection of MIMO-BC subsystems (as illustrated in Fig. 5.1). We denote $\mathcal{N}_{\mathrm{MBC}}(n)$ the set of nodes in a MIMO-BC subsystem where node n is the transmitter. As an example, for the ad hoc network in Fig. 5.1(a), we have $\mathcal{N}_{\mathrm{MBC}}(n) = \{n, a, b, d, e\}$ and $\mathcal{N}_{\mathrm{MBC}}(m) = \{m, b, c, d\}$, as shown in Fig. 5.1(b). Next, we apply DPC within each MIMO-BC subsystems. The interference between each MIMO-BC subsystems is avoided by some appropriate channel assignment. In other words, we eliminate the effect of co-channel interference from different MIMO-BC subsystems, but keep the freedom of managing the co-channel interference within each MIMO-BC subsystem by using DPC. The channel assignment satisfies the following conditions: 1) if node $k \in \mathcal{N}_{\mathrm{MBC}}(n) \cap \mathcal{N}_{\mathrm{MBC}}(m)$, the signals from node n and node m are orthogonal at k; and 2) for a node $m \notin \mathcal{N}_{\mathrm{MBC}}(n)$, there is no interference at m caused by node n.

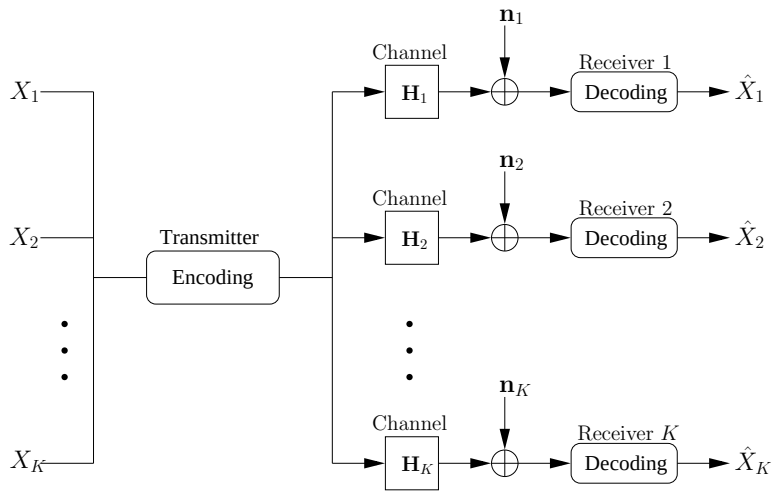


Figure 5.2: The link layer model for a MIMO-BC with DPC.

5.2.1 DPC-Based Link Layer

Before describing the DPC-based link layer, we first briefly introduce some basics of MIMO channel modeling. Due to multiple antennas on both sides of a channel, a MIMO link l can be represented by a matrix $\mathbf{H}_l$. The received complex base-band signal vector in such a MIMO channel can be written as $\mathbf{y}_l = \sqrt{\rho_l}\mathbf{H}_l\mathbf{x}_l + \mathbf{n}_l$, where $\mathbf{y}_l$ and $\mathbf{x}_l$ represent the received and transmitted signal vectors; ρ_l represents the signal-to-noise ratio (SNR); and $\mathbf{n}_l$ is a normalized additive white Gaussian noise vector. The transmit power of MIMO link l is represented by a matrix $\mathbf{\Gamma}_l$, defined as the covariance of the signal vector $\mathbf{x}_l$ (i.e., $\mathbf{\Gamma}_l = \mathbb{E}\{\mathbf{x}_l \cdot \mathbf{x}_l^\dagger\}$). The definition of $\mathbf{\Gamma}_l$ implies that it is Hermitian and PSD. Physically, $\mathbf{\Gamma}_l$ represents the power allocation in different antenna elements and $\mathrm{Tr}\{\mathbf{\Gamma}_l\}$ is the total transmission power at the transmitter of link l.

With these basics of MIMO channel modeling, we now proceed to describe the DPC-based link layer. Fig. 5.2 illustrates a K-user MIMO-BC with DPC, where messages $X_1, \ldots, X_K$ are jointly encoded and transmitted through K MIMO channels. Let $\mathbf{H}_i \in \mathbb{C}^{M \times M}$ represent the channel gain matrix from the transmitting node to the

i-th receiving node, where M represents the number of antennas at each node. Let π denote an encoding order of the set $\{1,\ldots,K\}$: $\pi(i)=j$ represents that the i-th position in π is user j. From the DPC encoding process, the achievable rate for the $\pi(i)$-th receiving node can be computed as (assuming unit bandwidth) [48]

$$R_{\pi(i)}^{\mathrm{DPC}}(\mathbf{\Gamma})=\log\left|\mathbf{I}+\rho_{\pi(i)}\mathbf{H}_{\pi(i)}\mathbf{\Gamma}_{\pi(i)}\mathbf{H}_{\pi(i)}^{\dagger}\mathbf{W}_{\pi(i)}^{-1}\right|, \tag{5.2.1}$$

where $\mathbf{W}_{\pi(i)}\triangleq\mathbf{I}+\sum_{j=i+1}^{K}\rho_{\pi(j)}\mathbf{H}_{\pi(j)}\mathbf{\Gamma}_{\pi(j)}\mathbf{H}_{\pi(j)}^{\dagger}$ represents the power of aggregated interference plus noise at node i. Physically, (5.2.1) represents that under DPC, the $\pi(i)$-th user only sees interference from users $\pi(i+1),\ldots,\pi(K)$. We point out that the term $\mathbf{W}_{\pi(i)}$ makes (5.2.1) *neither a concave nor a convex* function of the power allocation variables $\mathbf{\Gamma}_i$.

Let $\mathbf{H}^{(n)}=[\mathbf{H}_l:l\in\mathcal{O}(n)]^T$ be the collection of all outgoing channel gain matrices at node n and $\mathbf{\Gamma}^{(n)}=[\mathbf{\Gamma}_l:l\in\mathcal{O}(n)]^T$ be the collection of all matrix power allocation variables at node n. Due to finite transmit power at each node, we have the following power allocation constraint $\sum_{l\in\mathcal{O}(n)}\mathrm{Tr}\{\mathbf{\Gamma}_l\}\leq 1$, for all n. Let $\mathbf{R}^{\mathrm{DPC}}(\pi,\mathbf{\Gamma}^{(n)})\triangleq[R_{\pi(1)}^{\mathrm{DPC}}(\mathbf{\Gamma}^{(n)}),\ldots,R_{\pi(K)}^{\mathrm{DPC}}(\mathbf{\Gamma}^{(n)})]^T$ be the achievable rate vector under an encoding order π and some power allocation $\mathbf{\Gamma}^{(n)}$ at node n. Denote $P_{\max}^{(n)}$ the maximum transmit power limit at node n. The DPC rate region $\mathcal{C}_{\mathrm{DPC}}(P_{\max}^{(n)},\mathbf{H}^{(n)})$ is defined as the convex hull of the union of all rate vectors over all $\mathbf{\Gamma}^{(n)}$ satisfying the power allocation constraint and over all $K!$ possible encoding orders, i.e.,

$$\mathcal{C}_{\mathrm{DPC}}(P_{\max}^{(n)},\mathbf{H}^{(n)})\triangleq Conv\left(\cup_{\pi,\mathbf{\Gamma}^{(n)}}\{\mathbf{R}^{\mathrm{DPC}}(\pi,\mathbf{\Gamma}^{(n)})\}\right),$$

where $Conv(\cdot)$ represents the convex hull operation.

5.2.2 Network Layer

The routing problem at the network layer considers, for a session f, $1 \leq f \leq F$, how to relay its data from the source node $\mathrm{src}(f)$ to its destination node $\mathrm{dst}(f)$.

The network topology of $\mathcal{G}$ can be represented by a *node-arc incidence matrix* (NAIM) [9] $\mathbf{A} \in \mathbb{R}^{N \times L}$, where each entry a_{nl} associating with node n and arc l is defined as

$$a_{nl} = \begin{cases} 1 & \text{if } n \text{ is the transmitting node of arc } l \\ -1 & \text{if } n \text{ is the receiving node of arc } l \\ 0 & \text{otherwise.} \end{cases} \tag{5.2.2}$$

We define $\mathcal{O}(n)$ and $\mathcal{I}(n)$ as the sets of links that are outgoing from and incoming to node n, respectively. We use a multicommodity flow model for the routing of data packets across the network. Under this model, flow splitting and multi-path routing are both allowed. We define a *source-sink vector* $\mathbf{s}_f \in \mathbb{R}^N$, whose entries, other than at the positions of $\mathrm{src}(f)$ and $\mathrm{dst}(f)$, are all zeros. In order for the network to operate stably, the so-called flow balance condition must be satisfied. That is, the traffic that flows out of a source node will be completely absorbed by its corresponding destination node. Thus, we have that $(\mathbf{s}_f)_{\mathrm{src}(f)} = -(\mathbf{s}_f)_{\mathrm{dst}(f)}$ (the minus sign means the reverse direction). Without loss of generality, we can let $(\mathbf{s}_f)_{\mathrm{src}(f)} \geq 0$ and simply denote it as a scalar s_f. Thus, we can write the source-sink vector of flow f as

$$\mathbf{s}_f = s_f \begin{bmatrix} \cdots & 1 & \cdots & -1 & \cdots \end{bmatrix}^T \in \mathbb{R}^N, \tag{5.2.3}$$

where the dots represent zeros, and 1 and -1 are in the positions of $\mathrm{src}(f)$ and $\mathrm{dst}(f)$, respectively.[1] We use the notation "$=_{x,y}$" to represent the component-wise equality of a vector except at the x-th and the y-th entries. Thus, we can write (5.2.3) compactly

[1]For the source-sink vector of a session f, 1 does not necessarily appear before -1 as in (5.2.3), which is only for illustrative purpose.

as $\mathbf{s}_f =_{\mathrm{src}(f),\mathrm{dst}(f)} \mathbf{0}$. In addition, using a matrix $\mathbf{S} \triangleq \begin{bmatrix} \mathbf{s}_1 & \mathbf{s}_2 & \dots & \mathbf{s}_F \end{bmatrix} \in \mathbb{R}^{N\times F}$ to denote the collection of all source-sink vectors, we have

$$\mathbf{S}\mathbf{e}_f =_{\mathrm{src}(f),\mathrm{dst}(f)} \mathbf{0}, \quad 1 \le f \le F, \tag{5.2.4}$$

$$\langle \mathbf{1}, \mathbf{S}\mathbf{e}_f \rangle = 0, \qquad 1 \le f \le F, \tag{5.2.5}$$

$$(\mathbf{S}\mathbf{e}_f)_{\mathrm{src}(f)} = s_f, \qquad 1 \le f \le F, \tag{5.2.6}$$

where $\mathbf{e}_f$ is the f-th unit column vector.

On link l, we let $z_l^{(f)} \ge 0$ be the amount of flow of session f. We define $\mathbf{z}^{(f)} \in \mathbb{R}^L$ as the flow vector for session f. Due to the flow balance condition at each node n, the components of vectors $\mathbf{z}^{(f)}$ and $\mathbf{s}_f$ satisfy the following condition: $\sum_{l\in\mathcal{O}(n)} z_l^{(f)} - \sum_{l\in\mathcal{I}(n)} z_l^{(f)} = (\mathbf{s}_f)_n$, $1 \le n \le N$, $1 \le f \le F$. With NAIM, the flow balance condition across the entire network can be written compactly as $\mathbf{A}\mathbf{z}^{(f)} = \mathbf{s}_f, \quad 1 \le f \le F$. We use the matrix $\mathbf{Z} \triangleq \begin{bmatrix} \mathbf{z}^{(1)} & \mathbf{z}^{(2)} & \dots & \mathbf{z}^{(F)} \end{bmatrix} \in \mathbb{R}^{L\times F}$ to denote the collection of all flow vectors. With $\mathbf{Z}$ and $\mathbf{S}$, we can further write the flow balance condition as $\mathbf{A}\mathbf{Z} = \mathbf{S}$.

5.2.3 Problem Formulation

In this chapter, we wish to jointly optimize power allocation and multi-hop multi-path routing such that some network utility function is maximized. We adopt proportional fairness (i.e., $\ln(s_f)$ for session f [75]) as the network utility in our objective function. We note that our analysis in this chapter is general and other types of utility functions can also be adopted (e.g., the weighted proportional fairness (i.e., $w_f \ln s_f$) or sum rate capacity (i.e., $\sum_f s_f$), etc). Since the total amount of flow in each link l cannot exceed its capacity limit, we have, in matrix form, $\langle \mathbf{1}, \mathbf{Z}^T\mathbf{e}_l \rangle \le R_l^{\mathrm{DPC}}(\mathbf{\Gamma}^{(n)})$ for all $l \in \mathcal{O}(n)$. Putting together other constraints in the network and link layers, we have

the problem formulation as follows:

$$
\begin{array}{lll}
\underline{\textbf{CRPA:}} & & \\
\text{Maximize} & \sum_{f=1}^{F} \ln(s_f) & (5.2.7) \\
\text{subject to} & \mathbf{AZ} = \mathbf{S} & \\
& \mathbf{Z} \geq \mathbf{0} & \\
& \mathbf{S}\mathbf{e}_f =_{\mathrm{src}(f),\mathrm{dst}(f)} \mathbf{0} & \forall\, f \\
& \langle \mathbf{1}, \mathbf{S}\mathbf{e}_f \rangle = 0 & \forall\, f \\
& (\mathbf{S}\mathbf{e}_f)_{\mathrm{src}(f)} = s_f & \forall\, f \\
& \langle \mathbf{1}, \mathbf{Z}^T \mathbf{e}_l \rangle \leq R_l^{\mathrm{DPC}}(\mathbf{\Gamma}^{(n)}) & \forall\, l \in \mathcal{O}\,(n)\,, \forall n \\
& \mathbf{R}^{\mathrm{DPC}}(\mathbf{\Gamma}^{(n)}) \in \mathcal{C}_{\mathrm{DPC}}^{(n)}(P_{\max}^{(n)}, \mathbf{H}^{(n)}) & \forall\, n \\
& \sum_{l \in \mathcal{O}(n)} \mathrm{Tr}\{\mathbf{\Gamma}_l\} \leq 1 & \forall\, n \\
& \mathbf{\Gamma}_l \succeq 0 & \forall\, l \\
& \text{Variables : } \mathbf{S},\, \mathbf{Z},\, \mathbf{\Gamma}. &
\end{array}
$$

As pointed out earlier, the rate expression $R_l^{\mathrm{DPC}}(\mathbf{\Gamma})$ is non-convex with respect to $\mathbf{\Gamma}$. As a result, problem (5.2.7) is a challenging non-convex optimization problem.

5.3 Problem Reformulation

To overcome the non-convex difficulty, in this section, we propose an approach to reformulate and "convexify" problem (5.2.7). We first introduce a result called uplink-downlink duality [152, 161], which will play an important role in the reformulation.

Simply put, uplink-downlink duality says that for every MIMO-BC, there exists a dual MIMO multiple access channel (MIMO-MAC) that achieves the same capacity region as that of the MIMO-BC. Fig. 5.3 shows a way to construct a dual MIMO-MAC for a given MIMO-BC. First, we change the receivers in the MIMO-BC into transmitters and change the transmitter into a receiver. Next, we conjugate transpose

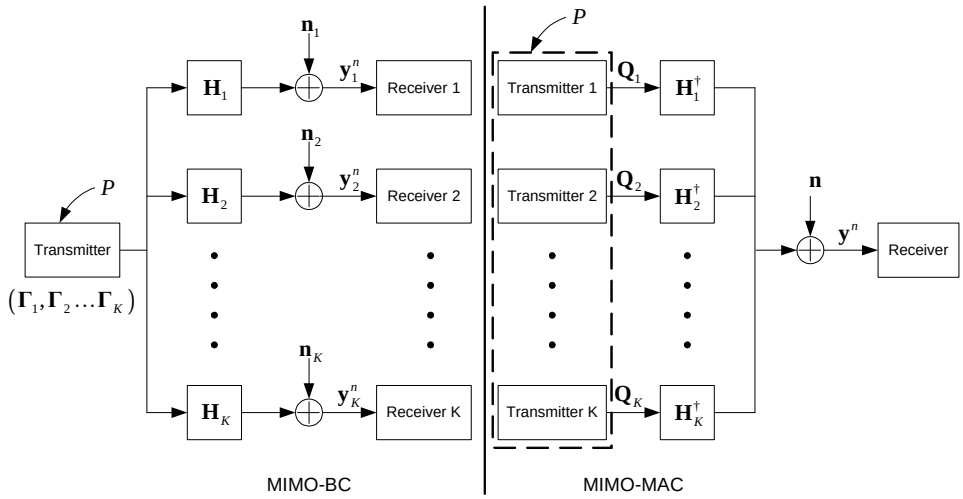


Figure 5.3: A MIMO-BC and its dual MIMO-MAC.

all the channel gain matrices. Finally, we let the sum of the maximum powers of the transmitters in the dual MIMO-MAC equal to the maximum transmit power P in the MIMO-BC. As shown in Fig. 5.3, we use $\mathbf{Q}$ to denote the covariance matrices (i.e., the power allocation variables) in the dual MIMO-MAC channels to differentiate from the covariance matrices $\mathbf{\Gamma}$ in the original MIMO-MAC.

Denote the capacity region of a dual MIMO-MAC as $\mathcal{C}_{\mathrm{MAC}}(P, \mathbf{H}^\dagger)$. We now state the uplink-downlink duality in the following lemma.

Lemma 5.3.1. *The DPC rate region of a MIMO-BC with maximum power constraint P is equal to the capacity region of the dual MIMO MAC with a sum power constraint P, i.e., $\mathcal{C}_{\mathrm{DPC}}(P, \mathbf{H}) = \mathcal{C}_{\mathrm{MAC}}(P, \mathbf{H}^\dagger)$.*

The proof of this lemma can be done in several ways [151, 152, 161]. It is also shown in [151] that any rate vector in a MIMO-BC with a particular encoding order can be achieved in its dual MIMO-MAC with the *reversed* successive decoding order.

In this chapter, we reformulate problem (5.2.7) by exploiting the fact that $\mathcal{C}_{\mathrm{DPC}}(P, \mathbf{H}) =$

$\mathcal{C}_{\mathrm{MAC}}(P, \mathbf{H}^{\dagger})$. More specifically, we replace $\mathcal{C}_{\mathrm{DPC}}(\cdot)$ in (5.2.7) by the capacity regions of the dual MIMO-MAC channels $\mathcal{C}_{\mathrm{MAC}}(\cdot)$ and optimize over $\mathbf{Q}$ instead of $\mathbf{\Gamma}$. The motivation of this reformulation is that we can show that the dual MIMO-MAC capacity region is convex. We formally state this result in the following lemma.

Lemma 5.3.2. *The capacity region of a K-user dual MIMO-MAC with a sum power constraint $\sum_{i=1}^{K} \mathrm{Tr}(\mathbf{Q}_i) \leq P_{\max}$ is convex with respect to the power allocation variables $\mathbf{Q}_1, \ldots, \mathbf{Q}_K$.*

Proof. Denote the input signals of the K users by $\mathbf{x}_1, \ldots, \mathbf{x}_K$, respectively, and denote the output of the dual MIMO-MAC channel by $\mathbf{y}$. For simplicity, we absorb ρ_i into $\mathbf{H}_i$ in this proof. From [29, Theorem 14.3.5], we can derive the capacity region of a dual MIMO-MAC as follows:

$$\mathcal{C}_{\mathrm{MAC}}(\mathbf{Q}_1, \ldots, \mathbf{Q}_K) = \left\{ (R_1, \ldots, R_K) \left| \begin{array}{l} \sum_{i \in \mathcal{S}} R_i(\mathbf{Q}) \leq \\ I(\mathbf{x}_i, i \in \mathcal{S}; \mathbf{y} | \mathbf{x}_i, i \in \mathcal{S}^c), \\ \forall \mathcal{S} \subseteq \{1, \ldots, K\} \\ \sum_{i=1}^{K} \mathrm{Tr}(\mathbf{Q}_i) \leq P_{\max} \end{array} \right. \right\}, \tag{5.3.1}$$

where the mutual information expression $I(;)$ can be bounded as follows:

$$I(\mathbf{x}_i, i \in \mathcal{S}; \mathbf{y} | \mathbf{x}_i, i \in \mathcal{S}^c) \leq \log \left| \mathbf{I} + \sum_{i \in \mathcal{S}} \mathbf{H}_i^{\dagger} \mathbf{Q}_i \mathbf{H}_i \right|. \tag{5.3.2}$$

Next, we show that $\mathcal{C}_{\mathrm{MAC}}(\mathbf{Q}_1, \ldots, \mathbf{Q}_K)$ in (5.3.1) is convex by using the definition of convex regions. We first arbitrarily choose two achievable rate vectors in $\mathcal{C}_{\mathrm{MAC}}(\mathbf{Q}_1, \ldots, \mathbf{Q}_K)$ and denote them as $[R_1, \ldots, R_K]$ and $[\hat{R}_1, \ldots, \hat{R}_K]$. Suppose that they are achieved by two feasible power vectors $[\mathbf{Q}_1, \ldots, \mathbf{Q}_K]$ and $[\hat{\mathbf{Q}}_1, \ldots, \hat{\mathbf{Q}}_K]$, accordingly. In other words, $[\mathbf{Q}_1, \ldots, \mathbf{Q}_K]$ and $[\hat{\mathbf{Q}}_1, \ldots, \hat{\mathbf{Q}}_K]$ satisfy $\sum_{i=1}^{K} \mathrm{Tr}(\mathbf{Q}_i) \leq P_{\max}$ and $\sum_{i=1}^{K} \mathrm{Tr}(\hat{\mathbf{Q}}_i) \leq P_{\max}$. Now, arbitrarily choose $0 \leq \alpha \leq 1$ and consider the convex

combination

$$[\bar{R}_1, \ldots, \bar{R}_K] = \alpha[R_1, \ldots, R_K] + (1-\alpha)[\hat{R}_1, \ldots, \hat{R}_K].$$

Also, let $\bar{\mathbf{Q}}_i = \alpha\mathbf{Q}_i + (1-\alpha)\hat{\mathbf{Q}}_i$, $i = 1, \ldots, K$. It is clear that $\sum_{i=1}^{K} \mathrm{Tr}(\bar{\mathbf{Q}}_i) \leq P_{\max}$. From (5.3.1) and (5.3.2), we have

$$\begin{aligned} &\alpha \textstyle\sum_{i\in\mathcal{S}} R_i + (1-\alpha) \sum_{i\in\mathcal{S}} \hat{R}_i \leq \alpha \log \left| \mathbf{I} + \sum_{i\in\mathcal{S}} \mathbf{H}_i^\dagger \mathbf{Q}_i \mathbf{H}_i \right| \\ &\quad + (1-\alpha) \log \left| \mathbf{I} + \textstyle\sum_{i\in\mathcal{S}} \mathbf{H}_i^\dagger \hat{\mathbf{Q}}_i \mathbf{H}_i \right|. \end{aligned}$$

Also, since the function $\log|\mathbf{A}|$ is concave for positive semidefinite matrix variable $\mathbf{A}$ [29], it follows from Jensen's inequality that

$$\alpha \textstyle\sum_{i\in\mathcal{S}} R_i + (1-\alpha) \sum_{i\in\mathcal{S}} \hat{R}_i \leq \frac{1}{2} \log \left| \mathbf{I} + \sum_{i\in\mathcal{S}} \mathbf{H}_i^\dagger \bar{\mathbf{Q}}_i \mathbf{H}_i \right|,$$

which means that the convex combination of rate vectors $[R_1, \ldots, R_K]$ and $[\hat{R}_1, \ldots, \hat{R}_K]$ can also be achieved by using the feasible power vector $[\bar{\mathbf{Q}}_1, \ldots, \bar{\mathbf{Q}}_K]$ directly. This completes the proof. □

The reformulated problem for CRPA is shown in (5.3.3) and denoted as CRPA-E. Lemma 5.3.2 immediately implies that problem (5.3.3) is a convex program. Moreover, since $\mathcal{C}_{\mathrm{DPC}}(P, \mathbf{H}) = \mathcal{C}_{\mathrm{MAC}}(P, \mathbf{H}^\dagger)$, it is clear that CRPA-E is equivalent to CRPA in the sense that they achieve the same optimal objective value. Thus, instead of solving an intractable non-convex problem, we can now solve an equivalent convex program. Further, upon solving CRPA-E, we can recover the optimal solution $\mathbf{\Gamma}^*$ for CRPA from the optimal solution $\mathbf{Q}^*$ by using the MAC-to-BC mapping from uplink-downlink duality [151]. Therefore, in the next section, we will focus on solving

CRPA-E.

CRPA-E:

$$
\begin{aligned}
\text{Maximize} \quad & \textstyle\sum_{f=1}^{F} \ln(s_f) && && (5.3.3)\\
\text{subject to} \quad & \mathbf{AZ} = \mathbf{S} \\
& \mathbf{Z} \geq \mathbf{0} \\
& \mathbf{Se}_f =_{\mathrm{src}(f),\mathrm{dst}(f)} \mathbf{0} && \forall\, f \\
& \langle \mathbf{1}, \mathbf{Se}_f \rangle = 0 && \forall\, f \\
& (\mathbf{Se}_f)_{\mathrm{src}(f)} = s_f && \forall\, f \\
& \langle \mathbf{1}, \mathbf{Z}^T \mathbf{e}_l \rangle \leq R_l(\mathbf{Q}^{(n)}) && \forall\, l \in \mathcal{O}(n), \forall\, n \\
& \mathbf{R}(\mathbf{Q}^{(n)}) \in \mathcal{C}^{(n)}_{\mathrm{MAC}}(P^{(n)}_{\max}, \mathbf{H}^{\dagger(n)}) && \forall\, n \\
& \textstyle\sum_{l \in \mathcal{O}(n)} \mathrm{Tr}\{\mathbf{Q}_l\} \leq 1 && \forall\, n \\
& \mathbf{Q}_l \succeq 0 && \forall\, l \\
& \text{Variables : } \mathbf{S},\ \mathbf{Z},\ \mathbf{Q}.
\end{aligned}
$$

5.4 A Solution to the Reformulated Problem

Since CRPA-E is convex, we may apply standard convex programming tools to solve it. However, due to the complex structure in (5.3.3) and the inherent ordering issue associated with DPC in MIMO-BC (correspondingly, the successive decoding orders in the dual MIMO-MAC), methods that do not exploit the special structure of (5.3.3) will likely be inefficient. In this chapter, we propose a solution based on a dual decomposition approach, which exploits the coupling structure in (5.3.3) to reduce its computational complexity. We also show that, through further transformations, the ordering difficulty can also be satisfactorily handled. In what follows, we introduce the key components of our proposed solution.

5.4.1 Dual Decomposition

Associating dual variables u_l, $l = 1, \ldots, L$, with the coupling constraints $\langle \mathbf{1}, \mathbf{Z}^T \mathbf{e}_l \rangle \leq R_l(\mathbf{Q}^{(n)})$, we can write the dual function as

$$\Theta(\mathbf{u}) = \sup_{\mathbf{S},\mathbf{Z},\mathbf{Q}} \left\{ L(\mathbf{S}, \mathbf{Z}, \mathbf{Q}, \mathbf{u}) | (\mathbf{S}, \mathbf{Z}, \mathbf{Q}) \in \Psi \right\}, \tag{5.4.1}$$

where the Lagrangian $L(\mathbf{S}, \mathbf{Z}, \mathbf{Q}, \mathbf{u}) = \sum_f \ln(s_f) + \sum_l u_l \left(R_l(\mathbf{Q}) - \langle \mathbf{1}, \mathbf{Z}^T \mathbf{e}_l \rangle \right)$ and the set Ψ is defined as

$$\Psi \triangleq \left\{ (\mathbf{S}, \mathbf{Z}, \mathbf{Q}) \left| \begin{array}{ll} \mathbf{AZ} = \mathbf{S} & \\ \mathbf{Z} \geq \mathbf{0} & \\ \mathbf{S}\mathbf{e}_f =_{\mathrm{src}(f),\mathrm{dst}(f)} \mathbf{0} & \forall\, f \\ \langle \mathbf{1}, \mathbf{S}\mathbf{e}_f \rangle = 0 & \forall\, f \\ (\mathbf{S}\mathbf{e}_f)_{\mathrm{src}(f)} = s_f & \forall\, f \\ \sum_{l \in \mathcal{O}(n)} \mathrm{Tr}\{\mathbf{Q}_l\} \leq 1 & \forall\, n \\ \mathbf{Q}_l \succeq 0 & \forall\, l \\ \mathbf{R}(\mathbf{Q}^{(n)}) \in \mathcal{C}_{\mathrm{MAC}}(P_{\max}^{(n)}, \mathbf{H}^{\dagger(n)}) & \forall\, n \end{array} \right. \right\}.$$

Then, the dual problem of CRPA can be written as:

$$\begin{array}{rll} \mathbf{D}^{\mathrm{CRPA-E}}: & \mathsf{Minimize} & \Theta(\mathbf{u}) \\ & \mathsf{subject\ to} & \mathbf{u} \geq \mathbf{0}. \end{array} \tag{5.4.2}$$

It is easy to recognize that, for a given $\mathbf{u}$, the dual function in (5.4.1) can be re-arranged and separated into two terms:

$$\Theta(\mathbf{u}) = \Theta_{\mathrm{net}}(\mathbf{u}) + \Theta_{\mathrm{link}}(\mathbf{u}),$$

where Θ_{net} and Θ_{link} denote a network layer subproblem and a link layer subproblem, respectively. More specifically,

$$\begin{aligned}
\Theta_{\text{net}}(\mathbf{u}) \triangleq\ & \mathsf{Maximize}\ \textstyle\sum_f \ln(s_f) - \sum_l u_l \langle \mathbf{1}, \mathbf{Z}^T \mathbf{e}_l \rangle \\
\mathsf{subject\ to}\ & \mathbf{AZ} = \mathbf{S} \\
& \mathbf{Z} \geq \mathbf{0} \\
& \mathbf{Se}_f =_{\text{src}(f),\text{dst}(f)} \mathbf{0} \quad \forall\, f \\
& \langle \mathbf{1}, \mathbf{Se}_f \rangle = 0 \quad \forall\, f \\
& (\mathbf{Se}_f)_{\text{src}(f)} = s_f \quad \forall\, f \\
\mathsf{Variables:}\ & \mathbf{S},\ \mathbf{Z}.
\end{aligned}$$

$$\begin{aligned}
\Theta_{\text{link}}(\mathbf{u}) \triangleq\ & \mathsf{Maximize}\ \textstyle\sum_l u_l R_l(\mathbf{Q}) \\
\mathsf{subject\ to}\ & \textstyle\sum_{l \in \mathcal{O}(n)} \text{Tr}\{\mathbf{Q}_l\} \leq 1 \quad \forall\, n \\
& \mathbf{Q}_l \succeq 0 \quad \forall\, l \\
& R_l(\mathbf{Q}) \in \mathcal{C}_{\text{MAC}}(P_{\max}^{(n)}, \mathbf{H}^{\dagger(n)}), \\
& \qquad \forall\, l \in \mathcal{O}(n), n \in N \\
\mathsf{Variables:}\ & \mathbf{Q}.
\end{aligned}$$

Then, the dual problem of CRPA-E can be re-cast into a master-dual problem:

$$\begin{aligned}
\mathbf{MD}^{\text{CRPA-E}}: \quad & \mathsf{Minimize} \quad \Theta_{\text{net}}(\mathbf{u}) + \Theta_{\text{link}}(\mathbf{u}) \\
& \mathsf{subject\ to} \quad \mathbf{u} \geq \mathbf{0}.
\end{aligned} \tag{5.4.3}$$

Thus, solving CRPA-E boils down to solving the master-dual problem $\mathbf{MD}^{\text{CRPA-E}}$ and two subproblems $\Theta_{\text{net}}(\mathbf{u})$ and $\Theta_{\text{link}}(\mathbf{u})$. In $\Theta_{\text{net}}(\mathbf{u})$, since the objective function is concave and all constraints are affine, $\Theta_{\text{net}}(\mathbf{u})$ can be readily solved by standard convex programming tools. Thus, in the remainder of this section, we only focus on solving $\mathbf{MD}^{\text{CRPA-E}}$ and $\Theta_{\text{link}}(\mathbf{u})$.

5.4.2 Solving the Master-Dual Problem

The master dual problem in (5.4.3) is piece-wise linear with respect to $\mathbf{u}$. Thus, it is convex but non-differentiable. For this non-differentiable problem, we propose two algorithms to solve it: one centralized and one distributed.

Cutting-Plane Method (Centralized): The basic idea of the cutting-plane method is based on *successive outer linear approximation*. Letting z be the optimal objective value of the dual function, we can rewrite the dual problem (5.4.2) in the following equivalent form:

$$\begin{array}{rl} \mathsf{Minimize} & z \\ \mathsf{subject\ to} & z \geq \sum_f \ln(s_f) + \sum_l u_l \left(R_l(\mathbf{Q}) - \langle \mathbf{1}, \mathbf{Z}^T \mathbf{e}_l \rangle\right) \\ & \mathbf{u} \geq 0, \end{array} \tag{5.4.4}$$

where $(\mathbf{S}, \mathbf{Z}, \mathbf{Q}) \in \Psi$. Although problem (5.4.4) is a linear program, it has an infinite number of constraints that are not known explicitly. Hence, we consider an *approximating* problem of (5.4.4):

$$\begin{array}{rl} \mathsf{Minimize} & z \\ \mathsf{subject\ to} & z \geq \sum_f \ln(s_f^{(j)}) + \sum_l u_l \left(R_l(\mathbf{Q}^{(j)}) - \right. \\ & \qquad \left. \langle \mathbf{1}, \mathbf{Z}^{(j)T} \mathbf{e}_l \rangle\right), \ j = 1, \ldots, K-1 \\ & \mathbf{u} \geq 0, \end{array} \tag{5.4.5}$$

where the points $(\mathbf{S}^{(j)}, \mathbf{Z}^{(j)}, \mathbf{Q}^{(j)}) \in \Psi$, $j = 1, \ldots, k-1$, are given. We can see that this approximating linear program has a finite number of constraints and can be solved explicitly. Let $(z^{(k)}, \mathbf{u}^{(k)})$ be an optimal solution to the approximating problem (5.4.5). It is evident that if this solution is feasible to (5.4.4), then it is already optimal to problem (5.4.4) since (5.4.5) is a relaxation of (5.4.4). To check the feasibility, we

Algorithm 5.4.1 Centralized Algorithm for Master-Dual Problem

1. Find a point $(\mathbf{S}^{(0)}, \mathbf{Z}^{(0)}, \mathbf{Q}^{(0)}) \in \Psi$. Let $k = 1$.
2. Solve the master program in (5.4.5). Let $(z^{(k)}, \mathbf{u}^{(k)})$ be an optimal solution.
3. Solve the subproblem in (5.4.6). Let $(\mathbf{S}^{(k)}, \mathbf{Z}^{(k)}, \mathbf{Q}^{(k)})$ be an optimal point, and let $\Theta^*(\mathbf{u}^{(k)})$ be the corresponding optimal objective value.
4. If $z^{(k)} \geq \Theta(\mathbf{u}^{(k)})$, then stop with $\mathbf{u}^{(k)}$ as the optimal dual solution. Otherwise, add constraint (5.4.7) to (5.4.5), replace k by $k+1$, and go to step 2.

consider the following subproblem:

$$\begin{array}{ll} \mathsf{Maximize} & \sum_f \ln(s_f) + \sum_l u_l^{(k)} \left(R_l(\mathbf{Q}) - \langle \mathbf{1}, \mathbf{Z}^T \mathbf{e}_l \rangle \right) \\ \mathsf{subject\ to} & (\mathbf{S}, \mathbf{Z}, \mathbf{Q}) \in \Psi. \end{array} \tag{5.4.6}$$

Suppose that $(\mathbf{S}^{(k)}, \mathbf{Z}^{(k)}, \mathbf{Q}^{(k)})$ is an optimal solution to (5.4.6) and $\Theta^*(\mathbf{u}^{(k)})$ is the corresponding optimal objective value. If $z_k \geq \Theta^*(\mathbf{u}^{(k)})$, then $\mathbf{u}^{(k)}$ is an optimal solution to (5.4.4) since $(z^{(k)}, \mathbf{u}^{(k)})$ is feasible. Otherwise, for $\mathbf{u} = \mathbf{u}^{(k)}$, the inequality constraint in (5.4.4) is not satisfied for $(\mathbf{S}^{(k)}, \mathbf{Z}^{(k)}, \mathbf{Q}^{(k)})$. Thus, we can add the following constraint (called a cutting-plane) to the approximating problem (5.4.5):

$$z \geq \sum_f \ln\left(s_f^{(k)}\right) + \sum_l u_l \left(R_l(\mathbf{Q}^{(k)}) - \langle \mathbf{1}, \mathbf{Z}^{(k)T} \mathbf{e}_l \rangle \right). \tag{5.4.7}$$

Then, we solve the expanded approximating program again. Obviously, in the expanded approximating problem, $(z^{(k)}, \mathbf{u}^{(k)})$ violates (5.4.7) and will be cut off. The cutting plane algorithm is summarized in Algorithm 5.4.1.

Subgradient Method (Distributed): Since the subgradient method is widely used for non-differentiable problems, we only provide a sketch in here and refer readers to standard texts for details (e.g., [10]). For $\Theta(\mathbf{u})$, starting with an initial $\mathbf{u}^{(1)}$ and after evaluating subproblems $\Theta_{\text{net}}(\mathbf{u})$ and $\Theta_{\text{link}}(\mathbf{u})$ with $\mathbf{u}^{(k)}$ in the k-th iteration, we update the dual variables by $\mathbf{u}^{(k+1)} = \left[\mathbf{u}^{(k)} - \omega_k \mathbf{d}^{(k)}\right]_+$, where the operator $[\cdot]_+$ projects a vector on to the nonnegative orthant; ω_k denotes a positive step size; and $\mathbf{d}^{(k)}$ is a subgradient of the Lagrangian dual function at point $\mathbf{u}^{(k)}$. It can be

shown that the subgradient method converges if the step size ω_k satisfies $\omega_k \to 0$ as $k \to \infty$ and $\sum_{k=0}^{\infty} \omega_k = \infty$ [10]. A simple and useful step size selection strategy is the divergent harmonic series $\{\omega_k\} = \beta\frac{1}{k}$, where β is a constant. The subgradient for the dual problem can be computed as

$$\mathbf{d}^{(k)} = R_l(\mathbf{Q}^*(\mathbf{u})) - \langle \mathbf{1}, \mathbf{Z}^*(\mathbf{u})^T \mathbf{e}_l \rangle, \quad l = 1, 2, \ldots, L. \tag{5.4.8}$$

We point out that since the subgradient computation in (5.4.8) involves only local traffic information, the subgradient method can be implemented in a distributed fashion.

5.4.3 Solving the Link Layer Subproblem $\Theta_{\text{link}}(\mathbf{u})$

A closer look shows that $\Theta_{\text{link}}(\mathbf{u})$ can be decomposed on a node-by-node basis as follows:

$$\begin{aligned} \Theta_{\text{link}}(\mathbf{u}) &= \max_{\Omega} \sum_{l} u_l R_l(\mathbf{Q}) \\ &= \sum_{n=1}^{N} \left(\max_{\Omega^n} \sum_{l \in \mathcal{O}(n)} u_l R_l(\mathbf{Q}) \right) = \sum_{n=1}^{N} \Theta_{\text{link}}^{(n)}(\mathbf{u}(n)), \end{aligned} \tag{5.4.9}$$

where $\mathbf{u}(n) = \{u_l | l \in \mathcal{O}(n)\}$; Ω represents the feasible region of $\Theta_{\text{link}}(\mathbf{u})$; and $\Omega_+(n) \triangleq \{\mathbf{Q}_l | \sum_l \text{Tr}\{\mathbf{Q}_l\} \leq 1, \mathbf{Q}_l \succeq 0,\ l \in \mathcal{O}(n)\}$. With this decomposition, we can see that $\Theta_{\text{link}}^{(n)}(\mathbf{u}(n)) \triangleq \max_{\Omega_+(n)} \sum_{l \in \mathcal{O}(n)} u_l R_l(\mathbf{Q})$ is a *maximum weighted sum rate* (MWSR) problem of a dual MIMO-MAC with dual variables u_l as weights. Thus, we only need to solve N small-sized MWSR problems, which is much easier than solving $\Theta_{\text{link}}(\mathbf{u})$ directly. More importantly, this decomposition enables a distributed implementation in practice.

Now, we consider how to solve each MWSR problem. For a dual MIMO-MAC

with K links, recall that there exist $K!$ DPC encoding orders in the original MIMO-BC. Hence, in the dual MIMO-MAC, there also exist $K!$ successive decoding orders according to uplink-downlink duality [151]. Thus, even for moderate values of K, enumerating all $K!$ possible decoding orders to solve the MWSR problem is extremely cumbersome, and the complexity grows *exponentially* as K gets large.

Surprisingly, we find that it is *not* necessary to enumerate all $K!$ orders to solve the MWSR problem because we can transform the problem into an equivalent one in which the ordering can be uniquely determined. We state this result in Theorem 5.4.1.

Theorem 5.4.1. *The MWSR problem* $\max_{\Omega_+(n)} \sum_{i=1}^{K} u_i R_i(\mathbf{Q})$ *is equivalent to the following problem:*

$$\begin{aligned} \mathsf{Maximize}\ & \textstyle\sum_{i=1}^{K}(u_{\pi(i)} - u_{\pi(i-1)}) \\ & \times \log\left|\mathbf{I} + \textstyle\sum_{j=i}^{K} \rho_{\pi(j)}\mathbf{H}_{\pi(j)}^{\dagger}\mathbf{Q}_{\pi(j)}\mathbf{H}_{\pi(j)}\right| \\ \mathsf{subject\ to}\ & \mathbf{Q}_i \succeq 0,\ \forall i,\ \textstyle\sum_{i=1}^{K} \mathrm{Tr}(\mathbf{Q}_i) \leq 1, \end{aligned} \tag{5.4.10}$$

where $u_{\pi(0)} \triangleq 0$, $\pi(i), i = 1, \ldots, K$ *is a permutation on* $\{1, \ldots, K\}$ *satisfying* $u_{\pi(1)} \leq \ldots \leq u_{\pi(K)}$. *Further, let* $(\mathbf{Q}_{\pi(1)}^{*}, \ldots, \mathbf{Q}_{\pi(K)}^{*})$ *be the optimal solution of (5.4.10). Then, the optimal rates are given by*

$$R_{\pi(K)}^{*} = \log\left|\mathbf{I} + \rho_{\pi(K)}\mathbf{H}_{\pi(K)}^{\dagger}\mathbf{Q}_{\pi(K)}^{*}\mathbf{H}_{\pi(K)}\right| \tag{5.4.11}$$

and

$$\begin{aligned} R_{\pi(i)}^{*} = & \log\left|\mathbf{I} + \textstyle\sum_{j=i}^{K} \rho_{\pi(j)}\mathbf{H}_{\pi(j)}^{\dagger}\mathbf{Q}_{\pi(j)}^{*}\mathbf{H}_{\pi(j)}\right| \\ & - \log\left|\mathbf{I} + \textstyle\sum_{j=i+1}^{K} \rho_{\pi(j)}\mathbf{H}_{\pi(j)}^{\dagger}\mathbf{Q}_{\pi(j)}^{*}\mathbf{H}_{\pi(j)}\right|, \end{aligned} \tag{5.4.12}$$

for $i = 1, 2, \ldots, K-1$.

Proof. Since the MWSR objective function is monotonically increasing, the optimal solution must be achieved on the border of the capacity region. Denote $\pi(\cdot)$ the

optimal decoding order. The objective gradient can be computed as

$$\left[\begin{array}{cccc} u_{\pi(1)} & u_{\pi(2)} & \cdots & u_{\pi(K)} \end{array} \right]^T.$$

Note that the objective function is linear and the dual MIMO-MAC capacity region is convex. Further, since the MWSR problem is strictly feasible over a dual MIMO-MAC capacity region (since an all-zero solution trivially holds), we have that the Slater condition is satisfied (cf. [10]). As a result, the KKT conditions are necessary and sufficient for optimality. Note that the active rate vector constraints at the optimal corner point corresponding to $\pi(\cdot)$ satisfy

$$\sum_{j=i}^{K} R_{\pi(j)} \leq \log \left| \sum_{j=i}^{K} \mathbf{H}_{\pi(j)}^{\dagger} \mathbf{Q}_{\pi(j)}^{*} \mathbf{H}_{\pi(j)} \right|, \tag{5.4.13}$$

where $\mathbf{Q}_{\pi(j)}^{*}$, $i = 1, 2, \ldots, K$ represent the optimal input covariance matrices that achieve the maximum weighted sum rate. Thus, by the KKT conditions, we can derive that

$$\left[\begin{array}{c} u_{\pi(1)} \\ \vdots \\ u_{\pi(K-1)} \\ u_{\pi(K)} \end{array} \right] = v_1 \left[\begin{array}{c} 0 \\ \vdots \\ 0 \\ 1 \end{array} \right] + v_2 \left[\begin{array}{c} 0 \\ \vdots \\ 1 \\ 1 \end{array} \right] + \cdots + v_K \left[\begin{array}{c} 1 \\ \vdots \\ 1 \\ 1 \end{array} \right],$$

where $v_i \geq 0, \forall i$ are the associated dual variables. Solving for v_i, we have $v_K = u_{\pi(1)}$ and $v_{K-i} = u_{\pi(i+1)} - u_{\pi(i)}$ for $i = 1, 2, \ldots, K-1$. Since $v_i \geq 0 \ \forall i$, we have that

$$u_{\pi(1)} \leq u_{\pi(2)} \leq \ldots \leq u_{\pi(K)},$$

i.e., the weights must be in ascending order.

Further, since the constraints in (5.4.13) are tight at optimality, we have

$$\begin{aligned} R_{\pi(i)} &= \log \left| \mathbf{I} + \sum_{j=i}^{K} \mathbf{H}_{\pi(K)}^{\dagger} \mathbf{Q}_{\pi(K)}^{*} \mathbf{H}_{\pi(K)} \right| \\ &- \log \left| \mathbf{I} + \sum_{j=i+1}^{K} \mathbf{H}_{\pi(K)}^{\dagger} \mathbf{Q}_{\pi(K)}^{*} \mathbf{H}_{\pi(K)} \right|, \ i = 1, \ldots, K-1, \end{aligned}$$

and $R_{\pi(K)} = \log\left|\mathbf{I} + \mathbf{H}_{\pi(K)}^{\dagger}\mathbf{Q}_{\pi(K)}^{*}\mathbf{H}_{\pi(K)}\right|$. Summing up all $u_{\pi(i)}R_{\pi(i)}$ and after rearranging the terms, we have

$$\begin{aligned}\sum_{i=1}^{K} u_{\pi(i)}R_{\pi(i)} = \sum_{i=1}^{K}(u_{\pi(i)} - u_{\pi(i-1)})\times \\ \log\left|\mathbf{I} + \sum_{j=i}^{K}\mathbf{H}_{\pi(j)}^{\dagger}\mathbf{Q}_{\pi(j)}\mathbf{H}_{\pi(j)}\right|,\end{aligned} \tag{5.4.14}$$

which is the objective function in (5.4.10). Thus, we can see that the MWSR problem of the dual MIMO-MAC is equivalent to maximizing (5.4.10) with a sum power constraint. This completes the proof. □

Remark 5.4.1. Theorem 5.4.1 is important because: 1) Problem (5.4.10) eliminates the exponential complexity of the MWSR problem; 2) Problem (5.4.10) is convex, meaning that designing efficient algorithms is possible; and 3) Theorem 5.4.1 shows how to determine the optimal decoding order in the dual MIMO-MAC. Thus, the DPC encoding order in the original problem can also be determined.

Now, we turn our attention to solving problem (5.4.10). Although problem (5.4.10) is convex, standard convex optimization tools are inefficient to use due to the problem's complex matrix structure. In what follows, we propose an efficient method. Our method utilizes a powerful tool called Hessian conjugate that deflects the gradients appropriately to achieve an asymptotic superlinear convergence rate [10]. We then project the conjugate gradient to find an *improving feasible direction* in each iteration. The complete algorithm for solving problem (5.4.10) is shown in Algorithm 5.4.2.

There are two key components Algorithm 5.4.2: 1) the gradient computation in Step 2; and 2) the projection in Step 4.

Gradient computation: By utilizing matrix differential calculus [98], we can derive the gradient expression for problem (5.4.10). We state the result in the following proposition.

Algorithm 5.4.2 Solution to the MWSR problem of MIMO-MAC

1. Choose $\mathbf{Q}^{(0)} = [\mathbf{Q}_1^{(0)}, \mathbf{Q}_2^{(0)}, \ldots, \mathbf{Q}_K^{(0)}]^T$. Let $k = 0$.
2. Calculate the conjugate gradients $\mathbf{G}_i^{(k)}$, $i = 1, 2, \ldots, K$.
3. Choose an appropriate step size s_k. Let $\mathbf{Q}_i^{'(k)} = \mathbf{Q}_i^{(k)} + s_k \mathbf{G}_i^{(k)}$, for $i = 1, 2, \ldots, K$.
4. Compute the projection of $\mathbf{Q}^{'(k)}$ onto $\Omega_+(n)$, denoted as $\bar{\mathbf{Q}}^{(k)}$.
5. Choose an appropriate step size α_k. Let $\mathbf{Q}_l^{(k+1)} = \mathbf{Q}_l^{(k)} + \alpha_k(\bar{\mathbf{Q}}_i^{(k)} - \mathbf{Q}_i^{(k)})$, $i = 1, 2, \ldots, K$.
6. Let $k = k + 1$. If the maximum absolute value of the elements in $\mathbf{Q}_i^{(k)} - \mathbf{Q}_i^{(k-1)} < \epsilon$, for $i = 1, 2, \ldots, L$, then stop; else go to Step 2.

Proposition 5.4.2. *The gradient of user $\pi(i)$, denoted as $\bar{\mathbf{G}}_{\pi(j)}$, can be computed as*

$$\begin{aligned}\bar{\mathbf{G}}_{\pi(j)} = 2\rho_{\pi(j)}\mathbf{H}_{\pi(j)} \Bigg[\textstyle\sum_{i=1}^{j} \left(u_{\pi(i)} - u_{\pi(i-1)}\right) \times \\ \left(\mathbf{I} + \textstyle\sum_{k=i}^{K} \rho_{\pi(k)} \mathbf{H}_{\pi(k)}^{\dagger} \mathbf{Q}_{\pi(k)} \mathbf{H}_{\pi(k)}\right)^{-1} \Bigg] \mathbf{H}_{\pi(j)}^{\dagger}.\end{aligned} \tag{5.4.15}$$

Proof. For convenience, we denote the objective function of (5.4.10) as $F(\mathbf{Q})$. To prove Proposition 5.4.2, we need the following lemma [98].

Lemma 5.4.3. *For matrices $\mathbf{A} \in \mathbb{C}^{p\times p}$, $\mathbf{B} \in \mathbb{C}^{p\times m}$, $rank(\mathbf{B}) = p$, $\mathbf{X} \in \mathbb{C}^{m\times n}$, $\mathbf{C} \in \mathbb{C}^{n\times p}$, and $rank(\mathbf{C}) = p$, if $(\mathbf{A} + \mathbf{BXC})$ is non-singular, then we have $\frac{\partial}{\partial \mathbf{X}} \ln\det(\mathbf{A} + \mathbf{BXC}) = [\mathbf{C}(\mathbf{A} + \mathbf{BXC})^{-1}\mathbf{B}]^T$.*

Assume that the channel gain matrices $\mathbf{H}$ are of full row rank (if not, we can always delete the linearly dependent rows). Applying Lemma 5.4.3 by letting $\mathbf{A} = \mathbf{I} + \sum_{k=i,k\neq j}^{K} \mathbf{H}_{\pi(k)}^{\dagger} \mathbf{Q}_{\pi(k)} \mathbf{H}_{\pi(k)}$, $\mathbf{B} = \mathbf{H}_{\pi(j)}^{\dagger}$, $\mathbf{X} = \mathbf{Q}_{\pi(j)}$, and $\mathbf{C} = \mathbf{H}_{\pi(j)}$, we can compute the partial derivative of the i-th term in the summation of $F(\mathbf{Q})$ with respect to $\mathbf{Q}_{\pi(j)}$, $j \geq i$, as follows:

$$\begin{aligned}&\tfrac{\partial}{\partial \mathbf{Q}_{\pi(j)}} F(\mathbf{Q}) = \rho_{\pi(j)} \left(u_{\pi(i)} - u_{\pi(i-1)}\right) \times \\ &\left[\mathbf{H}_{\pi(j)} \left(\mathbf{I} + \textstyle\sum_{k=i}^{K} \rho_{\pi(k)} \mathbf{H}_{\pi(k)}^{\dagger} \mathbf{Q}_{\pi(k)} \mathbf{H}_{\pi(k)}\right)^{-1} \mathbf{H}_{\pi(j)}^{\dagger}\right]^T.\end{aligned}$$

Note that when computing the gradient of $F(\mathbf{Q})$ with respect to $\mathbf{Q}_{\pi(j)}$, only the first j terms in $F(\mathbf{Q})$ involve $\mathbf{Q}_{\pi(j)}$. Thus, by the definition $\nabla_z f(z) = 2(\partial f(z)/\partial z)^*$

[58], we have

$$\bar{\mathbf{G}}_{\pi(j)} = 2\rho_{\pi(j)}\mathbf{H}_{\pi(j)}\Bigg[\sum_{i=1}^{j}\left(u_{\pi(i)} - u_{\pi(i-1)}\right)\times \left(\mathbf{I} + \sum_{k=i}^{K}\rho_{\pi(k)}\mathbf{H}_{\pi(k)}^{\dagger}\mathbf{Q}_{\pi(k)}\mathbf{H}_{\pi(k)}\right)^{-1}\Bigg]\mathbf{H}_{\pi(j)}^{\dagger}.$$

□

The deflected gradient direction in the m-th iteration can be computed as $\mathbf{G}_{\pi(j)}^{(m)} = \bar{\mathbf{G}}_{\pi(i)}^{(m)} + \kappa_m \mathbf{G}_{\pi(i)}^{(m-1)}$, where κ_m is a deflection parameter. In this chapter, we adopt Fletcher and Reeves' choice of deflection [10], which can be computed as $\kappa_m = \|\bar{\mathbf{G}}_{\pi(j)}^{(m)}\|^2 / \|\bar{\mathbf{G}}_{\pi(j)}^{(m-1)}\|^2$. As mentioned earlier, the purpose of deflecting the gradient is to find the Hessian conjugate $\mathbf{G}_{\pi(j)}^{(m)}$. By doing so, we can curtail the "zigzagging" phenomenon encountered in the conventional gradient projection method, and achieve an asymptotic superlinear convergence rate without actually storing a large Hessian approximation matrix as in quasi-Newton methods.

Projection: Note from (5.4.15) that $\mathbf{G}_{\pi(j)}$ is Hermitian. Thus, we have that $\mathbf{Q}_{\pi(j)}^{\prime(k)} = \mathbf{Q}_{\pi(j)}^{(k)} + s_k \mathbf{G}_{\pi(j)}^{(k)}$ is also Hermitian (Step 3 in Algorithm 5.4.2). This means that we need to simultaneously project $|\mathcal{O}(n)|$ Hermitian matrices onto the set $\Omega_+(n)$. To this end, we construct a block diagonal matrix $\mathbf{D} = \mathrm{Diag}\{\mathbf{Q}_{\pi(1)} \ldots \mathbf{Q}_{\pi(K)}\} \in \mathbb{C}^{KM \times KM}$. It is easy to see that $\mathbf{Q}_{\pi(j)} \in \Omega_+(n)$ if only if $\mathrm{Tr}(\mathbf{D}) = \sum_{j=1}^{K} \mathrm{Tr}\left(\mathbf{Q}_{\pi(j)}\right) \leq 1$ and $\mathbf{D} \succeq 0$. Thus, given a block diagonal matrix $\mathbf{D}$, we wish to find a matrix $\tilde{\mathbf{D}}$ satisfying the above two constraints and being the closest to $\mathbf{D}$. This can be formulated as the following optimization problem:

$$\begin{aligned} \text{Minimize} \quad & \tfrac{1}{2}\|\tilde{\mathbf{D}} - \mathbf{D}\|_F^2 \\ \text{subject to} \quad & \mathrm{Tr}(\tilde{\mathbf{D}}) \leq 1,\ \tilde{\mathbf{D}} \succeq 0, \end{aligned} \tag{5.4.16}$$

where $\|\cdot\|_F$ denotes the Frobenius norm. Problem (5.4.16) belongs to a special class of problems in semidefinite programming called "matrix nearness problem" [17, 99],

which are not easy to solve in general. However, by exploiting its quadratic structure, we show that (5.4.16) can be solved in polynomial-time. The trick to solve (5.4.16) in polynomial-time is stated in the following theorem.

Theorem 5.4.4. *Problem (5.4.16) is equivalent to the following optimization problem*

$$\begin{array}{ll} \mathsf{Maximize} & -\frac{1}{2}\sum_{j=1}^{KM}\left(\max\left\{0,\lambda_j-\mu\right\}\right)^2-\mu \\ \mathsf{subject\ to} & \mathbf{\Pi}\succeq 0, \mu\geq 0, \end{array} \tag{5.4.17}$$

where λ_j, $j=1,\ldots,KM$ are the eigenvalues of $\mathbf{D}$. Let μ^* be an optimal solution of (5.4.17). Let the eigenvalue decomposition of $\mathbf{D}$ be $\mathbf{D}=\mathbf{U}\mathbf{\Lambda}\mathbf{U}^\dagger$. Then, the solution to problem (5.4.16), denoted as $\tilde{\mathbf{D}}$, can be recovered as $\tilde{\mathbf{D}}=\mathbf{U}(\mathbf{\Lambda}-\mu^*\mathbf{I})_+\mathbf{U}^\dagger$.

Proof. We now show that problem (5.4.16) is equivalent to problem (5.4.17). First, we observe that in (5.4.16), the objective function is convex in $\tilde{\mathbf{D}}$, the constraint $\tilde{\mathbf{D}}\succeq 0$ represents the convex cone of positive semidefinite matrices, and the constraint $\mathrm{Tr}(\tilde{\mathbf{D}})\leq 1$ is a linear constraint. Thus, problem (5.4.16) is a convex minimization problem with Slater condition being hold (since an all-zero solution is strictly feasible), meaning that its dual problem has a zero duality gap. By associating a dual Hermitian matrix $\mathbf{\Pi}$ to $\tilde{\mathbf{D}}\succeq 0$ and a dual variable μ to $\mathrm{Tr}(\tilde{\mathbf{D}})\leq 1$, we can write the Lagrangian as

$$g(\mathbf{\Pi},\mu)=\min_{\tilde{\mathbf{D}}}\{\frac{1}{2}\|\tilde{\mathbf{D}}-\mathbf{D}\|_F^2-\mathrm{Tr}(\mathbf{\Pi}^\dagger\tilde{\mathbf{D}})+\mu(\mathrm{Tr}(\tilde{\mathbf{D}})-1)\}.$$

Since $g(\mathbf{\Pi},\mu)$ is an unconstrained quadratic minimization problem, the minimizer can be achieved by simply setting its derivative with respect to $\tilde{\mathbf{D}}$ to zero and then solving for the critical point, i.e., setting $(\tilde{\mathbf{D}}-\mathbf{D})-\mathbf{\Pi}^\dagger+\mu\mathbf{I}=0$. Noting that $\mathbf{\Pi}^\dagger=\mathbf{\Pi}$, we have $\tilde{\mathbf{D}}=\mathbf{D}-\mu\mathbf{I}+\mathbf{\Pi}$. Substituting $\tilde{\mathbf{D}}$ back into the Lagrangian, we have

$$g(\mathbf{\Pi},\mu)=-\frac{1}{2}\|\mathbf{D}-\mu\mathbf{I}+\mathbf{\Pi}\|_F^2-\mu+\frac{1}{2}\|\mathbf{D}\|_F^2.$$

Therefore, the dual problem can be written as

$$\begin{array}{ll} \text{Maximize} & -\frac{1}{2}\|\mathbf{D}-\mu\mathbf{I}+\mathbf{\Pi}\|_F^2-\mu+\frac{1}{2}\|\mathbf{D}\|_F^2 \\ \text{subject to} & \mathbf{\Pi}\succeq 0, \mu\geq 0. \end{array} \tag{5.4.18}$$

Upon solving (5.4.18), the optimal solution to (5.4.16) can be computed as $\tilde{\mathbf{D}}^* = \mathbf{D}-\mu^*\mathbf{I}+\mathbf{\Pi}^*$, where μ^* and $\mathbf{\Pi}^*$ are the optimal dual solutions to the dual problem in (5.4.18). We now consider the term $\mathbf{D}-\mu\mathbf{I}+\mathbf{\Pi}$, which is the only term involving $\mathbf{\Pi}$ in the dual objective function. From Moreau Decomposition [61], we immediately have $\min_{\mathbf{\Pi}}\|\mathbf{D}-\mu\mathbf{I}+\mathbf{\Pi}\|_F = (\mathbf{D}-\mu\mathbf{I})_+$, where the operation $(\mathbf{A})_+$ means performing eigenvalue decomposition on matrix $\mathbf{A}$, keeping the eigenvector matrix unchanged, setting all non-positive eigenvalues to zero, and then multiplying back. Thus, the matrix variable $\mathbf{\Pi}$ in the dual problem can be removed and the dual problem can be rewritten as

$$\begin{array}{l} \text{Maximize } \psi(\mu)\triangleq -\frac{1}{2}\left\|(\mathbf{D}-\mu\mathbf{I})_+\right\|_F^2-\mu \\ \text{subject to } \mu\geq 0. \end{array} \tag{5.4.19}$$

Since $\mathbf{U}$ is unitary, we have $(\mathbf{D}-\mu\mathbf{I})_+ = \mathbf{U}(\mathbf{\Lambda}-\mu\mathbf{I})_+\mathbf{U}^\dagger$. It then follows that $\left\|(\mathbf{D}-\mu\mathbf{I})_+\right\|_F^2 = \left\|(\mathbf{\Lambda}-\mu\mathbf{I})_+\right\|_F^2$. We denote the eigenvalues in $\mathbf{\Lambda}$ by λ_i, $i=1,2,\ldots,KM$. Suppose that we sort them in non-increasing order such that $\mathbf{\Lambda}=\text{Diag}\{\lambda_1\ \lambda_2\ldots\ \lambda_{KM}\}$, where $\lambda_1\geq\ldots\geq\lambda_{KM}$. It then follows that

$$\left\|(\mathbf{\Lambda}-\mu\mathbf{I})_+\right\|_F^2=\sum_{j=1}^{KM}\left(\max\{0,\lambda_j-\mu\}\right)^2.$$

So, we can rewrite $\psi(\mu)$ as

$$\psi(\mu)=-\frac{1}{2}\sum_{j=1}^{KM}\left(\max\{0,\lambda_j-\mu\}\right)^2-\mu, \tag{5.4.20}$$

which is the objective function stated in (5.4.17). This completes the proof. □

For problem (5.4.17), we develop a polynomial-time algorithm, which is summarized in Algorithm 5.4.3.

Algorithm 5.4.3 Performing Projection

1. Construct a block diagonal matrix $\mathbf{D}$. Perform the eigenvalue decomposition $\mathbf{D} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^{\dagger}$, sort the eigenvalues in non-increasing order.
2. Introduce $\lambda_0 = \infty$ and $\lambda_{KM+1} = -\infty$. Let $\hat{I} = 0$. Let the endpoint objective value $\psi_{\hat{I}}(\lambda_0) = 0$, $\phi^* = \psi_{\hat{I}}(\lambda_0)$, and $\mu^* = \lambda_0$.
3. If $\hat{I} > KM$, go to Step 6; else let $\mu^*_{\hat{I}} = (\sum_{j=1}^{\hat{I}} \lambda_j - 1)/\hat{I}$.
4. If $\mu^*_{\hat{I}} \in [\lambda_{\hat{I}+1}, \lambda_{\hat{I}}] \cap \mathbb{R}_+$, then let $\mu^* = \mu^*_{\hat{I}}$ and go to Step 6.
5. Compute $\psi_{\hat{I}}(\lambda_{\hat{I}+1})$. If $\psi_{\hat{I}}(\lambda_{\hat{I}+1}) < \phi^*$, then go to Step 6; else let $\mu^* = \lambda_{\hat{I}+1}$, $\phi^* = \psi_{\hat{I}}(\lambda_{\hat{I}+1})$, $\hat{I} \leftarrow \hat{I} + 1$ and go to Step 3.
6. Compute $\tilde{\mathbf{D}}$ as $\tilde{\mathbf{D}} = \mathbf{U}\left(\mathbf{\Lambda} - \mu^*\mathbf{I}\right)_+ \mathbf{U}^{\dagger}$.

We now show that Algorithm 5.4.3 is a polynomial-time algorithm. In essence, Steps 2 to 5 in Algorithm 5.4.3 is a searching process specifically tailored to the special structure that $\psi(\mu)$ is continuous and piece-wise concave in μ. For such a structure, we can search the optimal μ^* piece-by-piece as follows. Let $\hat{I}$ index the pieces of $\psi(\mu)$, $\hat{I} = 0, 1, \ldots, KM$. Initially we set $\hat{I} = 0$ and increase $\hat{I}$ subsequently. Also, we introduce $\lambda_0 = \infty$ and $\lambda_{KM+1} = -\infty$. We let the endpoint objective value $\psi_{\hat{I}}(\lambda_0) = 0$, $\phi^* = \psi_{\hat{I}}(\lambda_0)$, and $\mu^* = \lambda_0$. If $\hat{I} > KM$, the search stops. For a particular index $\hat{I}$, by setting $\frac{\partial}{\partial \mu}\psi_{\hat{I}}(\nu) \triangleq \frac{\partial}{\partial \mu}\left(-\frac{1}{2}\sum_{i=1}^{\hat{I}}(\lambda_i - \mu)^2 - \mu\right) = 0$, we have

$$\mu^*_{\hat{I}} = \frac{\sum_{i=1}^{\hat{I}} \lambda_i - 1}{\hat{I}}. \tag{5.4.21}$$

In searching μ^* using (5.4.21), two cases need to be considered:

1) If $\mu^*_{\hat{I}} \in \left[\lambda_{\hat{I}+1}, \lambda_{\hat{I}}\right] \cap \mathbb{R}_+$, where $\mathbb{R}_+$ denotes the set of non-negative real numbers, then $\mu^*_{\hat{I}}$ is the optimal solution because $\psi(\mu)$ is concave in μ. Thus, the point having zero-valued first derivative, if it exists, must be the unique global maximum solution. Hence, we can let $\mu^* = \mu^*_{\hat{I}}$ and the search is done.

2) If $\mu^*_{\hat{I}} \notin \left[\lambda_{\hat{I}+1}, \lambda_{\hat{I}}\right] \cap \mathbb{R}_+$, we must have that the local maximum in the interval $\left[\lambda_{\hat{I}+1}, \lambda_{\hat{I}}\right] \cap \mathbb{R}_+$ is achieved at one of the two endpoints. Note that we do not need to compute the objective value $\psi_{\hat{I}}(\lambda_{\hat{I}})$. This is because $\psi_{\hat{I}}(\lambda_{\hat{I}}) = \psi_{\hat{I}-1}(\lambda_{\hat{I}})$, which is

due to the continuity of the objective function $\psi(\mu)$. Thus, we only need to compute the other endpoint's objective value $\psi_{\hat{I}}\left(\lambda_{\hat{I}+1}\right)$. If $\psi_{\hat{I}}\left(\lambda_{\hat{I}+1}\right) < \psi_{\hat{I}}\left(\lambda_{\hat{I}}\right) = \phi^*$, then we know μ^* is the optimal solution; else let $\mu^* = \lambda_{\hat{I}+1}$, $\phi^* = \psi_{\hat{I}}\left(\lambda_{\hat{I}+1}\right)$, $\hat{I} \leftarrow \hat{I} + 1$ and continue.

Since there are $KM+1$ intervals in total, the search process takes at most $KM+1$ steps to find the optimal solution μ^*. Hence, this search is of polynomial-time complexity $O(KM)$.

5.5 Numerical Results

In this section, we present simulation results to provide insights on performance optimization in MIMO ad hoc networks with DPC. The network simulation setting is as follows. There are 15 MIMO nodes uniformly distributed in a square region, as shown in Fig. 5.4. Each node in the network is equipped with two antennas. The maximum transmit power for each node is 20 dBm. The total bandwidth allocated to the MIMO-BC subsystem at each node is 20 MHz. The path loss exponent is 4. There are three flows transmitting across the network: Session 1 from N14 to N1, Session 2 from N6 to N10, and Session 3 from N5 to N4, respectively.

For this 15-node example, after executing our proposed solution procedure, we find the optimal objective value is 8.28 (log(bps/Hz)). The optimal flow rates for the three sessions are 310.6Mbps, 321.2Mbps, and 317.8Mbps, respectively. We can see that the optimal session rates are very close to each other, which means that the session rates are fair. We point out that, when solving the dual problem with the cutting-plane and subgradient methods, we start with equal power allocation at each node and the initial objective is -4.60 (log(bps/Hz)). This shows that a MIMO network

with DPC, if not properly designed, may have a very poor network performance.

The solutions for multi-hop/multi-path routing and flow rates of Sessions 1, 2, and 3 are shown in Figs. 5.5, 5.6, and 5.7, respectively. We find that the optimized routes/paths under DPC tend to be "spread out." This is because DPC can fully exploit the capacity region of MIMO broadcast channels. As a result, all sessions are able to make use of every possible link in the network to transmit their data.

The optimal power allocation solution is shown in Table 5.1. We observe from Table 5.1 that not every node allocates its full power to transmit data. For example, N7 allocates 93.56 mW ($\mathrm{Tr}(\mathbf{\Gamma}_{(7,5)}) + \mathrm{Tr}(\mathbf{\Gamma}_{(7,10)}) = 93.56$ mW) to transmit data for Sessions 1 and 2. This is due to the existence of bottleneck nodes in the network. That is, even if the non-bottleneck nodes increase their transmit power, the session rates will not be further increased because the end-to-end session rates are bounded by the capacity of the links that pass through the bottleneck node. In this example, the bottleneck node turns out to be N5. It can be verified that $\mathrm{Tr}(\mathbf{\Gamma}_{(5,3)}) + \mathrm{Tr}(\mathbf{\Gamma}_{(5,6)}) + \mathrm{Tr}(\mathbf{\Gamma}_{(5,7)}) = 100$ mW, which means that the power at N5 has been fully utilized and cannot be further increased.

The convergence process for the cutting-plane method is illustrated in Fig. 5.8. The convergence process for the subgradient method is shown in Fig. 5.9. The step size selection for the subgradient method in this example is $\omega_k = 0.1/k$. From the convergence curves, we can see that the cutting-plane method is more efficient. It converges with 127 cuts only. As expected, the duality gap is zero because of the convexity of the reformulated problem. The subgradient method also achieves the same optimal solution and objective value upon convergence. However, the subgradient method takes approximately 1900 iterations to converge, much slower than the

Table 5.1: Power allocation in the 15-node ad hoc network (mW).

$\mathbf{Q}_{(1,4)}$	25	$0.00-0.00i$	$\mathbf{Q}_{(4,1)}$	21.74	$0.00+0.00i$	$\mathbf{Q}_{(1,12)}$	25	$0.00-0.00i$
	$0.00+0.00i$	25		$0.00-0.00i$	21.76		$0.00+0.00i$	25
$\mathbf{Q}_{(12,1)}$	7.82	$1.90-0.13i$	$\mathbf{Q}_{(2,3)}$	10.09	$0.00-0.00i$	$\mathbf{Q}_{(3,2)}$	14.61	$1.17-0.87i$
	$1.90+0.13i$	8.86		$0.00+0.00i$	10.09		$1.17+0.87i$	18.69
$\mathbf{Q}_{(2,8)}$	11.16	$-0.43-0.68i$	$\mathbf{Q}_{(8,2)}$	32.74	$-20.44+6.78i$	$\mathbf{Q}_{(2,12)}$	12.25	$0.42-0.04i$
	$-0.43+0.68i$	10.73		$-20.44+6.78i$	14.31		$0.42+0.04i$	11.07
$\mathbf{Q}_{(12,2)}$	9.16	$1.84+0.03i$	$\mathbf{Q}_{(2,13)}$	17.23	$0.02+0.00i$	$\mathbf{Q}_{(13,2)}$	10.21	$1.43-0.15i$
	$1.84-0.03i$	4.34		$0.02-0.00i$	17.19		$1.43+0.15i$	8.38
$\mathbf{Q}_{(3,5)}$	7.36	$0.59-0.22i$	$\mathbf{Q}_{(5,3)}$	3.43	$-1.81+1.81i$	$\mathbf{Q}_{(3,6)}$	6.57	$0.00-0.00i$
	$0.59+0.22i$	7.27		$-1.81-1.81i$	3.86		$0.00+0.00i$	6.57
$\mathbf{Q}_{(6,3)}$	10.90	$0.63-0.29i$	$\mathbf{Q}_{(3,11)}$	6.98	$-1.28+0.46i$	$\mathbf{Q}_{(11,3)}$	5.97	$0.00-0.00i$
	$0.63+0.29i$	10.88		$-1.28-0.46i$	11.18		$0.00+0.00i$	5.97
$\mathbf{Q}_{(3,12)}$	13.51	$0.43+1.98i$	$\mathbf{Q}_{(12,3)}$	6.88	$0.37-1.67i$	$\mathbf{Q}_{(4,12)}$	14.12	$0.00-0.00i$
	$0.43-1.98i$	7.21		$0.37+1.67i$	4.32		$0.00+0.00i$	14.12
$\mathbf{Q}_{(12,4)}$	17.68	$-5.27-6.00i$	$\mathbf{Q}_{(4,13)}$	14.12	$0.00-0.00i$	$\mathbf{Q}_{(13,4)}$	25.16	$0.96+0.58i$
	$-5.27+6.00i$	13.38		$0.00+0.00i$	14.12		$0.96-0.58i$	23.68
$\mathbf{Q}_{(5,6)}$	4.36	$10.95-3.57i$	$\mathbf{Q}_{(6,5)}$	14.69	$2.10+1.71i$	$\mathbf{Q}_{(5,7)}$	25.69	$2.85+22.52i$
	$10.95+3.57i$	41.67		$2.10-1.71i$	15.85		$2.85-22.52i$	20.98
$\mathbf{Q}_{(7,5)}$	14.75	$-6.87-4.32i$	$\mathbf{Q}_{(6,9)}$	11.08	$-0.34-0.55i$	$\mathbf{Q}_{(9,6)}$	16.67	$0.00-0.00i$
	$-6.87+4.32i$	9.02		$-0.34+0.55i$	9.50		$0.00+0.00i$	16.67
$\mathbf{Q}_{(6,11)}$	15.33	$2.58-0.39i$	$\mathbf{Q}_{(11,6)}$	5.85	$0.00-0.01i$	$\mathbf{Q}_{(7,10)}$	28.65	$2.46-2.56i$
	$2.58+0.39i$	11.76		$0.00+0.00i$	5.85		$2.46+2.56i$	41.14
$\mathbf{Q}_{(10,7)}$	25.00	$0.00-0.00i$	$\mathbf{Q}_{(7,14)}$	3.16	$0.00-0.00i$	$\mathbf{Q}_{(14,7)}$	50.05	$0.01+0.13i$
	$0.00+0.00i$	25.00		$0.00+0.00i$	3.16		$0.01-0.13i$	49.95
$\mathbf{Q}_{(8,10)}$	33.54	$10.92-19.91i$	$\mathbf{Q}_{(10,8)}$	25.00	$0.00-0.00i$	$\mathbf{Q}_{(8,13)}$	0.37	$0.14-0.77i$
	$10.92+19.91i$	16.41		$0.00+0.00i$	25.00		$0.14+0.77i$	2.63
$\mathbf{Q}_{(13,8)}$	7.86	$-0.46+0.40i$	$\mathbf{Q}_{(9,11)}$	16.67	$0.00+0.00i$	$\mathbf{Q}_{(11,9)}$	5.85	$0.00-0.00i$
	$-0.46+0.40i$	9.26		$0.00-0.00i$	16.77		$0.00+0.00i$	5.85
$\mathbf{Q}_{(9,15)}$	16.67	$0.00-0.00i$	$\mathbf{Q}_{(15,9)}$	25.00	$0.00+0.00i$	$\mathbf{Q}_{(11,12)}$	26.18	$-0.00-0.01i$
	$0.00+0.00i$	16.67		$0.00-0.01i$	25.00		$-0.00+0.01i$	26.20
$\mathbf{Q}_{(12,11)}$	3.45	$0.00+0.00i$	$\mathbf{Q}_{(11,15)}$	5.85	$0.00+0.00i$	$\mathbf{Q}_{(15,11)}$	25.00	$0.00-0.00i$
	$0.00-0.00i$	3.45		$0.00-0.00i$	5.85		$0.00+0.00i$	25.00
$\mathbf{Q}_{(12,13)}$	9.40	$4.73-4.75i$	$\mathbf{Q}_{(13,12)}$	7.61	$0.03+0.02i$			
	$4.73+4.75i$	11.10		$0.03-0.02i$	7.64			

cutting-plane method. This is in part due to the step size selection in the subgradient method, which cannot be too conservative or aggressive. Another reason of the slow convergence speed of the subgradient method is due to the cumbersomeness in recovering the primal feasible solution. In this example, it takes 1700 iterations for the dual upper bound to get close to the optimal value. However, the optimal primal feasible solution cannot be identified until 1900 iterations.

We now study how much performance gain DPC can achieve compared with transmission schemes based on interference avoidance. Since interference avoidance schemes such as frequency, time, code, or space divisions share a similar mathematical

Table 5.2: Power allocation, encoding order, or bandwidth allocation at N5.

		DPC		FDM	
Power allocation (mW)	$\mathbf{\Gamma}_{(5,3)}$	3.43	$-1.81+1.81i$	16.93	$-0.00+0.01i$
		$-1.81-1.81i$	3.86	$-0.00-0.01i$	16.94
	$\mathbf{\Gamma}_{(5,6)}$	4.36	$10.95-3.57i$	13.87	$-0.00+0.02i$
		$10.95+3.57i$	41.67	$-0.00-0.02i$	13.98
	$\mathbf{\Gamma}_{(5,7)}$	25.69	$2.85+22.52i$	19.14	$-0.00+0.02i$
		$2.85-22.52i$	20.98	$-0.00-0.02i$	19.14
Encoding order	$l_{(5,3)}$	3rd		—-	
	$l_{(5,6)}$	2nd		—-	
	$l_{(5,7)}$	1st		—-	
Bandwidth allocation ($\times$ 20MHz)	$W_{(5,3)}$	—-		0.34	
	$W_{(5,6)}$	—-		0.28	
	$W_{(5,7)}$	—-		0.38	
Capacity (bps/Hz)	$C_{(5,3)}$	11.92		6.37	
	$C_{(5,6)}$	6.04		5.88	
	$C_{(5,7)}$	5.30		7.38	

structure, we choose frequency division multiplexing (FDM) as an example in this chapter. For the same network in Fig. 5.4, we optimize the same objective (proportional fairness) under FDM scheme. Unlike DPC, each node under FDM will divide its allocated band into subbands for each of its outgoing links to avoid co-channel interference. For example, in Fig. 5.4, N6 will divide its bandwidth into four subbands for its outgoing links to N5, N3, N11, and N9, respectively. Under FDM, the problem is to find the optimal routing, subband bandwidth allocation, and power allocation to maximize the network utility function. It turns out that this problem under FDM is convex. Thus, the basic dual decomposition framework and all the solution techniques proposed in this chapter are applicable to FDM as well. The gradient expression under FDM can be computed as in (5.5.1) and (5.5.2):

$$\mathbf{G}_{\mathbf{Q}_l} = \frac{2W_l u_l \rho_l}{\ln 2} \mathbf{H}_l^\dagger \left(\mathbf{I} + \rho_l \mathbf{H}_l \mathbf{Q}_l \mathbf{H}_l^\dagger \right)^{-1} \mathbf{H}_l, \tag{5.5.1}$$

$$G_{W_l} = u_l \log_2 \det(\mathbf{I} + \rho_l \mathbf{H}_l \mathbf{Q}_l \mathbf{H}_l^\dagger), \tag{5.5.2}$$

where W_l represents the fraction of a bandwidth allocated to link l's transmission.

As a comparison, for the same network in Fig. 5.4, we find that the optimal objective value under FDM is 6.56 (in log(bps/Hz)). The rates for Sessions 1, 2, and 3 are 147 Mbps, 266.6 Mbps, and 221.6 Mbps, respectively. Thus, DPC provides a 26.2% fairness gain over FDM. Also, all session rates under DPC (310.6Mbps, 321.2Mbps, and 317.8Mbps) are much higher than those under FDM. A closer observation shows that this significant improvement is due to the enlarged capacity region at the bottleneck node N5. Table 5.2 shows the optimal power/bandwidth allocations and link capacity for N5 under DPC and FDM, respectively. Clearly, the capacity region at N5 under DPC is much larger than that under FDM. An interesting observation is that N5 is able to provide a fairly high link capacity for $l_{(5,3)}$ without spending too much power on this link (only 7.29 mW). This is because by encoding $l_{(5,3)}$ last, the transmission on $l_{(5,3)}$ becomes interference-free. In contrast, FDM tends to have an approximately equal power and bandwidth allocation for three outgoing links to insure fairness at N5.

5.6 Chapter Summary

In this chapter, we studied how to incorporate DPC into multi-hop MIMO ad hoc networks to improve network performance. Under DPC, the formulation of multi-hop MIMO ad hoc network optimization has a complex non-convex structure and an exponentially large search space due to the enumeration of encoding orders, a unique problem for DPC. We proposed to reformulate the original problem into an equivalent convex program. For the reformulated problem, which still has an exponentially large search space, we showed that enumerating all possible DPC encoding orders can be avoided by using dual decomposition and further simplifications. This finding allowed

us to develop an efficient algorithm to solve the reformulated problem. We presented extensive numerical study to show that, by optimally managing radio resources, DPC can provide a significant network performance gain compared to conventional transmission schemes based on interference avoidance. Our results in this paper offer a theoretical understanding on how to fully exploit DPC's potential in MIMO ad hoc networks, and also a set of algorithmic tools that can serve as a basis for DPC's practical implementation in networking environments.

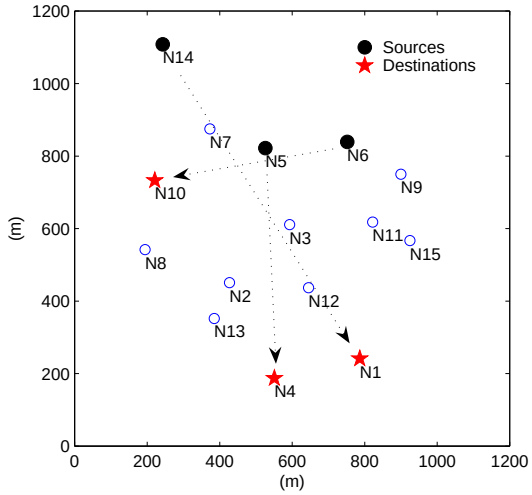


Figure 5.4: Network topology of a 15-node MIMO-based ad hoc network with 3 sessions.

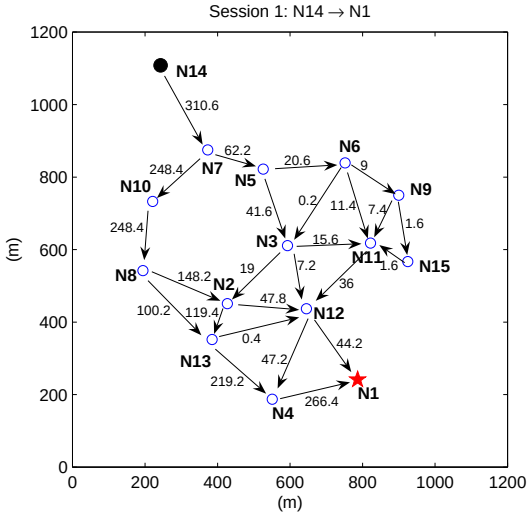


Figure 5.5: Routing directions and flow rates of session 1 (in Mbps).

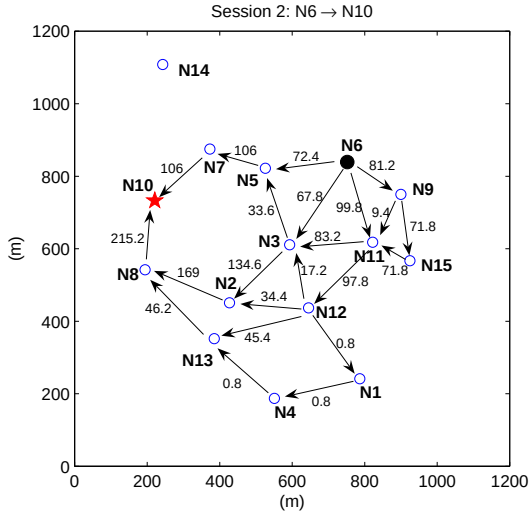


Figure 5.6: Routing directions and flow rates of session 2 (in Mbps).

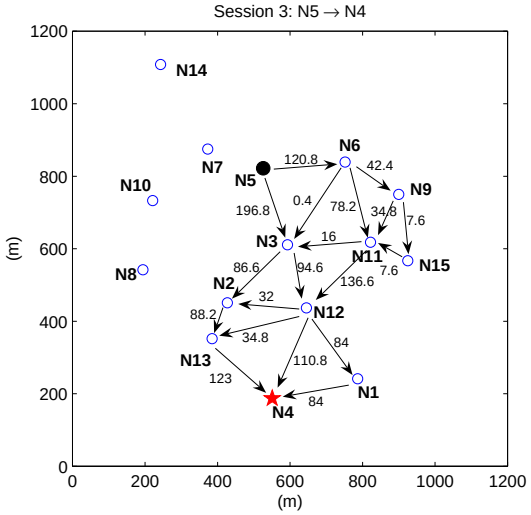


Figure 5.7: Routing directions and flow rates of session 3 (in Mbps).

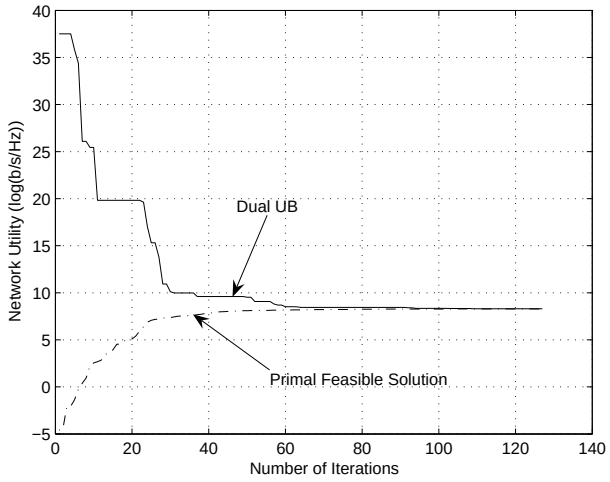


Figure 5.8: Convergence behavior of the centralized algorithm.

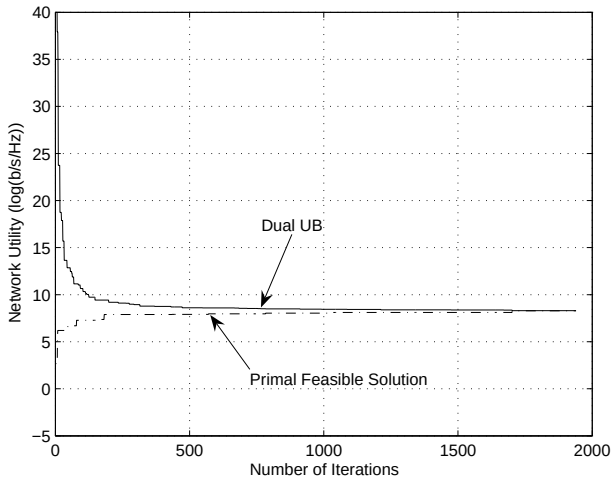


Figure 5.9: Convergence behavior of the distributed algorithm.

Chapter 6

A Tractable and Accurate Cross-Layer Model for Multi-hop MIMO Networks

6.1 Introduction

Although there have been extensive studies on MIMO at the physical layer for point-to-point and cellular communications (see, e.g., [13] for an overview), fundamental understanding and results on MIMO in multi-hop networks remain limited, particularly from a cross-layer perspective. This stagnation is mainly due to the lack of an accurate and more importantly, tractable model that is amenable for analysis by networking researchers. Traditional signal processing and channel models for MIMO in communications research are clogged with complex matrix representations and operations, rendering enormous challenges for multi-hop network optimizations. Due to these challenges, most efforts on multi-hop MIMO networks to date [12, 24, 53, 66, 77, 115, 116, 118, 140] fall into the following two approaches.

The first approach is to formulate the problems by faithfully incorporating the MIMO channel and signal models without any loss of accuracy. However, the problem

formulations under this approach soon become *intractable* due to the heavy burden from the underlying models. For example, Kim *et al.* studied a maxmin optimization problem in [77] for multi-hop MIMO backhaul networks where they formulated a nonlinear optimization problem to maximize the fair throughput of the access points in the network under the routing, MAC, and physical layer constraints. The physical layer in [77] is based on minimum mean square error (MMSE) beamforming. In [24], Chu and Wang also studied cross-layer algorithms for MIMO ad hoc networks where MMSE sequential interference cancellation technique (MMSE-SIC) was employed at the physical layer to maximize signal to interference and noise ratio (SINR). Due to the complex MMSE mechanics, the cross-layer optimization problems in [77] and [24] are intractable and the authors had to resort to heuristic algorithms.

The second approach is to simplify MIMO physical layer behavior so that tractable analysis can be developed for networking research. Although such approach is attractive, the problem with existing models under this approach suffer from "over simplification." That is, existing simple models ignore some important characteristics of MIMO and thus lead to results far from MIMO's achievable performance. In [12,53], a simplified MIMO cross-layer model was employed to study different throughput optimization problems. By using this model, the network throughput performance can be characterized simply by counting the number of degrees of freedom (DoF) in the network. However, this model does not consider transmit power constraint and power allocation at each node in the network. Also, although some ideas of zero-forcing beamforming (ZFBF) were employed to handle interference, the proposed interference cancellation scheme at the link layer was not designed efficiently, resulting in a small DoF region and inferior throughput performance. Also, in [66,115,116,118,140],

various studies on MAC designs and routing schemes are given based on very simple MIMO models that do not fully exploit MIMO physical capabilities.

The goal of this paper is to achieve the best of both approaches while avoiding their pitfalls. We want to construct a model for MIMO that is both *tractable* and *accurate* for cross-layer optimization. Our main contributions are as follows.

- At the physical layer, we devise a simple model for computing MIMO channel capacity. This model captures the essence of both spatial multiplexing and transmit power constraint. More importantly, this model does not require complex matrices computation and complicated water-filling process (which does not admit a close-form solution). We show that the gap between our proposed model and the exact capacity model is negligible.

- At the link layer, we construct a model that takes into account the interference nulling/supression by exploiting ZFBF. More specifically, we propose a space-time scheduling scheme called OBIC (abbreviation of order-based interference cancelation). The proposed OBIC employs simple algebraic computation on matrix dimensions to model ZFBF in a multi-hop network. Moreover, by carefully arranging the cancellation order among the nodes, OBIC does not waste unnecessary DoF resources on interference mitigation, thus offering superior throughput performance than those in [12, 53].

- As an application, we use the proposed new models to study a cross-layer utility maximization problem for multi-hop MIMO networks. We show that the resulting optimization problem no longer involve complex matrix variables and operations. Further, the formulated problem shares a lot of similarities with

those cross-layer optimization problems under single-antenna ad hoc networks, which have been actively studied in recent years. This suggests that new solutions to multi-hop MIMO networks may be developed by drawing upon the experiences gained for single-antenna ad hoc networks.

The remainder of this paper is organized as follows. Section 6.2 presents a new channel capacity model for MIMO at the physical layer. Section 6.3 presents a new link layer model called OBIC. In Section 6.4, as an application of our new models, we study a cross-layer optimization problem in a multi-hop MIMO network. Section 6.5 concludes this paper.

6.2 A Model for Physical Layer Capacity Computation

From networking research perspective, the most important aspect of physical layer modeling for MIMO is its channel capacity computation. In Section 6.2.1, we first give background on MIMO channel capacity computation and analyze why it is difficult to work with for networking research. Then, in Section 6.2.2, we propose a new model for MIMO channel capacity that is both simple and accurate.

6.2.1 Why Existing Physical Model for MIMO is Difficult to Use?

The channel of a MIMO link l is characterized by a matrix $\mathbf{H}_l$, as shown in Fig. 6.1. Communication over such a MIMO channel with n_t transmit antennas and n_r receive antennas can be described by

$$\mathbf{y}_l = \sqrt{\rho_l \alpha_l}\mathbf{H}_l\mathbf{x}_l + \mathbf{n}_l, \tag{6.2.1}$$

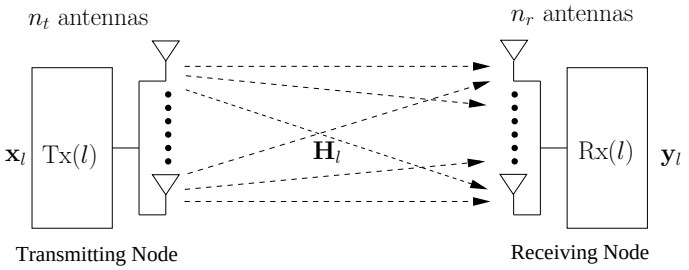


Figure 6.1: A MIMO channel.

where $\mathbf{x}_l$, $\mathbf{y}_l$ and $\mathbf{n}_l$ denote the vectors of transmitted signal, received signal, and white Gaussian noise with unit variance, respectively. In (6.2.1), ρ_l represents the received SNR of the channel, $\alpha_l \in [0, 1]$ represents the fraction of the transmit power that is assigned to link l (in the case when the source of link l also transmits on other links). As we shall see later in Section 6.4, α_l is useful to model the power allocation at each node if multi-hop multi-path routing is employed in the network. For the single link case in Fig. 6.1, we have $\alpha_l = 1$.

The channel gain matrix $\mathbf{H}_l$ is typically assumed to be a complex random matrix with each of its entries being i.i.d. Gaussian distributed [148] with zero mean and unit variance. From basic linear algebra, we know that by singular value decomposition (SVD), the channel model in (6.2.1) can be written as $\mathbf{y}_l = \sqrt{\rho_l \alpha_l}\mathbf{U}_l\mathbf{\Lambda}_l\mathbf{V}_l^\dagger\mathbf{x}_l + \mathbf{n}_l$, where $\mathbf{U}_l$ and $\mathbf{V}_l$ are unitary matrices, $\mathbf{\Lambda}_l$ is a diagonal matrix with the singular values of $\mathbf{H}_l$ on its main diagonal. By letting $\tilde{\mathbf{x}}_l = \mathbf{V}_l^\dagger\mathbf{x}_l$, $\tilde{\mathbf{y}}_l = \mathbf{U}_l^\dagger\mathbf{y}_l$, and $\tilde{\mathbf{n}}_l = \mathbf{U}_l^\dagger\mathbf{n}_l$, the channel model can be re-written as

$$\tilde{\mathbf{y}}_l = \sqrt{\rho_l \alpha_l}\mathbf{\Lambda}_l\tilde{\mathbf{x}}_l + \tilde{\mathbf{n}}_l, \tag{6.2.2}$$

which is equivalent to a set of parallel channels shown in Fig. 6.2. The number of non-zero singular values (i.e., non-zero diagonal entries in $\mathbf{\Lambda}_l$) is $d_l \leq \min\{n_t, n_r\}$, i.e., the rank of $\mathbf{H}_l$. The rank of $\mathbf{H}_l$ is also called the *degrees of freedom* (DoF), which

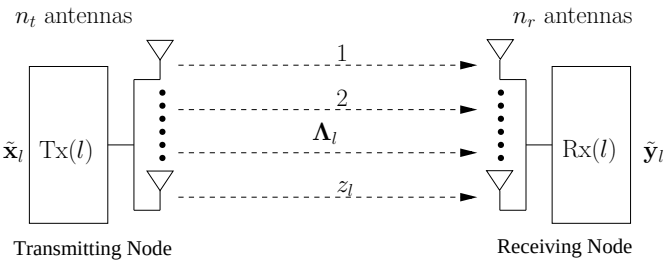


Figure 6.2: The equivalent parallel scalar channels after transformation.

measures the number of independent signaling dimensions that are available in the channel.

The capacity for the set of parallel channels in (6.2.2) can be found by the water-filling power allocation algorithm [146]:

$$\begin{aligned} C_l^{(\mathrm{wf})} &= \max_{\mathbf{Q}_l} W \log_2 \det(\mathbf{I} + \rho_l \alpha_l \mathbf{H}_l \mathbf{Q}_l \mathbf{H}_l^\dagger) \\ &= \sum_{i=1}^{d_l} W \left(\log_2(\rho_l \alpha_l \mu \lambda_i)\right)_+ , \end{aligned}$$

where W represents the bandwidth of the channel; $\mathbf{Q}_l = \mathbb{E}\{\mathbf{x}_l \mathbf{x}_l^\dagger\}$ is the input covariance matrix representing the power allocation of signal $\mathbf{x}_l$; $\det(\cdot)$ represents matrix determinant; $\mathbf{I}$ represents an $n_r \times n_r$ identity matrix; $(\cdot)_+$ represents $\max(0, \cdot)$; λ_i denotes an eigenvalue of matrix $\mathbf{H}_l \mathbf{H}_l^\dagger$ (having the same number of non-zero singular values in $\mathbf{\Lambda}_l$ and equal to the square of the singular values of $\mathbf{H}_l$); and μ is the optimal water-level satisfying $\sum_{i=1}^{d_l} (\mu - (\rho_l \alpha_l \lambda_i)^{-1})_+ = 1$.

Further, since $\mathbf{H}_l$ is a random matrix, the ergodic capacity of such a fading MIMO channel can be computed as [48]:

$$\begin{aligned} C_{l,\mathrm{ergodic}}^{(\mathrm{wf})} &= \mathbb{E}_{\mathbf{H}_l}[C_l^{(\mathrm{wf})}] = W \sum_{i=1}^{d_l} \mathbb{E}_{\lambda_i} \left[\left(\log_2(\rho_l \alpha_l \mu \lambda_i)\right)_+\right] \\ &= W \sum_{i=1}^{d_l} \int \left(\log_2(\rho_l \alpha_l \mu \lambda_i)\right)_+ f_{\lambda_i}(\lambda) d\lambda \ , \end{aligned} \tag{6.2.3}$$

where $\mathbb{E}_{\lambda_i}[\cdot]$ represents the expectation taken over the distribution of λ_i and $f_{\lambda_i}(\cdot)$ denotes the distribution of λ_i. Although (6.2.3) is the exact formula for computing

MIMO channel capacity, there are some issues that prevent (6.2.3) from being easily adopted in cross-layer optimization.

1) To determine the eigenvalues λ_i of $\mathbf{H}_l\mathbf{H}_l^\dagger$, one needs to solve the characteristic polynomial equation. However, it is known that there is no formula for roots of polynomials of degree 5 or greater. Even for a cubic or quartic polynomial equation (corresponding to 3×3 and 4×4 MIMO channels), the root formula is cumbersome to use and the equation is often solved approximately by numerical methods instead. Further, due to the complexity in computing λ_i, it is even harder to determine the distribution of $f_{\lambda_i}(\cdot)$ from $\mathbf{H}_l\mathbf{H}_l^\dagger$.

2) Even if we have solved λ_i's for a given $\mathbf{H}_l$, it remains to solve the optimal water-level μ for the optimal power allocation. However, due to the discontinuity property of the water-filling solution, there is no closed-form solution to determine μ. Instead, μ can only be evaluated numerically.

3) Since it is difficult to determine λ_i, $f_{\lambda_i}(\cdot)$, and μ, computing the integration in (6.2.3) becomes a challenging task. Instead of integrating $\left(\log_2(\rho_l\alpha_l\mu\lambda_i)\right)_+ f_{\lambda_i}(\cdot)$, we can calculate a sample mean of $\left(\log_2(\rho_l\alpha_l\mu\lambda_i)\right)_+$ as an approximation for the expectation. However, this calculation requires a large number of random samples of $\mathbf{H}_l$ (so as to obtain a good approximation).

Due to the above difficulties, Eq. (6.2.3) cannot be readily used to offer tractable analysis in cross-layer optimization.

6.2.2 A Simple and Accurate Model for MIMO Channel Capacity

To avoid the difficulties incurred in using (6.2.3), we propose a simple and yet non-trivial model to approximate the MIMO channel capacity computation as follows:

$$C_l^{(\text{sim})} = W \cdot d_l \cdot \log_2 \left(1 + \frac{\rho_l \alpha_l}{d_l}\right). \tag{6.2.4}$$

The construction of (6.2.4) is based on the following intuition. First, note that in (6.2.3), the capacity is determined by the averaging behavior of the eigenvalues of $\mathbf{H}_l\mathbf{H}_l^\dagger$. Although these eigenvalues are random, in practice they tend to be i.i.d. faded. As a result, when averaged over a large number of channel realizations, the mean channel gain for each parallel spatial channel (see Fig. 6.2) is roughly the same. Therefore, we approximate the random matrix $\mathbf{H}_l$ by a *deterministic* identity matrix (i.e., we replace $\mathbf{H}_l$ by $\mathbf{I}_{d_l}$), thus eliminating the expectation computation. With such a simplification, it is easy to verify that the optimal water-filling scheme degenerates into a trivial equal power allocation since all spatial channels have equal gain. It then follows that the channel capacity can be roughly approximated by (6.2.4). The main benefit of (6.2.4) is that we no longer need to explicitly compute the eigenvalues of $\mathbf{H}_l\mathbf{H}_l^\dagger$, the p.d.f. of λ_i, the optimal water level μ, and the expectation function. Note that when $d_l = 1$, (6.2.4) is reduced to Shannon formula for single-antenna case.

We now formally examine the accuracy of (6.2.4). First, we quantify the gap between (6.2.3) and (6.2.4) for one channel realization, for which we have the following lemma.

Lemma 6.2.1. *For a MIMO link with instantaneous channel gain $\mathbf{H}_l$ of rank d_l, $\Delta C_l \triangleq C_l^{(\text{wf})} - C_l^{(\text{sim})} \approx W \sum_{i=1}^{d_l} \log_2 \lambda_i$ under a high SNR regime.*

Proof. Under high SNR regime, $\rho_l\alpha_l$ is large enough such that all spatial channels are active with high probability, i.e., $\sum_{i=1}^{z_l}(\mu-(\rho_l\alpha_l\lambda_i)^{-1})_+ = \sum_{i=1}^{z_l}(\mu-(\rho_l\alpha_l\lambda_i)^{-1}) = 1$. Thus, we have

$$z_l\mu = 1+\textstyle\sum_{i=1}^{z_l}(\rho_l\alpha_l\lambda_i)^{-1}. \tag{6.2.5}$$

Also, we have that $C_l^{(\mathrm{wf})} = W\sum_{i=1}^{z_l}\log_2(\rho_l\alpha_l\mu\lambda_i)$. On the other hand, we can re-write the capacity formula for the simple model as $C_l^{(\mathrm{sim})} = W\sum_{i=1}^{z_l}\log_2\left(1+\frac{\rho_l\alpha_l}{z_l}\right)$. As a result, the capacity gap can be computed as

$$\begin{aligned}
\Delta C_l &= C_l^{(\mathrm{wf})} - C_l^{(\mathrm{sim})} = W\textstyle\sum_{i=1}^{z_l}\log_2\frac{\rho_l\alpha_l\mu\lambda_i}{1+\rho_l\alpha_l/z_l} \\
&\approx W\textstyle\sum_{i=1}^{z_l}\log_2 z_l\mu\lambda_i && (6.2.6)\\
&= W\textstyle\sum_{i=1}^{z_l}\log_2\left(1+(\rho_l\alpha_l)^{-1}\sum_{i=1}^{z_l}\lambda_i^{-1}\right)\lambda_i && (6.2.7)\\
&\approx W\textstyle\sum_{i=1}^{z_l}\log_2\lambda_i, && (6.2.8)
\end{aligned}$$

where (6.2.6) follows from the fact that in high SNR regime, $\rho_l\alpha_l/z_l \gg 1$ so that 1 can be ignored; (6.2.7) follows from substituting (6.2.5) into (6.2.6), and (6.2.8) follows from the fact that in high SNR regime, the term $(\rho_l\alpha_l)^{-1}\sum_{i=1}^{z_l}\lambda_i^{-1} \ll 1$ (none of λ_i can be too small since $\mathbf{H}_l$ is a well-conditioned matrix) and can be ignored. This completes the proof. □

Based on Lemma 6.2.1, we show the gap between (6.2.3) and (6.2.4) is small by showing $\mathbb{E}_{\mathbf{H}_l}[\sum_{i=1}^{d_l}\log\lambda_i]$ is negligibly small. We state the result in the following theorem.

Theorem 6.2.2. *Under a high SNR regime, for a MIMO link with Gaussian random channel matrix $\mathbf{H}_l$ of rank d_l, the approximation gap incurred by the simple model in (6.2.4) is close to zero.*

Proof. First of all, from Lemma 6.2.1, we have

$$\mathbb{E}_{\mathbf{H}_l}[\Delta C_l] \approx W \sum_{i=1}^{z_l} \mathbb{E}_{\lambda_i}[\log_2 \lambda_i] \leq W \sum_{i=1}^{z_l} \log_2 \mathbb{E}_{\lambda_i}[\lambda_i].$$

where the last inequality follows from the concavity of the log function and Jensen's inequality. The Mar**u**cenko-Pastur theorem [102] says that for a matrix $\mathbf{H}_l$ with $\frac{n_r}{n_t} = \beta$, the limiting p.d.f. of the eigenvalues of the corresponding Wishart matrix $\mathbf{H}_l\mathbf{H}_l^\dagger$ as $n_t, n_r \to \infty$ is:

$$f_\lambda^\beta(x) = \left(1 - \frac{1}{\beta}\right)_+ \delta(x) + \frac{\sqrt{(x-l)_+(u-x)_+}}{2\pi\beta x}, \tag{6.2.9}$$

where $l = (1-\sqrt{\beta})^2$, $u = (1+\sqrt{\beta})^2$, and $(\cdot)_+ = \max(0, \cdot)$, and $\delta(x)$ is the Dirac delta function. Moreover, even for small values of n_t and n_r, the p.d.f function in (6.2.9) can be used to serve as an excellent approximation [148].

Now, let us first consider the case with $\beta \leq 1$. In this case, the p.d.f. can be simplified to $f_\lambda^\beta(x) = \sqrt{(x-l)_+(u-x)_+}/2\pi\beta x$. Since all eigenvalues are i.i.d. distributed, we have

$$\begin{aligned}
&\sum_{i=1}^{z_l} \log_2 \mathbb{E}_{\lambda_i}[\lambda_i] = z_l \log_2 \mathbb{E}[\lambda] \\
&= z_l \log_2 \left(\frac{1}{2\pi\beta} \int_l^u \sqrt{-x^2 + 2(1+\beta)x - (1-\beta)^2} dx \right).
\end{aligned}$$

For convenience, let $R(x) = -x^2 + 2(1+\beta)x - (1-\beta)^2$. By using [50, Eq. 2.262], we can derive that

$$\int_l^u \sqrt{R(x)} dx = \frac{2x - 2(1+\beta)}{4} \sqrt{R(x)} \Big|_l^u + \frac{16\beta}{8} \int_l^u \frac{dx}{\sqrt{R(x)}}. \tag{6.2.10}$$

Note that the first term in the summation in (6.2.10) is zero. Then by using [50, Eqn. 2.261], we can further derive that

$$\begin{aligned}
\int_l^u \sqrt{R(x)} dx \ &= 2\beta \arcsin\left(\frac{-2x + 2(1+\beta)}{\sqrt{16\beta}} \right) \Big|_l^u \\
&= -2\beta(\arcsin(-1) - \arcsin(1)) = 2\pi\beta.
\end{aligned}$$

Table 6.1: Normalized gap versus the number of antennas.

Number of antennas	Normalized gap	
	SNR = 20 dB	SNR = 30 dB
2	0.96%	0.82%
3	2.23%	1.87%
4	2.89%	2.39%
5	3.21%	2.73%
6	3.46%	2.93%
7	3.46%	3.08%
8	3.70%	3.23%

It thus follows that

$$\mathbb{E}_{\mathbf{H}_l}[\Delta C_l] \leq W \sum_{i=1}^{z_l} \log_2 \mathbb{E}_{\lambda_i}[\lambda_i] = W z_l \log_2 \left(\frac{2\pi\beta}{2\pi\beta}\right) = 0.$$

For the case where $\beta > 1$, the first term in the p.d.f. function in (6.2.9) becomes relevant. Thus, we need to further evaluate the expectation of the first term. In this case, it is easy to see that

$$\int_l^u x \left(1 - \frac{1}{\beta}\right) \delta(x) dx = \left(1 - \frac{1}{\beta}\right) \int_l^u x\delta(x) dx = 0. \tag{6.2.11}$$

Combining two cases, we finally have $\mathbb{E}_{\mathbf{H}_l}[\Delta C_l] \approx 0$ for all β. □

To offer some quantitative insights on this gap, in Table 6.1, we show the normalized gap between (6.2.3) and (6.2.4) for a MIMO channel under $\rho_l = 20$ dB and $\rho_l = 30$ dB, respectively. We vary the number of antennas from 2 to 8 (range for practical MIMO systems). We can see that the gap between the approximation and the exact capacity is indeed negligibly small. For example, with 4 antennas under 30 dB, the gap is only 2.39%.

6.3 Link Layer Modeling for Multi-hop MIMO Networks

At the link layer, MIMO opens up new opportunities in space domain to mitigate interference. In Section 6.3.1, we first describe zero forcing beamforming (ZFBF), which is a powerful MIMO interference mitigation technique. We also discuss its benefits and challenges in multi-hop network setting. In Section 6.3.2, we propose a space-time scheduling scheme called OBIC and in Section 6.3.3, we construct its mathematical model.

6.3.1 Zero-Forcing Beamforming: Benefits and Challenges

In cellular MIMO systems, one of the most powerful interference mitigation technique is called ZFBF [23, 139]. ZFBF uses multi-antenna arrays to steer beams toward the intended receiver to increase SNR, while forming nulls to unintended receivers to avoid interference. In MIMO cellular systems, however, ZFBF is usually performed at the transmitter side. In this paper, we generalize ZFBF to multi-hop networks by allowing beamforming to be performed on both transmitter and receiver sides.

To see how the generalized ZFBF can be used in multi-hop MIMO networks, consider a network having L links among which L_0 links are active. We denote $\bar{\mathcal{I}}_l$ the set of links that interfere with the reception of link l's intended receiver, $l = 1, 2, \ldots, L_0$. We denote $\mathbf{H}_{\mathrm{Tx}(m),\mathrm{Rx}(l)}$ the interference channel gain matrix from transmitting node of interference link m ($\mathrm{Tx}(m)$) to receiving node of link l ($\mathrm{Rx}(l)$).

To extract the transmitted signal through a MIMO channel, a transmit beamforming matrix and a receive beamforming matrix are employed on the channel. Thus,

the received signal at link l can be written as

$$\mathbf{y}_l = \underbrace{\sqrt{\rho_l \alpha_l}\, \mathbf{R}^{\dagger}_{\mathrm{Rx}(l)} \mathbf{H}_l \mathbf{T}_{\mathrm{Tx}(l)} \mathbf{x}_l}_{\text{Desired signal part}} + \underbrace{\sum_{m \in \bar{\mathcal{I}}_l} \sqrt{\rho_{m,l} \alpha_m} \mathbf{R}^{\dagger}_{\mathrm{Rx}(l)} \mathbf{H}_{\mathrm{Tx}(m),\mathrm{Rx}(l)} \mathbf{T}_{\mathrm{Tx}(m)} \mathbf{x}_m}_{\text{Interference part}} + \mathbf{n}_l, \ \forall l,$$

where $\rho_{m,l}$ denotes the interference-to-noise ratio (INR) from node $\mathrm{Tx}(m)$ to node $\mathrm{Rx}(l)$.

By exploiting the multi-antenna array at each node, it is possible to cancel out all interferences by judiciously configuring $\mathbf{T}$'s and $\mathbf{R}$'s. Specifically, we can configure $\mathbf{T}$'s and $\mathbf{R}$'s in such a way that

$$\mathbf{R}^{\dagger}_{\mathrm{Rx}(l)} \mathbf{H}_{\mathrm{Tx}(m),\mathrm{Rx}(l)} \mathbf{T}_{\mathrm{Tx}(m)} = \mathbf{0}, \quad \forall l, m \in \bar{\mathcal{I}}_l. \tag{6.3.1}$$

Note that if there exist non-trivial solutions to (6.3.1) (i.e., $\mathbf{R}_{\mathrm{Rx}(l)} \neq \mathbf{0}$, $\mathbf{T}_{\mathrm{Tx}(m)} \neq \mathbf{0}$, $\forall l, m \in \bar{\mathcal{I}}_l$), then it means that all L_0 links can be active simultaneously in an *interference-free* environment. Moreover, the ranks of $\mathbf{T}_{\mathrm{Tx}(l)}$ and $\mathbf{R}_{\mathrm{Rx}(l)}$ determine the maximum number of data streams z_l that can be transmitted over link l, i.e., $z_l \leq \min\{\mathrm{rank}(\mathbf{T}_{\mathrm{Tx}(l)}), \mathrm{rank}(\mathbf{R}_{\mathrm{Rx}(l)})\}$.

Although ZFBF's benefits are appealing, there remain significant challenges to employ it in multi-hop networks. This is because finding an optimal set of $\mathbf{T}$'s and $\mathbf{R}$'s satisfying (6.3.1) requires solving a large number of *bilinear* equations. Unlike linear equation systems, a general solution to bilinear equation systems remains unknown [128]. Thus, it becomes an intractable problem to design scheduling schemes based on solving (6.3.1).

6.3.2 OBIC: Basic Idea

We find that the specific element configurations in $\mathbf{T}$'s and $\mathbf{R}$'s are more closely tied to beamforming design than to link layer scheduling. Therefore, instead of focusing on solving (6.3.1), we propose to reposition ourselves to exploit matrix dimension constraints that are sufficient for (6.3.1) to hold. By doing so, we can characterize the link layer scheduling performance without entangling with the details of beamforming designs.

To understand how we can extract the matrix dimension constraints for ZFBF-based scheduling, we first use a simple two-link network shown in Fig. 6.3 as an example. In this network, link l has 3 antennas on each side and link m has 5 antennas on each side. For this simple network, we can first choose a $\mathbf{T}_{\mathrm{Tx}(l)}$ arbitrarily without considering link m's existence. Suppose that $\mathbf{T}_{\mathrm{Tx}(l)}$ is full-rank (i.e., 3 data streams being transmitted). Next, we choose an $\mathbf{R}_{\mathrm{Rx}(m)}$ to cancel the interference from link l, while receiving data streams from its desired transmitter. This is equivalent to solving $(\mathbf{H}_{\mathrm{Tx}(l),\mathrm{Rx}(m)}\mathbf{T}_{\mathrm{Tx}(l)})^{\dagger}\mathbf{R}_{\mathrm{Rx}(m)} = \mathbf{0}$ with $\mathbf{T}_{\mathrm{Tx}(l)}$ already determined. This implies that all column vectors in $\mathbf{R}_{\mathrm{Rx}(m)}$ have to lie in the null space of $(\mathbf{H}_{\mathrm{Tx}(l),\mathrm{Rx}(m)}\mathbf{T}_{\mathrm{Tx}(l)})^{\dagger}$. The dimension of the null space in this case is $\dim(\mathrm{null}((\mathbf{H}_{\mathrm{Tx}(l),\mathrm{Rx}(m)}\mathbf{T}_{\mathrm{Tx}(l)})^{\dagger})) = 5-3 = 2$, meaning that $\mathrm{Rx}(m)$ can receive up to 2 streams. Note that links l and m are both active in an *interference-free* environment. Further, by varying the ranks of $\mathbf{T}_{\mathrm{Tx}(l)}$ and $\mathbf{R}_{\mathrm{Rx}(m)}$, it is easy to verify that the achievable DoF region under this ZFBF-based scheme is the trapezoid shown in Fig. 6.4.[1] Observe that the scheduling scheme in this two-link example is performed in an *ordered* fashion: we arbitrarily choose a

[1]Note that the achievable DoF region in Fig. 6.4 coincides with the maximum DoF region described in [67, Theorem 2]. Thus, for this two-link example, the proposed scheduling scheme is an optimal scheduling scheme.

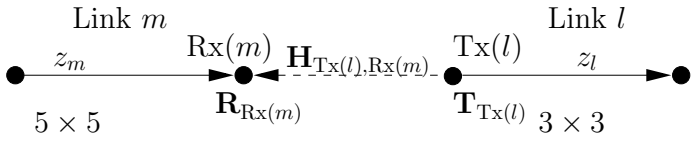


Figure 6.3: A two-link example.

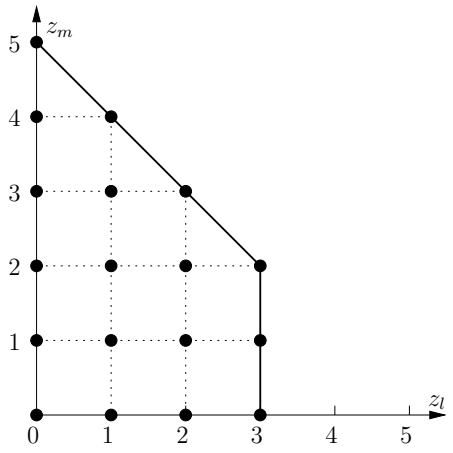


Figure 6.4: DoF region of two-link example.

$\mathbf{T}_{\mathrm{Tx}(l)}$ first, and then choose an $\mathbf{R}_{\mathrm{Rx}(m)}$ such that the interference can be eliminated.

We now extend this order-based interference cancellation idea to a three-link example shown in Fig. 6.5, which is more complicated than the previous example. Here, each receiving node of a link is being interfered by the transmitting nodes of other links. For this example, we can start with a scheduling order for the six nodes. Such a scheduling order will be subject to an optimization in Section 6.3.3. Suppose the scheduling order for the six nodes is $\mathrm{Tx}(l) \rightarrow \mathrm{Rx}(m) \rightarrow \mathrm{Rx}(l) \rightarrow \mathrm{Tx}(m) \rightarrow \mathrm{Tx}(n) \rightarrow \mathrm{Rx}(n)$. Then, through the following scheduling decision, we can show that the stream combination $(1, 1, 2)$ is achievable.

1) Tx(l): Since nodes Tx(l) is the first to be scheduled, it does not have any interference to concern about. Also, since Tx(l) has only 1 antenna, we can let Tx(l) transmit 1 data stream;

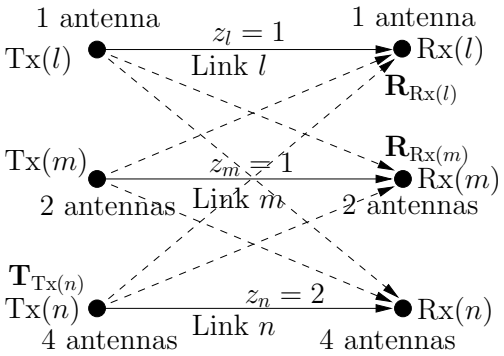


Figure 6.5: A three-link example.

2) Rx(m): Since Rx(m) is scheduled after Tx(l), it needs to suppress the interference from Tx(l), i.e., solving $(\mathbf{H}_{\mathrm{Tx}(l),\mathrm{Rx}(m)}\mathbf{T}_{\mathrm{Tx}(l)})^{\dagger}\mathbf{R}_{\mathrm{Rx}(m)} = \mathbf{0}$. We have

$$\dim(\mathrm{null}((\mathbf{H}_{\mathrm{Tx}(l),\mathrm{Rx}(m)}\mathbf{T}_{\mathrm{Tx}(l)})^{\dagger})) = 2 - 1 = 1,$$

i.e., we can let Rx(m) receive 1 stream in this case;

3) Rx(l): Since no interfering transmitting node is scheduled before Rx(l), Rx(l) does not need to concern about any interference. Given Rx(l) has only 1 antenna, we can let it receive 1 stream;

4) Tx(m): Following the similar argument as for Rx(m), we can let Tx(m) transmit 1 stream;

5) Tx(n): Since Tx(n)'s transmission should not interfere with Rx(l) and Rx(m), it follows that $\mathbf{T}_{\mathrm{Tx}(n)}$ should satisfy $\begin{bmatrix} \mathbf{R}^{\dagger}_{\mathrm{Rx}(l)}\mathbf{H}_{\mathrm{Tx}(n),\mathrm{Rx}(l)} \\ \mathbf{R}^{\dagger}_{\mathrm{Rx}(m)}\mathbf{H}_{\mathrm{Tx}(n),\mathrm{Rx}(m)} \end{bmatrix}\mathbf{T}_{\mathrm{Tx}(n)} = \mathbf{0}$. Since

$$\dim\left(\mathrm{null}\begin{bmatrix} \mathbf{R}^{\dagger}_{\mathrm{Rx}(l)}\mathbf{H}_{\mathrm{Tx}(n),\mathrm{Rx}(l)} \\ \mathbf{R}^{\dagger}_{\mathrm{Rx}(m)}\mathbf{H}_{\mathrm{Tx}(n),\mathrm{Rx}(m)} \end{bmatrix}\right) = 4 - (1+1) = 2,$$

we can schedule 2 data streams at Tx(n);

6) Rx(n): Following a similar analysis as in 5), it can be shown that 2 streams can be scheduled at Rx(n).

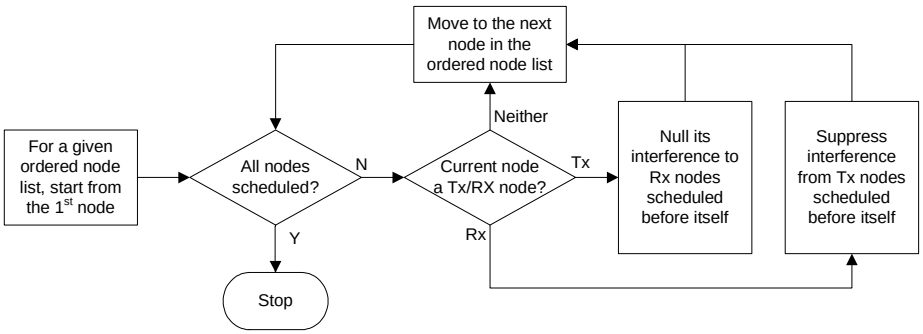


Figure 6.6: The flow chart of OBIC scheduling scheme.

The idea in the three-link example can be synthesized for a general multiple-link setting. The essence of this scheduling scheme is to perform interference cancellation successively based on an ordered node list:

- If a node is transmitting, then it is only necessary to ensure that its transmissions do not interfere with previously scheduled receiving nodes in the ordered node list. It does not need to expend precious DoF resources to null its interference to those receiving nodes to be scheduled after itself in the node list.

- If a node is receiving, it only needs to suppress interference from transmitting nodes scheduled before itself in the node list. It does not need to concern interfering transmitting nodes to be scheduled after itself.

The interference cancellation behavior described above offers the basic idea for a node-based scheduling scheme. For easy reference, we call this scheduling scheme OBIC (order-based interference cancelation). Additional quantitative constraints on DoF on each transmitting and receiving node (as shown in last two examples) will be discussed in the next section. Fig. 6.6 shows the flow chart of the OBIC scheme.

Remark 6.3.1. In [53], Hamdaoui and Shin proposed several interference avoidance schemes based on ZFBF. For the so-called CiM scheme (the best among the proposed

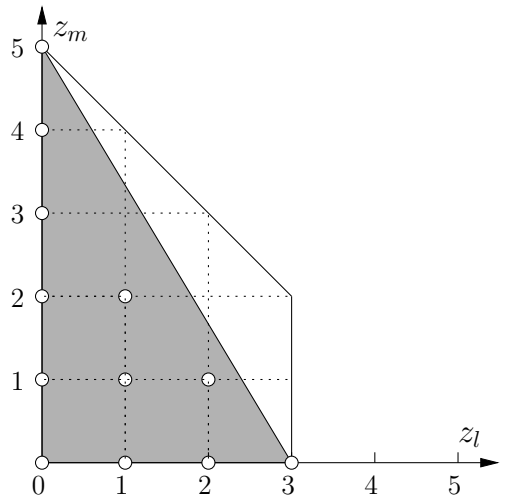


Figure 6.7: Achievable DoF region comparison between OBIC and CiM for the example in Fig. 6.3.

schemes in [53]), the authors also recognized that interference can be cancelled by either the transmitting or the receiving node of an interference link. However, without employing a node-based sequential scheduling, it is impossible to know which node should perform interference mitigation. As a result, the CiM scheme requires both the transmitting and receiving nodes of an interference link to expend precious DoFs for interference cancellation (c.f. [53, Eq. (10)]). This approach adversely leads to a much smaller DoF region. As an example, we compare the performance of OBIC and the CiM model on the simple two-link example in Fig. 6.3. Under the CiM model, it is not difficult to verify that the achievable DoF region is the shaded triangle in Fig. 6.7, representing the convex hull of the white dots, which are the DoF combinations directly achievable under CiM. It can be seen that this region is smaller than the achievable DoF region by OBIC. In general, it can be shown that the DoF region achieved under the CiM model is always a subset of that under the OBIC [89].

Remark 6.3.2. We point out that for the three-link network example in Fig. 6.5, a

larger DoF region can be achieved by interference alignment (IA) [19]. The basic idea of IA is that by aligning the interference from two interfering transmitters at each receiving node, the interference from different nodes becomes dependent, implying a smaller rank of the effective interference channel. This in turn leads to a higher dimensional null space that can be exploited for data transmissions. We note, however, that IA also has its limitations. IA was proposed for the classical interference channel in the context of network information theory [29], where the channel possesses a "fully-connected" structure. That is, each link in the network is interfered by all remaining links. For general multi-hop network topology where the full connectivity may not hold, the achievable DoF region of IA remains an open problem.

6.3.3 OBIC: A Mathematical Model

Having introduced the basic idea of OBIC, we now develop its mathematical model. We represent the topology of a MIMO ad hoc network by a directed graph, denoted by $\mathcal{G} = \{\mathcal{N}, \mathcal{L}\}$, where $\mathcal{N}$ and $\mathcal{L}$ are the sets of nodes and all possible MIMO links, respectively. Suppose that the cardinalities of the sets $\mathcal{N}$ and $\mathcal{L}$ are $|\mathcal{N}| = N$ and $|\mathcal{L}| = L$, respectively. In this chapter, we assume that scheduling operates in on period frame-by-frame system with T time slots in each frame.

Ordered Node List: An Optimization Model. Referring to Fig. 6.6 and the discussion in Section 6.3.2, before we start scheduling on a node, we must a model for an ordered node list. The node ordering in this list will be part of the optimization problem.

To model an ordered node list that can be optimized, we define the following

Table 6.2: Enumerating the transitivity relationship.

$\pi_{ij}(t)$	$\pi_{jk}(t)$	$\pi_{ik}(t)$
1	1	1
1	0	indefinite
0	1	indefinite
0	0	0

binary variable. For $i, j \in \mathcal{N}$, $i \neq j$, we let

$$\pi_{ij}(t) = \begin{cases} 1 & \text{if node } j \text{ is after node } i \text{ in time slot } t, \\ 0 & \text{if node } j \text{ is before node } i \text{ in time slot } t. \end{cases}$$

To model an ordered node list, $\pi_{ij}(t)$ must satisfy the following two properties.

1) *Mutual exclusiveness*: If node j is scheduled after node i (i.e., $\pi_{ij}(t) = 1$), then it also implies that node i is before node j (i.e., $\pi_{ji}(t) = 0$). This relationship can be modeled as

$$\pi_{ij}(t) + \pi_{ji}(t) = 1, \quad \forall i, j \in \mathcal{N} : i \neq j, \ \forall t. \tag{6.3.2}$$

Transitivity: If node j is scheduled after i (i.e., $\pi_{ij} = 1$) and node k is scheduled after node j (i.e., $\pi_{jk} = 1$), then it implies that node k is scheduled after node i (i.e., $\pi_{ik} = 1$). We enumerate all possible cases for π_{ij} and π_{jk} in Table 6.2 and show what π_{ik} will be. In Table 6.2, "indefinite" means that $\pi_{ik}(t)$ cannot be determined by the current settings of $\pi_{ij}(t)$ and $\pi_{jk}(t)$. Mathematically, the relationship in Table 6.2 can be modeled as

$$\pi_{ik}(t) \leq \pi_{ij}(t) + \pi_{jk}(t), \quad \forall i, j, k \in \mathcal{N}, \forall t, \tag{6.3.3}$$

$$\pi_{ik}(t) \geq \pi_{ij}(t) + \pi_{jk}(t) - 1, \quad \forall i, j, k \in \mathcal{N}, \forall t. \tag{6.3.4}$$

For example, for the second row in Table 6.2, when $\pi_{ij}(t) = 1$ and $\pi_{jk}(t) = 0$, (6.3.3) gives $\pi_{ik}(t) \leq 1$ and (6.3.4) gives $\pi_{ik}(t) \geq 0$. Given that $\pi_{ik}(t)$ is a binary variable, this is equivalent to saying that $\pi_{ik}(t)$ remains to be determined (or indefinite).

Note that according to (6.3.3) and (6.3.4), we can write 12 different constraints in 6 groups for three nodes i, j, $k \in \mathcal{N}$ as follows.

$$\begin{cases} \pi_{ij}(t) \leq \pi_{ik}(t) + \pi_{kj}(t) \\ \pi_{ij}(t) \geq \pi_{ik}(t) + \pi_{kj}(t) - 1, \end{cases} \tag{6.3.5}$$

$$\begin{cases} \pi_{ji}(t) \leq \pi_{jk}(t) + \pi_{ki}(t) \\ \pi_{ji}(t) \geq \pi_{jk}(t) + \pi_{ki}(t) - 1, \end{cases} \tag{6.3.6}$$

$$\begin{cases} \pi_{ik}(t) \leq \pi_{ij}(t) + \pi_{jk}(t) \\ \pi_{ik}(t) \geq \pi_{ij}(t) + \pi_{jk}(t) - 1, \end{cases} \tag{6.3.7}$$

$$\begin{cases} \pi_{ki}(t) \leq \pi_{kj}(t) + \pi_{ji}(t) \\ \pi_{ki}(t) \geq \pi_{kj}(t) + \pi_{ji}(t) - 1, \end{cases} \tag{6.3.8}$$

$$\begin{cases} \pi_{jk}(t) \leq \pi_{ji}(t) + \pi_{ik}(t) \\ \pi_{jk}(t) \geq \pi_{ji}(t) + \pi_{ik}(t) - 1, \end{cases} \tag{6.3.9}$$

$$\begin{cases} \pi_{kj}(t) \leq \pi_{ki}(t) + \pi_{ij}(t) \\ \pi_{kj}(t) \geq \pi_{ki}(t) + \pi_{ij}(t) - 1. \end{cases} \tag{6.3.10}$$

A closer look at these 6 groups of constraints show that any one group can be used to derive the other 5 groups of constraints (will be shown in Lemma 6.3.1). In other words, any one group from the six groups in (6.3.5)–(6.3.10) is sufficient to describe the transitivity property for a node triplet $\{i, j, k\}$.

To select only one group out of (6.3.5)–(6.3.10) to describe the transitivity property, we need to maintain certain consistency in such selection. Our approach to achieve such consistency is to create a mapping $\Omega(\cdot) : \mathcal{N} \to \mathbb{N}$, where each node in $\mathcal{N}$ is mapped to an integer number in $\mathbb{N} = \{1, 2, \ldots, N\}$. Now, without loss of generality, suppose that nodes i, j, and k satisfy $\Omega(i) < \Omega(j) < \Omega(k)$. We will select

group constraint (6.3.5), which is the same as selecting two constraints from (6.3.3) and (6.3.4) such that $\Omega(i) < \Omega(j) < \Omega(k)$. Using such a mapping and the constraint $\Omega(i) < \Omega(j) < \Omega(k)$, we can consistently and uniquely identify a group out of 6 groups for any node triplet. Now, combining the ordered mapping and (6.3.3) and (6.3.4), we have:

$$\pi_{ik}(t) \leq \pi_{ij}(t) + \pi_{jk}(t), \quad \forall i,j,k \in \mathcal{N} : \Omega(i) < \Omega(j) < \Omega(k), \forall t, \tag{6.3.11}$$

$$\pi_{ik}(t) \geq \pi_{ij}(t) + \pi_{jk}(t) - 1, \quad \forall i,j,k \in \mathcal{N} : \Omega(i) < \Omega(j) < \Omega(k), \forall t. \tag{6.3.12}$$

We can further show that (6.3.11) and (6.3.12) can be written in a more compact form: $1 \leq \pi_{ij}(t) + \pi_{jk}(t) + \pi_{ki}(t) \leq 2$. We summarize this result in the following lemma.

Lemma 6.3.1. *For nodes $i,j,k \in \mathcal{N}$ such that $\Omega(i) < \Omega(j) < \Omega(k)$, the following two constraints are sufficient to describe the transitivity relationship among nodes node triplet i, j, and k:*

$$1 \leq \pi_{ij}(t) + \pi_{jk}(t) + \pi_{ki}(t) \leq 2. \tag{6.3.13}$$

Proof. For simplicity, we drop the time slot index t of $\pi-$ variables in this proof. We first show that given any one group in (6.3.5)–(6.3.10), the remaining 5 are redundant. Without loss generality, let us consider the two constraints in group (6.3.5), i.e.,

$$\begin{cases} \pi_{ik} \leq \pi_{ij} + \pi_{jk} \\ \pi_{ik} \geq \pi_{ij} + \pi_{jk} - 1. \end{cases}$$

Multiply -1 and adding 1 on both sides of the above two inequalities, we have

$$\begin{cases} 1 - \pi_{ik} \geq 1 - \pi_{ij} - \pi_{jk} \\ 1 - \pi_{ik} \leq 1 - \pi_{ij} - \pi_{jk} + 1 \end{cases} \Rightarrow \begin{cases} \pi_{ki} \geq \pi_{ji} + \pi_{kj} - 1 \\ \pi_{ki} \leq \pi_{ji} + \pi_{kj}. \end{cases}$$

This shows that the set of constraints in group (6.3.6) can be derived from group (6.3.5) and the mutual exclusiveness constraints. Thus, group (6.3.6) is not independent and can be removed. Following the same token, we can also show that the groups (6.3.8) and (6.3.10) can derived from groups (6.3.7) and (6.3.9). So, they can also be removed. Now, let us consider the remaining 6 constraints associated with groups (6.3.5), (6.3.7), and (6.3.9). Note that

$$\begin{cases} \pi_{ik} \le \pi_{ij} + \pi_{jk} & \Rightarrow \quad \pi_{ik} \le \pi_{ij} + 1 - \pi_{kj} \\ & \Rightarrow \quad \pi_{ij} \ge \pi_{ik} + \pi_{kj} - 1 \\ \pi_{ik} \ge \pi_{ij} + \pi_{jk} - 1 & \Rightarrow \quad \pi_{ik} \ge \pi_{ij} - \pi_{kj} \\ & \Rightarrow \quad \pi_{ij} \le \pi_{ik} + \pi_{kj}. \end{cases}$$

This shows that group (6.3.7) can be derived from group (6.3.5) and can also removed. Following exactly the same token, we can show that group (6.3.9) can be derived from group (6.3.5) and also can be removed. So finally, we have only 2 remaining constraints associated with group (6.3.5).

Next, we show that (6.3.5) can be written in a more compact form. Note that

$$\begin{cases} \pi_{ik} \le \pi_{ij} + \pi_{jk} & \Rightarrow \quad 1 - \pi_{ki} \le \pi_{ij} + \pi_{jk} \\ & \Rightarrow \quad \pi_{ij} + \pi_{jk} + \pi_{ki} \ge 1 \\ \pi_{ik} \ge \pi_{ij} + \pi_{jk} - 1 & \Rightarrow \quad 1 - \pi_{ki} \ge \pi_{ik} + \pi_{jk} - 1 \\ & \Rightarrow \quad \pi_{ij} + \pi_{jk} + \pi_{ki} \le 2, \end{cases}$$

which give the constraints stated in Lemma 6.3.1. □

It is not hard to see that Lemma 6.3.1 will significantly reduce computational complexity in finding the optimal spatial scheduling order. Also note that loop is not allowed in the transitivity relationship in (6.3.13). This is because that (6.3.13) implies that at least 1 and at most 2 π-variables can be equal to one. The cases when loop occurs, i.e., either $\pi_{ij}(t) = \pi_{jk}(t) = \pi_{ki}(t) = 0$ or $\pi_{ij}(t) = \pi_{jk}(t) = \pi_{ki}(t) = 1$, are not allowed in (6.3.13).

In summary, to optimize the ordered node list, we need list (6.3.2) and (6.3.13) for every node triplet from $\mathcal{N}$.

Models for Transmitting and Reeving Nodes. Next, we model the block in Fig. 6.6 where node i is scheduled to be a transmitting node. Note that in each time slot t, $1 \leq t \leq T$, due to half-duplex limitation, each node either transmits, receives, or be idle. To model half-duplex, we introduce two groups of binary variables $g_i(t)$'s and $h_i(t)$'s as follows. $g_i(t) = 1$ if node i is transmitting in time slot t and 0 otherwise; $h_i(t) = 1$ if node i is receiving in time slot t and 0 otherwise. Then, the half-duplex constraint can be characterized by

$$g_i(t) + h_i(t) \leq 1, \quad \forall i, t. \tag{6.3.14}$$

We assume that scattering is rich enough in the environment such that all channel matrices are of full-rank. As a result, the number of data streams that a node can transmit or receive is limited by its number of antennas and we have the following two constraints:

$$g_i(t) \leq \sum_{l \in \mathcal{L}_i^{\text{Out}}} z_l(t) \leq g_i(t) A_i, \tag{6.3.15}$$

$$h_i(t) \leq \sum_{l \in \mathcal{L}_i^{\text{In}}} z_l(t) \leq h_i(t) A_i, \tag{6.3.16}$$

where $\mathcal{L}_i^{\text{Out}}$ and $\mathcal{L}_i^{\text{In}}$ represent the sets of outgoing and incoming links at node i, respectively; $z_l(t)$ denotes the number of data streams over link l in time slot t, and A_i represents the number of antennas at node i.

Recall that when node i is transmitting data streams, it should not interfere with those receiving nodes scheduled before node itself. This is equivalent to saying that the transmission beamforming vectors in $\mathbf{T}_i$ should lie in the null space of the stacked

matrix formed by stacking all $\mathbf{R}_j^{\dagger}\mathbf{H}_{i,j}$ matrices, where j denotes a previously scheduled receiving node that could be interfered by node i. That is,

$$\mathbf{T}_i \in \text{null}\begin{bmatrix} \vdots \\ \mathbf{R}_j^{\dagger}\mathbf{H}_{i,j} \\ \vdots \end{bmatrix}, \begin{array}{l} j \in \mathcal{I}_i \text{ and } j \text{ is} \\ \text{scheduled before } i \\ \text{(i.e., } \pi_{ji} = 1\text{)}, \end{array} \tag{6.3.17}$$

where $\mathcal{I}_i$ represents the set of nodes within the interference range of node i. For convenience, we let $\mathbf{S}$ denote the stacked matrix in (6.3.17). Note that $\sum_{l\in\mathcal{L}_i^{\text{Out}}} z_l(t)$ is the number of data streams that node i transmits in time slot t. Thus, from (6.3.17), we have that $\sum_{l\in\mathcal{L}_i^{\text{Out}}} z_l(t)$ should be less than or equal to the nullity of $\mathbf{S}$, i.e., $\sum_{l\in\mathcal{L}_i^{\text{Out}}} z_l(t) \leq \text{null}(\mathbf{S})$. Also, note that the rank of $\mathbf{S}$ is $\sum_{j\in\mathcal{I}_i} \pi_{ji}(t) \sum_{l:\text{Rx}(l)=j}^{\text{Tx}(l)\neq i} z_l(t)$. Therefore, according to rank-nullity theorem [63] (i.e., $\text{rank}(\mathbf{S}) + \text{null}(\mathbf{S})$ is equal to the number of columns in $\mathbf{S}$), we can model the dimensional constraint as follows. For all $i, j \in \mathcal{N}$ and for $t = 1, 2, \cdots, T$,

$$\sum_{l\in\mathcal{L}_i^{\text{Out}}} z_l(t) + \sum_{j\in\mathcal{I}_i} \pi_{ji}(t) \sum_{\substack{l:\text{Rx}(l)=j \\ \text{Tx}(l)\neq i}} z_l(t) \leq A_i + (1 - g_i(t))M. \tag{6.3.18}$$

In (6.3.18), M is a sufficiently large number (e.g., we can set $M = \sum_{j\in\mathcal{I}_i} A_i$). When node i is a transmission node (i.e., $g_i(t) = 1$), then (6.3.18) is reduced to the rank-nullity condition with respect to $\mathbf{S}$. Otherwise, if node i is scheduled to be a receiving node or in idle status (i.e., $g_i(t) = 0$), then (6.3.18) holds trivially due to the large value of M.

We note that the nonlinear terms $\pi_{ji}(t) \sum_{l:\text{Rx}(l)=j}^{\text{Tx}(l)\neq i} z_l(t)$ in (6.3.18) could complicate optimization algorithms design. To remove these nonlinear terms, we can introduce a new integer variable $\phi_{ji}(t)$ and reformulate (6.3.18) as follows:

$$\sum_{l\in\mathcal{L}_i^{\text{Out}}} z_l(t) + \sum_{j\in\mathcal{I}_i} \phi_{ji}(t) \leq A_i + (1 - g_i(t))M, \tag{6.3.19}$$

where $\phi_{ji}(t)$, $\forall i \in \mathcal{N}$, $j \in \mathcal{I}_i$, $\forall t$, satisfies the following constraints:

$$\phi_{ji}(t) \leq \sum_{l:\mathrm{Rx}(l)=j}^{\mathrm{Tx}(l)\neq i} z_l(t), \tag{6.3.20}$$

$$\phi_{ji}(t) \leq A_i \pi_{ji}(t), \tag{6.3.21}$$

$$\phi_{ji}(t) \geq A_i \pi_{ji}(t) - A_i + \sum_{l:\mathrm{Rx}(l)=j}^{\mathrm{Tx}(l)\neq i} z_l(t). \tag{6.3.22}$$

It can be verified that this set of new constraints (6.3.19)–(6.3.22) is equivalent to (6.3.18) as follows. First, when $\pi_{ji}(t) = 1$, we have that $\phi_{ji}(t) = \sum_{l:\mathrm{Rx}(l)=j}^{\mathrm{Tx}(l)\neq i} z_l(t)$ from (6.3.20) and (6.3.22). Thus, the terms in (6.3.19) with $\pi_{ji}(t) = 1$ are the same as those in (6.3.18). On the other hand, if $\pi_{ji}(t) = 0$, we have that $\phi_{ji}(t) = 0$ as well from (6.3.20)–(6.3.22). Thus, the terms in (6.3.19) with $\pi_{ji}(t) = 0$ are the same as those in (6.3.18).

The approach to model receiving node behavior is similar to the above model for the receiving node. When node i is scheduled to be a receiving node, we can derive the following constraints. For all $i, j \in \mathcal{N}$ and for $t = 1, 2, \cdots, T$,

$$\sum_{l \in \mathcal{L}_i^{\mathrm{In}}} z_l(t) + \sum_{j \in \mathcal{I}_i} \varphi_{ji}(t) \leq A_i + (1 - h_i(t))M, \tag{6.3.23}$$

$$\varphi_{ji}(t) \leq \sum_{l:\mathrm{Tx}(l)=j}^{\mathrm{Rx}(l)\neq i} z_l(t), \tag{6.3.24}$$

$$\varphi_{ji}(t) \leq A_i \pi_{ji}(t), \tag{6.3.25}$$

$$\varphi_{ji}(t) \geq A_i \pi_{ji}(t) - A_i + \sum_{l:\mathrm{Tx}(l)=j}^{\mathrm{Rx}(l)\neq i} z_l(t). \tag{6.3.26}$$

Link Capacity Computation under OBIC. In Section 6.2, we proposed a simple and accurate physical layer model to approximate the capacity of a single MIMO link. We are now ready to further extend (6.2.4) to approximate the capacity of each MIMO link under OBIC. First, we note that there is no interference among links due to ZFBF. Thus, as the single link case in Section 6.2, the capacity of each

link under OBIC is not affected by interference. However, compared with the single MIMO link case, a key difference in OBIC is that each active link l now transmits z_l data streams instead of d_l and $z_l \leq d_l$. In this case, one may conjecture that a simple way to modify (6.2.4) for OBIC is to replace d_l with z_l. Indeed, the following theorem says that such an extension is correct.

Theorem 6.3.2. *Under OBIC, each MIMO link's capacity in time slot t can be approximated as*

$$C_l(t) = W \cdot z_l(t) \cdot \log_2 \left(1 + \frac{\rho_l \alpha_l(t)}{z_l(t)}\right). \tag{6.3.27}$$

Moreover, under a high SNR regime, the approximation gap incurred by (6.3.27) is negligible.

Proof. For simplicity, we drop the time index t in this proof. Without loss of generality, we suppose that $\mathrm{Tx}(l)$'s transmission interferes with some receiving nodes $\mathrm{Rx}(j)$, $j = 1, \ldots, p$, and the channel between $\mathrm{Tx}(l)$ and each $\mathrm{Rx}(j)$ is $\mathbf{H}_{l,j}$. We also suppose that $\mathbf{R}_{\mathrm{Rx}(j)}$'s has been scheduled earlier than $\mathrm{Tx}(l)$. According to SUCCINCT, to avoid interference, $\mathbf{T}_{\mathrm{Tx}(l)}$ needs to stay in the null space of stacked effective interference channel $[\cdots \mathbf{H}_{l,j}^{\dagger} \mathbf{R}_{\mathrm{Rx}(j)} \cdots]^{\dagger}$. For this purpose, we can perform SVD on effective interference channel gain matrix $[\cdots \mathbf{H}_{l,j}^{\dagger} \mathbf{R}_{\mathrm{Rx}(j)} \cdots]^{\dagger}$ and choose $\mathbf{T}_{\mathrm{Tx}(l)}$ as all the right singular vectors corresponding to the zero singular values. Suppose that the null space size of $[\cdots \mathbf{H}_{l,j}^{\dagger} \mathbf{R}_{\mathrm{Rx}(j)} \cdots]^{\dagger}$ is z_l. This means that $\mathbf{T}_{\mathrm{Tx}(l)}$ has z_l column vectors that constitute a orthonormal basis.

For notation simplicity, from now on we use $\mathbf{T}_l$ as a shorthand for $\mathbf{T}_{\mathrm{Tx}(l)}$. Any data that link l transmits can be represented as $\mathbf{T}_l \mathbf{x}_l$, where $\mathbf{x}_l \in \mathbb{C}^{n_t}$ is the signal to be transmitted. Let $\mathbf{Q}_l \triangleq \mathbb{E}\{\mathbf{x}_l \mathbf{x}_l^{\dagger}\} \in \mathbb{C}^{z_l \times z_l}$. Clearly, $\mathbf{Q}_l$ is of rank-z_l. The link

capacity of l under the beamforming constraint of $\mathbf{T}_l$ can be written as

$$C_l = \max_{\mathbf{Q}_l} \mathbb{E}_{\mathbf{H}_l}[W \log_2 \det(\mathbf{I} + \rho_l \alpha_l \mathbf{H}_l \mathbf{T}_l \mathbf{Q}_l \mathbf{T}_l^\dagger \mathbf{H}_l^\dagger)], \tag{6.3.28}$$

Since all column vectors in $\mathbf{T}_l$ are orthonormal, we can derive a unitary matrix $\bar{\mathbf{T}}_l$ from $\mathbf{T}_l$. Also, let $\bar{\mathbf{Q}}_l$ represent the the appended block matrix $\begin{bmatrix} \mathbf{Q}_l & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}$. Then, it is clear that (6.3.28) is equivalent to

$$C_l = \max_{\mathbf{Q}_l} \mathbb{E}_{\mathbf{H}_l}[W \log_2 \det(\mathbf{I} + \rho_l \alpha_l \mathbf{H}_l \bar{\mathbf{T}}_l \bar{\mathbf{Q}}_l \bar{\mathbf{T}}_l^\dagger \mathbf{H}_l^\dagger)], \tag{6.3.29}$$

Since $\bar{\mathbf{T}}_l$ is unitary and $\mathbf{H}_l$ is Gaussian, $\mathbf{H}_l \bar{\mathbf{T}}_l$ has the same distribution as $\mathbf{H}_l$ [146, Lemma 5]. Thus, we have

$$\begin{aligned} C_l &= \max_{\mathbf{Q}_l} \mathbb{E}_{\mathbf{H}_l}[W \log_2 \det(\mathbf{I} + \rho_l \alpha_l \mathbf{H}_l \bar{\mathbf{Q}}_l \mathbf{H}_l^\dagger)] \\ &= \max_{\mathbf{Q}_l} \mathbb{E}_{\tilde{\mathbf{H}}_l}[W \log_2 \det(\mathbf{I} + \rho_l \alpha_l \tilde{\mathbf{H}}_l \mathbf{Q}_l \tilde{\mathbf{H}}_l^\dagger)], \end{aligned} \tag{6.3.30}$$

where $\tilde{\mathbf{H}}_l$ is the submatrix selected from $\mathbf{H}_l$'s first z_l columns. Now, it can be seen that we have transformed the problem in (6.3.28) into a form exactly the same as that of (6.2.3). The only difference is that in (6.3.30), the matrix in the determinant is of rank-z_l instead of rank-d_l. Since (6.2.4) is a tight approximation of (6.2.3), it follows that the replacing d_l with z_l in (6.2.4) (i.e., (6.3.27)) is also a tight approximation for (6.3.30). This completes the proof. □

6.3.4 Performance Comparison between OBIC and the H-S Model

In Remark 6.3.1, we pointed out that the CiM model in [53] can only achieve a portion of the DoF region achieved under our proposed OBIC scheme. In this subsection, we provide an in-depth analysis.

For the CiM scheme in [53], two sets of integer variables are defined: For every pair of mutually-interfered links i and j in time slot t, $\theta_{ij}(t)$ represents the number of DoFs assigned by $t(i)$ to null its interference at $r(j)$, and $\vartheta_{ij}(t)$ represents the number of DoFs assigned by $r(j)$ to suppress the interference coming from $t(i)$. The H-S link layer mode is given as follows (in this chapter's notation):

$$\sum_{l\in\mathcal{L}_i^{\mathrm{Out}}} z_l(t) + \sum_{l\in\mathcal{I}_i^{+}} \theta_{il}(t) \le A_{t(i)}, \tag{6.3.31}$$

$$\sum_{l\in\mathcal{L}_i^{\mathrm{In}}} z_l(t) + \sum_{l\in\mathcal{I}_i^{-}} \vartheta_{il}(t) \le A_{r(i)}, \tag{6.3.32}$$

$$z_i(t) \le \vartheta_{ij}(t) + \alpha_{t(i)}(1 - y_i(t)), \tag{6.3.33}$$

$$z_j(t) \le \theta_{ij}(t) + \beta_{r(j)}(1 - y_j(t)). \tag{6.3.34}$$

Now, let us compare the performance using the 2-link example shown in Fig. 6.3. Clearly, when both links i and j are active, i.e., $y_i(t) = y_j(t) = 1$, we have from (6.3.33) and (6.3.34) that $z_i(t) \le \vartheta_{ij}(t)$ and $z_j(t) \le \theta_{ij}(t)$. In Section 6.3.1, we can see that $z_i(t) = 3$ and $z_j(t) = 2$ can be scheduled without causing any mutual interference. However, $z_i(t) = 3$ and $z_j(t) = 2$ clearly violates (6.3.31)-(6.3.34). This is because $z_i(t) = 3 \Rightarrow \theta_{ij} = 0$ (by (6.3.31)) and $\theta_{ij} = 0 \Rightarrow z_j(t) = 0$ (by (6.3.34)), a contradiction to $z_j(t) = 2$. Thus, we can see that OBIC can schedule more data streams than CiM.

In fact, for the simple 2-link example in Fig. 6.3, we can derive the DoF region under the H-S model as follows. First, note that there is only one interference going from link i to link j. Thus, when both links are active, we can simplify (6.3.31)-(6.3.34) as: $z_i + \theta_{ij} \le 3$, $z_j + \vartheta_{ij} \le 5$, and $z_j \le \theta_{ij}$. Since $z_j \le \theta_{ij}$ is tighter than $z_j + \vartheta_{ij} \le 5$ (because $\theta_{ij} \le 3$), the latter can be ignored. As a result, we have DoF region as $z_i + z_j \le z_i + \theta_{ij} \le 3$. We plot this DoF region (as squares) in Fig. 6.7. It can be seen that the H-S model can only achieve a portion of the entire DoF region.

To formally state the superior performance of OBIC, we have the following fact.

Fact 6.3.3. *The achievable DoF region $\mathcal{C}_{\rm OBIC}$ contains the achievable DoF region $\mathcal{C}_{\rm HS}$ achieved by the H-S model, i.e., $\mathcal{C}_{\rm HS} \subseteq \mathcal{C}_{\rm OBIC}$.*

Proof. Recall that in the OBIC scheme, the number of data streams and the number of DoFs used for interference mitigation satisfy the following constraints:

$$\sum_{l\in\mathcal{L}_i^{\rm Out}} z_l(t) + \sum_{j\in\mathcal{I}_i} \pi_{ji}(t) \sum_{\substack{l:{\rm Rx}(l)=j\\ {\rm Tx}(l)\neq i}} z_l(t) \leq A_i + (1-g_i(t))M, \tag{6.3.35}$$

$$\sum_{l\in\mathcal{L}_i^{\rm In}} z_l(t) + \sum_{j\in\mathcal{I}_i} \pi_{ji}(t) \sum_{\substack{l:{\rm Tx}(l)=j\\ {\rm Rx}(l)\neq i}} z_l(t) \leq A_i + (1-h_i(t))M. \tag{6.3.36}$$

On the other hand, the link layer constraints for transmissions and interference mitigation in the H-S model can be re-written as:

$$\sum_{l\in\mathcal{L}_i^{\rm Out}} z_l(t) + \sum_{j\in\mathcal{I}_i} \sum_{\substack{l:{\rm Rx}(l)=j\\ {\rm Tx}(l)\neq i}} z_l(t) \leq A_i + (1-g_i(t))M, \tag{6.3.37}$$

$$\sum_{l\in\mathcal{L}_i^{\rm Out}} z_l(t) + \sum_{j\in\mathcal{I}_i} \sum_{\substack{l:{\rm Rx}(l)=j\\ {\rm Tx}(l)\neq i}} z_l(t) \leq A_i + (1-g_i(t))M. \tag{6.3.38}$$

It can see that the RHS are identical. However, in the OBIC scheme, the number of terms in the summation for interference mitigation for each node is strictly less than that in the H-S model except for the last node to be scheduled in the OBIC scheme. This is because for each node in the network, it only has to perform interference mitigation for the nodes that are scheduled earlier. Mathematically, this can be reflected by the fact that only some of γ-variables in the summation are equal to one.

By contrast, in the H-S model, each node has to perform interference mitigation task for *all* nodes within its interference range. As a result, the number of terms in the summation for interference mitigation is strictly larger than that in the OBIC

scheme. Mathematically, this is equivalent to setting all π-variables equal to 1 in (6.3.35) and (6.3.36).

Since more DoF resources are used for interference mitigation in the H-S model, we have that the number of DoF used for data transmissions is strictly less than that in the OBIC scheme, thus resulting in a strictly larger DoFs region. □

6.4 Application in Multi-hop Ad Hoc Networks

In Sections 6.2 and 6.3, we have developed two models for the physical layer and the link layer in multi-hop MIMO ad hoc networks, respectively. In this section, we will show how to apply them for cross-layer optimization in multi-hop MIMO ad hoc networks. We consider a generic utility maximization problem involving a set of sessions, $\mathcal{F}$, in an ad hoc network. Denote $\mathrm{src}(f)$ and $\mathrm{dst}(f)$ the source and destination nodes of session $f \in \mathcal{F}$, respectively. Denote $r(f)$ the rate of session f and $r_l(f)$ the amount of rate on link l that is attributed to session $f \in \mathcal{F}$, respectively. Denote $C_l(t)$ the capacity of link l in time-slot t. Here, $C_l(t)$ is calculated by using the simple MIMO physical layer model proposed in (6.2.4). Note that in different time slots, $C_l(t)$ may vary due to different scheduling ($z_l(t)$) and different power allocation ($\alpha_l(t)$) decisions. Thus, over a time frame containing T time slots, the average capacity of link l can be computed as $\frac{1}{T}\sum_{t=1}^{T} C_l(t)$. Further, for stability, the flow rate on each link l should not exceed its average link capacity to avoid severe delay or even packet loss. Thus, we have the following constraints on the flow rates:

$$\sum_{f\in\mathcal{F}} r_l(f) \le \frac{1}{T}\sum_{t=1}^{T} C_l(t), \quad \forall l. \tag{6.4.1}$$

At the network layer, different routing schemes can be adopted. But regardless of specific routing schemes, the flow balance constraints must hold at each node $i \in \mathcal{N}$

in the network.

$$\sum_{l\in\mathcal{L}_i^{\mathrm{Out}}} r_l(f) - \sum_{l\in\mathcal{L}_i^{\mathrm{In}}} r_l(f) = r(f), \text{ if } i = \mathrm{src}(f), \tag{6.4.2}$$

$$\sum_{l\in\mathcal{L}_i^{\mathrm{Out}}} r_l(f) = \sum_{l\in\mathcal{L}_i^{\mathrm{In}}} r_l(f), \text{ if } i \neq \mathrm{src}(f), \mathrm{dst}(f), \tag{6.4.3}$$

$$\sum_{l\in\mathcal{L}_i^{\mathrm{In}}} r_l(f) - \sum_{l\in\mathcal{L}_i^{\mathrm{Out}}} r_l(f) = r(f), \text{ if } i = \mathrm{dst}(f). \tag{6.4.4}$$

It can be easily verified that if (6.4.2) and (6.4.3) are satisfied, then (6.4.4) is automatically satisfied. As a result, there is not necessary to list (6.4.4) in a formulation once we have both (6.4.2) and (6.4.3).

When a node is transmitting simultaneously on more than one outgoing links, it is necessary to consider power allocation among $\mathcal{L}_i^{\mathrm{Out}}$ at node i. Recall that $\alpha_l(t) \in [0, 1]$ represent a fraction of transmit power allocated onto link l in time-slot t. Then, for each node i in time-slot t, we have

$$\sum_{l\in\mathcal{L}_i^{\mathrm{Out}}} \alpha_l(t) \leq g_n(t), \quad \forall n, t. \tag{6.4.5}$$

The constraint in (6.4.5) ensures that the sum of transmit power of all outgoing links at node i does not exceed the transmit power limit. In the case when node i is not in transmission mode, then $g_i(t) = 0$ and $\alpha_l(t) = 0$ for all $l \in \mathcal{L}_i^{\mathrm{Out}}$.

Consider an objective function for each session, $u\big(r(f)\big)$, which we assume is concave. Then a general MIMO network utility maximization (MIMO–NUM) problem

can be formulated as follows.

MIMO–NUM

$$\max \quad \sum_{f=1}^{F} u\big(r(f)\big)$$

s.t. Network layer flow-balance routing constraints in (6.4.2) and (6.4.3);
Coupling constraints between network layer and link layer in (6.4.1);
OBIC based link layer modeling constraints
in (6.3.14), (6.3.15), (6.3.16), (6.3.2), (6.3.13) and (6.3.19)–(6.3.26);
Simplified MIMO physical layer model in (6.2.4) and (6.4.5).

Two remarks for the MIMO–NUM problem are in order. 1) *Tractability.* Recall that a MIMO cross-layer optimization involves many matrix variables in the capacity calculation and ZFBF scheduling, making network level research quite challenging. With our simple models, matrix variables no longer appear in MIMO–NUM problem, which significantly simplifies formulation and reduce computational complexity. 2) *Solvability.* By using our simple models, MIMO–NUM problem is reduced to a similar mathematical form as in NUM problems for single-antenna ad hoc networks. Note that although OBIC part is unique, it is of linear form and can be easily handled mathematically. This suggests that new solutions to MIMO–NUM problems may be developed by drawing upon the rich experiences gained for single-antenna ad hoc networks.

Next, we use some numerical examples to illustrate how our proposed physical layer and link layer models can be used to solve cross-layer optimization problems for multi-hop MIMO ad hoc networks. To show the details in our proposed models, we first study a six-node network as shown in Fig. 6.8. In this network, six nodes are distributed in a square region of 1000m $\times$ 1000m. Each node is equipped with four antennas and the maximum power for each node is 100 mW. The channel bandwidth is 20 MHz. The path-loss index is set to 3.5. There are two sessions in the network:

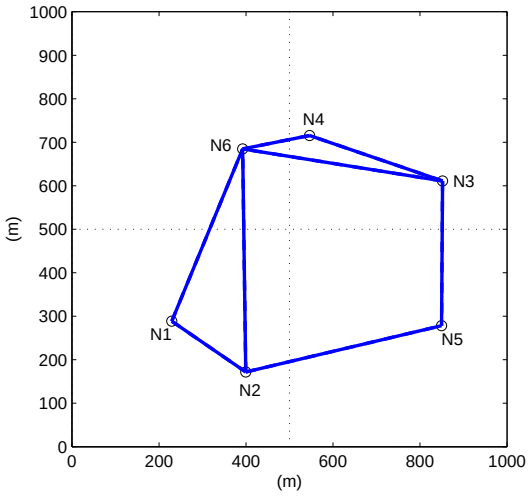


Figure 6.8: A six-node MIMO ad hoc network.

N2 to N4 and N5 to N6. The optimization objective is to maximize the sum of these two end-to-end session rates. Suppose that each time frame consists of 3 time slots (i.e., $T = 3$). For each session in the network, multi-hop multi-path routing is employed at the network layer. With these settings, we can define the MIMO-NUM problem explicitly.

We use CPLEX integer solver combined with LINDO nonlinear solver to solve this problem. The optimal routing paths and scheduling decisions for session N2 $\rightarrow$ N4 and N5 $\rightarrow$ N6 are plotted in Fig. 6.9 and Fig. 6.10, respectively. In Fig. 6.9 and Fig. 6.10, the $\bigstar$ marker and the $\blacksquare$ marker denote the source node and the destination node of the session, respectively.

It can be seen that the optimal routing solution for session N2 $\rightarrow$ N4 is single-path and of three-hops (N2 $\rightarrow$ N1 $\rightarrow$ N6 $\rightarrow$ N4), while the optimal routing solution for session N5 $\rightarrow$ N6 is multi-hop and multi-path (N5 $\rightarrow$ N2 $\rightarrow$ N1 $\rightarrow$ N6 $\rightarrow$ N4; N5 $\rightarrow$ N3 $\rightarrow$ N6; and N5 $\rightarrow$ N3 $\rightarrow$ N4 $\rightarrow$ N6). The optimal session rates for N2 $\rightarrow$ N4

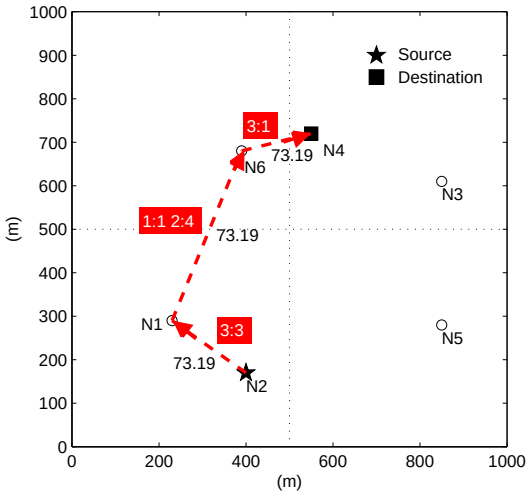


Figure 6.9: The optimal end-to-end session rate (in Mbps), routing, and scheduling solutions for session N2 → N4.

and N5 → N6 and the optimal sum-rate are 73.19 Mbps, 212.65 Mbps, and 285.84 Mbps, respectively. Also shown in Fig. 6.9 and Fig. 6.10 are the optimal scheduling decisions. Take session N5 → N6 as an example (Fig. 6.10), the shaded box next to the link from N1 to N6 contains "1 : 1 2 : 4", which means that in time slots 1 and 2, there are 1 and 4 data streams scheduled on this link, respectively. In time slot 3, however, link (N1→N6) is not transmitting. The optimal scheduling order for each node in each time slot is shown in Table 6.3. In this table, each column gives the node ordering for scheduling in a given time slot of the frame. For example, in the first time slot, the optimal ordering of the nodes is N3 → N5 → N4 → N6 → N1 → N2.

At the physical layer, the power allocation solutions for sessions N2 → N4 and N5 → N6 are plotted in Fig. 6.11 and Fig. 6.12, respectively. Take session N5 → N6 as an example (Fig. 6.12), the shaded box next to the link (N1→N6) contains "1 : 1 2 : 1",

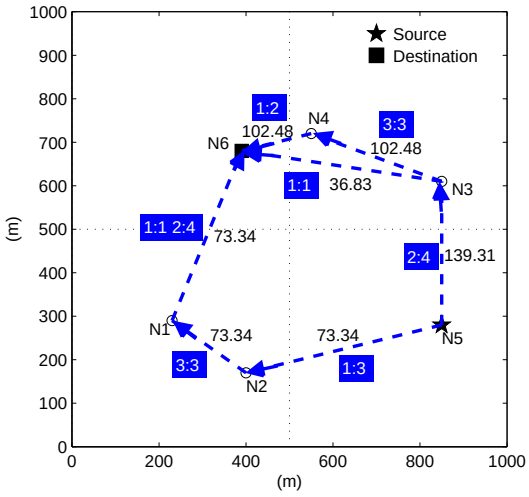


Figure 6.10: The optimal end-to-end session rate (in Mbps), routing, and scheduling solutions for session N5 → N6.

Table 6.3: Optimal node ordering in each time slot of a frame for the six-node network example in Fig. 6.8.

	Time Slot 1	Time Slot 2	Time Slot 3
1st	N3	N4	N3
2nd	N5	N3	N5
3rd	N4	N1	N4
4th	N6	N6	N6
5th	N1	N2	N1
6th	N2	N5	N2

which means that in time slots 1 and 2, N1 is transmitting at full power ($\alpha = 1$). In time slot 3, however, no transmission power is allocated to N1. Note that since session N5 → N6 uses multi-hop multi-path routing, some nodes along the paths may split its transmission power. Indeed, it can be seen from Fig. 6.12 that in time slot 3, N3 splits its transmission power to links (N3→N6) and (N3→N4), and the optimal power allocation in this time slot is 0.37 and 0.63, respectively. We also observe from Fig. 6.11 and Fig. 6.12 that even if a node is not simultaneously transmitting

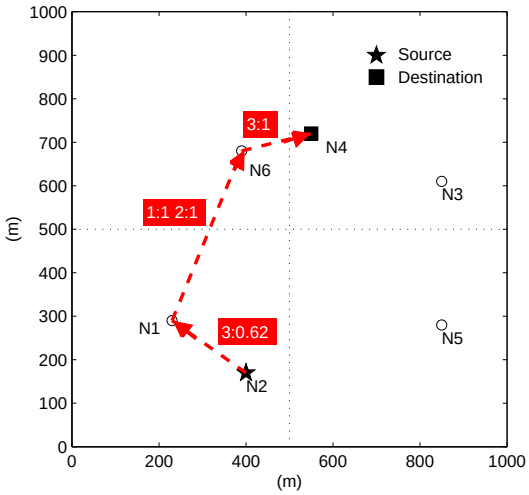


Figure 6.11: The optimal power allocation solution for session N2 → N4.

to multiple neighboring nodes, it does not necessarily transmit at its full power. For example, it can be seen both in Fig. 6.11 and Fig. 6.12 that N2 only allocates 0.62 of its full power to transmit to N1 in time slot 3 ($\alpha_{(\mathrm{N2}\rightarrow\mathrm{N1})}(3) = 0.62$). This is because the path passing through N1 is bottlenecked by the weaker link (N1→N6). Therefore, N2 does not need to transmit at its full power as doing so would not increase the end-to-end session rates. We can thus see that after optimization on power allocation, we not only maximize the network sum-rate, but also conserve energy for those end nodes of non-bottleneck links. With the values of α and z in each time slot, we can then compute the MIMO link capacity of each link based on the simple physical layer model in (6.2.4). The value of each link's capacity for the six-node example is shown in Table 6.4.

As another example to illustrate the solvability of our MIMO–NUM formulation, we consider a larger MIMO ad hoc network consisting of 50 nodes that are uniformly distributed in a square region of 1500m × 1500m (see Fig. 6.13). We assume that

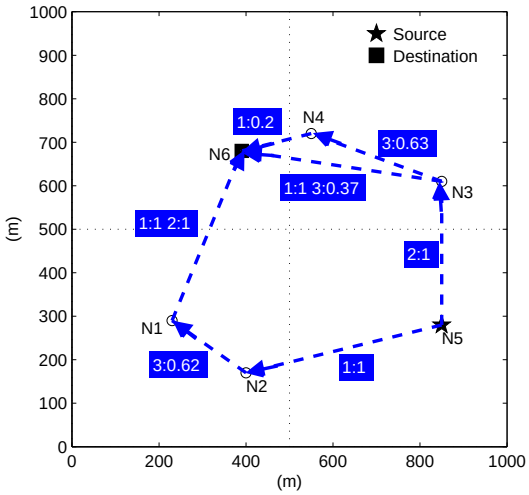


Figure 6.12: The optimal power allocation solution for session N5 → N6.

Table 6.4: Links' capacities for the six-node network in Fig. 6.8 (in Mbps).

Link	Capacity	Link	Capacity
(N1→N2)	0	(N2→N1)	146.53
(N1→N6)	146.53	(N6→N1)	0
(N2→N5)	0	(N5→N2)	80.61
(N2→N6)	0	(N6→N2)	0
(N3→N4)	102.48	(N4→N3)	0
(N3→N5)	0	(N5→N3)	139.31
(N3→N6)	36.83	(N6→N3)	0
(N4→N6)	102.48	(N6→N4)	73.19

minimum-hop routing is employed at the network layer. Each node in the network is equipped with 4 antennas and the maximum power for each node is 100 mW. The channel bandwidth is 20 MHz. The path-loss index is 3.5. There are 5 sessions in the network: N26 to N19, N44 to N18, N24 to N15, N48 to N2, and N9 to N32, respectively. The objective is to maximize the sum of the end-to-end session rates, i.e., $u\big(r(f)\big) = r(f)$. Suppose that there are 4 time slots in each time frame, i.e., $T = 4$. Given these parameters and network settings, the MIMO–NUM problem can be completely specified.

Again, we use CPLEX and LINDO solver to obtain an optimal solution. The

Table 6.5: Optimal node ordering in each time slot of a frame.

	Time Slot 1	Time Slot 2	Time Slot 3	Time Slot 4
1st	N19	N48	N24	N24
2nd	N18	N27	N22	N32
3rd	N44	N32	N44	N26
4th	N15	N15	N9	N48
5th	N26	N9	N15	N18
6th	N22	N19	N27	N24
7th	N5	N44	N32	N9
8th	N48	N26	N26	N22
9th	N6	N18	N19	N5
10th	N27	N2	N18	N6
11th	N24	N24	N6	N15
12th	N32	N22	N5	N19
13th	N3	N5	N48	N2
14th	N1	N3	N1	N27
15th	N9	N1	N3	N1
16th	N2	N6	N2	N3

optimal scheduling ordering for each node in each time slot is listed in Table 6.5. For example, in the first time slot, the optimal ordering of the nodes is N19 $\rightarrow$ N18 $\rightarrow \cdots \rightarrow$ N2. Fig. 6.14 shows the routing paths for each session and optimal scheduling solution (shown in shaded boxes). As an example, the shaded box next to the link from N6 to N18 contains "$1:1 \quad 2:1 \quad 3:2$", which means that in time slots 1, 2, 3, there are 1, 1, and 2 streams on this link, respectively. In time slot 4, the link is not transmitting. Based on the number of streams, the simple physical layer model (6.2.4), and the constraint in (6.4.1), the optimal session rates are found to be (in Mbps): 60.4 for N26 $\rightarrow$ N19, 151 for N44 $\rightarrow$ N18, 102 for N24 $\rightarrow$ N15, 36.6 for N48 $\rightarrow$ N2, and 57.2 for N9 $\rightarrow$ N32.

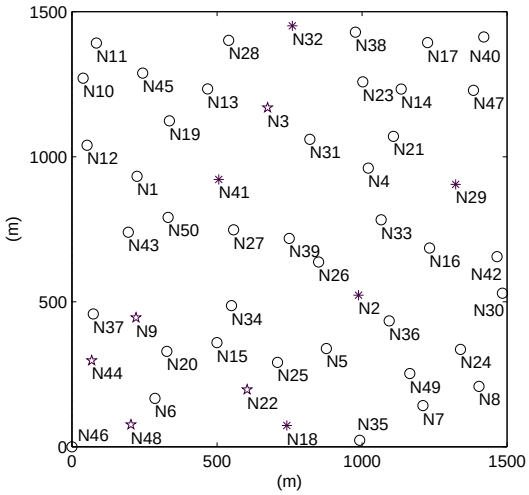


Figure 6.13: A 50-node 5-session MIMO-based ad hoc network.

6.5 Chapter Summary

Existing models for MIMO suffer from either intractability or inaccuracy when they are employed for multi-hop ad hoc network. In this chapter, we proposed a tractable and accurate model for MIMO that is amenable for cross-layer analysis in multi-hop ad hoc networks. Our contributions included a model at the physical layer and a model at the link layer. At the physical layer, we proposed a simple model to compute MIMO channel capacity that captures the essence of spatial multiplexing and transmit power limit without involving complex matrix operations and the water-filling algorithm. We proved that the approximation gap in this physical layer model is negligible. At the link layer, we proposed a scheduling scheme called OBIC that is based on ZFBF interference mitigation. The proposed OBIC scheme cuts through the complexity associated with beamforming designs in a multi-hop ad hoc network by using simple numeric computation on matrix dimensions. As a result, we can characterize the link layer scheduling performance without entangling with beamforming details. By

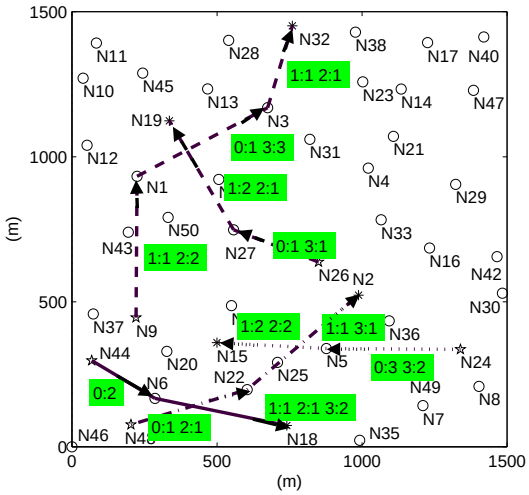


Figure 6.14: Scheduling result on each link.

applying the proposed cross-layer model to a general network utility maximization problem, we validate its efficacy in practice. The results in this chapter offer an important analytical tool to fully exploit the potential of MIMO in multi-hop ad hoc networks.

Chapter 7

Summary and Future Work

7.1 Summary

Compared with simple point-to-point MIMO channels, the network design and performance optimization for MIMO-based wireless networks is much more challenging and there remain a lot of open problems. Due to the complex characteristics of MIMO physical layer technology, it is not only desirable but also necessary to consider models and constraints at multiple layers (e.g., physical, link, and network) when exploring performance limits of MIMO-based wireless networks. In this dissertation, our focus is on developing novel algorithmic design and optimization techniques that provide optimal or near-optimal solutions. The main contributions of this dissertation are summarized as follows.

- In Chapter 2, we investigated the weighted proportional fair scheduling problem for MIMO broadcast channels (MIMO-BC). To handle the difficulty in determining the optimal DPC encoding order, we derived a set of optimality conditions that the optimal decoding order must satisfy. Based on these optimality conditions, we designed an efficient algorithm called iterative gradient sorting (IGS) to determine the optimal decoding order by iteratively sorting the gradient entries and moving

across corner points. We also showed that this method can be geometrically interpreted as sequential gradient projections. For the power allocation subproblem, we again proposed to use CGP technique to determine a local optimal solution. By comparing this solution with an upper bound of the optimal WPF value, we showed the tightness of the CGP solution to the true optimum.

- In Chapter 3, our main contributions are the mathematical developments of the solution procedure to solve the problem and its convergence speedup techniques. Specifically, we derived tight upper and lower bounds for each potential partitioning variable used for the problem. Each nonlinear term was relaxed with a set of linear constraints based on the bounds we develop to generate a higher dimensional upper-bounding problem. We also utilized a polyhedral outer approximation method to accurately approximate the logarithmic function. During each iteration of the branch-and-bound procedure, we proposed a variable selection policy based not only on the relaxation error, but also on the relative significance of the variables in our problem. Our proposed method guarantees the finding of a global optimal solution.

- In Chapter 4, we considered the problem of jointly optimizing power and bandwidth allocation at each node and multihop/multipath routing in a multi-hop MIMO ad hoc networks where links operate in orthogonal channels. For this cross-layer optimization problem, we showed that it has some special structure which allows us to decompose the original problem into a set of subproblems in its dual domain. Specifically, our solution procedure first decouples the dual problem into a network layer subproblem and a link-physical layer subproblem. For the dual problem, we proposed two strategies, i.e., a cutting-plane method based on outer-linearization

and the subgradient-based scheme. Our proposed cutting-plane method allows an easy recovery of primal feasible and optimal solutions. Finally, based on the subgradient-based approach, we design a distributed algorithm that achieves the same optimal solution as that of the centralized algorithm. We showed that the excess link capacity of each link can be used for message exchange in our distributed algorithm.

- In Chapter 5, we relaxed the orthogonal assumption in the previous work and studied the performance optimization with dirty paper coding (DPC). It turns out that this optimization problem is a challenging non-convex problem. To address this difficulty, we transformed the original problem to an equivalent problem by exploiting the uplink-downlink duality. For the transformed problem, we again proposed a solution procedure that integrates Lagrangian dual decomposition, conjugate gradient projection based on matrix differential calculus, and cutting-plane methods.

- In Chapter 6, we proposed a tractable and accurate model for MIMO that is amenable for cross-layer analysis in multi-hop setting. Our contributions included a model at the physical layer and a model at the link layer. At the physical layer, we proposed a simple model to compute MIMO channel capacity that captures the essence of spatial multiplexing and transmit power limit without involving complex matrix operations and the water-filling algorithm. We proved that the approximation gap in this physical layer model is negligible. At the link layer, we proposed a scheduling scheme called OBIC that is based on ZFBF interference mitigation. The proposed OBIC scheme cuts through the complexity associated with beamforming designs in a multi-hop network by using simple algebraic computation. This allows us to explore the link layer scheduling performance without entangling

with beamforming details. By applying the proposed cross-layer model to a general NUM problem, we validated its efficacy in practice. The results in this chapter offer an important analytical tool to fully exploit the potential of MIMO in multi-hop networks.

7.2 Future Research Direction

Due to the high complexity in MIMO technology, there remain many open problems in applying MIMO in ad hoc networks. In particular, the following open problems will form my future research directions.

- *Multi-criteria Optimization for MIMO Networks:* In addition to throughput/capacity, there are a number of other performance metrics that are also important. For example, maximizing energy efficiency, optimizing the number of antennas (hardware cost), minimizing delay, and maintaining network stability are all important aspects of a network system and should all be considered. To date, most of the state-of-the-art network optimization algorithms are limited to a single-criterion (see, e.g., [1]). But for real-world network systems, multi-criteria optimization is necessary. The technical challenges associated with multi-criteria optimization are significant because: 1) multiple objectives are likely to be conflicting with each other, meaning that we need to determine a pareto-efficient solution set rather than a single optimal solution; 2) problem formulations are likely non-convex or NP-hard. So far, results on multi-criteria network optimization are extremely limited and the few available results remain heuristic [6, 79]. We plan to explore recent advances in this area [36] and study some interesting multi-criteria optimization problems for MIMO ad hoc networks based on our simple physical and link layer models.

- *Approximation Algorithms with Performance Guarantee:* In [88], we applied a perturbation function and dual-based approach to solve optimization problem for multi-carrier MIMO ad hoc networks. In other problems, the perturbation function may not be concave and there exists a non-zero duality gap. In such cases, we propose to design approximation (or near-optimal) solutions with a *provable* performance guarantee. In this project, we plan to explore two such types of approximation algorithms: constant factor and $(1-\epsilon)$-optimal solutions. In particular, the discretization technique with $(1-\epsilon)$-optimal performance guarantee proposed in [136] appears promising to address the nonlinear non-convex problem here. Further, the "traveling salesman"-like structure associated with node ordering constraint at the link layer in problem (6.4) suggests that some advanced techniques in the area of approximation algorithms (e.g., primal-dual-based approximation [62]) may be promising. We will fully study these approximation techniques for MIMO ad hoc networks.

- *Network Optimization under Partial CSI* Our work in this dissertation has been developed by assuming perfect channel state information (CSI). In environments with noisy feedback or fast fading channels, only limited or partial knowledge of CSI may be available. In this case, a problem of interest is to optimize network performance with only channel distribution information (CDI) or channel statistics (e.g., mean and variance). One possible approach we are considering is to exploit techniques from stochastic approximation [81, 100]. In our recent work in [88], we applied stochastic approximation techniques to develop an *online adaptive algorithm* under a partial CSI setting and showed that the online algorithm achieves the same optimal solution *almost surely* (with probability 1) as that under a full

CSI case. We will extend this approach to address a broader class of problems with partial CSI.

- *Asymptotic Analysis and Capacity Scaling Laws:* In addition to performance optimization for finite-sized network instances, another important line of research is to develop asymptotic results (e.g., [52]) to reveal scaling laws when network size becomes large. Such analysis offers a theoretical understanding of on the trends (or asymptotics) when network size grows. Note that the tractability of our proposed model offers a good start to characterize throughput performance for MIMO ad hoc networks, which is otherwise impossible under traditional MIMO models. This paves the way for us to incorporate advanced techniques (such as percolation [42], mobility [51], and hierarchical cooperation [117]). One interesting problem that we aim to explore is whether or not the long-sought "super-linear" scaling law is indeed achievable in MIMO ad hoc networks.

Appendix A

Some Relevant Results for MIMO Networks: A Background

The field of MIMO communications has flourished for more than a decade. An attempt to review this rich topic in one chapter is doomed to omit some significant results in this area. Rather than scanning the entire field, in this appendix, we focus on some key information-theoretic MIMO results that are most relevant to this dissertation. These results will play an important role throughout this dissertation. In Section A.1, we review the fundamental information-theoretic property of point-to-point MIMO channel. Then, in Section A.2, we review some results in two important MIMO networking components, namely MIMO broadcast channels (MIMO-BC) and MIMO multiple-access channels (MIMO-MAC). Uplink-downlink duality will be reviewed in Section A.2.3.

A.1 Point-to-Point MIMO Channel

The fundamental idea in MIMO is the so-called *space-time processing* technique [13, 45], which is inherent in the use of multiple spatially-distributed antennas. Popular space-time processing techniques include *spatial multiplexing* and *space-time coding.* Under space-time processing, the time dimension is coupled with the spatial dimension in order to achieve *spatial multiplexing gain* or to achieve transmit-receive *diversity gain.* Diversity gain provided by space-time coding [4, 142, 143] mainly aims to combat multi-path fading and improve transmission reliability [45]. On the other hand, through spatial multiplexing gain (provided by exploiting the rick scattering environment in multi-antenna systems), multiple data streams can be transmit simultaneously to boost the transmit data rate of the system. However, there exists a fundamental trade-off between these two types of gains [166]: One cannot achieve one type of the gains without hurting the other.

Among these types of gains, diversity gain contributes to a gain in SNR. As a result, capacity increase due to diversity gain follows logarithmic law. On the other hand, multiplexing gain from spatial multiplexing directly contributes to a *linear* increase of link capacity. Due to spatial multiplexing's important role in capacity increase, we give it a brief review in this section from information theoretic perspective.

The basic idea of spatial multiplexing is to exploit a rich scattering environment with multiple transmit and receive antennas. In contrast to a conventional antenna system that is characterized by a scalar channel gain, the channel under a MIMO system is characterized by a *matrix,* within which the element on the j-th row and i-th column represents the channel gain between the i-th antenna at the transmitter and the j-th antenna at the receiver, as shown in Fig. A.1.

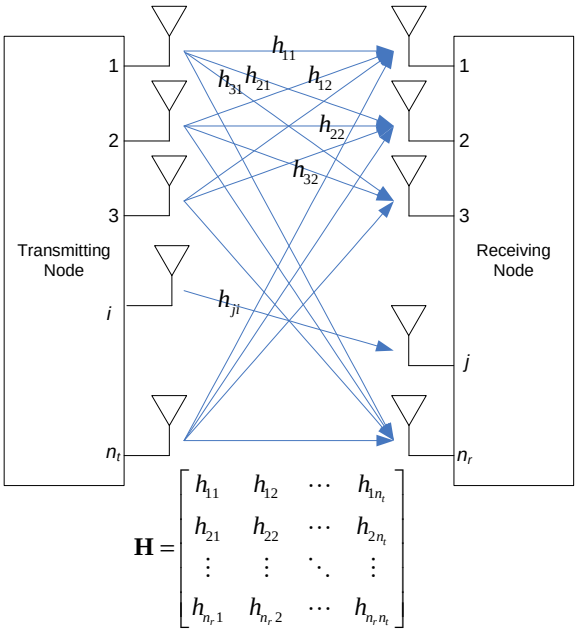


Figure A.1: A matrix representation of spatial channels for a MIMO link.

A point-to-point MIMO channel can be described by

$$\mathbf{y} = \sqrt{\rho}\mathbf{H}\mathbf{x} + \mathbf{n}, \tag{A.1.1}$$

where $\mathbf{x} \in \mathbb{C}^{n_t}$, $\mathbf{y} \in \mathbb{C}^{n_r}$ and $\mathbf{n} \in \mathbb{C}^{n_r}$ denote the transmitted signal, received signal, and white Gaussian noise, respectively. As is standard, there is a maximum transmit power limit P at the transmitting node.

In the literature, the channel gain matrix $\mathbf{H} \in \mathbb{C}^{n_r \times n_t}$ is usually assumed to be constant during a symbol's transmission and and varies independently from symbol to symbol. In $\mathbf{H}$, $[\mathbf{H}]_{ij}$ is the channel gain from transmit antenna j to receive antenna i. It is usually assumed that each entry in $\mathbf{H}$ is i.i.d. Gaussian distributed with zero mean and unit variance. Further, $\mathbf{H}$ is assumed to satisfy the circularly symmetric property [146]. For simplicity, in this section, we only review the case where $\mathbf{H}$ is constant. Further capacity results on fading MIMO channels can be found in [146].

The aforementioned MIMO channel is also referred to as a vector Gaussian channel. The capacity of such a vector Gaussian can be found by decomposing the vector channel into a set of parallel and independent scalar Gaussian subchannels as follows. From basic matrix theory, matrix $\mathbf{H}$ has the so-called *singular value decomposition* (SVD):

$$\mathbf{H} = \mathbf{U}\mathbf{\Lambda}\mathbf{V}^{\dagger}, \tag{A.1.2}$$

where $\mathbf{U} \in \mathbb{C}^{n_r \times n_r}$ and $\mathbf{V} \in \mathbb{C}^{n_t \times n_t}$ are unitary matrices and $\mathbf{\Lambda} \in \mathbb{C}^{n_r \times n_t}$ is a rectangular matrix whose diagonal elements are non-negative real numbers and whose off-diagonal elements are zero. Geometrically, singular value decomposition means that every linear transformation $\mathbf{H}$ can be represented as a composition of three operations: a rotation operation, a scaling operation, and another rotation operation.

The diagonal elements $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_{n_{\min}}$ are ordered singular values of the matrix $\mathbf{H}$, where $n_{\min} = \min(n_t, n_r)$. We can rewrite the SVD as

$$\mathbf{H} = \sum_{i=1}^{n_{\min}} \lambda_i \mathbf{u}_i \mathbf{v}_i^{\dagger}, \tag{A.1.3}$$

i.e., the sum of ran-one matrices $\lambda_i \mathbf{u}_i \mathbf{v}_i^{\dagger}$. Thus, it can be seen that the rank of $\mathbf{H}$ is precisely the number of non-zero singular values.

With the SVD in (A.1.2), the vector Gaussian channel becomes

$$\mathbf{y} = \mathbf{U}\mathbf{\Lambda}\mathbf{V}^{\dagger}\mathbf{x} + \mathbf{n}. \tag{A.1.4}$$

Pre-multiplying both sides by $\mathbf{U}^{\dagger}$, we have

$$\mathbf{U}^{\dagger}\mathbf{u} = \mathbf{\Lambda}\mathbf{V}^{\dagger}\mathbf{x} + \mathbf{U}^{\dagger}\mathbf{n} \tag{A.1.5}$$

Now let us define

$$\tilde{\mathbf{x}} = \mathbf{V}^{\dagger}\mathbf{x}, \quad \tilde{\mathbf{y}} = \mathbf{U}^{\dagger}\mathbf{y}, \quad \tilde{\mathbf{n}} = \mathbf{U}^{\dagger}\mathbf{n}, \tag{A.1.6}$$

the vector Gaussian can be rewritten as

$$\tilde{\mathbf{y}} = \mathbf{\Lambda}\tilde{\mathbf{x}} + \tilde{\mathbf{n}}, \tag{A.1.7}$$

where $\tilde{\mathbf{n}}$ has the same distribution as $\mathbf{n}$ since $\mathbf{U}^\dagger$ is a unitary transmission. Thus, it can be seen that by pre-multiplying $\mathbf{U}^\dagger$, we have transformed the original channel into a parallel Gaussian channel

$$\tilde{y}_i = \lambda_i \tilde{x}_i + \tilde{n}_i, \quad i = 1, 2, \ldots, n_{\min}. \tag{A.1.8}$$

Hence, the capacity can be computed by

$$C = \sum_{i=1}^{n_{\min}} \log\left(1 + \frac{P_i^* \lambda_i^2}{N_0}\right), \tag{A.1.9}$$

where $P_1^*, \ldots, P_{n_{\min}}^*$ are the waterfilling power allocations $P_i^* = \left(\mu - \frac{N_0}{\lambda_i^2}\right)_+$, with μ chosen to satisfy the total power constraint $\sum_i P_i^* = P$.

In fact, the roles played by $\mathbf{U}$ and $\mathbf{V}$ can be interpreted as the optimal receive and transmit beamformers that align the transmitted signals to the eigen-directions of the channel matrix $\mathbf{H}$. The transmitter forms $n_{\min}$ beams by using $n_{\min}$ beamformers (the columns in $\mathbf{V}$). Meanwhile, the receiver uses $n_{\min}$ beamformers to reconstruct the transmitted signals. While forming the beams, the transmitter allocates more power to stronger eigen-modes simply using waterfilling.

Alternatively, the capacity of point-to-point Gaussian MIMO channel can be derived using eigenvalue decomposition. Note that the mutual information $I(\mathbf{x}; \mathbf{y})$ can be written as

$$I(\mathbf{x}; \mathbf{y}) = H(\mathbf{y}) - H(\mathbf{y}|\mathbf{x}) = H(\mathbf{y}) - H(\mathbf{n}). \tag{A.1.10}$$

Thus, maximizing $I(\mathbf{x}; \mathbf{y})$ is equivalent to maximizing $H(\mathbf{y})$. Since $\mathbf{x}$ satisfies $\mathbb{E}[\mathbf{x}^\dagger \mathbf{x}] \leq P$, so does $\mathbf{x} - \mathbb{E}[\mathbf{x}]$, we can restrict ourselves to zero-mean $\mathbf{x}$. Moreover, if $\mathbf{x}$ is zero-mean with covariance $\mathbb{E}[\mathbf{x}\mathbf{x}^\dagger] = \mathbf{Q}$, then $\mathbf{y}$ is zero-mean with covariance $\mathbb{E}[\mathbf{y}\mathbf{y}^\dagger] =$

$\mathbf{I} + \mathbf{HQH}^\dagger$. It can be shown that when $\mathbf{y}$ is circularly symmetric complex Gaussian, which is the case when $\mathbf{x}$ is circularly symmetric complex Gaussian, the entropy $H(\mathbf{y})$ is the largest [146]. Thus, we can further restrict ourselves to circularly symmetric complex Gaussian $\mathbf{x}$. In this case, the mutual information is given by

$$I(\mathbf{x};\mathbf{y}) = \log \left| \mathbf{I} + \mathbf{HQH}^\dagger \right|. \tag{A.1.11}$$

It is easy to verify that (A.1.9) and (A.1.11) are equivalent.

A.2 MIMO Broadcast Channel and MIMO Multiple Access Channel

In this dissertation, MIMO Gaussian broadcast channels and MIMO Gaussian multiple access channels will occur frequently. Thus, it is beneficial to quickly review the basic definitions of MIMO Gaussian broadcast and multiple access channels and related topics, such as dirty paper coding, successive decoding, and uplink-downlink duality.

A.2.1 MIMO Broadcast Channel

In network information theory, a communication system where a single transmitter sends independent information to multiple *uncoordinated* receivers is referred to as a broadcast channel. If the transmitter and the receivers in the channel are equipped with multiple antennas, then we refer to this channel as a MIMO broadcast channel (MIMO-BC). In MIMO-BC, the channel gain of each link in the broadcast channel is in matrix form and the noise to each link is a Gaussian random vector. Hence, MIMO-BC is sometimes referred to as "Gaussian vector broadcast channel." Figure A.2 illustrates a K-user Gaussian vector broadcast channel, where independent messages

$W_1, \ldots, W_K$ are jointly encoded by the transmitter, and the receivers are trying to decode $W_1, \ldots, W_K$, respectively.

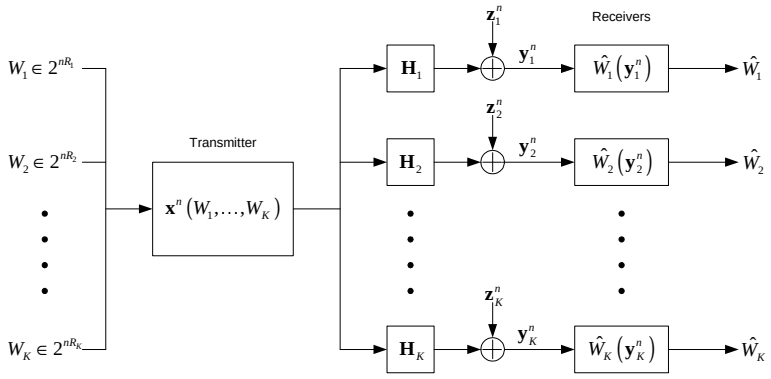
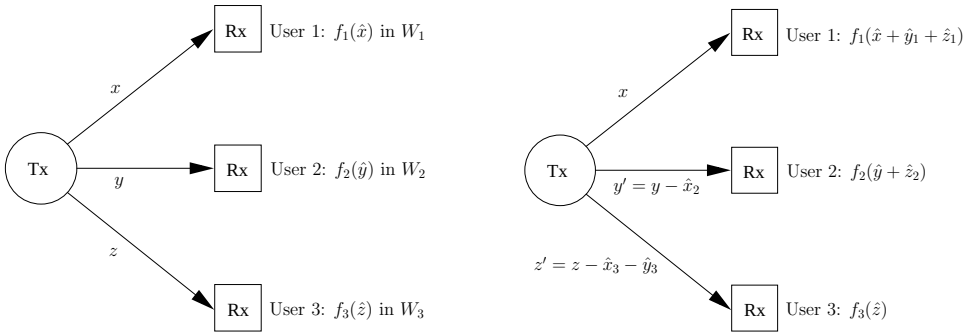


Figure A.2: A MIMO Gaussian broadcast channel.

Since MIMO Gaussian broadcast channel is a multi-user communication system, its capacity is characterized by a capacity region. To define the capacity region more precisely, we introduce a $(n, 2^{nR_1}, \ldots, 2^{nR_K})_{BC}$ codebook for a broadcast channel consists of an encoding function $\mathbf{x}^n(W_1, \ldots, W_K)$ where $W_i \in \{1, \ldots, 2^{nR_i}\}$, $i = 1, 2, \ldots, K$. The decoding function of receiver i is $\hat{W}_i(\mathbf{y}_i^n)$. An error occurs when $\hat{W}_i \neq W_i$. A rate vector $\mathbf{R} = [R_1, \ldots, R_K]^T$ is said to be achievable if there exists a sequence of $(n, 2^{nR_1}, \ldots, 2^{nR_K})_{\mathrm{BC}}$ codebooks for which the average probability of error $P_e \to 0$ as the code length $n \to \infty$. The capacity region of a broadcast channel is defined as the union of all achievable rate vectors [29]. Gaussian vector broadcast channel can be used to model many different types of systems [29].

It has been shown by Weigarten *et al.* that the capacity region of MIMO-BC is achieved by the so-called "dirty paper coding" (DPC). Dirty paper coding (DPC) was first discovered by Costa [27], who showed a rather surprising result that if



(a) Frequency or time division (b) DPC transmission strategy

Figure A.3: A three-user broadcast channel example.

interference is known to the transmitter, DPC achieves the full capacity of a point-to-point Gaussian channel as if the interference does not exist. DPC was first extended to broadcast channels by Caire and Shamai [20], who showed that DPC achieves the sum-rate capacity of 2-user MIMO-BC with single antenna at each receiving node. Subsequently, this result was extended to MIMO-BC with arbitrary number of antennas on both sides of broadcast channels [151, 152, 162]. Finally, it is recently shown by Weigarten *et al.* [154] that DPC does indeed achieve the entire capacity region of broadcast channels.

To begin with, it is beneficial to review the basic idea of DPC in MIMO-BC. As an example, Fig. A.3(a) shows a simple MIMO-BC where there are three uncoordinated users and a single transmitting node, each of which is equipped with multiple antennas. The transmitter is assumed to have full knowledge of channel state information (CSI). Suppose that messages x, y, and z need to be transmitted to user 1, 2, and 3, respectively. Also, suppose that the received signals subject to ambient noise are $\hat{x}$, $\hat{y}$, and $\hat{z}$, and the decoding functions are $f_1(\cdot)$, $f_2(\cdot)$, and $f_3(\cdot)$, respectively. The conventional wisdom is to divide the transmission media into three portions (either in

time slots or frequency sub-bands) W_1, W_2, and W_3, and determine the optimal sub-band bandwidth or time slot duration for transmissions. The benefit of this strategy is that interference can be completely eliminated.

Now, consider the following different strategy as shown in Fig. A.3(b): We jointly encode the messages for all the users in some order and then broadcast the resulting codeword *simultaneously*. Suppose that we pick user 1 to be encoded first, then followed by user 2, and finally user 3. We choose the original codeword x for user 1. Since the transmitter has full CSI, the interference seen by user 2 due to user 1 (denoted by $\hat{x}_2$) is known at the transmitter. So, the transmitter can pre-subtract the interference and encode user 2 as $y' = y - \hat{x}_2$ rather than y itself. As a result, user 2 does not see any interference from the signal intended for user 1. Likewise, after encoding user 2, the interferences seen by user 3 due to user 1 and 2 (denoted by $\hat{x}_3$ and $\hat{y}_3$) are known at the transmitter. Then, the transmitter can pre-subtract the interferences and encode user 3 as $z' = z - \hat{x}_3 - \hat{y}_3$ rather than z itself. Therefore, user 3 does not see any interferences from the signals intended for user 1 and 2. In the end, the transmitter adds all the codewords together and broadcasts the sum to all users simultaneously. From Fig. A.3(b), it can be seen that the received signal at user 1 is $\hat{x} + \hat{y}_1 + \hat{z}_1$, i.e., user 1 will experience the interference from the signals intended for users 2 and 3; the received signal at user 2 is $\hat{y} + \hat{z}_2$, i.e., user 2 only experiences the interference from the signal intended for user 3; and finally, the received signal at user 3 is $\hat{z}$, i.e., user 3 does *not* experience any interference.

This encoding process operates like writing on a dirty paper, hence the name "dirty paper coding." It can be shown that the DPC rate region that allows interference could be much larger than those of time or frequency division schemes [29].

In fact, it has been shown that DPC achieves the full capacity region of a broadcast channel [154]. In addition to information-theoretic study, initial attempts on practical implementations of DPC have also been reported [119, 164, 165].

In Fig. 5.2, let $\mathbf{H}_i \in \mathbb{C}^{n_r \times n_t}$ represent the channel gain matrix from the transmitting node to the i-th receiving node, where n_t and n_r are the number of transmit and receive antennas, respectively. Denote $\mathbf{\Gamma}_i$ the input covariance matrix (i.e., power allocation) for link i. From DPC encoding process and interference relationships among the users, the achievable rate for receiving node i can be computed as [48]

$$R_{\pi(i)}^{\mathrm{BC}} = \log \frac{\left| \mathbf{I} + \mathbf{H}_{\pi(i)} \left(\sum_{j \geq i} \mathbf{\Gamma}_{\pi(j)} \right) \mathbf{H}_{\pi(i)}^{\dagger} \right|}{\left| \mathbf{I} + \mathbf{H}_{\pi(i)} \left(\sum_{j > i} \mathbf{\Gamma}_{\pi(j)} \right) \mathbf{H}_{\pi(i)}^{\dagger} \right|}, \tag{A.2.1}$$

where π denotes an encoding order of the set $\{1, \ldots, K\}$. $\pi(i)$ represents the i-th position in permutation π. It is not hard to see that (A.2.1) is neither a concave nor a convex function of the input covariance matrices $\mathbf{\Gamma}_i$, $i = 1, 2, \ldots, K-1$.

A.2.2 MIMO Multiple Access Channel

In network information theory, a communication system where multiple *uncoordinated* transmitters send independent information to a single receiver is referred to as a multiple access channel. If the transmitter and the receivers in the channel are equipped with multiple antennas, then we refer to this channel as a MIMO multiple access channel (MIMO-MAC). In MIMO-MAC, the channel gain of each link in the multiple access channel is in matrix form and the noise to each link is a Gaussian random vector. Hence, MIMO-MAC is also sometimes referred to as "Gaussian vector multiple access channel". Figure A.4 illustrates a K-user Gaussian vector multiple access channel, where independent messages W_1, ..., W_K, are encoded by transmitters 1 to K, respectively, and the receiver is trying to decode $W_1, \ldots, W_K$.

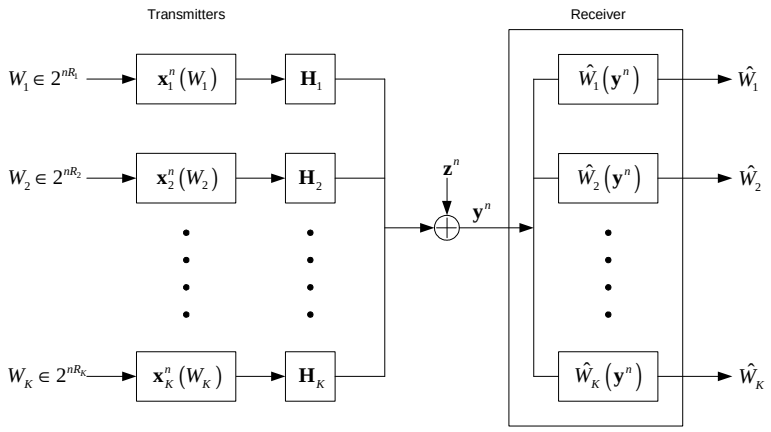


Figure A.4: A MIMO Gaussian multiple access channel.

Similar to MIMO-BC, the capacity of MIMO-MAC is characterized by a capacity region. To define the capacity region more precisely, we define a $(n, 2^{nR_1}, \ldots, 2^{nR_K})_{\text{MAC}}$ codebook for a multiple access channel consists of encoding functions $\mathbf{x}_1^n(W_1)$, ..., $\mathbf{x}_K^n(W_K)$ where $W_i \in \{1, \ldots, 2^{nR_i}\}$, $i = 1, 2, \ldots, K$. The decoding functions at the receiver are $\hat{W}_i(\mathbf{y}_i^n)$, $i = 1, 2, \ldots, K$. An error occurs when $\hat{W}_i \neq W_i$. A rate vector $\mathbf{R} = [R_1, \ldots, R_K]^T$ is said to be achievable if there exists a sequence of $(n, 2^{nR_1}, \ldots, 2^{nR_K})_{\text{MAC}}$ codebooks for which the average probability of error $P_e \to 0$ as the code length $n \to \infty$. The capacity region of a multiple access channel, denoted by $\mathcal{C}_{\text{MAC}}$ is defined as the union of all achievable rate vectors [29].

A.2.3 Uplink-Downlink Duality

There exists a nice duality between MIMO-BC and MIMO-MAC. Uplink-downlink duality relies on the notion of dual MIMO multiple access channel (MIMO-MAC), which is the reverse uplink of a broadcast channel. As shown in Fig A.5, the dual MIMO-MAC of a MIMO-BC can be constructed by changing the receivers in the MIMO-BC into transmitters and changing the transmitter in the MIMO-BC into the

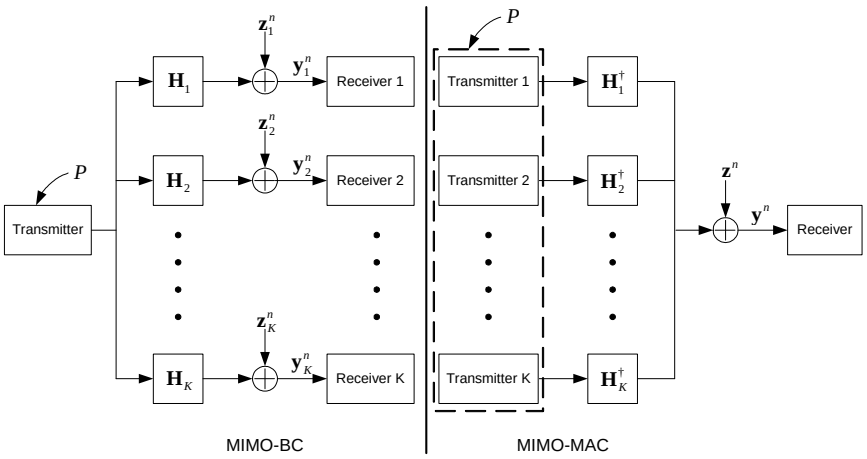


Figure A.5: Uplink-Downlink Duality

receiver. The channel gain matrices in dual MIMO-MAC are the conjugate transpose of the channel gain matrices in MIMO-BC. The maximum sum power in the dual MIMO-MAC is the same maximum power level as in MIMO-BC. We denote its capacity region as $\mathcal{C}_{\mathrm{MAC}}(P, \mathbf{H}^{\dagger})$.

For MIMO-BC and its dual MIMO-MAC, we have the following important result, which states the relationship between the capacity regions of a MIMO-BC and its dual MIMO-MAC.

Theorem A.2.1. *The DPC rate region of a MIMO-BC with maximum power constraint P is equal to the capacity region of the dual MIMO MAC with sum power constraint P, i.e.,*

$$\mathcal{C}_{\mathrm{DPC}}(P, \mathbf{H}) = \mathcal{C}_{\mathrm{MAC}}(P, \mathbf{H}^{\dagger}).$$

The above theorem is usually referred to as "uplink-downlink duality." The proof of this theorem can be arrived in various ways [151, 152, 161]. The most straightforward approach is to show that any MIMO-BC achievable rate vector is also achievable in its dual MIMO-MAC and vice versa. The MAC-to-BC and BC-to-MAC mappings

can be found in [151]. It is also shown in [151] that any rate vector in a MIMO-BC with a particular encoding order can be achieved in its dual MIMO-MAC with the *reversed* successive decoding order. As will be shown later, uplink-downlink duality plays an important role in optimizing MIMO-BC related problems because of the nice convexity of the capacity region of MIMO-MAC.

Bibliography

[1] "Special issue on nonlinear optimization of communication systems," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 8, Aug. 2006.

[2] A. Adya, P. Bahl, J. Padhye, A. Wolman, and L. Zhou, "A multi-radio unification protocol for IEEE 802.11 wireless networks," in *Proc. IEEE International Conference on Broadband Networks*, San Jose, CA, Oct. 5-9, 2004, pp. 344–354.

[3] N. Al-Dhahir, "Single-carrier frequency-domain equalization for space-time block-coded transmissions over frequency-selective fading channels," *IEEE Commun. Lett.*, vol. 5, no. 7, pp. 304–306, Jul. 2001.

[4] S. M. Alamouti, "A simple transmit diversity technique for wireless communications," *IEEE J. Sel. Areas Commun.*, vol. 16, no. 8, pp. 1451–1458, Oct. 1998.

[5] M. Alard and R. Lassalle, "Principles of modulation and channel coding for digital broadcasting for mobile receivers," *EBU Technical Review*, no. 224, pp. 168–190, Aug. 1987.

[6] E. Alba, B. Dorronsoro, F. Luna, A. J. Nebro, P. Bouvry, and L. Hogie, "A cellular multi-objective genetic algorithm for optimal broadcasting strategy in metropolitan MANETs," *Elsevier Computer Communications Journal*, vol. 30, no. 4, pp. 685–697, Feb. 2007.

[7] M. Alicherry, R. Bhatia, and L. Li, "Joint channel assignment and routing for throughput optimization in multi-radio wireless mesh networks," in *Proc. ACM Mobicom*, Cologne, Germany, Aug. 2005, pp. 58–72.

[8] S. Baro, G. Bauch, and A. Hansmann, "Improved codes for space-time trellis-coded modulation," *IEEE Commun. Lett.*, vol. 4, no. 1, pp. 20–22, Jan. 2000.

[9] M. S. Bazaraa, J. J. Jarvis, and H. D. Sherali, *Linear Programming and Network Flows*, 3rd ed. New York: John Wiley & Sons Inc., 2005.

[10] M. S. Bazaraa, H. D. Sherali, and C. M. Shetty, *Nonlinear Programming: Theory and Algorithms*, 3rd ed. New York, NY: John Wiley & Sons Inc., 2006.

[11] M.-G. D. Benedetto, T. Kaiser, A. F. Molisch, I. Oppermann, C. Politano, and D. Procino, Eds., *UWB Communication Systems: A Comprehensive Overview*. New York, NY: Hindawi Publishing Corporation, 2006.

[12] R. Bhatia and L. Li, "Throughput optimization of wireless mesh networks with MIMO links," in *Proc. IEEE INFOCOM*, Anchorage, AK, May 6-12, 2007, pp. 2326–2330.

[13] E. Biglieri, R. Calderbank, A. Constantinides, A. Goldsmith, A. Paulraj, and H. V. Poor, *MIMO Wireless Communications*. Cambridge University Press, Jan. 2007.

[14] J. A. C. Bingham, "Multicarrier modulation for data transmission: An idea whose time has come," *IEEE Commun. Mag.*, vol. 28, no. 5, pp. 5–14, May 1990.

[15] R. S. Blum, "MIMO capacity with interference," *IEEE J. Sel. Areas Commun.*, vol. 21, no. 5, pp. 793–801, Jun. 2003.

[16] H. Bolcskei, M. Borgmann, and A. J. Paulraj, "Impact of the propagation environment on the performance of space-frequency coded MIMO-OFDM," *IEEE J. Sel. Areas Commun.*, vol. 21, no. 3, pp. 427–439, Apr. 2003.

[17] S. Boyd and L. Xiao, "Least-squares covariance matrix adjustment," *SIAM Journal on Matrix Analysis and Applications*, vol. 27, no. 2, pp. 532–546, Nov. 2005.

[18] R. M. Buehrer, S. Arunachalam, K. Wu, and A. Tonello, "Spatial channel models and measurements for IMT-2000 systems," in *Proc. IEEE VTC*, May 2001, pp. 342–346.

[19] V. R. Cadambe and S. A. 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.

[20] G. Caire and S. Shamai (Shitz), "On the achievable throughput of a multiantenna Gaussian broadcast channel," *IEEE Trans. Inf. Theory*, vol. 49, no. 7, pp. 1691–1706, Jul. 2003.

[21] B. Chen and M. J. Gans, "MIMO communications in ad hoc networks," *IEEE Trans. Signal Process.*, vol. 54, no. 7, pp. 2773–2783, Jul. 2006.

[22] Z. Chen, J. Yuan, and B. Vucetic, "Improved space-time trellis coded modulation scheme on slow Rayleigh fading channels," vol. 37, no. 7, pp. 440–441, Mar. 2001.

[23] L.-U. Choi and R. D. Murch, "A transmit preprocessing technique for multiuser MIMO systems using a decomposition approach," *IEEE Trans. Wireless Commun.*, vol. 3, no. 1, pp. 20–24, Jan. 2004.

[24] S. Chu and X. Wang, "Opportunistic and cooperative spatial multiplexing in MIMO ad hoc networks," in *Proc. ACM Mobihoc*, Hong Kong SAR, China, May 26-30, 2008, pp. 63–72.

[25] L. J. Cimini, Jr, "Analysis and simulation of a digital mobile channel using orthogonal frequency division multiplexing," *IEEE Trans. Commun.*, vol. 33, no. 7, pp. 665–675, Jul. 1985.

[26] C. Cook and H. Marsh, "An introduction to spread spectrum," *IEEE Commun. Mag.*, vol. 21, no. 2, pp. 8–16, Mar. 1983.

[27] M. Costa, "Writing on dirty paper," *IEEE Trans. Inf. Theory*, vol. 29, no. 3, pp. 439–441, May 1983.

[28] T. M. Cover, "Broadcast channels," *IEEE Trans. Inf. Theory*, vol. 18, no. 1, pp. 2–14, Jan. 1972.

[29] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York: John Wiley & Sons, Inc., 1991.

[30] O. Damen, A. Chkeif, and J.-C. Belfiore, "Lattice code decoder for space-time codes," *IEEE Commun. Lett.*, vol. 4, no. 5, pp. 161–163, May 2000.

[31] O. M. Damen, A. Tewfik, and J.-C. Belfiore, "A construction of a space-time code based on number theory," *IEEE Trans. Inf. Theory*, vol. 48, no. 3, pp. 753–760, Mar. 2002.

[32] M. F. Demirkol and M. A. Ingram, "Power-controlled capacity for interfering MIMO links," in *Proc. IEEE Veh. Technol. Conf.*, Atlantic City, NJ, U.S.A., Oct. 2001, pp. 187–191.

[33] ——, "Stream control in network with interfereing MIMO links," in *Proc. IEEE Wireless Commun. and Networking Conf.*, New Orleans, LA, U.S.A., Mar. 2003, pp. 343–348.

[34] R. Draves, J. Padhye, and B. Zill, "Routing in multi-radio, multi-hop wireless mesh networks," in *Proc. ACM Mobicom*, Philadelphia, PA, Sep. 2004, pp. 114–128.

[35] P. C. F. Eggers, J. Toftgard, and A. M. Oprea, "Antenna systems for base station diversity in urban small and micro cells," *IEEE J. Sel. Areas Commun.*, vol. 11, no. 7, pp. 1046–1057, Sep. 1993.

[36] M. Ehrgott, *Multicriteria Optimization*, 2nd ed. Berlin/Heidelberg/New York: Springer, 2005.

[37] R. H. Etkin, D. N. C. Tse, and H. Wang, "Gaussian interference channel capacity to within one bit," *IEEE Trans. Inf. Theory*, vol. 54, no. 12, pp. 5534–5562, Dec. 2008.

[38] M. P. Fitz and J. V. Krogmeier, "Further results on space-time codes for Rayleigh fading," in *Proc. Allerton Conf. on Commun., Control, and Comput.*, Sep. 1998, pp. 391–400.

[39] G. J. Foschini, "Layered space-time architecture for wireless communication in a fading envorinment when using multi-element antennas," *Bell Labs Tech. J.*, vol. 1, no. 2, pp. 41–59, 1996.

[40] G. J. Foschini and M. J. Gans, "On limits of wireless communications in a fading environment when using multiple antennas," *Wireless Personal Commun.*, vol. 6, pp. 311–355, Mar. 1998.

[41] C. Fragouli, N. Al-Dhahir, S. N. Diggavi, and W. Turin, "Prefiltered space-time M-BCJR equalizer for frequency-selective channels," *IEEE Trans. Commun.*, vol. 50, no. 5, pp. 742–753, May 2002.

[42] M. Francheschetti, O. Dousse, D. Tse, and P. Thiran, "Closing the gap in the capacity of random wireless networks via percolation theory," *IEEE Trans. Inf. Theory*, vol. 53, no. 3, pp. 1009–1018, Mar. 2007.

[43] G. Ganesan and P. Stoica, "Differential modulation using space-time block codes," *IEEE Trans. Signal Process.*, vol. 9, no. 2, pp. 57–60, Feb. 2002.

[44] D. Gesbert, H. Bolcskei, D. Gore, and A. Paulraj, "Outdoor MIMO wireless channels: Models and performance prediction," *IEEE Trans. Commun.*, vol. 50, no. 12, pp. 1926–1934, Dec. 2002.

[45] D. Gesbert, M. Shafi, D. Shiu, P. J. Smith, and A. Naguib, "From theory to practice: An overview of MIMO space-time coded wireless systems," *IEEE J. Sel. Areas Commun.*, vol. 21, no. 3, pp. 281–302, Apr. 2003.

[46] K. S. Gilhousen, I. M. Jacobs, R. Padovani, and L. A. Weaver Jr, "Increased capacity using cdma for mobile satellite communications," *IEEE J. Sel. Areas Commun.*, vol. 8, no. 4, pp. 503–514, May 1990.

[47] G. D. Golden, C. J. Foschini, R. A. Valenzuela, and P. W. Wolniansky, "Detection algorithm and initial laboratory results using V-BLAST space-time communication architecture," *Electronics Letter*, vol. 35, no. 1, pp. 14–16, Jan. 1999.

[48] A. Goldsmith, S. A. Jafar, N. Jindal, and S. Vishwanath, "Capacity limits of MIMO channels," *IEEE J. Sel. Areas Commun.*, vol. 21, no. 1, pp. 684–702, Jun. 2003.

[49] D. Gore and A. Paulraj, "MIMO antenna subset selection with space-time coding," *IEEE Trans. Signal Process.*, vol. 50, no. 10, pp. 2580–2588, Oct. 2002.

[50] I. S. Gradshteyn and I. M. Ryzhik, *Table of Integrals, Series, and Products*. San Diego: Academic Press, 2000.

[51] M. Grossglauser and D. Tse, "Mobility increases the capacity of adhoc wireless networks," *IEEE/ACM Trans. Netw.*, vol. 10, no. 4, pp. 477–486, Aug. 2002.

[52] P. Gupta and P. R. Kumar, "The capacity of wireless networks," *IEEE Trans. Inf. Theory*, vol. 46, no. 2, pp. 388–404, Mar. 2000.

[53] B. Hamdaoui and K. G. Shin, "Characterization and analysis of multi-hop wireless MIMO network throughput," in *Proc. ACM Mobihoc*, Montréal, Québec, Canada, Sep. 2007, pp. 120–129.

[54] A. R. Hammons and H. E. Gamal, "On the theory of space-time codes for PSK modulation," *IEEE Trans. Inf. Theory*, vol. 46, no. 2, pp. 524–542, Mar. 2000.

[55] T. S. Han and K. Kobayashi, "A new achievable rate region for the interference channel," *IEEE Trans. Inf. Theory*, vol. 27, no. 1, pp. 49–60, Jan. 1981.

[56] B. Hassibi and B. Hochwald, "High rates codes that are linear in space and time," *IEEE Trans. Inf. Theory*, vol. 48, no. 7, pp. 1804–1824, Sep. 2002.

[57] B. Hassibi and M. Sharif, "Fundamental limits in MIMO broadcast channels," *IEEE J. Sel. Areas Commun.*, vol. 25, no. 7, pp. 1333–1344, Sep. 2007.

[58] S. Haykin, *Adaptive Filter Theory*. Englewood Cliffs, NJ: Prentice-Hall, 1996.

[59] ——, "Cognitive radio: Brain-empowered wireless communications," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 2, pp. 201–220, Feb. 2005.

[60] R. W. Heath, S. Sandhu, and A. Paulraj, "Antenna selection for spatial multiplexing systems with linear receivers," *IEEE Commun. Lett.*, vol. 5, no. 4, pp. 142–143, Apr. 2001.

[61] J.-B. Hiriart-Urruty and C. Lemaréchal, *Fundamentals of Convex Analysis*. Berlin: Springer-Verlag, 2001.

[62] D. S. Hochbaum, *Approximation Algorithms for NP-Hard Problems.* Boston: PWS Publishing Company, 1997.

[63] R. A. Horn and C. R. Johnson, *Matrix Analysis.* New York: Cambridge University Press, 1990.

[64] R. Horst and H. Tuy, *Global Optimization: Deterministic Approaches*, 3rd ed. Berlin/Heidelberg/New York: Springer-Verlag, 2003.

[65] Y. T. Hou, Y. Shi, and H. D. Sherali, "Optimal spectrum sharing for multi-hop software defined radio networks," in *Proc. IEEE INFOCOM*, Anchorage, Alaska, USA, May 6-12, 2007, pp. 1–9.

[66] M. Hu and J. Zhang, "MIMO ad hoc networks: Medium access control, saturation throughput, and optimal hop distance," *Special Issue on Mobile Ad Hoc Networks, Journal of Communications and Networks*, pp. 317–330, Dec. 2004.

[67] S. A. Jafar and M. J. Fakhereddin, "Degrees of freedom for the MIMO interference channel," *IEEE Trans. Inf. Theory*, vol. 53, no. 7, pp. 2637–2642, Jul. 2007.

[68] H. Jafarkhani, "A quasi orthogonal space time block code," *IEEE Trans. Commun.*, vol. 49, no. 1, pp. 1–4, Jan. 2001.

[69] H. Jafarkhani and N. Seshadri, "Super-orthogonal space-time trellis codes," *IEEE Trans. Inf. Theory*, vol. 49, no. 4, pp. 937–950, Apr. 2003.

[70] N. Jain, S. R. Das, and A. Nasipuri, "A multichannel CSMA MAC protocol with receiver-based channel selection for multihop wireless networks," in *Proc. IEEE ICCCN*, Scottsdale, AZ, Oct. 15-17, 2001, pp. 432–439.

[71] J. Jiang, R. M. Buehrer, and W. H. Tranter, "Greedy scheduling performance for a zero-forcing dirty paper coded system," *IEEE Trans. Commun.*, vol. 54, no. 5, pp. 789–793, May 2006.

[72] N. Jindal, "MIMO broadcast channels with finite rate feedback," *IEEE Trans. Inf. Theory*, vol. 52, no. 11, pp. 5045–5059, Nov. 2006.

[73] N. Jindal, W. Rhee, S. Vishwanath, S. A. Jafar, and A. Goldsmith, "Sum power iterative water-filling for multi-antenna Gaussian broadcast channels," *IEEE Trans. Inf. Theory*, vol. 51, no. 4, pp. 1570–1580, Apr. 2005.

[74] E. A. Jorswieck and H. Boche, "Performance analysis of capacity of MIMO systems under multiuser interference based on worst-case noise behavior," *EURASIP Journal on Wireless Communications and Networking*, vol. 2004, no. 2, pp. 273–285, Feb. 2004.

[75] F. P. Kelly, A. K. Malullo, and D. K. H. Tan, "Rate control in communications networks: Shadow prices, proportional fairness and stability," *Journal of the Operational Research Society*, vol. 49, pp. 237–252, 1998.

[76] B. Khoshnevis, W. Yu, and R. Adve, "Grassmannian beamforming for MIMO amplify-and-forward relaying," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 8, pp. 1397–1407, Oct. 2008.

[77] S.-J. Kim, X. Wang, and M. Madihian, "Cross-layer design of wireless multihop backhaul networks with multiantenna beamforming," *IEEE Trans. Mobile Comput.*, vol. 6, no. 11, pp. 1259–1269, Nov. 2007.

[78] M. Kodialam and T. Nandagopal, "Characterizing the capacity region in multi-radio multi-channel wireless mesh networks," in *Proc. ACM Mobicom*, Cologne, Germany, Aug. 2005, pp. 73–87.

[79] O. Komolafe and J. Sventek, "RSVP performance evaluation using multi-objective evolutionary optimisation," in *Proc. IEEE INFOCOM*, Miami, FL, Mar. 13-17, 2005, pp. 2447–2457.

[80] S. Kozono, T. Tsuruhara, and M. Sakamoto, "Base station polarization diversity reception for mobile radio," *IEEE Trans. Veh. Technol.*, vol. 33, no. 4, pp. 301–306, Nov. 1984.

[81] H. J. Kushner and G. G. Yin, *Stochastic Approximation and Recursive Algorithms and Applications*. New York: Springer, 2003.

[82] P. Kyasanur and N. H. Vaidya, "Capacity of multi-channel wireless networks: Impact of number of channels and interfaces," in *Proc. ACM Mobicom*, Cologne, Germany, Aug.28–Sep.2, 2005, pp. 43–57.

[83] J. N. Laneman, D. N. Tse, and G. W. Wornell, "Cooperative diversity in wireless networks: Efficient protocols and outage behavior," *IEEE Trans. Inf. Theory*, vol. 50, no. 12, pp. 3062–3080, Dec. 2004.

[84] J. Liu and Y. T. Hou, "Weighted proportional fairness capacity for Gaussian MIMO broadcast channels," *Technical Report, Dept. of ECE, Virginia Tech*, Jun. 2007. [Online]. Available: http://filebox.vt.edu/users/kevinlau/publications

[85] J. Liu, Y. T. Hou, and H. D. Sherali, "Conjugate gradient projection approach for MIMO Gaussian broadcast channels," in *Proc. IEEE International Symposium on Information Theory (ISIT)*, Nice, France, Jun. 24-29, 2007, pp. 781–785.

[86] ——, "On the maximum weighted sum-rate of MIMO Gaussian broadcast channels," in *Proc. IEEE ICC*, Beijing, China, May 19-23, 2008, pp. 3664–3668.

[87] J. Liu, Y. T. Hou, Y. Shi, and H. D. Sherali, "Cross-layer optimization for MIMO-based wireless ad hoc networks: Routing, power allocation, and bandwidth allocation," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 6, pp. 913–926, Aug. 2008.

[88] ——, "On performance optimization for multi-carrier MIMO ad hoc networks," in *Proc. ACM Mobihoc*, New Orleans, LA, May 18-21, 2009, pp. 43–54.

[89] J. Liu, Y. Shi, and Y. T. Hou, "A tractable and accurate cross-layer model for multi-hop MIMO ad hoc networks," *Technical Report, Department of ECE, Virginia Tech*, Jul. 2009. [Online]. Available: http://filebox.vt.edu/users/kevinlau/publications

[90] Y. Liu, M. P. Fitz, and O. Y. Takeshita, "A rank criterion for QAM space-time codes," *IEEE Trans. Inf. Theory*, vol. 48, no. 12, pp. 3062–3079, Dec. 2002.

[91] Z. Liu, G. B. Giannakis, S. Barbarossa, and A. Scaglione, "Transmit antenna space-time block coding for generalized OFDM in the presence of unknown multipath," *IEEE J. Sel. Areas Commun.*, vol. 19, no. 7, pp. 1352–1364, Jul. 2001.

[92] D. J. Love and R. W. Heath, "Diversity performance of precoded orthogonal space-time block codes using limited feedback," *IEEE Commun. Lett.*, vol. 8, no. 5, pp. 305–307, May 2004.

[93] ——, "Limited feedback unitary precoding for orthogonal space-time block codes," *IEEE Trans. Signal Process.*, vol. 53, no. 1, pp. 64–73, Jan. 2005.

[94] ——, "Limited feedback unitary precoding for spatial multiplexing systems," *IEEE Trans. Inf. Theory*, vol. 51, no. 8, pp. 2967–2976, Aug. 2005.

[95] ——, "Necessary and sufficient conditions for full diversity order in correlated Rayleigh fading beamforming and combining systems," *IEEE Trans. Wireless Commun.*, vol. 4, no. 1, pp. 20–23, Jan. 2005.

[96] D. J. Love, R. W. Heath, W. Santipach, and M. Honig, "What is the value of limited feedback for MIMO channels?" *IEEE Commun. Mag.*, vol. 42, no. 10, pp. 54–59, Oct. 2004.

[97] D. J. Love, R. W. Heath, and T. Strohmer, "Grassmannian beamforming for multiple-input multiple-output wireless systems," *IEEE Trans. Inf. Theory*, vol. 49, no. 10, pp. 2735–2747, Oct. 2003.

[98] J. R. Magnus and H. Neudecker, *Matrix Differential Calculus with Applications in Statistics and Economics*. New York: Wiley, 1999.

[99] J. Malick, "A dual approach to semidefinite least-squares problems," *SIAM Journal on Matrix Analysis and Applications*, vol. 26, no. 1, pp. 272–284, Sep. 2005.

[100] K. Marti, *Stochastic Optimization Methods*. Berlin/Heidelberg/New York: Springer, 2005.

[101] T. May, H. Rohling, and V. Engels, "Performance analysis of Viterbi decoding for 64-DAPSK and 64-QAM modulated OFDM signals," *IEEE Trans. Commun.*, vol. 46, no. 1, pp. 182–190, Sep. 1998.

[102] M. L. Metha, *Random Matrices*, 3rd ed. London, UK: Academic Press, 2004.

[103] J. Mitola III, "Cognitive radio: An integrated agent architecture for software defined radio," Ph.D. dissertation, KTH Royal Institute of Technology, 2000.

[104] A. F. Molisch, M. Z. Win, Y.-S. Choi, and J. H. Winters, "Capacity of MIMO systems with antenna selection," *IEEE Trans. Wireless Commun.*, vol. 4, no. 4, pp. 1759–1772, Jul. 2005.

[105] A. F. Molisch, "Ultrawideband propagation channels – Theory, measurement, and modeling," *IEEE Trans. Veh. Technol.*, vol. 54, no. 5, pp. 1528–1545, Sep. 2005.

[106] P. H. Moose, "A technique for orthogonal frequency division multiplexing frequency offset correction," *IEEE Trans. Commun.*, vol. 42, no. 10, pp. 2908–2914, Oct. 1994.

[107] S. H. Müller, R. W. Bäuml, R. F. H. Fischer, and J. B. Huber, "OFDM with reduced peak-to-average power ratio by multiple signal representation," *Annual of Telecommunications*, vol. 52, no. 1-2, pp. 58–67, Feb. 1997.

[108] O. Muñoz-Medina, J. Vidal, and A. Agustín, "Linear transceiver design in nonregenerative relays with channel state information," *IEEE Trans. Signal Process.*, vol. 55, no. 6, pp. 2593–2604, Jun. 2007.

[109] R. Nabar, H. Bolcskei, V. Erceg, D. Gesbert, and A. Paulraj, "Performance of multiantenna signaling techniques in the precense of polarization diversity," *IEEE Trans. Commun.*, vol. 50, no. 10, pp. 2553–2562, Oct. 2002.

[110] A. Naguib, N. Seshadri, and R. Calderbank, "Increasing data rate over wireless channels," *IEEE Signal Process. Mag.*, vol. 17, no. 3, pp. 76–92, May 2000.

[111] R. Naraimhan, "Spatial multiplexing with transmit antenna and constellation selection for correlated MIMO fading channels," *IEEE Trans. Inf. Theory*, vol. 51, no. 11, pp. 2829–2838, Nov. 2003.

[112] A. Nasipuri, J. Zhuang, and S. R. Das, "A multichannel CSMA MAC protocol for multihop wireless networks," in *Proc. IEEE WCNC*, Hong Kong, P. R. China, Sep. 21-24, 1999, pp. 1402–1406.

[113] M. J. Neely, E. Modiano, and C. E. Rohrs, "Power allocation and routing in multibeam satellites with time-varying channels," *IEEE/ACM Trans. Netw.*, vol. 11, no. 2, pp. 138–152, Feb. 2003.

[114] ——, "Dynamic power allocation and routing for time varying wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 1, pp. 89–103, Jan. 2005.

[115] S. Y. Oh, M. Gerla, and J.-S. Park, "MIMO and TCP: A case for cross-layer design," in *Proc. IEEE MILCOM*, Orlando, FL, Oct. 29-31, 2007.

[116] S. Y. Oh, M. Gerla, P. Zhao, B. Daneshrad, G. Pei, and J. H. Kim, "MIMO-CAST: A cross-layer ad hoc multicast protocol using MIMO radios," in *Proc. IEEE MILCOM*, Orlando, FL, Oct. 29-31, 2007.

[117] A. Ozgur, O. Leveque, and D. Tse, "Hierarchical cooperation achieves optimal capacity scaling in ad hoc networks," *IEEE Trans. Inf. Theory*, vol. 53, no. 10, pp. 3549–3572, Oct. 2007.

[118] J.-S. Park, A. Nandan, M. Gerla, and H. Lee, "SPACE-MAC: Enabling spatial reuse using MIMO channel-aware MAC," in *Proc. IEEE ICC*, Seoul, Korea, May 16-20, 2005, pp. 3642–3646.

[119] C. B. Peel, B. Hochwald, and A. L. Swindlehurst, "A vector perturbation technique for near capacity multi-antenna multi-user communication – Part I: Channel inversion and regularization," *IEEE Trans. Commun.*, vol. 53, no. 1, pp. 195–202, Jan. 2005.

[120] R. Pickholtz, D. Schilling, and L. Milstein, "Theory of spread-sprectrum communications – A tutorial," *IEEE Trans. Commun.*, vol. 30, no. 5, pp. 855–884, May 1982.

[121] M. B. Pursley, "Performance evaluation for phase-coded spread sprectrum multiple-access communication – Part I: System analysis," *IEEE Trans. Commun.*, vol. 25, no. 8, pp. 795–799, Aug. 1977.

[122] ——, "The role of spread sprectrum in packet radio networks," *Proc. IEEE*, vol. 75, no. 1, pp. 116–134, Jan. 1987.

[123] M. B. Pursley and D. V. Sarwate, "Performance evaluation for phase-coded spread sprectrum multiple-access communication – Part II: Code sequence analysis," *IEEE Trans. Commun.*, vol. 25, no. 8, pp. 800–803, Aug. 1977.

[124] A. Raniwala and T. Chiueh, "Architecture and algorithms for an IEEE 802.11-based multi-channel wireless mesh network," in *Proc. IEEE INFOCOM*, Miami, FL, Mar. 13-17, 2005, pp. 2223–2234.

[125] J. H. Reed, *Software Radio: A Modern Approach to Radio Engineering*. Upper Saddle River, NJ: Prentice Hall, 2002.

[126] J. H. Reed, Ed., *An Introduction to Ultra Wideband Communication Systems*. Upper Saddle River, NJ: Prentice Hall, 2005.

[127] H. Rohling, T. May, K. Brüninghaus, and R. Grünheid, "Broad-band OFDM radio transmission for multimedia applications," *Proc. IEEE*, vol. 87, no. 10, pp. 1778–1789, Oct. 1999.

[128] S. Roman, *Advanced Linear Algebra*. New York, NY: Springer, 2007.

[129] S. Sandhu and A. Paulraj, "Unified design of linear space-time block codes," in *Proc. IEEE GLOBECOM*, 2001, pp. 1073–1077.

[130] W. Santipach and M. L. Honig, "Asymptotic performance of MIMO wireless channels with limited feedback," in *Proc. IEEE Mil. Commun. Conf.*, vol. 1, Oct. 2003, pp. 141–146.

[131] W. Santipach and M. Honig, "Signature optimiation for CDMA with limited feedback," *IEEE Trans. Inf. Theory*, vol. 51, no. 10, pp. 3475–3492, Oct. 2005.

[132] H. D. Sherali and W. P. Adams, *A Reformulation-Linearization-Technique for Solving Discrete and Continuous Nonconvex Problems*. Boston, MA: Kluwer Academic Publishing, 1999.

[133] H. D. Sherali and H. Wang, "Global optimization of nonconvex factorable programming problems," *Mathematical Programming*, vol. 89, no. 3, pp. 459–478, 1997.

[134] H. D. Sherali and C. H. Tuncbilek, "A global optimization algorithm for polynomial programming problems using a reformulation-linearization technique," *Journal of Global Optimization*, vol. 2, no. 1, pp. 101–112, 1992.

[135] Y. Shi and Y. T. Hou, "Optimal power control for programmable radio networks," in *Proc. IEEE INFOCOM*, Anchorage, Alaska, USA, May 6-12, 2007, pp. 1694–1702.

[136] ——, "Theoretical results on base station movement problem for sensor network," in *Proc. IEEE INFOCOM*, Phoenix, AZ, Apr. 13-18, 2008, pp. 376–384.

[137] M. Simon, *Spread Spectrum Communications*. Rockvill, MD: Computer Science Press, 1985.

[138] S. Siwamogsatham and M. P. Fitz, "Robust space-time codes for correlated rayleigh fading channels," *IEEE Trans. Signal Process.*, vol. 50, no. 10, pp. 2408–2416, Oct. 2002.

[139] Q. H. Spencer, A. L. Swindlehurst, and M. Haardt, "Zero-forcing methods for downlink spatial multiplexing in multiuser MIMO channels," *IEEE Trans. Signal Process.*, vol. 52, no. 2, pp. 461–471, Feb. 2004.

[140] K. Sundaresan and R. Sivakumar, "Routing in ad hoc networks with MIMO links," in *Proc. IEEE International Conf. on Network Protocols*, Boston, MA, U.S.A., Nov. 2005, pp. 85–98.

[141] X. Tang and Y. Hua, "Optimal design of non-regenerative MIMO wireless relays," *IEEE Trans. Wireless Commun.*, vol. 6, no. 4, pp. 1398–1407, Apr. 2007.

[142] V. Tarokh, H. Jafarkhani, and A. R. Calderbank, "Space-time block codes for wireless communications: Performance results," *IEEE Trans. Inf. Theory*, vol. 17, no. 3, pp. 451–460, Mar. 1999.

[143] V. Tarokh, N. Seshadri, and A. R. Calderbank, "Space-time codes for high data rate wireless communication: Performance criterion and code construction," *IEEE Trans. Inf. Theory*, vol. 44, no. 2, pp. 744–765, Mar. 1998.

[144] L. Tassiulas and A. Ephremides, "Stability properties of constrained queuing systems and scheduling policies for maximum throughput in multihop radio networks," *IEEE Trans. Autom. Control*, vol. 37, no. 2, pp. 466–478, Mar. 1993.

[145] J. D. Taylor, Ed., *Introduction to Ultra-Wideband Radar Systems*. Boca Raton, FL: CRC Press, 1995.

[146] I. E. Telatar, "Capacity of multi-antenna Gaussian channels," *European Trans. Telecomm.*, vol. 10, no. 6, pp. 585–596, Nov. 1999.

[147] D. Tse and S. Hanly, "Multi-access fading channels: Part I: Polymatroid structure, optimal resource allocation and throughput capacities," *IEEE Trans. Inf. Theory*, vol. 44, no. 7, pp. 2796–2815, Nov. 1998.

[148] A. M. Tulino and S. Verdú, *Random Matrix Theory and Wireless Communications*. Hanover, MA: now Publishers Inc., 2004.

[149] A. M. D. Turkmani, A. A. Arowojolu, P. A. Jefford, and C. J. Kellett, "An experimental evaluation of the performance of two-branch space and polarization of the performance of two-branch space and polization diversity schemes at 1800 MHz," *IEEE Trans. Veh. Technol.*, vol. 44, no. 2, pp. 318–326, May 1995.

[150] S. Verdú, *Multiuser Detection*. Cambridge, UK: Cambridge University Press, 1998.

[151] S. Vishwanath, N. Jindal, and A. Goldsmith, "Duality, achievable rates, and sum-rate capacity of MIMO broadcast channels," *IEEE Trans. Inf. Theory*, vol. 49, no. 10, pp. 2658–2668, Oct. 2003.

[152] P. Viswanath and D. N. C. Tse, "Sum capacity of the vector Gaussian broadcast channel and uplink-downlink duality," *IEEE Trans. Inf. Theory*, vol. 49, no. 8, pp. 1912–1921, Aug. 2003.

[153] H. Viswanathan, S. Venkatesan, and H. Huang, "Downlink capacity evaluation of cellular networks with known-interference cancellation," *IEEE J. Sel. Areas Commun.*, vol. 21, no. 5, pp. 802–811, Jun. 2003.

[154] H. Weingarten, Y. Steinberg, and S. Shamai (Shitz), "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.

[155] S. B. Weinstein and P. M. Ebert, "Data transmission by frequency-division multiplexing using the discrete fourier transform," *IEEE Trans. Commun.*, vol. 19, no. 5, pp. 628–634, Oct. 1971.

[156] M. Z. Win and R. A. Scholtz, "Impulse radio: How it works," *IEEE Commun. Lett.*, vol. 2, no. 2, pp. 36–38, Feb. 1998.

[157] J. H. Winters, "Smart antenna techniques and their application to wireless ad hoc networks," *IEEE Wireless Commun. Mag.*, vol. 13, no. 4, pp. 77–83, Aug. 2006.

[158] Q. Yan and R. S. Blum, "Improved space-time convolutional codes for quasi-static slow fading channels," *IEEE Trans. Commun.*, vol. 52, no. 7, pp. 1136–1144, Jul. 2004.

[159] S. Ye and R. S. Blum, "Optimized signaling for MIMO interference systems with feedback," *IEEE Trans. Signal Process.*, vol. 51, no. 11, pp. 2839–2848, Nov. 2003.

[160] W. Yu, "A dual decomposition approach to the sum power Gaussian vector multiple-access channel sum capacity problem," in *Proc. Conf. Information Sciences and Systems (CISS)*, Baltimore, MD, U.S.A., Mar. 2003.

[161] ——, "Uplink-downlink duality via minimax duality," *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 361–374, Feb. 2006.

[162] W. Yu and J. M. Cioffi, "Sum capacity of Gaussian vector broadcast channels," *IEEE Trans. Inf. Theory*, vol. 50, no. 9, pp. 1875–1892, Sep. 2004.

[163] W. Yu, W. Rhee, S. Boyd, and J. M. Cioffi, "Iterative water-filling for Gaussian vector multiple-access channels," *IEEE Trans. Inf. Theory*, vol. 50, no. 1, pp. 145–152, Jan. 2004.

[164] W. Yu, D. Varodayan, and J. M. Cioffi, "Trellis and convolutional precoding for transmitter-based interference presubtraction," *IEEE Trans. Inf. Theory*, vol. 53, no. 7, pp. 1220–1230, Jul. 2005.

[165] R. Zamir, S. Shamai (Shitz), and U. Erez, "Nested linear/lattice codes for structured multiterminal binning," *IEEE Trans. Inf. Theory*, vol. 48, no. 6, pp. 1250–1277, Jun. 2002.

[166] L. Zheng and D. N. C. Tse, "Diversity and multiplexing: A fundamental tradeoff in multiple-antenna channels," *IEEE Trans. Inf. Theory*, vol. 49, no. 5, pp. 1073–1096, May 2003.