Cross-layer Modeling and Algorithm Design for MIMO Ad hoc Networks

A Dissertation Presented

by

Shan Chu

to

The Graduate School

in Partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy

in

Electrical Engineering

Stony Brook University

December 2011

Stony Brook University

The Graduate School

Shan Chu

We, the dissertation committee for the above candidate for the Doctor of Philosophy degree, hereby recommend acceptance of this dissertation.

Xin Wang – Dissertation Advisor
Associate Professor, Department of Electrical and Computer Engineering

Yuanyuan Yang – Chairperson of Defense
Professor, Department of Electrical and Computer Engineering

Alex Doboli
Associate Professor, Department of Electrical and Computer Engineering

Xianfeng Gu
Associate Professor, Department of Computer Science

This dissertation is accepted by the Graduate School.

Lawrence Martin
Dean of the Graduate School

Abstract of the Dissertation

Cross-layer Modeling and Algorithm Design for MIMO Ad hoc Networks

by

Shan Chu

Doctor of Philosophy

in

Electrical Engineering

Stony Brook University

2011

With the fast progress of MIMO technology and its growing applications in networks, it is important to develop techniques to enable more efficient MIMO network communications. However, in a dynamic ad hoc network with changing channel conditions and topology, it is very challenging to coordinate channel evaluations and data transmissions as well as fully exploit the advantages of MIMO technology. Having observed that most of the literature works lack realistic model of the interaction between physical layer and the upper layers, we are trying to model the realistic physical layer characteristics and constraints to assist the design of algorithms and protocols that enhance the performance and efficiency of MIMO ad hoc networks.

This dissertation focuses on modeling and algorithm design of MIMO ad hoc networks and addresses several important issues spanning over several network layers. First, we formulate a concrete physical model for MIMO ad hoc networks, and propose cross-layer algorithms which take advantage of physical layer

channel information to opportunistically schedule cooperative spatial multiplexed transmissions between nodes to maximize the network throughput. Second, we propose a series of scheduling algorithms to support opportunistic and cooperative MIMO transmission in different scenarios, including adaptive scheduling in heterogeneous transmission environment, seamless use of cooperative relay to cope with harsh channel condition and distributed interference management to better leverage MIMO advantages. Third, we formulate the MIMO-enabled multi-source multi-destination multi-hop routing problem into a multi-commodity flow problem and developed a polynomial time approximation solution that maximizes the scaling factor for the concurrent flows as well as a distributed algorithm to minimize the congestion in the network links. Finally, we study the deployment of MIMO nodes as relays to assist weak links in wireless networks, with the aim of reducing the number of relay nodes and providing performance provisioning. We provide a polynomial-time approximation scheme algorithm, as well as centralized and distributed algorithms to effectively determine the MIMO relay nodes positions over the network and flexibly select various transmission strategies to further leverage the advantages brought by MIMO.

To my parents and husband.

Contents

List of Figures

List of Tables

Acknowledgements

I would like to show my gratitude to numerous people, who helped and supported me during my Ph.D. study.

First and foremost, I deeply and sincerely appreciate my advisor, Prof. Xin Wang, for her enthusiastic supervision and guidance to my research and professional growth over the past five years. Without her broad vision and deep insight, valuable advice and strong support, this dissertation would not have been possible. I can never forget the countless times she worked so hard and carefully on our papers, and even stayed up till 5:00am with me before some deadlines. I hope I could learn more from her persistence, optimism and strong-mindness, which will greatly benefit my career development as well as the attitude towards life.

I would like to express my thanks to my defense committee members Prof. Yuanyuan Yang, Prof. Alex Doboli and Prof. Xianfeng Gu for their precious time to read my thesis and attend the defense. I appreciate Prof. Yuanyuan yang for being my defense committee chair. Her deep insight and valuable advice have been of great help on our collaborated work, and her warm support as the graduate program director has helped make my life here smooth and pleasant. I would like to thank Prof. Alex Doboli for his valuable suggestions to improve my dissertation quality. I also would like to thank Prof. Xianfeng Gu, for the inspiring discussions and insightful advice.

Many thanks to the past and present members of the WINS Lab, Ziyi Zhang, Qiang Ma, Peng Wei, Qiang Liu, Cunhao Gao, Ying Li, Zhiyuan Weng and Jie Zhao, for their friendship and help over the years.

Last but not least, I would like to dedicate this thesis to my dear parents Xiaolian Liu and Genrong Chu, and my beloved husband Ji Li, who have provided me constant love and tremendous support all through the ups and downs of life. I also would like to thank my son, for making my life so delightful, and other family members for their support and encouragement in these years.

Chapter 1

Introduction

There are increasing interests and use of mobile ad hoc networks with the proliferation of mobile, network-enabled wireless devices, and the fast progress of computing techniques and wireless networking techniques. In a mobile ad-hoc network (MANET), wireless devices could self-configure and form a network with an arbitrary topology. The network's topology may change rapidly and unpredictably. Such a network may operate in a stand-alone fashion, or may be connected to the larger Internet. Mobile ad-hoc networks became a popular subject for research in recent years, and various studies have been made to increase the performance of ad hoc networks and support more advanced mobile computing and applications.

1.1 MIMO-based Ad hoc Networks

As the number, CPU power and storage space of wireless devices continue to grow, there is a significant increase in data transmission demand to support data intensive mobile computing and applications, such as multimedia streaming, gaming, transmission of a large amount of event data during environmental monitoring, and distributed and collaborative processing among a set of wireless devices. Wireless communication systems using multiple antennas at the transmitter and/or receiver have recently emerged as one of the most significant advances in wireless communications. Multiple-input multiple-output (MIMO) systems are presently at the leading edge of wireless systems research and are considered as one of the best approaches for increasing the capacity of wireless networks. MIMO technology is also prominently regarded as a technology of choice for next generation commercial wireless networks such as IEEE 802.16 [1], IEEE 802.11 [2] and cellular third generation (3G) systems such as Universal Mobile Telecommunications System (UMTS) or 1xEV-DO system [3]. Moreover, to fully support cellular environments MIMO research consortia including IST-MASCOT have proposed to develop advanced MIMO techniques, i.e., multi-user MIMO (MU-MIMO).

A MIMO wireless communication system is defined as a transmission link where the transmitter and the receiver are equipped with multiple antenna elements to supplement traditional time processing with spatial signal processing, with the aim of improving transmission reliability and providing higher raw data rates. As a rich scattering environment can provide independent transmission paths (multi-channels) between different transmitting and receiving antenna pairs, an intended receiver node can separate and decode its received data streams based on their unique spatial signatures. This allows MIMO systems to efficiently take advantage of random fading and multi-path propagation to improve the performance of wireless transmission links by several orders of magnitude without requiring any additional bandwidth. A transmitter node can divide its data into multiple data streams and transmit them simultaneously over multiple antenna elements, which is known as spatial multiplexing [4, 5] to increase the transmission rate and/or by space-time codes [6–8] to exploit the MIMO channel diversity. In a network with multiple users, the channels between different users and antenna pairs are different and vary over time. In cellular networks, multiuser diversity could be exploited by scheduling the user with the best channel condition to communicate with the base station [9–11].

Since the pioneering work on MIMO wireless systems that predicted a remarkable spectral efficiency [4], research has been mostly focused on the development of physical layer algorithms and coding techniques for reaching the theoretical MIMO capacity [4, 7]. However, wireless communication systems generally consist of several layers, and the use of multiple antennas does not affect only the physical and coding layers, but also impacts the higher layers. By designing MIMO-aware algorithms for the upper layers, one can envision that the global system performance can significantly improve.

1.2 Motivation

With the fast progress of MIMO technology, it is now being adopted in 802.11n [2] and is also considered for ad hoc networks, where all nodes are peer-to-peer in nature and connected through a mesh topology. Different from an infrastructure-based single-hop cellular network, it is difficult for nodes to coordinate in channel evaluations and transmissions in a dynamic meshed ad hoc network. Different nodes may have different number of antennas, and the peer relationship changes as network topology changes. The quick variation of channel condition and network topology as well as the inconsistency in node density would lead to more challenges in ad hoc network design.

Traditional networking research has modeled the physical layer by constructing simplified and, in many cases, unrealistic abstractions that make it easier to perform both analytical and simulation-oriented studies of the protocols developed. The transmission pattern using a multiplicity of antennas is complex and difficult to model. Furthermore, a multiplicity of communicating pairs that are in close proximity to one another could share the available bandwidth simultaneously. Given this, the design of a MAC protocol is challenging and should have to account for physical-layer dependencies. Although there are many recent efforts in developing MAC protocols for applying MIMO technique to ad hoc networks [12–18], there is very limited work to fully exploit the meshed topology of ad hoc networks and consider both multiuser diversity and spatial diversity to maximize network capacity. In addition, the traffic at each node may be different and the user packets may have different service requirements, which lead to more open problems for the MAC protocol design in MIMO-based ad hoc networks. Instead of simply extending the algorithms used in cellular networks, an efficient algorithm is needed to better exploit the peer-to-peer nature of the network and the varying channel condition to maximize the data rate of the network. Some important issues that need to be better addressed include:

- Integration of MIMO communication features with network design principles for optimal system performance. Existing MIMO MAC schemes are often decoupled from the physical model, either simply treating a node pair with N antennas as having N equal-gain frequency channels, or assuming that many physical parameters are known. It is very difficult to coordinate channel evaluations and transmissions in presence of multiple users, and the coordination is even harder in a network with meshed topology. Ignoring physical layer conditions will reduce transmission quality and even result in delivery failure. On the other hand, many theoretical MIMO gains cannot be achieved without network schemes to coordinate the transmissions.

- Consideration of node heterogeneity. Nodes in the network could have different number of antennas. The existence of a node with a smaller-size antenna array thus limited decoding capability could significantly limit the advantage of nodes with larger-size antenna arrays in the same neighborhood.

- Enabling of efficient MIMO communications in different environments. The number of simultaneous flows that can be transmitted not only depends on the number of antennas but also depends on the number of orthogonal channels (also called degree of freedom) (DoF) an environment allows. The network scheme that is designed to be stove-piped under a specific environment assumption (e.g., line-of-sight or rich scattering) will fail as a node's transmission environment changes.

- Flexible network design to handle extremely weak channels while ensuring a high transmission capacity. The capability of MIMO transmissions will change with channel conditions. Due to channel fading or temporary network topology change, the channel condition could become very weak. Special strategies are required to handle extremely weak channel for improved transmission reliability and stability while not significantly compromising network capacity.

- MIMO-aware routing design. Existing routing protocols cannot exploit MIMO features to construct more efficient routing paths, and adapt network paths based on changes of environment and topology. There is also a lack of theoretical studies from optimization perspective on the achievable routing performance by exploiting the opportunities and addressing the constraints imposed by the incorporation of MIMO.

1.3 Dissertation Outline

In this dissertation, we made extensive efforts to address the aforementioned issues. The rest of the dissertation is organized as follows. Chapter 2 proposes Opportunistic and Cooperative Spatial Multiplexing, a scheme taking advantage of the meshed topology of ad hoc networks to fully exploit the multiuser diversity and spatial diversity in order to maximize the data rate of the network while supporting different transmission priorities, reducing transmission delay, and ensuring fair transmissions among nodes [19, 20]. Chapter 3 proposes a holistic scheduling algorithm that can adaptively select different transmission strategies based on the node types and channel conditions to effectively relieve the bottleneck effect caused by nodes with smaller antenna arrays, and avoid the transmission failure due to the violation of lower degree of freedom constraint resulted from the channel dependency [21, 22]. Chapter 4 studies the problem of exploiting cooperative relay transmission in a MIMO-based ad hoc network to cope with harsh channel condition. The proposed scheduling scheme can efficiently invoke relay transmission without introducing significant signaling overhead as conventional relay schemes, and seamlessly integrate relay transmission with multiplexed MIMO transmission [23, 24]. Chapter 5 investigates the physical layer characteristics of MIMO transmission to identify the opportunities and constraints for the design of adaptive MIMO multiplexing with interference management, and propose a distributed scheduling algorithm which adaptively determines the transmission scheme for nodes with limited spatial DoF in MIMO-based wireless mesh networks [25]. Chapter 6 proposes the concept of MIMO-aware routing and investigate how it can further leverage the advantages brought by MIMO.We first present constraints that capture the characteristics of MIMO transmissions. Algorithms are proposed to provide an approximate solution to achieve maximum concurrent flow in the network minimizes the maximum congestion of link/MIMO-channels [26][27]. Chapter 7 studies the problem of deploying MIMO nodes as relays to assist weak links in wireless networks, with the aim of reducing the number of relay nodes and providing performance provisioning [28]. Finally Chapter 8 concludes the dissertation.

Chapter 2

Opportunistic and Cooperative Spatial Multiplexing

In this chapter, we propose an integrated scheduling scheme to improve the network throughput and transmission quality in MIMO-based ad hoc networks by jointly considering traffic demands, service requirements, network load, multiuser diversity, and channel condition. In our scheme, a sender node can transmit to multiple downstream nodes using different antennas, while a receiver node can receive packets from multiple upstream nodes. Therefore, a group of neighboring nodes can take advantage of the meshed network topology to cooperate in transmission and form a *virtual* MIMO array. In a transmission duration, transmitter nodes and antenna sets are selected opportunistically to exploit the multiuser diversity and spatial diversity to a large degree, while supporting different transmission priorities, reducing transmission delay and ensuring fair transmissions among nodes. Our scheduling scheme is *cross-layer*, with the consideration of physical channel condition and transmission power in MAC design. The main contributions of this chapter are summarized as follows.

- We formally formulate the multiuser MIMO scheduling problem using a graph approach, and divide it into two subproblems.

- We propose a centralized algorithm to use as performance benchmark, and a distributed algorithm for practical implementation. Both algorithms take advantage of the multiuser diversity and spatial diversity by opportunistically selecting the nodes and antennas with good channel conditions to form virtual transmission array and maximize the spatial multiplexing gain.

- We develop schemes to specifically consider the service requirements of the user traffic, the transmission delay, and the fairness among nodes.

- We form a concrete physical layer model, and provide efficient methodologies to evaluate channel coefficients and interference, in the presence of a large number of nodes competing in transmission. This can reduce the gap between physical layer theoretical studies and practical implementation of the algorithm in network to improve performance.

- We propose a new MAC scheme to better work in a MIMO-based multi-packet reception network, and to support our distributed algorithm design.

The rest of the chapter is organized as follows. Section 2.1 discusses the related work. We introduce the system architecture in Section 2.2 and describe our physical model in Section 2.3. We formally formulate the problem in Section 2.4, and propose our centralized and distributed algorithms in Section 2.5 and 2.6 respectively. In Section 2.7, we present our MAC protocol. Simulation results are given in Section 2.8 and the chapter is concluded in Section 2.9.

2.1 Related Work

Over the past several years, the application of MIMO technology in networks has undergone a fast development. Many studies have been performed to develop scheduling schemes to select the best user to transmit based on certain criteria in a multiuser MIMO-based cellular network. In [9], an overview of scheduling algorithms in MIMO-based fourth-generation wireless systems is given, and the relationship of spatial and multiuser diversity is also investigated. The work [10] addresses the design of the optimal space time scheduler for multiuser MIMO system based on an information theory approach. In [11], the authors argue that both multiuser and spatial diversity can be exploited with more bits of feedback information.

In recent years, many efforts have been made to support MIMO transmission in ad hoc networks. In [12], spatial diversity (e.g. space time coding (STC)) is explored to combat fading and achieve robustness. SPACE-MAC, proposed in [13], enables denser spatial reuse patterns with the aid of transmitter and receiver beamforming. Authors in [15] introduce a distributed scheduling (DSMA) scheme within the CSMA/CA framework where the stream allocation depends on the transmitter-receiver distance. Layered space-time multiuser detection and its role in PHY-MAC cross-layer design are analyzed in [14]. A high-level discussion about cross-layer issues in MAC protocols design for MIMO ad hoc networks is further presented in [29]. In [18], spatial multiplexing with antenna subset selection for data packet transmission is proposed. In [14, 15, 18], a user can only be scheduled to transmit to one receiver node, and the selected user is allowed to use all or a subset of its antennas for transmission. In [30], the physical layer approximation is studied to facilitate cross-layer design of MIMO-BLAST ad hoc network. However, it does not provide a complete algorithm/protocol that can be actually implemented. In [16], the authors discuss key optimization considerations for MAC layer design in ad hoc networks with MIMO links, and develop a centralized algorithm and a distributed algorithm. However, there is no description on how to obtain the parameters necessary for stream selection and performance optimization, while these parameters are critical for MIMO network design and challenging to gain in ad hoc networks. A unified representation of the physical layer capabilities of different types of smart antennas, and unified medium access algorithms are presented in [17]. In these literature works, spatial diversity and multiuser diversity are not fully exploited. There are no support of QoS and consideration of the difference in node traffic demands. We have made an effort to address some of these issues in [19]. In this chapter, we more clearly formulate the problem based on network graph, and further design the distributed algorithm to better support packet transmission priority and user service requirements. We also perform more extensive simulations to demonstrate the functionality of the proposed algorithms.

2.2 System Architecture

We consider an ad hoc network where each node is equipped with an antenna array. The number of antenna elements may vary from node to node. Our MAC design is TDMA based, in which the time domain is divided into transmission durations (TD). A TD consists of several time slots and covers one round of control signal exchange and fixed-size data frame transmission. The data transmission rate within a frame can vary based on the channel condition. For a channel with higher quality, more efficient coding can be used to encode data at a higher rate. Due to the peer-to-peer nature of nodal interaction in ad hoc networks, the total transmit power at each node is considered to be fixed, while the transmit power of each antenna is different when a node uses a different number of antennas for transmission. A link between a transmitter-receiver pair is half-duplex, so that a node can either transmit or receive but not at the same time. A node can transmit multiple streams to several downstream nodes or receive multiple streams from several upstream nodes simultaneously. Therefore, a *virtual* MIMO array can be formed among a group of nodes.

Spatial diversity can be adopted to further improve the transmission gain thus reliability and capacity. There are different types of diversity techniques. Without channel information, dependent streams can be transmitted on different antenna elements over multiple time slots and improve transmission quality through space time coding. When channel information is available, a subset of antennas that can transmit signals at better quality could be selected for transmissions through selection diversity, which is shown to outperform space-time coding in [31]. In this work, antenna selection diversity is exploited at a node to select a subset of stronger streams for transmission. In addition, the proposed many-to-many transmission with use of virtual MIMO array also helps to select stronger streams from candidate transmission node pairs, taking advantage of multi-user diversity to provide additional reliability and throughput. As we focus on spatial multiplexing instead of topology control in this chapter, spatial diversity is only used for diversity gain and transmission range is assumed to be uniform.

A stream is identified by a triplet $(I_{TX}, I_{RX}, I_{ANT})$, where I_{TX} is the index of the transmitter node, I_{RX} is the index of the receiver node, and I_{ANT} is the index of the transmitter antenna. At a transmitter node, independent data streams are transmitted from selected antenna elements. The total number of transmitted streams from a node is obviously limited by the total number of antenna elements of the node. Due to the broadcast nature of wireless links, a stream transmitted from a node i to its one-hop neighbor j is also received by all other one-hop neighbor nodes of i, which causes interference at these nodes. To differentiate the streams received at a node j, we call the streams targeted for j as *data streams*, and the streams not for j as *interference streams*. Thanks to multiple antennas, a node is endowed with multiple packet reception (MPR) capability so that it can receive data streams and suppress interference streams concurrently. Note that the total number of data streams and interference streams received at a node is also constrained by its degree of freedom (DOF), which is approximately equal to its number of antennas in a rich scattering environment [32].

As it is difficult to maintain a central controller in a practical ad hoc network and a node can not be a transmitter and receiver at the same time, our distributed scheduling algorithm has two phases, namely *transmitter node selection* and *stream allocation*. A set of nodes are first selected to be transmitter nodes based on their priority and the current network topology, then the streams with higher priority and/or better quality are allocated from the selected set of transmitter nodes to appropriate antennas.

In the first phase of the scheduling, instead of randomly selecting a set of transmitter nodes, our scheduling algorithm only selects active nodes that have packets for transmissions, and the selection is based on the priority of a node which depends on both the service type and the delay time of its queued packets. In the second phase of scheduling, stream allocation is performed so that data packets of the transmitter nodes are allocated to a selected set of antennas for transmission. In this phase, a selected transmitter node first determines a set of packets to transmit based on their priority and the allowed number of streams to transmit in the neighborhood. As discussed later in Section 2.3, multiple antennas at both ends of a link create multiple independent spatial channels with different channel gains in a multi-path or rich scattering environment, which makes channel capacities or achievable data rates of the streams different. It is thus beneficial to allocate the selected packets to transmit over channels that have stronger channel gain thus higher data rate, i.e. with *opportunistic stream allocation*, in order to maximize the temporal throughput of the network.

To capture the characteristics of a stream p, two parameters are defined below.

- **stream priority** $P(p)$: It depends on the type of the data to be sent with the stream and the delay time of the current data packet. A higher value of $P(p)$ indicates the priority of the stream p is higher. In other words, the stream whose data packet has a higher service priority and/or experiences a longer delay is given a higher priority for transmission.

- **stream quality** $Q(p)$: It describes the reliability of a stream transmission, which depends on the transmission power of the stream (which will reduce when more streams are selected from the same sending node) and the channel condition between the transmitter antenna and the receiver node of this stream (which can be represented by a vector function as discussed later).

2.3 Physical Model

In wireless communications, time-varying fading is commonly observed due to user mobility or the variation of propagation environments [32]. A fading channel can generally be expressed as

$$h = ae^{j\phi} + b, \tag{2.1}$$

where $ae^{j\phi}$ denotes the LOS component and b denotes the time-varying component of the fading. When the LOS component is very weak, the channel can be well modeled by Rayleigh fading.

Consider two nodes i and k which are within the transmission ranges of each other, and the numbers of antenna elements are n_i and n_k respectively. The spatial channel between i and k can be represented as an $n_k \times n_i$ matrix $\mathbf{H}_{ki}$:

$$\mathbf{H}_{ki} = \begin{pmatrix} h_{11} & h_{12} & \dots & h_{1n_i} \\ h_{21} & h_{22} & \dots & h_{2n_i} \\ \vdots & \vdots & \ddots & \vdots \\ h_{n_k1} & h_{n_k2} & \dots & h_{n_kn_i} \end{pmatrix}, \tag{2.2}$$

where h_{mn} is the spatial channel coefficient between the m-th antenna of node k and n-th antenna of node i, and can be represented as in (2.1). In general cases, the number of independent eigenchannels [32] between i and k is equal to the number of non-zero eigenvalues of the matrix $\mathbf{H}_{ki}^*\mathbf{H}_{ki}$. In a rich scattering environment and if the separation of antenna elements at each node is large enough, the spatial channels between node i and k undergo i.i.d fading and there are $\min\{n_i, n_k\}$ eigenchannels in total. For the convenience of discussion, we assume the rich scattering environment all through this chapter, and our results can be easily extended to scenarios with less scattering, i.e. with LOS, by calculating the actual number of eigenchannels.

Let node i be the transmitter node in a particular time slot, then the transmitted signal can be represented as a vector

$$\mathbf{s}_i = \begin{pmatrix} s_1 & s_2 & \dots & s_{n_i} \end{pmatrix}^T, \tag{2.3}$$

where $s_1, s_2, \dots, s_{n_i}$ are signals transmitted from antenna $1, 2, \dots, n_i$. Note that $s_1, s_2, \dots, s_{n_i}$ may have different target receiver nodes.

Consider an active node k with n_k antennas within the transmission range of node i. A receiving node is considered active if it is either a target receiver or a passive listening node of a transmission. Therefore, the faded signal from node i received at node k can be represented as:

$$\mathbf{r}_{ki} = \mathbf{H}_{ki}\mathbf{s}_i = \begin{pmatrix} \sum_{p=1}^{n_i} h_{1p}s_p \\ \sum_{p=1}^{n_i} h_{2p}s_p \\ \vdots \\ \sum_{p=1}^{n_i} h_{n_kp}s_p \end{pmatrix}$$

$$= \begin{pmatrix} \sum_{p \in X_{ki}} h_{1p} s_p \\ \sum_{p \in X_{ki}} h_{2p} s_p \\ \vdots \\ \sum_{p \in X_{ki}} h_{n_k p} s_p \end{pmatrix} + \begin{pmatrix} \sum_{p \notin X_{ki}} h_{1p} s_p \\ \sum_{p \notin X_{ki}} h_{2p} s_p \\ \vdots \\ \sum_{p \notin X_{ki}} h_{n_k p} s_p \end{pmatrix}$$

$$= \mathbf{r}_{ki,sig} + \mathbf{r}_{ki,int}, \tag{2.4}$$

where X_{ki} is the set of streams from node i that transmit signals to node k. Due to the broadcast nature of wireless channels, all signal streams transmitted by node i are received at node k. Therefore, node k has to differentiate streams targeted for itself (data streams) from streams targeted for other nodes (interference streams). Denote the signal to interference and noise ratio (SINR) of received stream p at node k as $SINR_p$, the sum data rate that receiver node k gets from transmitter node i is:

$$R_{ki} = \sum_{p \in X_{ki}} \log(1 + SINR_p). \tag{2.5}$$

Denote the set of transmitting nodes that are within the receiving range of node k as J_k, the total sum rate at receiver node k is therefore the summation over all transmitter nodes in J_k:

$$R_k = \sum_{i \in J_k} R_{ki} = \sum_{i \in J_k} \sum_{p \in X_{ki}} \log(1 + SINR_p). \tag{2.6}$$

The calculation of $SINR_p$ depends on the decoding capacity at the receiver node. According to [32], a way to get optimum performance for multiple stream decoding is using Minimum Mean Square Error Sequential Interference Cancelation (MMSE-SIC) receiver. In this case, the linear MMSE receiver for a stream p is represented by the vector:

$$\mathbf{v}_p = \mathbf{K}_{\mathbf{z}_p}^{-1} \mathbf{h}_p. \tag{2.7}$$

The corresponding SINR achieved is

$$SINR_p = \sigma_p^2 \mathbf{h}_p^* \mathbf{K}_{\mathbf{z}_p}^{-1} \mathbf{h}_p, \tag{2.8}$$

where $\mathbf{h}_p$ is the $n_k \times 1$ channel vector for stream p to a receiver k with n_k antennas, $\mathbf{K}_{\mathbf{z}_p}$ is the covariance of $\mathbf{z}_p$, which is the noise plus interference faced by data stream p: $\mathbf{z}_p = \sum_{q>p}^{N_k} \mathbf{h}_q s_q + \mathbf{n}$. Here N_k is the number of transmission streams (including both data and interference streams) around the receiver k. In SIC decoding, received streams are initially sorted according to their received strength, and the strongest stream is first recovered and subtracted from the received vector. Therefore, only the weaker streams create interference at a stream p. Although the quality of SIC decoding may be impacted by error propagation and the accuracy of channel estimation, it works well if the streams are well coded and the data block length is large [32]. As the design of receiver structure is beyond the scope of this chapter, we do not deal with the problems due to channel estimation and decoding errors.

In point-to-point transmissions, when channel information is known, a transmitter node can assign different power to different transmission streams based on their channel conditions using water-filling method [32] to maximize the data rate. As described in Section 2.5, our centralized algorithm schedules a stream for transmission by comparing its priority and channel gain with those of other candidate streams in the network, thus the streams to select from a node are not known in a scheduling step, and water-filling could not be easily applied to divide the total power among multiple streams in advance. However, water-filling can be used to assign transmission power in our distributed algorithm proposed in Section 2.6 where the determination of the number of streams to use and the allocation of streams are decoupled. As the transmissions of multiple streams from one transmitter node would lead to lower channel gain for individual streams, in many-to-many transmission scenario, it may help to schedule transmissions from multiple nodes than transmitting multiple streams from the same node given the same degree of freedom constraints. Therefore, there is a lower likelihood for a node to transmit multiple streams and the need of power splitting among streams using water-filling. For better performance comparison between our centralized algorithm and distributed algorithm, we consider equal power allocation in this chapter. The performance of our distributed scheduling would be further improved without much change to the algorithm if water-filling is used, but extra processing complexity is required for power assignment considering multiple channel matrices for transmissions to multiple receiver nodes.

As we consider each node has a fixed transmitting power, the transmitting power of a stream only depends on the number of streams allocated from this node. For instance, denote the total transmitting power of node i as P_i, the number of allocated streams of node i as n_i^{allo}, then the transmitting power of a single stream p is $P_p = P_i/n_i^{allo}$ if the total power is uniformly allocated to each stream. With power P_q associated with data stream q and N_0 as the noise variance, we can explicitly calculate $\mathbf{K}_{\mathbf{z}_p}$ as

$$\mathbf{K}_{\mathbf{z}_p} = N_0 \mathbf{I}_{n_r} + \sum_{q>p}^{N_k} P_q \mathbf{h}_q \mathbf{h}_q^*, \tag{2.9}$$

which is invertible. Note that in order to avoid significant signaling overhead, nodes are assumed to perform channel estimation through communications with their one-hop neighbors using MAC protocol in Section 2.7. As nodes are only able to estimate the channels between themselves and nodes in their receiving range, the signals coming from non-estimated channels may constitute a noise floor. Moreover, the channel estimation capacity of a node is always limited in any channel access strategy. The noise floor could potentially reduce the achievable receiving rate, as does in any transmission scheme. However, our MAC design exploits multi-user diversity and antenna selection diversity to significantly increase the transmission signal strength, which helps to increase the received SINR and thus mitigate the problem due to noise floor. Substitute (2.9) into (2.8), the output SINR for stream p can be calculated as:

$$SINR_p = P_p \mathbf{h}_p^* \left(N_0 \mathbf{I}_{N_r} + \sum_{q>p}^{N_k} P_q \mathbf{h}_q \mathbf{h}_q^* \right)^{-1} \mathbf{h}_p. \tag{2.10}$$

Substitute (2.10) into (2.6), we can calculate the data rate for each receiver node. Therefore, the aggregate data rate of the network is $R = \sum_{k \in S_r} R_k$, where S_r is the set of all receiver nodes.

Based on the analysis above, stream quality $Q(p)$ introduced in Section 3 can be quantitatively specified here. From (2.10), it is obvious that the larger the value of $\|\mathbf{h}_p\|^2 = \mathbf{h}_p^* \mathbf{h}_p$ is, the higher is the strength of stream p. So a straightforward way to define $Q(p)$ is to simply use the channel vector and the transmitting power:

$$Q(p)_{sim} = P_p(\mathbf{h}_p^* \mathbf{h}_p) = P_p \|\mathbf{h}_p\|^2. \tag{2.11}$$

However, in order to achieve better aggregate data rate of the whole network, the strength of interference streams caused by a data stream should also be taken into consideration. Unfortunately, it is very difficult to estimate the complete interference formation before scheduling is performed. Thus we define a normalized stream quality index to capture the interference a stream creates to its neighbors hence the impact of interference streams on scheduling:

$$Q(p)_{nor} = \frac{P_p \|\mathbf{h}_p\|^2}{\sum_{q \in X_{int}} P_q \|\mathbf{h}_q\|^2} = \frac{\|\mathbf{h}_p\|^2}{\sum_{q \in X_{int}} \|\mathbf{h}_q\|^2}, \tag{2.12}$$

where X_{int} is the set of interference streams towards neighboring active receivers caused by the transmission of data stream p. By normalizing the strength of a data stream with the strength of interference stream(s) it results in, streams that have higher channel gain yet cause smaller interference in the neighborhood are preferred during scheduling. The definition of stream quality is then used in the following sections for stream allocation.

So far, we have formulated a concrete physical model and provided a stream quality metric to facilitate scheduling. Although the above analysis is based on MMSE-SIC receiver, which helps investigate the impact of physical layer parameters on network performance, the scheduling algorithms we propose next do not depend on a specific receiver model. Other receiver strategies can be easily adopted using our algorithms without much modification.

2.4 Problem Formulation

In this section, we use graph representation to formally formulate the two-phase scheduling problem described in Section 2.2. We first describe graph construction guideline and constraints for scheduling, and then formulate the problem formally.

2.4.1 Graph Construction

A directed graph $G = (V, E)$ is used to model the topology and traffic demand of the network. Each node is represented by a vertex $v \in V$. A directed edge in the graph denotes a candidate transmission stream between a sender and a receiver. Specifically, the source/destination vertex of an edge is the transmitter/receiver of the corresponding stream, solid edges represent data streams, and dashed edges represent interference streams. At the beginning of a transmission duration, if node i has one packet targeted for node k, there is a candidate transmission stream from i to k. A solid edge appears in G with i as the source vertex and k as the destination vertex; meanwhile, if $j_1, \ldots, j_n$ are nodes in the one-hop neighborhood of i, a set of dashed edges are formulated from i to $j_1, \ldots, j_n$. If node i has n_i packets for transmission, there are a total of n_i solid edges originated from i to some of its neighbor nodes.

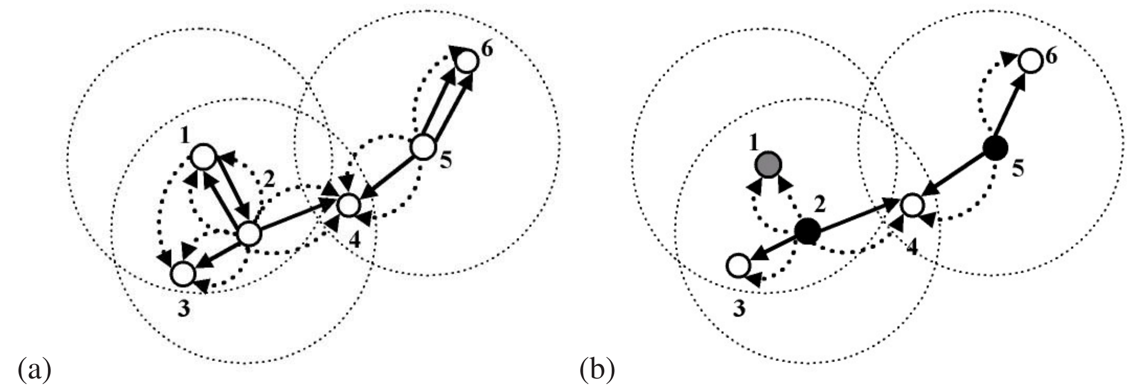


Figure 2.1: Examples: (a) graph representation of the network; (b) feasible scheduling.

Figure 2.1 (a) shows an example of the graph construction. In the figure, a dotted circle represents the transmission range of the centered node. The network consists of six nodes. As node 2 has a packet for each of the nodes 1, 3 and 4, there is one solid edge between node 2 and each of these nodes. A solid edge from node 2 to a target receiver node(e.g., node 1) is accompanied by a set of dashed edges to other neighbors (e.g., nodes 3 and 4). Therefore, nodes 1, 3, and 4 each has two incoming dashed edges from node 2 as a result of the three solid edges originated from node 2. A node may have multiple incoming solid edges from the same neighbor node which has more than one packet for it. For example, node 6 in the figure has two incoming solid edges from node 5, accompanied by two dashed edges from node 5 to node 4.

Edges are scheduled in sets. Each set $\{e\}_j$ consists of one solid edge $e_j = (s(j), t(j))$ and $N_j - 1$ dashed edges $\{\bar{e}\}_j$, where $s(j)$ / $t(j)$ are the source/destination vertices of e_j and N_j is the number of nodes within the transmission range of transmitter node $s(j)$.

The stream parameters defined in Section 2.2 thus become the parameters of edges. The stream priority depends on the data packet, thus one solid edge and its corresponding dashed edge(s) share the same stream priority parameter. The stream quality of an edge depends on the spatial channel between the transmit antenna of the stream and the target receiver node, and is associated with the stream triplet $(I_{TX}, I_{RX}, I_{ANT})$ described in Section 2.2. It is obvious that e_i and $\{\bar{e}\}_i$ have different stream quality as they are associated with different stream triplets. The assignment of the triplet to a data stream is decided by the scheduling algorithm, while the interference streams are caused by the existence of the data stream. The achievable data rate of a data stream e_i, $C(e_i)$ can be calculated based on the stream quality of e_i and all the interfering streams received at node i_t.

Table 1: Degree Constraints (DC).

At a transmission duration k, one and only one of the three constraints is satisfied for a vertex i in subgraph G_k^{opt}:
(1) $0 < d_i(out) \leq n_i$;
(2) $d_i(out) = 0$, $d_i(in.data) \neq 0$, and $d_i(in.data)$ + $d_i(in.int) \leq (1 + \alpha)n_i$;
(3) $d_i(out) = 0$ and $d_i(in.data) = 0$.

In a certain transmission duration k, all candidate streams of the network form a graph G_k. The scheduling is performed to select a subset of the data streams for transmission. The selected data streams and their resulted interference streams along with their senders and receivers form a subgraph of G_k, which is called G_k^{opt}. Denote the number of outgoing solid edges connected to a vertex i as $d_i(out)$, the number of incoming solid edges connected to a vertex i as $d_i(in.data)$, the number of incoming dashed edges connected to a vertex i as $d_i(in.int)$, and the number of antennas at the node of vertex i as n_i. Due to the limitation of decoding capability of nodes and the half-duplex characteristic of links, the degrees of nodes are subject to the constraints shown in Table 1.

If constraint (1) is satisfied, the node is classified as a *transmitter node*, and the total number of outgoing streams at a certain time cannot exceed its number of antennas. If constraint (2) is satisfied, a node receives some streams targeted for it, so it is an active *receiver node*. The parameter $\alpha \geq 0$ is called overload factor, which depends on the decoding capacity of the receiver node [33], and the condition $d(in.data)+d(in.int) \leq (1+\alpha)n$ is used to constrain the total number of incoming streams at a receiver node so that data streams can be decoded while interference streams can be suppressed. If constraint (3) is satisfied, the node is an *idle node*, and is not currently involved in either transmitting or receiving in the network. A node is called *fully loaded* if $d_i(out) = n_i$ for a transmitter node or $d_i(in.data) + d_i(in.int) = (1 + \alpha)n_i$ for a receiver node.

Figure 2.1 (b) shows an example of the feasible scheduling, where the degree constraint is satisfied for every node. In the figure, nodes 2 and 5, which have non-zero number of outgoing edges, are colored black and scheduled as transmitter nodes. Nodes 3, 4 and 6, with zero outgoing edges and non-zero number of solid incoming edges, are colored white and serve as receiver nodes. Node 1 has neither outgoing edge nor solid incoming edge, so it is colored grey as an idle node. Assume $n_i = 4, i = 1, \ldots, 6$ and $\alpha = 0$ here, then all the receiver nodes satisfy the third inequation of the constraint (2), which indicates that the data streams can be correctly decoded. Node 4, which has 4 incoming edges, is an example of being a fully

Table 2: Optimum Subgraph Problem (OSGP).

Select a subgraph G^{opt} of graph G, with antenna allocation L^{opt}, such that: **(1)** G^{opt} satisfies **constraint DC**; **(2) Optimum Priority**: Denote a residual graph $G^{-} = G - G^{opt}$. For any edge e_x in G^{-} whose stream priority is higher than the lowest stream priority of the edges in G^{opt}, **DC** cannot be held if e_x is added to G^{opt}. **(3) Optimum Capacity**: Denote the set of solid edges in G^{opt} as $\{e\}_{data}$. The total achievable data rate of G^{opt} is therefore $C(G^{opt}) = \sum_{p \in \{e\}_{data}} C(p)$. There does not exist another subgraph $G(k)$ with antenna allocation $L(k)$, which also satisfies (1) and (2), such that $C(G(k)) > C(G^{opt})$.

loaded receiver node.

2.4.2 Problem Definition

In dynamic networks, a node gets data packets from its upper layers from time to time, and it is impossible to have the information of all data packets in advance. Moreover, the spatial channels between nodes may vary over time. Rather than scheduling transmissions over time dimension, it is more practical to model the scheduling problem as an iterative optimum subgraph selection problem in each transmission duration (TD), where the temporal network performance is optimized. To consider interactions between consecutive TDs, the residual graph of each TD is updated and left to the next TD for processing. The stream priority metric is calculated accumulatively based on increasing delay time and the stream quality metric is updated according to the channel variation. We first define the optimum subgraph problem as in Table 2.

Basically, OSGP is to find a solution that satisfies all three conditions. First, the subgraph selected should meet the degree constraints. Second, the higher priority streams are preferably selected to form the subgraph. Third, the subgraph selected should achieve optimum aggregate capacity. If OSGP can be solved, the multi-user multi-stream scheduling can be performed in an iterative way as shown in Table 3.

In many cases, the channel associated with the transmitter and receiver of a packet with the highest priority may not have the best quality. There is a tradeoff between optimizing priority and optimizing capacity. In our problem, the optimality of priority is satisfied before the optimality of capacity condition is checked in order to assure the transmissions of high-

Table 3: Multi-User Multi-Stream Scheduling (MUMSS).

Initialization: $G_0 \leftarrow G$ **for** transmission duration $k = 1, 2, \ldots$ Update G_{k-1} according to new traffic demands and updated priority/quality, the new graph is G_k; OSGP(G_k), get graph G_k^{opt}; Send data frames according to G_k^{opt}; $G_k \leftarrow G_k - G_k^{opt}$; **end**

priority streams first.

Our scheme is TDMA based by scheduling transmissions in each transmission time duration. Although promising [34], the application of TDMA in ad-hoc networks leads to the known NP-complete Broadcast Scheduling Problem (BSP) [35]. Therefore, we will provide suboptimal solution with our centralized and distributed scheduling algorithms next.

2.5 Centralized Algorithm

In this section, we propose a centralized algorithm (CMUMSS) to solve the MUMSS problem where all the stream information is assumed to be known at a central controller. The design of the centralized algorithm provides a basis for the distributed algorithm.

In the algorithm, directed graphs are formed as described in Section 2.4.1 with each data edge associated with a candidate transmission between a transmitter and a receiver. The scheduling algorithm ranks all the packets in the system (in implementation, only the priority of the head of line packets of different nodes need to be compared) and greedily schedules transmissions from higher priority to lower priority. For packets with the same priority, transmissions are scheduled from the higher channel quality to lower quality, while ensuring that the overall scheduled network transmissions satisfy the degree constraints. The scheduling therefore meets the constraints (1) and (2) of OSGP formulated in Section 2.4.2 and provides performance with a fixed approximation ratio in terms of constraint (3). The centralized algorithm is given as below.

CMUMSS: Centralized MUMSS Algorithm

1. Initialization

The central controller checks the queue of data packets at every node, and constructs a graph

G_0 according to Section 2.4.1. G_0 contains the data edges and interference edges to be scheduled at the initial phase and is updated to form graph G_k in a subsequent transmission duration k ($k = 1, 2, \ldots$).

2. Greedy Scheduling

For a transmission duration k ($k = 1, 2, \ldots$), perform the following steps in sequence based on graph G_{k-1}.

I. Pre-scheduling Update

This step is performed at the beginning of a transmission duration. Each vertex i keeps a list L_i^{prio} where its outgoing solid edges (associated with to-be scheduled data streams) are ordered in decreasing sequence according to the priority of the corresponding data packets, with the priority calculated based on the service type and delay time of a packet. After checking the new data packets from upper layers for every node, the list for each vertex is updated according to the priority of the new packets. The new edges from all the lists are then added to graph G_{k-1} and the existing weights of G_{k-1} are updated based on the queuing delay of corresponding packets. The updated graph is denoted as G_k. Let the optimum subgraph $G_k^{opt} = NULL$. Create another subgraph called blocked graph G_k^b, used to save edge sets that cannot be scheduled in the current duration, and set $G_k^b = NULL$. Each node is allowed either to be a transmitter node or a receiver node at this stage.

II. Stream Allocation

Select the edge with the highest priority in L_i^{prio} from each vertex i to form a set $\{e\}^h$. The j-th element in $\{e\}^h$ is a solid edge associated with a candidate data transmission denoted as $e_j = (s(j), t(j))$, where $s(j)$ and $t(j)$ are the source and destination vertices of edge e_j respectively. e_j and its corresponding dashed edges form a set $\{e\}_j$. Sort all the elements in $\{e\}^h$ according to their priority. The set $\{e\}^h$ can then be partitioned into a series of subsets $\{e\}_1^h, \{e\}_2^h, \ldots, \{e\}_{N_{prio}}^h$, where N_{prio} is the number of different priority values in $\{e\}^h$ and elements in the set $\{e\}_l^h$ are edges in $\{e\}^h$ that have the same priority P_l.

for $l = 1 \rightarrow N_{prio}$

- Denote the q-th element e_q in $\{e\}_l^h$ along with its dashed edges as $\{e\}_q$. For an edge e_q, data transmission can be scheduled from any of the unassigned antennas of its transmitter and the scheduler assigns the antenna based on the channel quality.

Construct a set consisting of the channel quality factors associated with all the possible stream allocations for solid edges with priority P_l: $S_l = \{Q(a, B)|e_q = (s(q), t(q)), a \in A_{s(q)}, B = t(q), \forall e_q \in \{e\}_l^h\}$ where $A_{s(q)}$ is the set of unused antennas at $s(q)$, $Q(a, B)$ is the stream quality factor for a stream between antenna a and node B;

- **for** $q = 1 \rightarrow |\{e\}_l^h|$
  - Find the largest element in S_l, denote it as Q_{max}, and the corresponding transmitter node, receiver node and antenna as s_{max}, t_{max} and a_{max} respectively. $e_{max} = (s_{max}, t_{max})$ is the corresponding edge of Q_{max} that has s_{max}/t_{max} as its source/destination node.
  - If s_{max} is marked as a receiver node or t_{max} is marked as a transmitter node from previous scheduling steps, the edge e_{max} is not eligible for scheduling. Remove the set $\{e\}_{max}$ containing e_{max} and its corresponding interference edges from G_k and add it to G_k^b;
  - Else:
    Tentatively add $\{e\}_{max}$ to G_k^{opt}. Check whether DC is still satisfied for G_k^{opt}.
    - If no, remove $\{e\}_{max}$ from G_k and add it to G_k^b;
    - Else, the edge e_{max} is eligible for scheduling. Mark s_{max} as a transmitter node and t_{max} as a receiver nodes if they are not currently marked. Assign e_{max} to the antenna a_{max}, add $\{e\}_{max}$ along with the allocation information to G_k^{opt}. Update $A_{s_{max}}$ to remove a_{max} from the unused antenna set. Meanwhile, if any vertex associated with $\{e\}_{max}$ becomes fully loaded, remove all edge sets that may overload it from G_k and add them to G_k^b. Remove elements associated with a_{max} from S_l.
  - Delete Q_{max} from S_l.
- **end**

end

III. End Check

Check whether there is still any edge set in G_k. If yes, go to (II); else got to (IV).

IV. Post-scheduling update

The optimum subgraph for this transmission duration is generated. Schedule the transmissions according to graph G_k^{opt}. Add the edges in G_k^b back to G_k, which will be used for scheduling in the next transmission duration.

Next, we use the example in Figure 2.1 to explain our CMUMSS algorithm. In a specific transmission duration k, the graph G_k is constructed as in Figure 2.1(a). Assume the data edges (solid edges) in the figure, from left to right, have index numbers and priorities as in the following table.

Table 4: An example to explain CMUMSS algorithm.

data edge index	1	2	3	4	5	6	7
source vertex	2	2	1	2	5	5	5
destination vertex	3	1	2	4	4	6	6
stream priority	5	2	1	3	4	3	1

Initially, $\{e\}^h$ consists of edges e_1, e_3, e_5, which are respectively the highest priority edge from candidate transmitter nodes 2, 1, and 5. As e_1 has the highest priority 5, it is scheduled first, with node 2 and 3 identified as transmitter and receiver nodes respectively. Similarly, e_5 with priority 4 is scheduled next, with node 5 and 4 assigned as transmitter and receiver nodes. When scheduling e_3 in $\{e\}^h$, which has the lowest priority, as its destination node 2 has already been scheduled as a transmitter (when scheduling e_1), e_3 cannot be scheduled for transmission any more in duration k, and is deleted from G_k and added to G_k^b. In the second run of stream allocation, $\{e\}^h$ consists of edges e_4 and e_6, which have the same priority 3. The stream quality set S_1 is then constructed based on the two data edges and the available antennas at their source nodes. Assume e_4 is the edge corresponding to the largest element Q_{max} in S_1, it is then scheduled first. As the addition of e_6 into G_k^{opt} does not violate the DC, it is also scheduled. At this moment, node 4 has two incoming data edges and two incoming interference edges and is fully loaded. Therefore, the rest two edges e_2 and e_7 are removed from the candidate scheduling set G_k and added to G_k^b as it would overload node 4 if they are scheduled for transmissions. Finally, the optimum subgraph G_k^{opt} for the current transmission duration is formulated as in Figure 2.1(b).

As stated at the beginning of this section, the CMUMSS algorithm is optimum to the first two constraints of OSGP problem defined in Section 2.4.2. Suppose the stream quality factor $Q(p)$ directly reflects the value of stream capacity $C(p)$, we can further prove that CMUMSS achieves a fixed approximation ratio compared with the optimum solution that obtains the highest aggregate data rate, similar as the proof presented later in Section 3.4 . The centralized algorithm is used as a benchmark to evaluate the performance of the distributed algorithm presented next.

2.6 Distributed Algorithm

As introduced in Section 3, the scheduling algorithm includes two phases, namely transmitter nodes selection and stream allocation. The two phases are obviously dependent on each other. Although the two problems can be considered together in the centralized algorithm to achieve better overall performance, in the distributed case without a central controller, a node always has to decide whether it is a transmitter node first. Then the candidate outgoing streams of the selected transmitter nodes are compared, and the streams with higher priority and/or better quality are allocated for transmissions. For better stream selection, the channel condition between the selected transmitter nodes and their target receiver nodes need to be evaluated. To avoid transmission collision from the selected set of transmitting nodes, the channel measurement signals are encoded using pseudo-random codes as discussed in Section 2.7. In this section, we describe our algorithms for distributed transmitter nodes selection (DTNS) and distributed stream allocation (DSA) in detail, and DTNS and DSA jointly form the distributed MUMSS solution (DMUMSS).

2.6.1 Transmitter Nodes Selection

As the transmission is half-duplex and a node cannot be a transmitter and a receiver at the same time, there is a need to select a subset of the nodes to serve as transmitters in a transmission duration. Instead of randomly selecting the transmitter nodes, our DTNS supports service differentiation and reduces transmission delay by giving higher transmission priority to the packets that are in higher service class and/or have larger queuing delay. By reducing the transmission delay of each node, DTNS can balance the load in a neighborhood and ensure transmission fairness. In addition, adaptively selecting a subset of nodes in a neighborhood to participate in channel estimations based on the decoding capabilities of nodes in the neighborhood would help reduce the estimation complexity and avoid unnecessary channel estimations.

We consider a node with packets to transmit an *active* node. To select a subset of nodes to be transmitter nodes in a neighborhood, we introduce a probability P^{TX}, below which an active node can be selected as a transmitter node. The parameter P^{TX} is estimated by each node based on the number of active nodes around each neighboring node j and the maximum number of simultaneous flows allowed by j in its neighborhood. That is, a node estimates P^{TX} based on its two-hop information announced through a Hello message at network layer. In a neighborhood with n nodes, in order to not exceed the decoding capacity of any node at data transmission time, the number of streams that can be simultaneously transmitted in the neighborhood is constrained. Therefore, we constrain the number of transmitter nodes as well to this value to avoid unnecessary channel measurement, reduce processing complexity at a receiver, and better serve higher priority packets. For each active node i, denote the number of its neighboring nodes as n_i^n, the number of streams that can be decoded at its neighboring node j as N_j^{dec}, and the number of active nodes around j as n_j^a, P^{TX} at node i is calculated as follows:

$$P^{TX} = \min\{1, \min_{j=1}^{n_i^n} \left(\frac{N_j^{dec}}{n_j^a} \right)\} \tag{2.13}$$

Note that our selection is more conservative for a node to consider the decoding capability of all its neighbors instead of only the selected receiver nodes, whose information is not available at the selection time.

An active node will then decide if it can be selected as a transmitter node based on P^{TX} and the priority of its packets, which depends on the service type and delay time of the packets. A possible way to integrate both factors into the priority calculation is to let a packet to have its initial priority equal to its service priority number, and the priority of the packet will be increased as its queuing time increases. Assume node i has N_i^{pkt} packets and the priority of the m-th packet in queue is $p_i^{pkt}(m)$, the priority of node i can be calculated as $p(i) = \sum_{m=1}^{N_i^{pkt}} p_i^{pkt}(m)/N_i^{pkt}$. Before a node has any data transmission, it can attach its initial priority with the Hello message sent out. Thereafter, the updated priority is attached with each packet it sends out. A node with priority 0 is idle.

A node can calculate the average priority, $\bar{p}$, of all the active nodes in its neighborhood as $\bar{p} = (\sum_{i=1}^{n^a} p(i))/n^a$. Nodes with higher priority should be given higher transmission opportunity. To avoid extra signaling and control overhead, an active node i has to *self-decide* if it should be selected as a transmitter node by calculating an index number r_i^{TX} as follows:

$$r_i^{TX} = \frac{\bar{p} - p(i)}{\bar{p}} + \gamma_i = P_i + \gamma_i \tag{2.14}$$

where γ_i is a uniformly distributed random number with value in the range [0,1], which is generated at a node i at each transmission duration. The random number γ_i is introduced to provide some fairness among nodes, while the factor $\frac{\bar{p}-p(i)}{\bar{p}}$ is used to give a higher priority node the larger probability of transmission. If $r_i^{TX} < P^{TX}$, node i is selected as a transmitter node in the current transmission duration; otherwise, it has no right of transmission. Therefore, a node with higher service level and/or larger load and hence longer delay has higher chance of being selected as a transmitter node, and our selection algorithm supports QoS and load balancing while ensuring certain fairness. The distributed transmitter nodes selection algorithm is therefore summarized as in Algorithm 1.

Algorithm 1 DTNS: Distributed Transmitter Nodes Selection.

1: **for** Each node i **do**
2: Calculate P_i^{TX} based on Eq. 2.13;
3: Calculate r_i^{TX} based on Eq. 2.14;
4: If $r_i^{TX} \leq P_i^{TX}$, i determines itself to be a transmitter node.
5: **end for**

2.6.2 Stream Allocation

In the distributed allocation algorithm, we first assume that nodes can receive RTSs/CTSs from multiple transmitter/receiver nodes simultaneously and decode them correctly if the number of simultaneous RTSs/CTSs is less than a certain limit number. The feasibility of this assumption will be discussed in Section 2.7.

In distributed scheduling, as there is no centralized control mechanism, the stream allocation decision can be made either at the transmitter nodes or at the receiver nodes. However, there is a tradeoff for taking either of the options. If the decisions are made at the transmitter nodes, channel information should be made available at the transmitter side first. A transmitter node can properly allocate streams to transmit antennas through pre-coding and cancel the interference partially. However, if all the transmitter nodes make the stream allocation independently, it is very likely that the total number of streams (including data streams and interference streams) arriving at a receiver node exceeds the node's decoding capability. If the decisions are made at the receivers, as a receiver node has full knowledge of all data and interference streams it will receive, it can better select the set of streams to turn off so as to maximize the throughput locally. The disadvantage is that different receivers may decide to turn off different streams and lead to conflicting decisions, so extra coordination is still needed at transmitter nodes to finalize the decision. Additionally, the cost for feeding back the selected stream set is much higher compared with feeding back only a small number of relevant parameters, i.e. each receiver only has to feed back two parameters in our scheme.

In this section, we propose a distributed stream allocation algorithm (DSA) which makes decision first at the transmitter nodes, then at the receiver nodes and finalizes the decision at the transmitter nodes (based on the channel estimation from the reverse direction) to concurrently consider the priority and quality of the streams and constrain the number of transmission streams to be within the decoding capability of the receivers. In each transmission duration, the DSA takes the following steps in sequence.

(1) Step 1: actions at the transmitter nodes

At this step, a transmitter node i selects n_i^0 data packets from its queue. Denote the number of antennas at a transmitter node i as n_i. If the total number of packets in the queue is less than n_i, all of them are selected, i.e., $n_i^0 < n_i$; otherwise, only the n_i packets with the highest priority are selected. The IDs of the target receiver nodes of the selected packets, the value n_i^0, and a training signal are then rotationally broadcasted through each antenna of the transmitter node.

(2) Step 2: actions at the receiver nodes

After a receiver node k decodes the information sent from all the selected transmitter nodes in its neighborhood, it learns the number of streams it may receive in the current duration, N_k^0, including the data streams targeted to itself and the interference streams targeted to other nodes. Assume there are n_k^t transmitter nodes in the one-hop neighborhood of k, we have:

$$N_k^0 = \sum_{j=1}^{n_k^t} n_j^0 \tag{2.15}$$

In the reply slot, if a node is the target receiver of any data stream, it will broadcast N_k^0 and the maximum number of streams it can decode N_k^{dec} along with a training sequence.

(3) Step 3: actions at the transmitter nodes

Upon the reception of messages from neighboring receiver nodes, a transmitter node estimates the channel coefficients using the training sequence inserted in the messages, and make the final decision for stream allocation based on the receiving stream information at all its neighboring receivers. Denote the number of receiver nodes within the transmission range of a transmitter node i as n_i^r. Each receiver k sends back the total number of streams it may receive, N_k^0, and the maximum number of streams it can decode, N_k^{dec}. In order to ensure all the receiver nodes in its neighborhood to have high probability of meeting degree constraint, node i constrains its number of sending streams to a number $n_i^{allo} = n_i^0 \min_{k=1}^{n_i^r} \left(N_k^{dec}/N_k^0 \right)$. The value n_i^{allo} may be a fraction number. Instead of directly calculating n_i^{allo}, in our algorithm, n_i^{allo} is estimated based on the probability that one stream can be allocated, which is: $P_i^{allo} = \min_{k=1}^{n_i^r} \left(N_k^{dec}/N_k^0 \right)$. The stream allocation scheme of a selected transmitter node is then as follows.

1. Determine the number of streams that can be allowed for transmission n_i^{allo}, as in Algorithm 2.

Algorithm 2 Determine the value of n_i^{allo}.

```
1: textbfInitialize: Set n_i^allo = 0;
2: for j = 1 → n_i^0 do
3:     Generate a uniformly distributed random variable β_j in the range [0, 1];
4:     If β_j ≤ P_i^allo, n_i^allo ++;
5: end for
```

2. Allocate streams to antennas. Since node i can transmit up to n_i^{allo} number of streams, it needs to select n_i^{allo} packets among the n_i^0 packets selected at step (1) and assign them to the n_i^{allo} best antennas.

   The selection gives preference to packets with higher priority. For packets of the same priority, the selection is solely based on the stream quality in order to achieve a higher data rate. Denote the set of antennas that node i has as $\{a_i\}$, the set of priority levels of the n_i^0 packets as $\{P_i\}$, and the set of receiver nodes which the n_i^0 pre-selected packets are targeted for as $\{B_i\}$. The set $\{B_i\}$ is partitioned into subsets $\{B_i^1\}, \{B_i^2\}, \ldots, \{B_i^{|\{P_i\}|}\}$ according to the descending priorities of the packets. The j-th subset $\{B_i^j\}$ contains the target receiver nodes of the packets with priority $P_i(j)$.

   Recall that a stream p is identified by its transmitter node, transmitter antenna and receiver node, and each stream p has a unique stream quality parameter $Q(p)$, which depends on the transmission power and channel condition of the specific spatial channel. If the normalized stream quality parameter defined in (2.12) is used here, X_{int} only includes the interference streams towards the active receivers in the neighborhood, i.e. those that have sent back CTSs but are not the targeted receivers of stream p. For transmitter node i, there is a set S_i^0 consisting of all the stream quality parameters of the candidate streams:
   $S_i^0 = \{Q(a_i(p), B_i(q)) | a_i(p) \in \{a_i\}, B_i(q) \in \{B_i\}, p = 1, \ldots, |\{a_i\}|, q = 1, \ldots, |\{B_i\}|\}$

Algorithm 3 Subroutine: *stream_allocation* ($\{a'_i\}, \{B_i^j\}, k$).

Initialize: $l = 0$;
$S_i^j = \{Q(a'_i(p), B_i^j(q)) | a'_i(p) \in \{a'_i\}, B_i^j(q) \in \{B_i^j\}, p = 1, \ldots, |\{a'_i\}|, q = 1, \ldots, |\{B_i^j\}|\}$;
while $l < k$ **do**
    Find the largest element in S_i^j, denote it as Q_{max}, and the corresponding antenna and receiver node as $\{a_{max}, B_{max}\}$;
    Allocate the packet for the receiver B_{max} to the antenna a_{max};
    Remove $\{Q(a_{max}, B_i^j(q)) | B_i^j(q) \in \{B_i^j\}, q = 1, \ldots, |\{B_i^j\}|\}$ from S_i^j, as a_{max} is no longer available; if there is no other packet target for the receiver node B_{max}, also remove $\{Q(a'_i(p), B_{max}) | a'_i(p) \in \{a'_i\}, p = 1, \ldots, |\{a'_i\}|\}$ from S_i^j;
    Remove a_{max} from $\{a'_i\}$;
    $l++$;
end while

Assume $\{a'_i\}$ contain the set of available antennas of node i that can be used for stream allocation, the set S_i^j contain the quality parameters of the streams formulated between the antennas in $\{a'_i\}$ and the receivers in the set $\{B_i^j\}$. Let l represent the number of streams currently allocated. The subroutine *steam_alloation* as in Algorithm 3 is used to allocate k streams to transmit the packets which are targeted for the receivers in the priority set $\{B_i^j\}$. Note that $\{a'_i\}$, the set of available antennas of node i, is updated as the subroutine is executed. Let j be the index of the priority level and n_a be the number of streams that have been allocated. Based on the subroutine *stream_allocation*, n_i^{allo} streams can be allocated to appropriate antennas in a loop as in Algorithm 4.

Algorithm 4 Allocate streams to antennas.

Initialize: $j = 1$, $n_a = 0$, $\{a'_i\} = \{a_i\}$;
while $n_a < n_i^{allo}$ **do**
    If $|\{B_i^j\}| \leq n_i^{allo} - n_a$, do *stream_allocation* ($\{a'_i\}, \{B_i^j\}, |\{B_i^j\}|$), $n_a = n_a + |\{B_i^j\}|$;
    else, do *stream_allocation* ($\{a'_i\}, \{B_i^j\}, n_i^{allo} - n_a$), $n_a = n_i^{allo}$;
    $j++$;
end while

The data packets that cannot be scheduled in the current transmission duration will be kept in the transmission queue and wait to be scheduled in the next duration. Due to the increase in delay time, the unscheduled packets will have their priority increased, and hence have higher chance of being scheduled.

2.7 Protocol Description

In order to realize our distributed algorithm, we devise a MAC protocol based on the RTS/CTS mechanism of the IEEE802.11 distributed coordination function (DCF). As mentioned in Section 2.2, a transmission duration (TD) consists of several time slots and covers one round of control signal exchange and fixed-size data frame transmission. Follow the paradigm of IEEE802.11, a TD consists of four slots, namely RTS, CTS, DATA and ACK, which have different slot lengths. The duration of each slot is fixed and long enough for the corresponding messages to complete their tasks. Note that slot synchronization is currently achievable in the IEEE802.11 family of protocols [14]. Although distributed transmissions may increase the asynchronicity at the symbol level and impact decoding quality, as our scheme could effectively increase the SINR of received signals by taking advantage of the antenna selection diversity and multi-user diversity, it would help improve the accuracy of synchronization as well as mitigate the impact of asynchronicity in a distributed scenario. As a node has to decode multiple control signals from nodes in its neighborhood, a multiple-access scheme is required for multiuser detection. Generally, TDMA and CDMA are two commonly used schemes. In our design, we combined both schemes to facilitate multi-user and multi-antenna access. The protocol consists of the following five phases.

(1) RTS transmission

In this phase, nodes which determine themselves to be the selected transmitter nodes as in Section 2.6.1 broadcast RTSs to receiver nodes in its one-hop neighborhood at the beginning of an RTS slot. An RTS contains the ID of node k and the IDs of node k's targeted set of receiver nodes selected by step (1) in Section 2.6.2. The preamble of an RTS can be used as a training sequence for channel estimation at the receiver nodes. An RTS is masked by another random code, called ID code, which is assigned to each node according to its node ID. ID Codes for different nodes are almost orthogonal, which means that the cross-correlation of different nodes' codes is close to zero. Such code series can be constructed in a similar way as in CDMA systems, e.g. using OVSF code. The code length is related to the node density of the network. Recall that we assume the neighbor density is limited to ensure the possibility of channel estimations and hence decoding performance. Each node keeps a set of random codes, where the size of the set is large enough to cover the maximum number of nodes in its neighborhood. The assignment of codes can be done in a similar way as [36]. An RTS signal from node i is rotationally transmitted through node i's antennas $1 \sim n_i$, and there are a short notice signal between two antennas' transmissions to separate them.

(2) RTS reception and CTS transmission

In an RTS slot, a receiver node is in listening mode using all its antenna elements. Upon the reception of multiple RTSs, a receiver correlates its received signal with each element in its set of random codes to differentiate training sequences from different transmitter nodes and estimate spatial channels. Then information included in RTSs can be extracted to be used in receiver action as in step (2) of Section 2.6.2. In a CTS slot, a node k that is the targeted receiver in any RTS request broadcasts a CTS signal masked by the ID code of k, which includes its ID, the number of total streams it may receive N_k^0, and the number of streams it is able to decode N_k^{dec}. Similarly, the preamble of CTSs can be used for training and channel estimation purpose. To inform the transmitter nodes of full channel condition information, a CTS is rotationally transmitted from node k's antennas $1 \sim n_k$, as in the case of RTS. Therefore, each independent spatial channel between a transmitter/receiver pair can be estimated at transmitter nodes.

(3) CTS reception and DATA transmission

In a CTS slot, transmitter nodes are in listening mode. Similar to the case at receivers, a transmitter node has to extract the information included in multiple CTSs. Specifically, as described in step (3) of Section 2.6.2, it has to extract N_k^{dec} and N_k^0 from all its neighbor receiver nodes to determine the number of streams allowed for transmission, and estimates all spatial channels to construct the set S_i of stream quality parameters, which are used to allocate streams to antennas. After stream allocation is completed, spatial multiplexed data streams are transmitted through the selected antennas in a DATA slot.

(4) DATA reception and ACK transmission

In a DATA slot, receiver nodes receive streams from the neighboring transmitter nodes. With channel coefficients estimated in phase (2), streams are decoded using MMSE-SIC as described in Section 2.3. If a data stream is decoded correctly, the receiver node has to confirm with the transmitter node through ACK broadcast. An ACK thus includes the IDs of the transmitter nodes whose streams have been correctly received and is also masked by the ID code of the receiver.

(5) ACK reception

In an ACK slot, all transmitter nodes are in listening mode. Using channel coefficients estimated in phase (3), a transmitter node extracts information in ACKs and checks whether the streams it transmits in this transmission duration are all received correctly. Correctly received data packets are removed from the queue of the node, and erroneously received or lost data packets remain in the queue, waiting to be scheduled in the next transmission duration.

Note that random ID codes are only used for differentiation in control signal transmission. As control signals are relatively short and sent at the maximum power, there is no significant overhead induced for packet encoding and decoding and there is no need for power control.

2.8 Performance Evaluation

In this section, we evaluate the performance of our proposed algorithms through simulations. We consider an ad hoc network with random topology. Nodes are distributed uniformly over a $1250m \times 1250m$ area. Each node has a transmission range of $250m$. The MIMO channel between node pair is modeled based on the distance between nodes and the small-scale fading coefficients following Rayleigh model. White Gaussian noise with $SNR = 10dB$ is added to include environment noise and interference that cannot be canceled. A simulation result is obtained by averaging over several runs of simulations with different seeds.

The distributed multi-user multi-stream scheduling algorithms (DMUMSS) is implemented based on the MAC framework described in Section 2.7 and the algorithms proposed in Section 2.6. The centralized multi-user multi-stream scheduling algorithm (CMUMSS) described in Section 2.5 is also implemented, which serves as a benchmark for performance comparison. To demonstrate the benefit of using many-to-many cooperative transmission by fully taking advantage of multiuser diversity in a meshed network and through antenna selection, the performance of our algorithms is compared with corresponding centralized and distributed schemes of single-user multi-stream scheduling (SUMSS), which is based on conventional multiuser selection. In SUMSS, only one pair of transmitter/receiver nodes is allowed to communicate in the neighborhood, and both transmitter and receiver nodes use all their antenna elements. In each transmission duration, the node pair with the best channel quality is selected, and transmitter node selection is also implemented in SUMSS to reduce collision.

The metrics we use for comparison are aggregate data rate, average drop rate and normalized delay. Aggregate data rate is the total data rates of the network averaged over the number of transmission durations. Packets are dropped due to erroneous decoding when the total number of streams received at a receiver exceeds its decoding capability, i.e. overloaded. The drop rate is defined to be the total number of dropped packets divided by the total number of transmitted packets. For the convenience of comparison, the results of drop rate are normalized to a maximum value. Delay time is defined as the number of transmission durations a packet waits in the queue before it is successfully transmitted. The two phases of distributed scheduling, namely Distributed Transmitter Nodes Selection (DTNS) and Distributed Stream

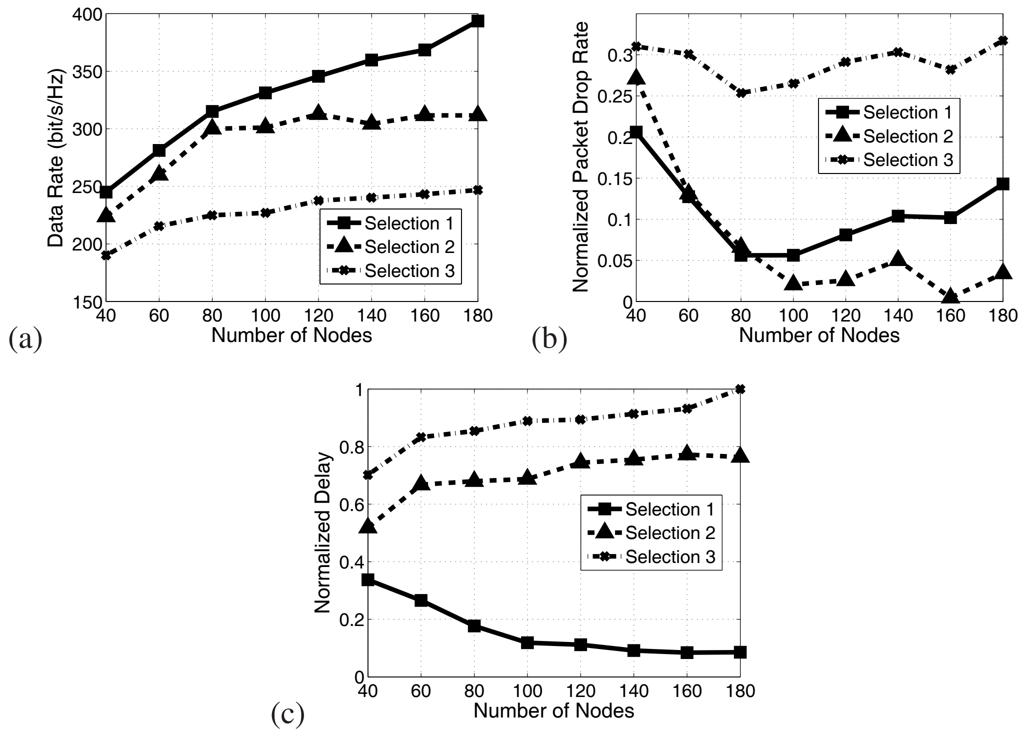


Figure 2.2: Performance of DTNS with different types of transmitter nodes selection: (a) data rate; (b) packet drop rate; (c) normalized delay.

Allocation (DSA), are first studied separately; then the overall performance of DMUMSS is evaluated and compared with CMUMSS, centralized SUMSS (CSUMSS) and distributed SUMSS (DSUMSS). If not otherwise specified, the number of nodes in the network is 100, the number of antenna elements at each node is 4, and the overload factor α defined in Section 2.4.2 is 0.

(1) Performance of DTNS

We first evaluate the performance of DTNS by varying the node density. We consider three types of distributed transmitter nodes selection:

- *Selection 1*: Use DTNS as described in 2.6.1;
- *Selection 2*: Use P^{TX} as described in 2.6.1, but does not consider node priority in r^{TX} calculation;
- *Selection 3*: Use a fixed P^{TX}, which is 0.5 in the simulation, and does not consider node priority in r^{TX} calculation.

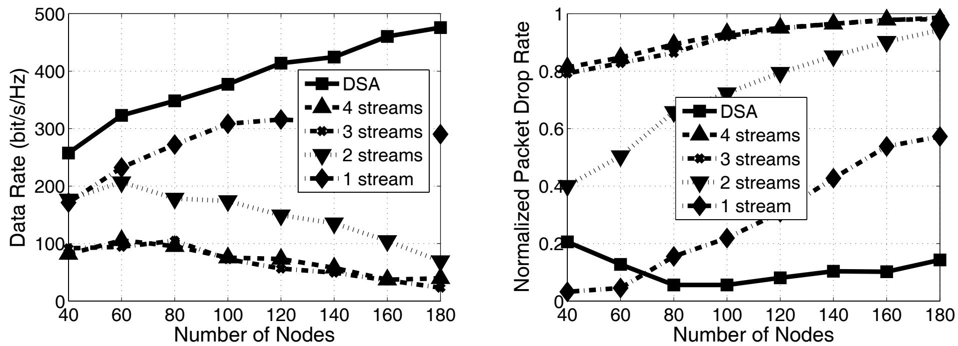


Figure 2.3: Performance of DSA: (a) data rate with DSA and non-adaptive distributed stream allocation; (b) packet drop rate with DSA and non-adaptive distributed stream allocation.

Aggregate data rate, average packet drop rate and normalized delay for the three selection schemes are compared in Figure 2.2. Selection scheme 3 is seen to have the lowest aggregate rate and the highest dropping rate and normalized delay, as it does not consider node density and load condition in node selection. By considering the active node density and traffic load in a neighborhood to reduce collision and delay, selection scheme 1 is seen to achieve more than 60% higher aggregate rate at the highest node density studied while reducing the delay up to 90%. In Figure 2.2 (b), scheme 2 achieves the lowest drop rate in high density case, as its r^{TX} calculation is not impacted by the priority factor which depends on network load and can hence better control the transmission node selection based on the number of active nodes in a neighborhood. As a tradeoff, Figure 2.2 (c) shows that scheme 1 has much lower average delay compared to scheme 2, as packets with longer queuing delay are favored for transmission in scheme 1. Although scheme 2 has lower packet drop rate than scheme 1 at high node density, its aggregate data rate is lower than scheme 1. This is because the scheduling decision of scheme 1 can better adapt to the traffic demands of nodes and increase the total transmission rate.

(2) Performance of DSA

In Section 2.6.2, the number of streams allocated is adaptively adjusted according to the traffic condition in the neighborhood. To demonstrate its advantage, we implement an alternative of DSA where the number of streams allocated is fixed. The number of streams is fixed to different values in the simulation. The performance of DSA and the alternative scheme is illustrated in Figure 2.3 (a) and (b). It is evident that by adjusting the number of streams according to traffic condition, DSA outperforms its alternative by providing significantly higher data rate and lower packet drop rate. As the node density increases, data rate for the alternative scheme reduces and the rate is lower when the fixed stream is set at a larger number, for more collisions are induced. In Section 2.3, a normalized stream quality factor is introduced, which is demonstrated to outperform the simple stream quality factor as in Figure 2.4.

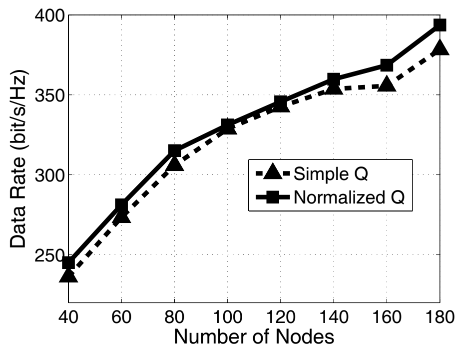


Figure 2.4: Data rate with simple and normalized stream quality factor.

(3) Performance of DMUMSS

The overall performance of DMUMSS is evaluated in Figure 2.5, with CMUMSS, CSUMSS and DSUMSS as references. According to Figure 2.5(a), the aggregate rates of DMUMSS and CMUMSS are close, but the rate of DMUMSS is more than double that of CSUMSS and almost eight times of the rate of DSUMSS. This demonstrates that the data rate can be greatly increased in a meshed network through many-to-many cooperative transmissions by fully exploiting multiuser diversity and spatial diversity. Moreover, as the number of nodes in the network increases, the data rates of both CMUMSS and DMUMSS increase, while the data rate of CSUMSS saturates at a maximum value and the rate of DSUMSS even decreases, as it cannot fully take advantage of the multiuser diversity to achieve higher rate. We also present the performance of centralized and distributed single-user single-stream scheduling algorithms, denoted as CSUSSS and DSUSSS respectively, where each node only has one antenna and best user pairs are selected opportunistically over the network. As expected, using MIMO transmission especially with multi-user multi-stream scheduling can significantly improve data rate. Figure 2.5(b) illustrates the changing of data rate with varied values of DoF (degree of freedom). Again, data rate of MUMSS increases almost linearly. In comparison, limited by the single user constraint, the increasing of data rate of SUMSS, especially DSUMSS, is much slower as the number of antennas grows. This figure indicates that

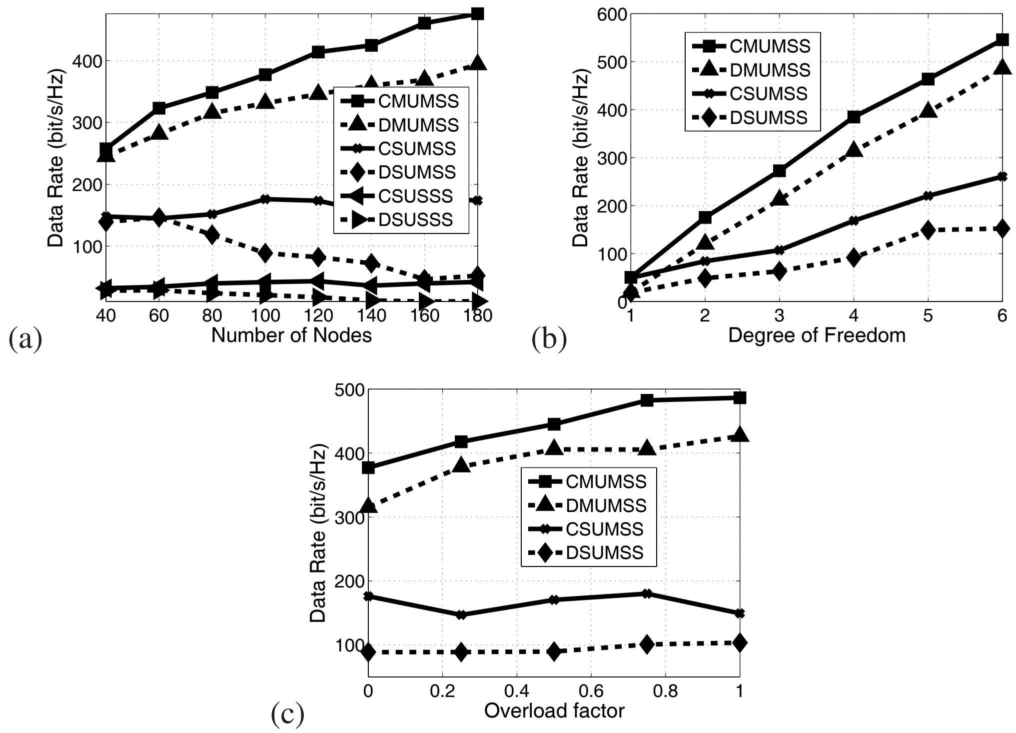


Figure 2.5: Performance of DMUMSS: (a) data rate with different number of nodes in the network; (b) data rate with different value of DoF; (c) data rate with different value of overload factor.

MUMSS can be expected to outperform SUMSS as long as there exists some level of DoF. In Section 2.4.1, we have mentioned overload factor α, which allows more streams to be correctly decoded than the number of antenna elements at receiver nodes. The impact of factor α is studied in Figure 2.5(c). SUMSS can not take advantage of the higher decoding capability to improve data rate, since only interference-free one-to-one communication is allowed in a neighborhood, and the number of streams transmitted between a node pair is constrained by the number of antennas at the transmitter node. Both CMUMSS and DMUMSS achieve higher data rates as overload factor increases from 0 to 1; however, the increasing slope reduces due to the limitation in the number of antennas at transmitter nodes, and the aggregate data rate becomes flat when the overload factor is between 0.75 and 1.

(4) Robustness to Topology Change Rate

In Figure 2.6, the aggregate data rate achieved by DMUMSS is further investigated under the different topology update rate v. The topology of the network changes every v number of transmission durations. For all the three representative values of node density simulated, the aggregate data rate remains almost constant with only slight variations. The result shows that our DMUMSS algorithm is robust to topology changes in the network, as it is always able to coordinate the transmissions based on traffic demand and schedule high-quality streams in any topology. This indicates that our scheme will perform well in a mobile ad hoc network with frequent topology change.

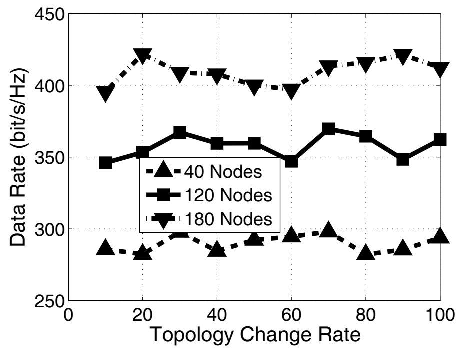


Figure 2.6: Data rate of DMUMSS with different topology change rate.

2.9 Conclusions

In this chapter, we propose a centralized and a distributed scheduling algorithms in MIMO-based ad hoc networks by concurrently considering traffic demand, service requirements, network load, multiuser diversity, and spatial diversity. Our algorithms fully exploit multiuser diversity and spatial diversity to opportunistically select transmitter nodes and transmission antennas while supporting QoS and fairness. Nodes in a neighborhood can cooperate in transmission and form a many-to-many virtual MIMO array. We form a concrete physical layer model, and apply the physical model in our MAC design to efficiently optimize network performance. Our performance results demonstrate that our proposed algorithms are very efficient in coordinating transmissions in a MIMO-based MPR network. Up to eight times data rate is achieved as compared to the scheme of selecting only one user pair at a time as often used in cellular networks, while the transmission delay is reduced up to 90%.

Besides spatial multiplexing, several other techniques can be utilized to further exploit the advantage of MIMO to improve network performance. For instance, space-time coding can be used to increase the reliability of transmissions. It would be intricate but promising to design a cross-layer scheme to adaptively utilize these techniques. These issues will be studied as part of our future work.

Chapter 3

Adaptive Scheduling in Heterogeneous MIMO networks

Although MIMO techniques have been widely studied in a more centralized and infrastructure-based cellular system, there are very limited work and big challenges in extending MIMO technique into a fully distributed system over an infrastructure-free wireless ad hoc network. Different from an infrastructure-based system, it is difficult for nodes to coordinate in channel evaluations and transmissions in a distributed manner. The fast variation of channel condition and network topology, the inconsistency in node density as well as the different traffic demands and service requirements of nodes lead to more open challenges to coordinating distributed node transmissions. Moreover, in a mobile computing environment, the network could be *heterogeneous*, which incurs additional challenges to MIMO MAC design. First, network nodes may be equipped with different number of antennas. The existence of nodes with smaller antenna array sizes may lead to significant network performance reduction. Either the concurrent number of transmissions in a neighborhood needs to be limited in order to meet the decoding constraint of receivers equipped with a lower number of antennas, or the lower-antenna nodes will be significantly interfered by neighboring nodes transmitting a larger number of streams at the same time. Second, the transmission environment could be heterogeneous, with channel conditions different between each node pair and varying over time, leading to the variation of the simultaneous streams allowed between a node pair. These two factors jointly determine the number of orthogonal channels (i.e., *degree of freedom*) an environment allows. It is critical for transmitter nodes to be aware of the allowed degree of freedom of a link for the correct decoding at receiver nodes.

Although there are some recent efforts in developing algorithms and protocols for applying MIMO techniques to ad hoc networks [12, 13, 16–19, 29, 37–39], to the best of our knowledge, there has not been any effort to specifically alleviate the transmission limit thus performance degradation due to the network heterogeneity in a distributed, peer to peer, ad hoc transmission environment. In a distributed system, the extension of solution from homogeneous cases to heterogeneous cases is far from trivial.

To enable more powerful mobile computing and applications in a practical system, the objective of this chapter is to design a holistic distributed scheduling algorithm to adaptively coordinate sharing of transmission resources among heterogeneous nodes in a varying physical operational scenario. The scheduling concurrently considers antenna array size, channel condition, traffic demand and multiuser diversity. In each transmission duration, the algorithm *opportunistically and distributively* schedules the nodes to transmit and determines the set of antennas to use at a selected node, by fully exploiting multiuser diversity and selection diversity to significantly improve transmission throughput and reliability. Through priority-aware scheduling, our algorithm also supports service differentiation while reducing the transmission delay and ensuring the fairness among nodes. Specifically, in order to alleviate the constraints caused by node heterogeneity and the lower-rank channel, our algorithm adaptively selects different transmission strategies based on both the antenna array sizes of nodes in a neighborhood and the degree-of-freedom the transmission environment allows. We also mathematically formulate the problem to maximize the weighted network throughput, and propose a centralized scheduling algorithm as a performance benchmark. Different from the literature work [37–39] which are based on the simple antenna model, our formulation takes into account the different transmission rates between different nodes and antenna pairs and the constraint on the degree of freedom due to the channel condition between a node pair. This facilitates the scheduling to take advantage of multi-user diversity for a higher data rate while considering transmission priority and balancing the network load, and also helps to avoid transmission failure by not overloading a lower-rank channel with more streams.

The rest of the chapter is organized as follows. Section 3.2 discusses the background information including MIMO technologies and their application in heterogeneous networks. In Section 3.3, the system model is defined and the problem is mathematically formulated, followed by Section 3.4, where a centralized algorithm is proposed to solve it. Section 3.5 presents the adaptive distributed scheduling algorithm and the protocol to implement the algorithm. Simulation results are provided in Section 3.6 and the paper is concluded in Section 3.7.

3.1 Related Work

Due to the difficulty of modeling the benefits and constraints of MIMO transmissions, only a limited number of efforts focus on the network performance from the optimization perspective. A centralized algorithm is presented in [37] to solve the joint routing, scheduling and stream control problem subject to the fairness constraint in mesh networks with MIMO links. In [38], the authors characterize the radio and interference constraints in multi-hop wireless MIMO networks and formulate a multi-hop joint routing and MAC problem to study the maximum achievable throughput subject to these constraints. The problem of jointly optimizing power and bandwidth allocation at each node and multi-hop/multi-path routing in a MIMO-based ad hoc network is studied in [39], and a solution procedure is developed to solve this cross-layer optimization problem. Although these efforts are important, they are often based on simple antenna models without considering the opportunities and constraints due to the difference in the physical channels and network heterogeneity. There is also a lack of distributed solutions to efficiently coordinate node transmissions in a practical MIMO network. The features and performance of a few antenna techniques are presented in [17], however there is no design to enable the selection of a specific antenna technique, which is the major challenge in MAC design.

A number of distributed schemes have also been proposed for MIMO MAC designs. In [16], the authors discuss key considerations for MIMO MAC design, and develop a centralized algorithm and a distributed algorithm to improve the transmission fairness. Based on CSMA/CA for control signal exchanges, it is hard for the algorithm to support cooperative transmissions. In [18], spatial multiplexing with antenna subset selection for data packet transmission is proposed, based on nodes with two antennas and a simple network topology. In [29], a transmitter is allowed to transmit to more than one node when there are a sufficient number of antennas and a receiver can also use its antenna to cancel interference. Each transmitter or receiver greedily accommodates the number of data and interference streams up to a pre-determined maximum number, and the decision at a receiver is based on the receiving power of RTS requests, which is transmitted from only one antenna of a transmitter. However, with different distances between different node pairs, different receivers may get the same request from a specific transmitter at different power and make conflicting decisions on the transmission requests to accept. As a result, either the decoding capability of a receiver will be wasted when it accepts an interfered transmission whose transmitting request was not accepted by its targeted receiver, or the decoding limit of the receiver will be exceeded when some unwanted transmissions are scheduled by their target receivers but not considered by this receiver. Moreover, based on only the received power of RTS, the channel difference between each sending antenna and receiving antenna is not considered in the stream selection which may lead to a lower transmission rate. In our scheduling, nodes in a neighborhood coordinate in selecting both the transmitters and antennas to use without the need of explicit signaling. To reduce the transmission conflicts and interference, our scheduling scheme is probabilistic and adaptive, taking into account the decoding capabilities and transmission requests of all nodes in a neighborhood.

In addition to issues associated with each scheme discussed above, most MIMO MAC schemes implicitly assume that the channel condition is known. In practice, coordinating channel measurement itself is a big challenge in presence of a group of competing nodes. Existing MAC schemes often ignore or cannot support channel measurement in a meshed network, which may lead to performance reduction and even failure in supporting basic MIMO communications. It would be more difficult to support antenna selection diversity and multi-user diversity distributively based on channel conditions in a dynamic ad hoc network. There is also no specific support of priority transmission. In existing work, the number of antennas or the pre-determined decoding limit is often used as the constraint of transmission and receiving without considering the actual physical channel variation thus the simultaneous streams allowed by a channel. This may result in the transmission failure. There is no specific design to mitigate the constraints due to the low degree of freedom of channels and the lower antenna number at receivers. In [19], a cooperative multiplexing scheme is proposed, however, it does not consider the heterogeneity of antenna arrays in ad hoc networks. We have made an effort to provide an adaptive and distributed solution considering the heterogeneity of antenna array sizes of network nodes in [21]. In this chapter, we further consider the impact of channel condition on the degree of freedom of MIMO channels. We remodel the problem to more accurately capture the transmission constraints due to both the number of antennas and channel conditions. We also modify the distributed algorithms and perform more extensive simulations to demonstrate the functionality of the proposed distributed algorithms. In addition, we propose a centralized solution with a proved approximation ratio to serve as the benchmark of the distributed algorithm.

3.2 Background and Motivation

With multiple antennas at the transmitter and/or receiver, multiple data streams may be transmitted between a transmission node pair, which is called *spatial multiplexing*. At the receiver, each antenna receives a superposition of all of the transmitted data streams. In a rich scattering environment where the transmission channels for different stream are differentiable and independent, i.e. orthogonal, an intended receiver node can separate and decode its received data streams based on their unique spatial signatures. This multiplexing gain can provide a linear increase (in the number of antenna elements) in the asymptotic link capacity. With multiple transmission paths, the transmission quality could be very different. Instead of sending different data through each transmitting antenna, *spatial diversity* may be exploited to improve transmission reliability. There are different types of diversity techniques. Without channel information, dependent streams can be transmitted on different antenna elements over multiple time slots and improve transmission quality through *space time coding*. When channel information is available, a subset of antennas that can transmit signals at better quality can be selected for transmissions through *selection diversity*, which is shown to outperform space-time coding [31]. As a more powerful yet more sophisticated scheme, data streams can be properly coded according to the channel information and sent through different transmit antennas, i.e., through *precoding*, to achieve the maximum throughput at the receiver.

In this section, we first present the problems due to the limitation of channel degree of freedom and heterogeneous number of network nodes, we then introduce the potential strategies to address the issues, and the tradeoff between different strategies.

In MIMO communications, the spatial channels between two neighboring nodes n_i and n_k which have N_i^{ant} and N_k^{ant} antenna elements respectively can be represented as a $N_k^{ant} \times N_i^{ant}$ matrix:

$$\mathbf{H}_{ki} = \begin{pmatrix} h_{11} & h_{12} & \dots & h_{1N_i^{ant}} \\ h_{21} & h_{22} & \dots & h_{2N_i^{ant}} \\ \vdots & \vdots & \ddots & \vdots \\ h_{N_k^{ant}1} & h_{N_k^{ant}2} & \dots & h_{N_k^{ant}N_i^{ant}} \end{pmatrix}. \tag{3.1}$$

The (p, q)-th entry of $\mathbf{H}_{ki}$, h_{pq}, is the spatial channel coefficient between the p-th antenna of node n_k and q-th antenna of node n_i. Each h_{pq} can generally be represented as [32]:

$$h_{pq} = \sqrt{\frac{\kappa}{\kappa+1}}\sigma_l e^{j\theta} + \sqrt{\frac{1}{\kappa+1}}\mathcal{CN}(0, \sigma_l^2), \tag{3.2}$$

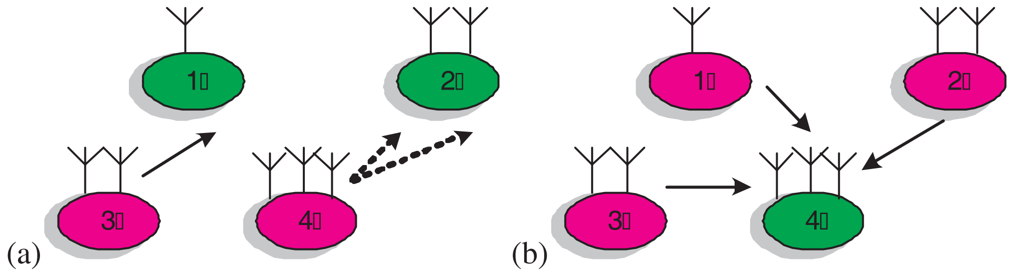


Figure 3.1: Illustration of a heterogeneous MIMO network.

where the first term denotes the line-of-sight (LOS) component with a uniform phase θ, and the second term corresponds to the aggregation of reflected and scattered paths, usually modeled as a circular symmetric random variable with variance σ_l. The parameter κ is called K-factor, which is the ratio of the energy in the LOS path to the energy in the scattered paths. When the LOS component is very weak, i.e. the propagation medium is rich scattering, the channel can be well modeled by Rayleigh fading. When the LOS component between transmitter and receiver is strong and/or there exist fixed scatters/signal reflectors in addition to random main scatters, Rician fading conditions hold and a higher correlation is observed between the elements of $\mathbf{H}_{ki}$.

The *degree-of-freedom* of a MIMO channel is an important metric to describe the dimension of space that the transmitted signals can be projected onto (so the receiver can differentiate the signals), and the number of streams allowed to simultaneously transmit between a pair of nodes. The degree-of-freedom is defined as the *rank* of the channel matrix $\mathbf{H}_{ki}$, or equivalently the number of non-zero eigenvalues of $\mathbf{H}_{ki}$. From (3.1), it is obvious that the degree-of-freedom of the channel between n_i and n_k depends on the number of antennas at nodes n_i and n_k, and the linear independency of the matrix which depends on the scattering conditions between n_i and n_k.

Instead of only allowing multiplexed transmission of multiple streams between a node pair as in traditional MIMO schemes, it is possible to allow multiple nodes to simultaneously transmit to a receiver that has multiple antennas, i.e. forming a *virtual MIMO array* [32], and a sender with multiple antennas can also transmit multiple streams to a set of nodes. The number of simultaneous transmissions allowed in a neighborhood is determined by the degree of freedom of MIMO channels and the decoding capabilities of the receivers. In order to both decode the data streams and cancel the interference streams, the antenna number of a receiver should generally be larger than the total number of data streams and interference streams. In a practical distributed wireless system, the antenna number of different devices is often different. In addition, the physical channel condition could vary over time. The constraint and variation of the degree of freedom under different channel conditions are often not considered in the literature MAC design, and the impact of low degree of freedom associated with specific devices and channels on the overall network performance are largely unexplored [12, 13, 16–19, 29, 37–39]. This will not only lead to the throughput reduction, but more seriously, transmission failures.

In Fig. 3.1, the four nodes, each equipped with an antenna array, are in the transmission range of each other. In (a), if node 1 is a selected receiver in a time slot, in order to ensure its correct decoding only one stream targeted to 1 is allowed to transmit in its neighborhood. Moreover, when both node 1 and 2 are selected as receivers, even though node 4 would be able to transmit up to 2 streams to node 2 (as shown in dashed lines), if simply scheduling the transmissions based on the minimum number of streams allowed in a neighborhood [19], only one stream is allowed to be transmitted around node 2 at a transmission time (e.g., either transmitting from node 3 or from node 4). That is, without differentiating node types, the maximum number of streams allowed to transmit at any time slot is constrained by the candidate receiver which has the smallest array. On the other hand, if every receiver simply considers its own decoding constraint [29], a higher number of transmissions could lead to serious interference and potential decoding failure at nodes with a lower number of antennas. In addition, when the channel between node 4 and 2 can only support one transmission, i.e. the degree-of-freedom is 1, but two streams are transmitted, the streams cannot be decoded at the receiver. The examples indicate that it would lead to either significant throughput reduction in order to not interfere with a node with lower number of antennas or transmission failure if the node heterogeneity and channel rank constraint are not considered in the MAC design. Additional issues will arise if some of the channels are weak, and cannot support good quality transmission.

These practical problems indicate that effective scheduling algorithms need to be designed to alleviate the bottleneck effect and to provide good system performance under any transmission environment. A few strategies may help. First, when the receiver has multiple antennas, the constraints to transmissions due to the lower degree-of-freedom between node pairs may be mitigated with the formulation of cooperative virtual MIMO array. In Figure 3.1(b), node 1, 2 and 3 can transmit concurrently to node 4 and exploit multiplexing gain to improve the throughput. Second, additional capacity gain can be achieved with the exploration of multi-user diversity and antenna selection diversity, in which case, the transmitter nodes and the antenna to use from a node are opportunistically selected based on the channel conditions between different nodes and antennas. Third, when the receiver has very few antennas (Node 1 in Fig. 3.1 (a)), its transmitter could employ precoding to optimally weight the transmissions from multiple antennas to improve the data rate.

As precoding is difficult to apply across multiple transmitters and receivers, it is not used simultaneously with the cooperative multiplexing. In Fig. 3.2, a simple experiment is performed to compare the data rates achieved by opportunistic transmitter precoding (OTPC) and opportunistic and cooperative spatial multiplexing (OCSM) [19] under a topology where two transmitter nodes are around one receiver node with i.i.d faded channels. The performances of the two are compared with the variation of the number of antennas at each node. When the receiver antenna array size is small, transmitter precoding is seen to outperform multi-user multiplexing as power gain is more significant. However, with more receiving antennas, cooperative multiplexing starts to outperform precoding. From this simple example, we can see that it is important to select an appropriate transmission strategy according to specific constrains, in order to achieve optimum possible performance. Instead of transmitting the same signal from multiple antennas with appropriate weighting to increase the rate of one stream as done in conventional beam-forming scheme, in this work each data packet is transmitted only through one selected stream and selected streams from all candidate antenna pairs form many-to-many cooperative MIMO transmissions to improve the total network capacity. Therefore, precoding is only assumed to weight the transmissions when multiple streams are selected to transmit between a node pair.

3.3 System Model

We consider channel resource allocation among an ad hoc network of nodes which have different number of antenna elements and experience different channel conditions. For a group of nodes that share the transmission resource, one node pair is often scheduled to transmit at a time in the traditional MIMO schemes. However, the chance of having multiple strong spatial paths between a node pair is small, which limits the transmission rate. Instead, our scheduling schemes support many-to-many transmissions between nodes using virtual MIMO arrays, and take advantage of multi-user diversity and antenna selection diversity to significantly improve the transmission reliability and throughput. Specifically, to address the challenges due to the network heterogeneity, our algorithms *adaptively and flexibly* schedule node transmissions using different MIMO techniques, including spatial multiplexing, selection diversity, and precoding, based on the node constraints and channel conditions. For the convenience

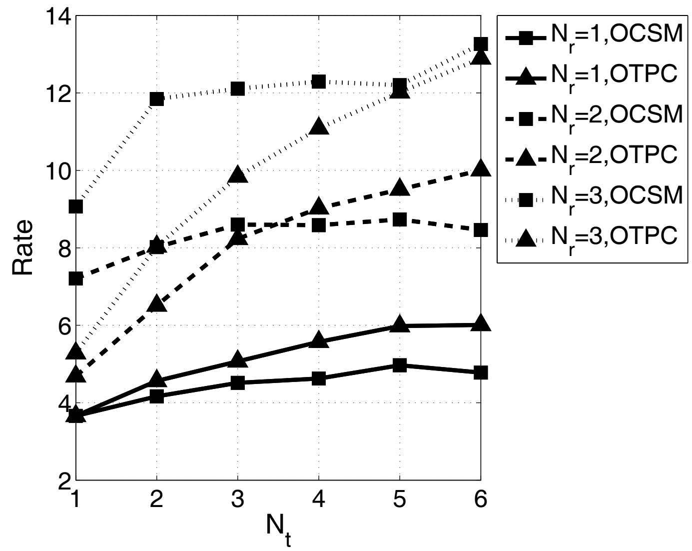


Figure 3.2: Comparison of multiplexing and transmitter precoding with varied transmitter/receiver antenna array size.

of presentation, in this section, we first introduce some notations used in this paper, and then formulate the problem mathematically and prove its NP-hardness.

3.3.1 Stream and Stream Characteristics

A stream is defined to be an *independent* flow of signals transmitted from a transmit antenna to a target node and identified by a triplet $(I^{tx}, I^{rc}, I^{ant})$, where I^{tx}/I^{rc} is the index of the transmitter/receiver node, and I^{ant} is the index of the antenna that involves in the transmissions of the stream. With the exploitation of selection diversity, the antennas with the strongest channel conditions among the candidate ones are selected to transmit the data streams. For a transmitter node with several streams selected, if the streams target for the same receiver, precoding is performed among the selected transmitting antennas with the power optimally allocated to achieve the maximum data rate between a node pair; otherwise, the power is evenly distributed over the selected antennas for streams targeting for different receivers for the processing simplicity.

In order for receiver nodes to decode data streams and suppress interference streams concurrently, the number of streams transmitted or received at a node is subject to certain constraint. Due to the broadcast nature of wireless channels, streams are categorized as *data* *streams* and *interference streams*. A data stream from node n_i to node n_k is received by n_i's neighboring node n_j as an interference stream. Denote the degree-of-freedom of the channel between n_i and n_k as $DoF(i,k)$, it is clear that n_k can differentiate streams from n_i only if the number of streams is no more than $DoF(i,k)$, which depend on both the antenna numbers of n_i and n_k and the correlation level of the channel between the two nodes. Denote the set of all active receiving nodes (i.e., the target receivers of some transmitter nodes) around node n_i's transmission range as R_i^{active}, as the *transmitting constraint*, the number of transmitting streams from n_i should be no larger than $\mathcal{N}_i^{tx} = \min_{k \in R_i^{active}} DoF(i,k)$. Similarly, to avoid erroneous decoding at a receiver node n_k, the number of simultaneous received streams N_k^{rc} (including both data streams and interference streams) should be limited. With use of virtual MIMO array, the size of antenna array N_k^{ant} generally provides the metric of spatial resolution at a receiver n_k, and hence the total received streams should not exceed the *receiving constraint* $\mathcal{N}_k^{rc} = N_k^{ant}$.

The characteristics of a stream are captured by two parameters, *stream priority* $\mathscr{P}(s)$ and *stream capacity* $\mathscr{C}(s)$. The stream priority depends on the service type and queuing delay of the data packet to be sent with the stream. The value of $\mathscr{P}(s)$ is initially set to the service priority of the associated packet, and increases as the queuing time of the packet increases. The stream capacity describes the maximum achievable rate of a stream transmission, which depends on the transmission power of the stream and the channel condition between the transmitter antenna(s) and the receiver node. $\mathscr{C}(s)$ can be estimated at a transmitter based on the estimated channel condition during the scheduling.

3.3.2 Types of Nodes and Slots

Our algorithm is TDMA-based, in which the time domain is divided into transmission durations (TD). A TD consists of several time slots and covers one round of control signal exchange and fixed-size data frame transmission. The data transmission rate within a frame can vary based on the channel condition. For a channel with higher quality, more efficient coding can be used to encode the symbols at a higher rate. A link between a transmitter-receiver pair is half-duplex, so that a node can either transmit or receive but not at the same time.

Denote the set of nodes in the transmission range of node n_i as $\mathscr{V}_i^A$, the receiving constraint of node n_k as $\mathcal{N}_k^{rc}$. Since a node with a higher value of $\mathcal{N}_k^{rc}$ can generally decode more streams, we use $\mathcal{N}_k^{rc}$ as a metric for measuring the receiving capability of n_k. The average receiving capability of nodes in $\mathscr{V}_i^A$ is then represented as $\bar{\mathcal{N}}_i^{rc} = \frac{1}{|\mathscr{V}_i^A|}\sum_{k\in\mathscr{V}_i^A}\mathcal{N}_k^{rc}$. Compared with $\bar{\mathcal{N}}_i^{rc}$, if $\mathcal{N}_k^{rc} \geq \bar{\mathcal{N}}_i^{rc}$, node n_i considers n_k as a *rich* node as it has relatively higher receiving capability among the neighboring nodes of n_i; otherwise, n_k is considered as a *poor* node and could potentially become a receiving bottleneck in the neighborhood. Note that when all nodes have the same number of antennas, the network contains only rich nodes, and it is degenerated to the homogeneous network case.

As discussed in Section 3.2, the limited decoding capability of a poor receiver constrains the maximum number of streams (including both data streams and interference streams) allowable in its neighborhood. To reduce the constraint, we divide the transmission slots into *P-slots* and *R-slots* and assume different transmission strategies towards poor nodes and rich nodes respectively. In a P-slot, the number of concurrent transmission streams is limited by the receiving constraint of the targeted poor node, and transmitter precoding may be utilized to optimize the link rate. In an R-slot, as only rich nodes serve as the receivers, multiuser spatial multiplexed transmissions are opportunistically scheduled for a higher throughput.

3.3.3 Problem Formulation

In a TDMA-based MIMO ad hoc network, packets are generated constantly. It is thus practical to schedule the transmission of packets in each transmission duration (TD) with the purpose of optimizing temporary network performance. Suppose there is a set of nodes $N = \{n_1, n_2, \ldots, n_{N_n}\}$ in the network. Based on their queuing packets, node n_i has a set of candidate streams S_i, where the destination node of the q-th stream $s_{iq} \in S_i$ is denoted as $d(s_{iq})$. Let the parameter set $\{y_{iq}\}$ ($y_{iq} \in \{0,1\}, i = 1,\ldots,N_n, q = 1,\ldots,|S_i|$) denote whether the q-th candidate stream of node i is transmitted in the current TD. If a stream s_{iq} is transmitted, $y_{iq} = 1$; otherwise, $y_{iq} = 0$. Similarly, $\{t_i\}$ and $\{h_i\}$ ($t_i, h_i \in \{0,1\}, i = 1,\ldots,N_n$) are used to denote the transmitter and receiver node assignment in the current TD respectively. If node n_i is selected as a transmitter/receiver node, we have $t_i = 1/h_i = 1$, otherwise $t_i = 0/h_i = 0$. If $t_i = h_i = 0$, node n_i is recognized as an idle node. The assignment of a stream to a specific antenna of a transmitter is represented by the parameter a_{iqk} ($a_{iqk} \in \{0,1\}, i = 1,\ldots,N_n, q = 1,\ldots,|S_i|$ and $k = 1,\ldots,N_i^{ant}$), where $a_{iqk} = 1$ if stream s_{iq} is assigned to transmit from antenna k of node n_i. The transmission rate of stream s_{iq} is impacted by both the strength of the stream $(i, d(s_{iq}), k)$ (denoted as $\mathcal{S}(s_{iq})$), and the interference level at receiver node $d(s_{iq})$ (denoted as $\mathcal{I}(d(s_{iq}))$). The priority of stream s_{iq} depends on the priority of its associated packet and is denoted as $\mathscr{P}(s_{iq})$.

The scheduling process selects a set of streams to transmit among all the candidate ones in the current TD. The objective of the scheduling is to maximize the sum of priority-weighted capacity of the scheduled streams, so that both data rate and priority can be jointly optimized. The problem is formulated as follows:

$$\max U = \sum_{n_i \in N} \sum_{s_{iq} \in S_i} y_{iq} \mathscr{C}(\mathcal{S}(s_{iq}), \mathcal{I}(d(s_{iq}))) \mathscr{P}(s_{iq}); \tag{3.3}$$

$$\sum_{s_{iq} \in S_i} a_{iqk} \leq 1, i = 1, 2, \ldots, N_n, k = 1, \ldots, N_i^{ant}; \tag{3.4}$$

$$\sum_{s_{iq} \in S_i} y_{iq} \leq \mathcal{N}_i^{tx}, i = 1, 2, \ldots, N_n; \tag{3.5}$$

$$h_i \sum_{m \in \mathscr{V}_i} \sum_{\substack{s_{mq} \in S_m \\ d(s_{mq}) = i}} y_{mq} + h_i \sum_{m \in \mathscr{V}_i} \sum_{\substack{s_{mq} \in S_m \\ d(s_{mq}) \neq i}} y_{mq} \leq \mathcal{N}_i^{rc}, \quad i = 1, 2, \ldots, N_n; \tag{3.6}$$

$$t_i + h_i \leq 1, i = 1, 2, \ldots, N_n; \tag{3.7}$$

$$a_{iqk} \leq y_{iq} \leq t_i, a_{iqk} \leq y_{iq} \leq h_{d(s_{iq})}, \quad i = 1, \ldots, N_n, q = 1, \ldots, |S_i|, k = 1, \ldots, N_i^{ant}; \tag{3.8}$$

$$t_i, h_i, y_{iq}, a_{iqk} \in \{0, 1\}.$$

Constraint (3.4) ensures that an antenna can only transmit one scheduled stream at most in each slot. Equation (3.5) constrains the total number of transmitted streams from n_i should be no more than its transmitting constraint value $\mathcal{N}_i^{tx}$, which depends on the antenna numbers of n_i and all its neighboring receivers as well as the channel independency level between n_i and every receiver. Equation (3.6) provides the constraint at receiver n_i where the total number of receiving streams including data streams (the first term on the left side) and interference streams (the second term on the left side) is restricted to be no more than its receiving constraint value $\mathcal{N}_i^{rc}$ in order to decode the receiving packet. Equation (3.7) represents that nodes in the network are half-duplex; and equation (3.8) ensures the parameters to have the correct relationship. So far, we formulate the problem of heterogeneous stream scheduling as an integer programming problem with the objective function in (3.3) subject to constraints (3.4)-(3.8).

Note that the the strength of a stream $\mathcal{S}(s_{iq})$ will reduce if more streams are scheduled to transmit from n_i, which will be incorporated during the stream scheduling process. As the interference $\mathcal{I}(d(s_{iq}))$ will not be known until the scheduling is completed, we will use an average interference level estimated from the past transmissions. In addition, a receiver cannot cancel the interference when the total number of streams it receives is beyond its decoding capability or it does not have channel knowledge, or the interference is due to decoding errors as a result of inaccurate channel knowledge or a-synchronization [29]. The last two types of interference is included in the measured interference. As our MAC design ensures that the number of concurrent transmissions from one-hop transmitters is below the decoding capability of each receiver and with channel estimation, so the un-cancelable interference is from transmitters two hops or more away and is thus weaker. Also, the number of antennas of nodes in the ad-hoc network is generally small, so the number of steps needed for interference cancelation and the error propagation is also limited. Based on the actual decoding quality, the estimated interference level can be adjusted, and set higher to select stronger streams for more reliable decoding at the cost of possible reduction of the number of concurrent streams. Further, as our algorithms schedule stronger streams, it helps to significantly increase the signal to interference plus noise ratio and mitigate the interference impact.

Proposition I: The heterogeneous stream scheduling (HSS) problem described above is NP-hard.

Proof: First we introduce a simplified version of HSS problem represented by a graph $G = (V, E)$. A vertex $v_i \in V$ represents a node n_i, and an edge $e = (v_i, v_k)$ denotes that n_i and n_k are neighbors in the network. Assume each node has a candidate stream s for each of its neighbors, and the gain of scheduling $\mathscr{C}(s)\mathscr{P}(s)$ is 1 for all s. The transmitting and receiving constraints for all n_i are $\mathcal{N}_i^{tx} = \mathcal{N}_i^{rc} = 1$. The optimum scheduling solution of the simplified HSS problem is a maximum set of vertices that can transmit simultaneously while $\mathcal{N}_i^{tx}$ and $\mathcal{N}_i^{rc}$ are satisfied for transmitter and receiver nodes respectively. The simplified HSS problem can be proved to be NP-hard by reducing the NP-complete maximum independent set (MIS) problem to it. For any instance of MIS represented by a graph $G' = (V', E')$, form a new graph $G = (V, E)$ in the following way. Keep the vertex set V' and replace each edge in E' with a dummy vertex, denoted as a set V_d, so that $V = V' \bigcup V_d$. Connect each dummy vertex in V_d to the two original end vertices in V'. The dummy vertices that represent edges connected to the same vertex in G' are also connected in G. It is then straightforward to see that the optimum scheduling solution of the simplified HSS problem in G gives an equivalent solution of MIS problem in G'.$\square$

3.4 Centralized Algorithm

Due to the NP-hardness of the problem, an efficient heuristic algorithm is required to solve the scheduling problem. In algorithm 5, we propose a centralized algorithm. In lines 1-7, a set W is constructed to include all the candidate streams from every node in the network. In lines 8-17, the centralized algorithm greedily schedules the stream with the highest weight for transmission in a TD, while meeting the constraints in equations (3.4)-(3.8). In line 11-12, the selected stream is assigned to be transmitted from the corresponding transmitter to the receiver. As a node cannot be a transmitter or receiver at the same time, in line 13, all the candidate streams that have transmission conflict with the scheduled stream $s = (i^*, d(s_{i^*q^*}), k^*)$ are removed from the set W, including the candidate streams that have the node n_{i^*} as the receiver, have $n_{d(s_{i^*q^*})}$ as the transmitter, or have node n_{i^*} as the transmitter and are associated with the antenna k^*. Next we prove that this simple algorithm has an approximation ratio to the optimal solution.

Algorithm 5 Centralized Scheduling

1: **Initialize:** $W \Leftarrow \oslash$
2: **for** $i = 1$ to N_n **do**
3:   **if** $\exists s_{iq}, \forall q \in \{1, \ldots, |S_i|\}$ **then**
4:     $w(iqk) \Leftarrow \mathscr{R}(iqk)\mathscr{P}(iq), \forall k \in \{1, \ldots, N_i^{ant}\}$
5:     $W \Leftarrow W \bigcup \{w(iqk)\}$
6:   **end if**
7: **end for**
8: **while** $W \neq \oslash$ **do**
9:   $(i^*, q^*, k^*) = \arg\max_{\{i,q,k\}} W$, the corresponding destination node is $d(s_{i^*q^*})$
10:   **if** Selecting $(i^*, d(s_{i^*q^*}), k^*)$ satisfies $\mathcal{N}_{i^*}^{tx}$ and $\mathcal{N}_{d(s_{i^*q^*})}^{rc}$ **then**
11:     Assign n_{i^*}/$n_{d(s_{i^*q^*})}$ as the transmitter/receiver node
12:     Schedule the stream $(i^*, d(s_{i^*q^*}), k^*)$
13:     $W \Leftarrow W \setminus \{w(iqk)|\forall i, q \ s.t. \ d(iq) = i^*, k = 1, \ldots, N_i^{ant}\} \bigcup \{w(iqk)|i = d(s_{i^*q^*}), q = 1, \ldots, |S_i|, k = 1, \ldots, N_i^{ant}\} \bigcup \{w(iqk)|i = i^*, k = k^*, q = 1, \ldots, |S_{i^*}|\}$
14:   **else**
15:     $W \Leftarrow W \setminus w(i^*q^*k^*)$
16:   **end if**
17: **end while**

Proposition 2: The centralized scheduling algorithm can achieve an approximation ratio of $1/\left((2 + \mathcal{D}) \max_i\{N_i^{ant}\} + 2\right)$, where $\mathcal{D}$ is the maximum node degree in the network.

Proof: Let *sol* be our solution, and *opt* be the optimum solution that satisfies equations (3.3)-(3.8). As shown above, in *sol*, some of the candidate streams are suppressed by the selection of a stream s (i.e., removed from W) due to their conflicting with transmission of s, but these streams may be in the set of selected streams in *opt*. Let $\max_i\{N_i^{ant}\}$ be the maximum antenna array size of nodes in the network. According to the constraints (3.4)-(3.8), the selection of the specific antenna k^* suppresses 1 stream as any other stream cannot be transmitted from k^*, the selection of the transmitter node i^* suppresses $\mathcal{N}_{i^*}^{rc}$ streams as i^* can no longer be scheduled as a receiver node, and the selection of the receiver node $d(s_{i^*q^*})$ suppresses $\mathcal{N}_{d(s_{i^*q^*})}^{tx}$ streams as $d(s_{i^*q^*})$ can not be a transmitter node in the current TD. The sum of suppressed streams in the three cases has the upper bound $2\max_i\{N_i^{ant}\}+1$, as both $\mathcal{N}_{i^*}^{rc}$ and $\mathcal{N}_{d(s_{i^*q^*})}^{tx}$ have values no larger than $\max_i\{N_i^{ant}\}$. Moreover, the assignment of transmitter/receiver eliminates their opportunity of being an idle node, while an idle node does not constrain the number of streams it perceives in the neighborhood. Denote the maximum node degree in the network as $\mathcal{D}$, the number of suppressed streams due to this reason should be no more than $\mathcal{D}\max_i\{N_i^{ant}\}$. Therefore, the number of suppressed streams that may be transmitted in one TD should be no more than $(2+\mathcal{D})\max_i\{N_i^{ant}\}+1$.

A stream $s' \in opt$ is considered to be associated with a stream $s'' \in sol$ either because they are identical or because s' is suppressed by s'' during the process of greedy selection. For each stream s_l in *sol*, there is a set Q_l containing the streams in *opt* that are associated with it, and $\bigcup_{s_l \in sol} Q_l = opt$. The number of streams in Q_l, $|Q_l|$, has an upper limit $(2+\mathcal{D})\max_i\{N_i^{ant}\}+2$. As the selection of stream in *sol* is greedy and looks for the one with the largest weight at a time, thus $w(s_l) \geq w(s_m), \forall s_m \in Q_l$. With U defined in equation (3.3), we have:

$$
\begin{aligned}
\frac{U(sol)}{U(opt)} &= \frac{\sum_{s_l \in sol} w(s_l)}{\sum_{s_l \in opt} w(s_l)} = \frac{\sum_{s_l \in sol} w(s_l)}{\sum_{s_l \in sol}\sum_{s_m \in Q_l} w(s_m)} \\
&\geq \frac{\sum_{l \in sol} w(s_l)}{\sum_{s_l \in sol}\sum_{s_m \in Q_l} w(s_l)} = \frac{\sum_{l \in sol} w(s_l)}{\sum_{s_l \in sol} |Q_l| w(s_l)} \\
&\geq \frac{\sum_{l \in sol} w(s_l)}{((2+\mathcal{D})\max_i\{N_i^{ant}\}+2)\sum_{l \in sol} w(s_l)} \\
&= \frac{1}{(2+\mathcal{D})\max_i\{N_i^{ant}\}+2}. \square
\end{aligned}
$$

The centralized algorithm with the proved approximation ratio serves as a benchmark for performance comparison. From the formulation (3.3)-(3.8), it is clear that the scheduling problem has to determine the values of the parameter sets: $\{t_i\}$, $\{h_i\}$, $\{y_{iq}\}$ and $\{a_{iqk}\}$ to assign a packet to an appropriate transmitter antenna in order to maximize the total weighted rate of the network. In a practical distributed half-duplex network, it is reasonable to divide the problem into two parts: transmitter selection and stream allocation, where the first phase determines the values of $\{t_i\}$ and $\{h_i\}$, and the second phase determines the value of $\{y_{iq}\}$ and $\{a_{iqk}\}$ to assign a packet to a specific transmission stream. In the next section, the two subproblems are solved separately.

3.5 Distributed Algorithm and Protocol

In order to address the network heterogeneity, our algorithm groups transmissions into two types, transmissions to poor nodes using *P-slots* and to rich nodes using *R-slots*. The current slot type is determined in a distributed manner by each node and the nodes in a neighborhood reach a consensus through signaling exchange. In both types of slots, spatial multiplexing, selection diversity and transmitter precoding are adaptively utilized to deal with varying traffic demands and channel conditions to improve the overall network performance.

The distributed scheduling algorithm consists of two phases, namely *transmitter node selection / slot request* and *stream allocation*. In the first phase, a set of nodes are first selected to be transmitter nodes, and each node differentiates its packets for poor nodes and rich nodes to determine its current preference of transmission slot type. In the second phase, stream allocation is performed to allocate the data packets of the transmitter nodes to a selected set of antennas with an appropriate MIMO strategy.

In the rest of this section, we first present our scheduling algorithm in sequence of the two phases mentioned above. The complete protocol is then introduced, where we explain the detailed procedures taken to implement the algorithm and calculate the required parameters in a distributed environment.

3.5.1 Transmitter Node Selection and Slot Request

In this phase, nodes are distributively selected as transmitter nodes and their preference of slot type is decided. Instead of randomly selecting the transmitter nodes, the transmitter selection phase supports service differentiation and reduces transmission delay by giving a higher transmission priority to the streams that are with packets in higher service class and/or have larger queuing delay. Additionally, the type of transmission slots is differentiated to support transmissions to heterogeneous nodes. We first give the main idea and define parameters used for the selection, then we discuss the details of the selection process.

Basic Plot

In MIMO transmissions, in order to not exceed the decoding capacity of nodes, the number of streams that can be simultaneously transmitted in a neighborhood is constrained. Therefore, the number of transmitter nodes selected in our algorithm also has a limit, which will avoid unnecessary channel measurement. In addition, the decoding capabilities of receivers, represented by their receiving constraints in Section 3.3.1, are different in a heterogeneous MIMO network. In our algorithm, each node distributively determines if it can serve as a transmitter node in a transmission duration, and selects the type of slot used for transmission based on the decoding capacity of its neighboring receivers.

Based on the receiving constraint, an active node n_i which has data to send groups its neighboring nodes into poor node set $\mathscr{V}_i^p$ and rich node set $\mathscr{V}_i^r$ based on the receiving constraint $\mathcal{N}_k^{rc}$ of a neighbor n_k, which is broadcast with the Hello messages sent periodically at the network layer. We introduce a threshold value $\mathcal{T}_i^{TX}$, which is calculated separately for each of the two sets. Denote the set of neighboring nodes *in concern* as $\mathscr{V}_i$, where $\mathscr{V}_i$ can correspond to $\mathscr{V}_i^p$ or $\mathscr{V}_i^r$ depending on which set is concerned at the calculation time. The parameter $\mathcal{T}_i^{TX}$ of n_i is estimated based on the number of active nodes around a neighboring node $n_j \in \mathscr{V}_i$ (denoted as N_j^{active}) and the receiving constraint of node n_j (denoted as $\mathcal{N}_j^{rc}$) as $\mathcal{T}_i^{TX} = \min\{1, \min_{j\in\mathscr{V}_i}\left(\mathcal{N}_j^{rc}/N_j^{active}\right)\}$. To support some transmission fairness, the neighboring transmitters of n_j can be evenly allocated the transmission opportunities based on the decoding constraint of n_j. Therefore, $\mathcal{T}_i^{TX}$ represents the probability of a node n_i being a transmitter in order to ensure all neighbors in $\mathscr{V}_i$ to perform the correct decoding. A node n_i can be selected as a transmitter if the value of an appropriately calculated random variable is below $\mathcal{T}_i^{TX}$.

Recall that we use stream priority to represent how urgent a stream transmission is. It is therefore natural to use the average stream priority to reflect the level of priority for a node to be a transmitter. Denote all candidate streams (i.e. the head-of-queue packets with the number constrained by the number of antennas of n_i) of n_i as a set S_i and the priority of a stream s_{iq} as $\mathscr{P}(s_{iq})$, the priority of a node n_i can be represented by the average priority of its candidate streams as $\mathscr{P}_i = \sum_{s_{iq}\in S_i}\mathscr{P}(s_{iq})/|S_i|$. A node n_i can calculate the average priority $\bar{\mathscr{P}}_i$ of all the N_i^{active} active nodes in its neighborhood as $\bar{\mathscr{P}}_i = \sum_{j=1}^{N_i^{active}} \mathscr{P}_j / N_i^{active}$. The priority of a node can be attached with periodic Hello messages sent at the network layer, and updated with the data packets sent. The priority of nodes not having packets sent in a TD can be predicted as time moves forward.

To avoid extra signaling and control overhead, an active node n_i *self-decides* if it should be selected as a transmitter node by calculating an index number $\mathcal{X}_i^{TX} = (\bar{\mathscr{P}}_i - \hat{\mathscr{P}}_i)/\bar{\mathscr{P}}_i + \gamma_i$. Here the parameter γ_i is a random number uniformly distributed in the range [0,1] and generated by a node n_i at each transmission duration (TD) to provide some fairness among nodes. $\hat{\mathscr{P}}_i$ is the average priority of candidate streams at node n_i that are targeted for nodes in $\mathscr{V}_i$. The factor $(\bar{\mathscr{P}}_i - \hat{\mathscr{P}}_i)/\bar{\mathscr{P}}_i$ is used to give the higher priority node a larger probability for transmission. In a TD, if $\mathcal{X}_i^{TX} < \mathcal{T}_i^{TX}$, node n_i is selected as a transmitter node for receiver nodes in $\mathscr{V}_i$; otherwise, it has no right of transmission. Our transmitter selection algorithm prefers a node with a higher service level and/or a larger load and hence longer delay, and thus supports QoS and load balancing while ensuring certain fairness. Our selection is conservative as it considers the decoding capability of all the neighboring nodes instead of only that of the actually selected receiver nodes known only after the scheduling.

Selection Process

To give priority to transmissions towards poor nodes, at the beginning of a transmission duration, an active node n_i first determines whether it needs to initiate a transmission using P-slot based on the priority of its streams targeted for poor nodes in $\mathscr{V}_i^p$. For the subset of candidate streams in S_i destined to poor nodes in $\mathscr{V}_i^p$, their average priority can be calculated as $\mathscr{P}_i^p = (\sum_{k\in\mathscr{V}_i^p} \sum_{m\in S_{i,k}} \mathscr{P}(m)) / \sum_{k\in\mathscr{V}_i^p} |S_{i,k}|$, where $S_{i,k}$ is the set of candidate streams from node n_i to the poor node n_k. Let $\mathscr{V}_i = \mathscr{V}_i^p$ and substitute $\hat{\mathscr{P}}_i$ by $\mathscr{P}_i^p$ for calculating the index $\mathcal{X}_i^{TX}$, which is compared with $\mathcal{T}_i^{TX}$ calculated based on nodes in $\mathscr{V}_i^p$. If $\mathcal{X}_i^{TX} < \mathcal{T}_i^{TX}$, node n_i can be a transmitter node and initiate a P-slot transmission. The P-slot streams are selected so that the receiving constraints are satisfied at a targeted poor receiver. Otherwise, node n_i checks if it can be a transmitter using R-slot. Similar to the previous step, $\mathcal{T}_i^{TX}$ is calculated concerning nodes in $\mathscr{V}_i^r$ and $\mathcal{X}_i^{TX}$ is obtained by letting $\hat{\mathscr{P}}_i$ equal to $\mathscr{P}_i^r = (\sum_{k\in\mathscr{V}_i^r} \sum_{m\in S_{i,k}} \mathscr{P}(m)) / \sum_{k\in\mathscr{V}_i^r} |S_{i,k}|$, where $S_{i,k}$ is the set of candidate streams which are from node n_i to a rich node n_k. Node n_i is selected as a transmitter node for receiver nodes in $\mathscr{V}_i^r$ if the updated parameters satisfy $\mathcal{X}_i^{TX} < \mathcal{T}_i^{TX}$.

If a node determines to be a transmitter node, it broadcasts an RTS message indicating the slot type as discussed in 3.5.3. After the transmitters and the slot types are confirmed by the receiver nodes through CTS transmission, the transmitter nodes proceed to the second phase of the scheduling described next.

3.5.2 Stream Allocation

Stream allocation is performed distributively at each of the selected transmitter nodes. The selection gives preference to streams with higher priority. For streams of the same priority, to achieve a higher data rate, the allocation process is solely based on the stream capacity by *opportunistically* assigning a channel with good condition to a selected stream. For a high-priority stream that does not have high-quality channel, the selection process reserves more of the total transmitting power for the stream to ensure a higher transmission reliability.

For a selected transmitter, there is a limit on the number of streams it is allowed to transmit, in order to meet the receiving constraints at all neighboring receivers. For a selected transmitter n_i, let N_i^0 be the number of pre-selected streams to be transmitted and N_i^{allo} be the number of streams node n_i is allowed to transmit, which is calculated based on feedbacks from neighboring receivers as described in Section 3.5.3. Suppose the N_i^0 candidate streams have L_i distinct priority levels. The receiver nodes that the candidate streams are targeted for are then partitioned into subsets $\{D_i^1\}, \{D_i^2\}, \ldots, \{D_i^{L_i}\}$ according to the descending priorities of the streams, where the set $\{D_i^j\}$ contains the target receiver nodes of the streams with the j-th highest priority level, and the q-th element in $\{D_i^j\}$ is denoted as $D_i^j(q)$. Recall that a stream s is identified by its transmitter node, receiver node and transmitter antenna. Denote the set of antennas that node n_i has as $\{A_i\}$, and the p-th element is $A_i(p)$. For a stream of n_i which has the receiver $D_i^j(q)$ and transmitting antenna $A_i(p)$, the stream capacity $\mathscr{C}(i, D_i^j(q), A_i(p))$ depends on the stream strength and the estimated interference level at the receiver node $D_i^j(q)$, as discussed in Section 3.3.3. For transmitter node n_i, there is a set W_i^0 consisting of all the capacity parameters of the candidate streams $W_i^0 = \bigcup_{j=1}^{L_i}\{\mathscr{C}(i, D_i^j(q), A_i(p))|A_i(p) \in \{A_i\}, D_i^j(q) \in \{D_i^j\}, p = 1, \ldots, |\{A_i\}|, q = 1, \ldots, |\{D_i^j\}|\}$.

The procedure of stream allocation is described in the algorithm 6, where j is the index of the priority level, $\{A_i\}^{res}$ is the set of remaining available antennas of node n_i and N_i^{res} is the residual number of streams to allocate. The initial value of N_i^{res} is set to be the total number of streams for allocation N_i^{allo}. As in lines 2-11, the algorithm starts from the set of candidate streams which have the highest priority ($j = 1$), and calls the subroutine *OPPORTUNISTIC_ALLOCATION* as in algorithm 7 for each priority level, until all the allowed streams have been allocated or the antennas of node n_i have all been assigned or reserved for streams. In lines 12-16, power is allocated to the selected antennas based on the transmission pattern. As described in section 3.3.1, precoding is used to maximize the data rate between a node pair if all streams are scheduled to transmit towards the same receiver where optimal power allocation is performed through water-filling; when streams are towards different receivers, power is simply distributed evenly among the antennas.

Algorithm 6 Distributed Scheduling

1: **Initialize:** $j = 1$, $\{A_i\}^{res} = \{A_i\}$, $N_i^{res} = N_i^{allo}$
2: **while** $N_i^{res} > 0$ **do**
3:     **if** $|\{D_i^j\}| \leq N_i^{res}$ **then**
4:         *OPPORTUNISTIC_ALLOCATION*$(\{A_i\}^{res}, \{D_i^j\}, |\{D_i^j\}|, N_i^{res})$
5:         $N_i^{res} = N_i^{res} - |\{D_i^j\}|$
6:     **else**
7:         *OPPORTUNISTIC_ALLOCATION*$(\{A_i\}^{res}, \{D_i^j\}, N_i^{res}, 0)$
8:         $N_i^{res} = 0$
9:     **end if**
10:     $j \Leftarrow j + 1$
11: **end while**
12: **if** All streams are towards one receiver **then**
13:     Use precoding and optimal power allocation
14: **else**
15:     Power is evenly distributed
16: **end if**

The subroutine *OPPORTUNISTIC_ALLOCATION* is described in algorithm 2 to allocate k antennas to transmit the streams of the j-th highest priority level that are targeted for the receiver set $\{D_i^j\}$. The parameter N_i^{res} is the residual number of antennas available for allocation, the set $\{A_i\}^{res}$ contains the candidate antennas of node n_i for stream allocation, W_i^j contains the capacity parameters of the streams formulated between the antennas in $\{A_i\}^{res}$ and the receivers in $\{D_i^j\}$ and l represents the number of streams currently allocated. The allocation is based on spatial multiplexing and selection diversity, and in sequence of descending stream quality. As the allocation scheme favors stream priority than stream quality, in some cases, although the channel condition is severe, a transmission with a high priority is still permitted. To reduce erroneous decoding thus packet loss under the severe channel condition, when a selected stream does not have good enough quality as indicated by a weak channel

Algorithm 7 *OPPORTUNISTIC_ALLOCATION* $(\{A_i\}^{res}, \{D_i^j\}, k, N_i^{res})$

Initialize: $l = 0$, $W_i^j = \{\mathscr{C}(i, D_i^j(q), A_i^{res}(p)) | A_i^{res}(p) \in \{A_i\}^{res}, D_i^j(q) \in \{D_i^j\}, p = 1, \ldots, |\{A_i\}^{res}|, q = 1, \ldots, |\{D_i^j\}|\}$

while $l < k$ **do**

  $W_{max} \Leftarrow \max W_i^j$, $\{A_{max}, D_{max}\} \Leftarrow \arg\max W_i^j$

  Allocate the stream for the receiver D_{max} to the antenna A_{max};

  $W_i^j \Leftarrow W_i^j \setminus \{W(A_{max}, D_i^j(q)) | D_i^j(q) \in \{D_i^j\}, q = 1, \ldots, |\{D_i^j\}|\}$; if there is no other stream target for the receiver node D_{max}, also remove $\{W(A_i^{res}(p), D_{max}) | A_i^{res}(p) \in \{A_i\}^{res}, p = 1, \ldots, |\{A_i\}^{res}|\}$;

  if D_{max} has sent indicator of weak channel **then**

    if $N_i^{res} > 0$ **then**

      $k \Leftarrow k - 1$, $l \Leftarrow l + 1$, $N_i^{res} \Leftarrow N_i^{res} - 1$;

    else $\{N_i^{res} = 0\}$

      $k \Leftarrow k - 1$

    end if

  end if

  $\{A_i\}^{res} \Leftarrow \{A_i\}^{res} \setminus A_{max}$

  $l \Leftarrow l + 1$

end while

indicator include in the CTS (Section 3.5.3), the total number of antennas available for allocation of this stream is decreased by one to reserve extra transmitting power for the weak stream to improve its quality, as in lines 6-12.

3.5.3 Implementation of Distributed Scheduling

To enable the proposed many-to-many transmission and better exploit various diversity techniques for higher capacity and reliability, the implementation of the distributed scheduling algorithm is TDMA-based, where the time is divided into a serials of transmission duration consisting of four phases with different lengths. The duration of each phase is fixed and enough for the corresponding message transmission. Following the convention of IEEE 802.11 DCF, signaling messages are named RTS, CTS, DATA and ACK, which are transmitted during phase I, II, III and IV respectively. Note that slot synchronization is currently achievable in the IEEE802.11 family of protocols. By taking advantage of the selection diversity and multi-user diversity, our scheme could effectively increase the SINR of a received signal, which would help improve the accuracy of synchronization as well as mitigate the impact of a-synchronicity in a distributed scenario. The procedure of signal exchange and information acquisition for heterogeneous MIMO scheduling is as follows.

Phase I: Transmission Request and Slot Conservation. At the beginning of phase I, a node n_i which selects itself as a transmitter node as in Section 3.5.1 broadcasts an RTS. Before sending out the RTS, node n_i selects a set of highest-priority data packets from its queue to form $N_i^0 \leq N_i^{max}$ candidate streams, where N_i^{max} is the maximum number of streams that can be transmitted in a transmission duration depending on the number of antennas of n_i, and the amount of data queued. The IDs of the target receiver nodes of the selected packets, the value N_i^0, as well as the ID of node n_i are then included in the RTS. If n_i wants to request a P-slot towards node n_k, an RTS should further carry an indicator of P-slot and the calculated average priority $\mathscr{P}_i^p$.

The preamble of a packet is used as the training sequence for the channel estimation purpose. After the RTS is transmitted from the first antenna of the transmitter node, for both types of slot, the preamble is rotationally broadcasted through the remaining antennas of the transmitter node with a short notice signal separating two antennas' transmissions, so that the spatial channels between each antenna of the transmitter nodes and the receiver nodes can be differentiated and estimated. An RTS is masked by another random code, called ID code, which are almost orthogonal for different nodes and assigned similarly to that in [36], so a receiver node can get the channel information of different transmitter nodes from concurrently received RTSs. Our transmitter node selection algorithm in Section 3.5.1 adaptively selects a subset of nodes in a neighborhood to participate in channel estimations based on the decoding capabilities of nodes in the neighborhood, which not only reduces the channel estimation complexity and avoids unnecessary channel estimations but also constrains the total interference in a neighborhood for better decoding.

Phase II: Transmission Confirmation. Upon receiving multiple RTSs, a receiver correlates its received signals with each element in its set of random codes to differentiate the training sequences from different transmitter nodes, estimates spatial channels and extracts other information included in RTSs.

If a node n_k receives a request for P-slot transmission to itself, it sorts all P-slot requests it receives (for itself or for other receiver nodes) based on the request priorities. When multiple requests have the same priority, the request for the receiver with a higher ID is preferred. The receiver n_k then checks the number of P-slot transmissions allowed in the neighborhood from higher priority to lower priority until all the requests are accommodated or n_k is fully-loaded with data and/or interference streams. Denote the number of P-slot requests accommodated at node n_k as $N_{k,p}^{dec}$, which does not exceed the receiving constraint of n_k, $\mathcal{N}_k^{rc}$. If n_k is a target receiver of some of the accommodated requests, it considers the current transmission duration as P-slot and broadcasts a CTS with its list of confirmed P-slot requests.

If n_k is only the target receiver of some R-slot requests, while it may overhear some P-slot requests for other receivers, it checks whether it has enough residual stream $N_{k,r}^{dec} = \max\{0, \mathcal{N}_k^{rc} - N_{k,p}^{dec}\}$ for R-slot transmission. If $N_{k,r}^{dec} > 0$, it considers the current transmission duration as R-slot. Different from P-slot transmission in which a transmitter node pre-selects a target receiver, transmission streams are flexibly selected for different receivers in R-slot based on the channel condition to improve the aggregate data rate. After node n_k decodes the information in RTSs from all the selected transmitter nodes in its neighborhood, it learns the number of R-slot streams it may receive in the current duration, $N_{k,r}^0$, including the data streams targeted to itself and the interference streams targeted to other nodes. Denote all transmitter nodes in the one-hop neighborhood of n_k as $\mathscr{V}_k^t$, and each transmitter n_j requires N_j^0 R-slot streams for transmission, we have $N_{k,r}^0 = \sum_{j \in \mathscr{V}_k^t} N_j^0$. Node n_k then broadcasts $N_{k,r}^0$ and $N_{k,r}^{dec}$ through CTS.

A stream may have poor quality, when there is a long distance or deep-fading channel between a transmitter and a receiver. A receiver estimates the strength of a data stream based on the signal-to-noise-ratio (SNR) of the training signal. If the received SNR is lower than a threshold, it includes a weak-channel indicator in the CTS to inform the transmitter to select a more reliable transmission scheme.

To allow the transmitter to estimate the spatial channels to the receiver, the preamble of CTS is utilized as a short training sequence and rotationally broadcast from node n_k's antennas $1 \sim N_k^{ant}$, as in the case of RTS. A CTS signal is also masked by the ID code of n_k.

Phase III: Stream Allocation and Transmission. By differentiating multiple CTSs and extracting the information included, a node n_i estimates the channel matrix $\mathbf{H}_{ki}$ between itself and each active receiver node n_k, and obtains its transmitting constraint value $\mathcal{N}_i^{tx}$. Specifically, if node n_i sends out a P-slot request in the RTS phase, it checks if its P-slot request has been confirmed by all the CTSs. Denote the number of streams confirmed by CTSs as $\tilde{N}_i^0$, so the number of streams allowed for transmission can be calculated as $N_i^{allo} = \min\{\mathcal{N}_i^{tx}, \tilde{N}_i^0\}$. Node n_i allocates the stream following the procedure in 3.5.2 according to the estimated spatial channels.

If n_i receives a confirmation for its R-slot request, it has to determine N_i^{allo} based on the total R-slot confirmations included in the CTSs from rich neighboring receivers in the set $\mathscr{V}_i^r$. Each responding receiver n_k sends back the total number of streams it may receive, $N_{k,r}^0$, the maximum number of streams it can decode, $N_{k,r}^{dec}$, and possibly weak-channel indicators. In order to ensure all the receiver nodes in its neighborhood to have a high probability of correct decoding, node n_i constrains its number of sending streams to a rounded integer number as $N_i^{allo} = \min\{\mathcal{N}_i^{tx}, N_i^0 \min_{k=1}^{N_i^r} \left(N_{k,r}^{dec}/N_{k,r}^0\right)\}$.

With the estimation of all spatial channels between n_i and its target nodes, the set W_i^0 of stream capacity factors is constructed and the stream allocation described in 3.5.2 is then performed to transmit the data streams through the selected antennas. Meanwhile, receiver nodes decode streams from the neighboring transmitter nodes using channel coefficients estimated in phase I.

Phase IV: Acknowledgement. If a data stream is decoded correctly, the receiver node has to confirm the reception. An ACK is masked with the ID code of the receiver and broadcast, carrying the IDs of the transmitter nodes whose streams have been correctly received.

In phase IV, all transmitter nodes are in listening mode. A transmitter node extracts the information in ACKs and removes the correctly received data packets from the queue, and keeps the erroneously received or lost data packets in the queue for scheduling in the next transmission duration.

Note that random ID codes are only used for differentiation in control signal transmission. As control signals are relatively short and sent at the maximum power, there is no significant overhead induced for packet encoding and decoding and there is no need for power control.

3.5.4 Examples

In this section, we give two examples to illustrate the process of the stream allocation algorithm based on the simple topology as in Fig. 3.1.

Suppose that each of nodes 2 and 3 has packets for both nodes 1 and 4. Depending on the antenna array sizes, both 2 and 3 regard node 1 as a poor node and node 4 as a rich node. Following the transmitter selection scheme, both node 2 and 3 may select themselves to be transmitters. As poor nodes have higher priority to receive packets, node 2 and 3 may both initiate P-slot transmission towards node 1 in the first TD. As node 1 can only receive one stream, it confirm the request from the node with the higher priority, say it is node 3. To avoid interference, node 2 cannot transmit at this TD. In the second TD, node 2 still initiates a P-slot transmission to node 1. After confirmed by node 1, one stream is transmitted in this TD. In the third TD, both 2 and 3 start to initiate R-slot transmission towards node 4, each with two streams in the RTS request. Node 4 then feeds back $N^0_{4,r} = 4$ and $N^{dec}_{4,r} = 3$ in the CTS. Following the procedures of Phase III, nodes 2 and 3 transmit three streams in total, i.e. two from node 2 and one from node 3, all towards node 4. As the streams from each sender are transmitted towards one receiver in each TD, the streams are precoded with the power allocated optimally among transmitting antennas. Note that in the counterpart algorithms where heterogeneity is not considered, nodes 2 and 3 can only transmit one stream in total in each TD, as node 1 is an active receiver and it always restricts the number of streams in the neighborhood.

Consider another scenario for the same topology that only nodes 1 and 4 have packets for nodes 2 and 3. As node 2 and 3 are considered as poor nodes by node 1 and rich node by node 4, node 1 first initiates P-slot transmission towards 2 and 3 respectively in two consecutive TDs. As node 1 only has one antenna, it can only transmits one stream in each TD. As a result, node 2 and 3 still have one DoF which can be used to receive a stream from node 4. This stream is selected by node 4 as described in 3.5.2, and the stream with the highest capacity in the first candidate stream subset $\{D^1_4\}$ is selected. So there are 2 streams transmitted in the network in each of the first two TDs. Suppose that node 4 still has packets for 2 and 3 in the following TD, R-slot transmission requests are thus sent towards node 2 and node 3 simultaneously. If the channels are rich-scattered, the DoF of the channel between 4 and any of the two receivers is $\min\{N^{ant}_4, N^{ant}_2\} = \min\{N^{ant}_4, N^{ant}_3\} = 2$. According to the transmitting degree constraint, only two streams can be transmitted. Node 4 therefore selects two candidate streams following

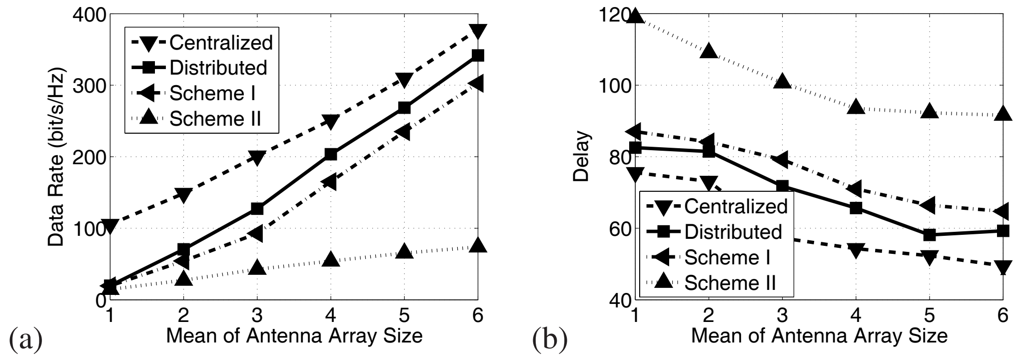


Figure 3.3: Impact of mean of antenna array size: (a) data rate; (b) delay.

the procedure in 3.5.2. If the two streams selected are towards two different receivers 2 and 3 respectively, power is distributed evenly over the two antennas; if the two streams selected are towards the same receiver, precoding and optimal power allocation are assumed. If there is strong LOS component presented between node 4 and node 1, and $DoF(4,1) = 1$, we have $\mathcal{N}_4^{tx} = 1$ and node 4 can only transmit one stream as a result. The stream selection is done by node 4 similarly as that in the first two TDs.

3.6 Performance Evaluation

In this section, we evaluate the performance of our proposed algorithms through simulations. Nodes are distributed uniformly over a $1250m \times 1250m$ area and form an ad-hoc network with random topology. Each node has a transmission range of 250m. The bandwidth of the channel is $20MHz$, and the length for data slot of a TD is $2.5ms$ so the MIMO channel can be considered as quasi-static during a TD [40]. To model a heterogeneous MIMO ad hoc network, we assume the antenna array sizes of nodes in the network are normally distributed with a given mean and variance. The channel is modeled based on the antenna array sizes, the distance between nodes and the small-scale fading coefficients following Rayleigh/Ricean model. The incoming traffic is Poisson distributed with a given mean value λ and the sources and destinations are chosen at random. The size of a packet is 1000 bytes. A result is obtained by averaging over 10 runs of simulations with different random seeds.

The distributed scheduling algorithm proposed in Section 3.5, including both transmitter nodes selection 3.5.1 and stream allocation 3.5.2, is implemented based on the protocol described in Section 3.5.3. Compared with conventional scheduling strategies in MIMO ad hoc

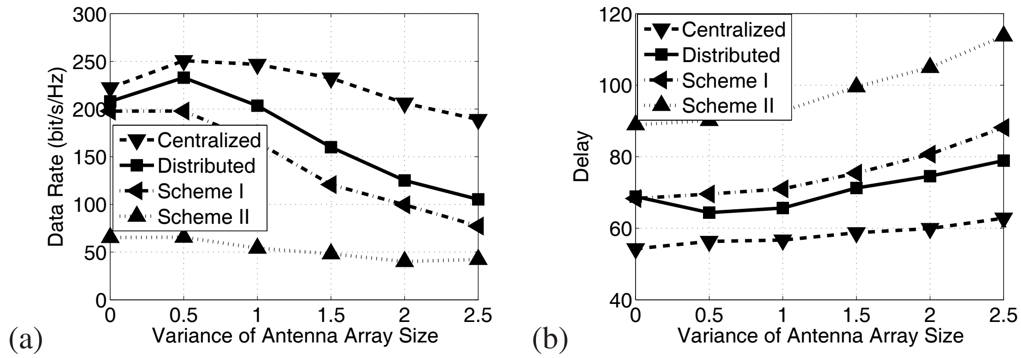
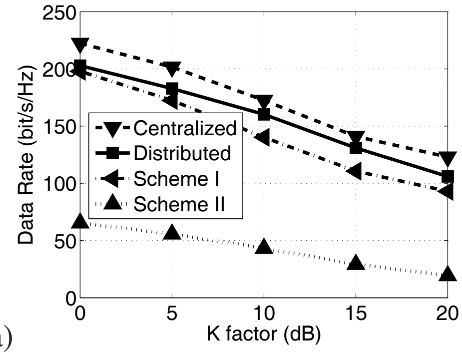


Figure 3.4: Impact of variance of antenna array size: (a) data rate; (b) delay.

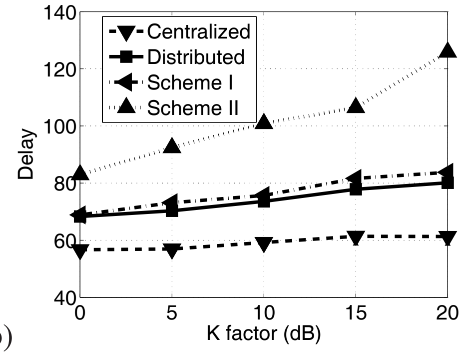
networks, our distributed algorithm has the following unique features: adaptive use of different transmission strategies based on node types and channel conditions, and enabling multi-user to multi-user transmissions exploiting both cooperative multiplexing and selective diversity. To demonstrate the benefits of these features, we design two alternative schemes here for reference. *Scheme I* is based on the opportunistic and cooperative spatial multiplexing scheme proposed in [19], which supports many-to-many cooperative transmission, but does not have specific strategies to handle the heterogeneity of nodes and channels. *Scheme II* takes the conventional scheduling strategy in MIMO ad hoc networks, where during each TD only one pair of transmitter/receiver nodes is allowed to communicate in a neighborhood with as many streams as possible. In each transmission duration, the node pair with the best channel quality is selected, and transmitter node selection is also implemented here to reduce collision. To provide a benchmark for performance comparison, we also implemented the centralized scheduling algorithm proposed in Section 3.4.

The metrics we use for comparison are aggregate data rate and average delay. Aggregate data rate is the total data rate of the network averaged over the number of transmission durations. Average delay is the average number of transmission durations a packet waits in the queue before it is successfully transmitted. We investigate the impact of a set of factors on performance, namely the mean value and variance of antenna array size, the LOS component, node density, traffic arrival rate and mobility. For each factor, the centralized algorithm and distributed algorithm as well as the two reference schemes are implemented, and both data rate and average delay are compared. If not otherwise specified, the value of K-factor is 0, the number of nodes in the network is 100, the mean and variance of degree-of-freedom are 4 and 1 respectively, the average packet arrival rate λ is 5 packets per link and the network is static.

(1) Impact of the Mean of Antenna Array Size. The mean value of antenna array size



(a)



(b)

Figure 3.5: Impact of LOS component: (a) data rate; (b) delay.

determines the average node transmission capability, and it impacts the overall capacity of the network. In Fig. 3.3, as the mean value grows, all schemes except scheme II obtain significantly higher data rate and lower delay, as these schemes can better exploit the multiplexing gain and diversity gain allowed by a larger antenna array to form many-to-many transmissions with a larger number of selected streams. Particularly, the formulation of cooperative multiplexing transmissions alleviates the limitation of the number of transmit antennas. In contrast, the performance of scheme II is constrained by both the number of transmit antennas and receive antennas, and a poor node could lead to more severe impact on the network capacity. Thus, its performance improvement is not as high as other schemes. Both the centralized algorithm and the distributed algorithm obtain consistently higher data rate and lower delay than scheme I under all mean values, as our schemes better alleviate the constraints due to the heterogeneity of the nodes.

(2) Impact of the Variance of Antenna Array Size. The variance of antenna array size reflects the degree of heterogeneity of nodes in the network. The larger the variance is, the greater the variety of antenna array size is and the portion of poor nodes may become higher. As shown in Fig. 3.4, when the variance is 0, which is the homogeneous case, the distributed algorithm and scheme I have very close performance. The slightly higher data rate achieved by the distributed algorithm is due to the use of pre-coding. When the variance increases to 0.5, the total rate of the network increases taking advantage of the receiver nodes with larger antenna arrays. The performances of all the distributed algorithms start to decrease when the variance increases beyond 0.5. As the lowest antenna number in the neighborhood is used to constrain the total allowable number of transmission streams in scheme I, its performance degrades faster than scheme II. Our distributed scheme achieves 23% higher rate than scheme I when variance equals 1. As the variance increases further, the performance of

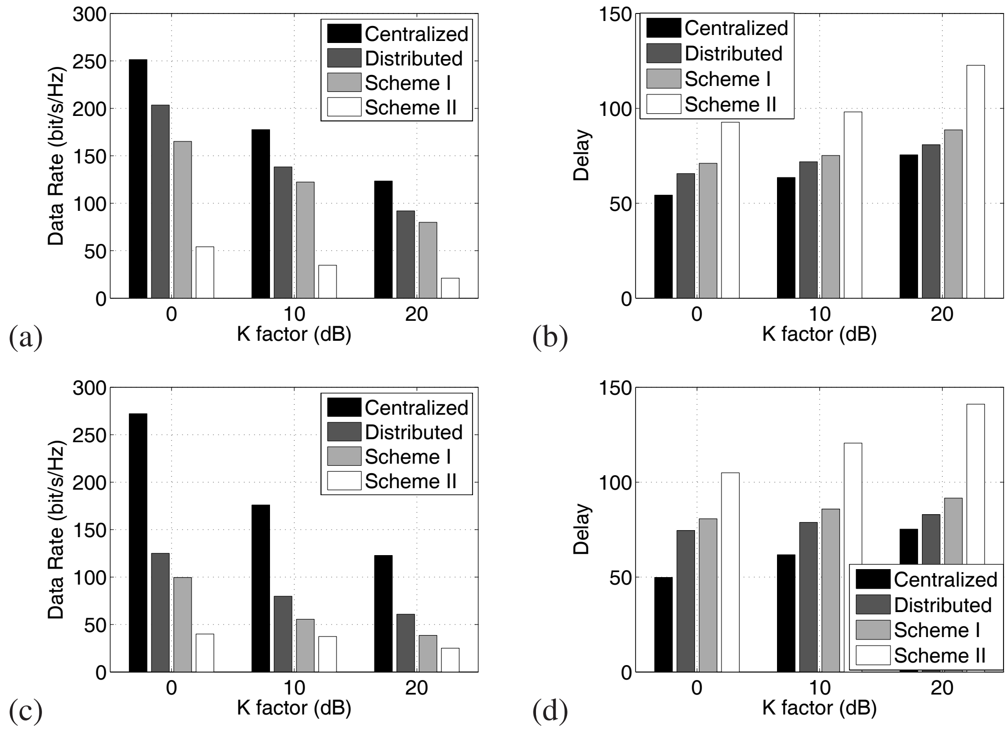


Figure 3.6: Impact of LOS component and variance of antenna array size. With variance = 1: (a) data rate; (b) delay. With variance = 2: (c) data rate; (d) delay.

scheme I is constrained more by the bottleneck effect at receiver nodes, so the gain of our distributed algorithm constantly increases, achieving up to 36% higher rate and 12% lower delay. This demonstrates that by differentiating between poor nodes and rich nodes and adaptively scheduling transmissions in the network based on the number of antennas at the receiver nodes and channel conditions, our distributed algorithm can effectively alleviate the impact of node heterogeneity and channel variations to achieve better performance. Without being limited by the poorest receiver in the neighborhood, the performance of scheme II reduces slower than other schemes when the variance is extremely high. However, the chance of having extremely high heterogeneity in the network is low, and all the schemes exploiting many-to-many transmissions still achieve significantly higher data rate and lower delay compared to scheme II at the highest variance studied.

(3) Impact of LOS Component. As described in section 3.2, the degree-of-freedom of a MIMO channel not only depends on the antenna array size of the end nodes, but also the channel condition of the link. When a LOS component exists, it can impact the correlation between the spatial channels of a link and possibly decrease the degree-of-freedom of the MIMO channel. The impact of the LOS component is described by the K-factor, whose value is generally in the range of $0 \sim 20dB$ in practice. In Fig. 3.5, the performance of the algorithms under different values of K-factor is studied. Nodes are all equipped with antenna array of size 4. With the increased value of K-factor, i.e., the increase of the strength of the LOS component, the degree-of-freedom of MIMO channels all over the network tends to decrease, which results in fewer orthogonal spatial channels over a link and therefore the degradation in rate and delay for all the schemes. As scheme II mainly relies on the degree of freedom thus the multiplexing gain of a single-link, its performance is impacted most significantly by a stronger LOS component thus lower degree of freedom of the channel, with 70% degradation in data rate and 40% increase in delay. The other algorithms can take advantage of multiuser diversity to schedule streams opportunistically, so the impact of LOS component on a single link is mitigated with the support of concurrent transmissions among multiple node pairs. Thus the degradations of data rate and delay of scheme I reduce to 55% and 18% respectively. Taking consideration of the impact of the LOS component and channel degree of freedom constraint, our distributed scheduling algorithm adjusts the number of streams accordingly to avoid transmission failure, and obtains up to 18% improvement in data rate and 5% decrease in delay compared to scheme I.

In practice, the degree-of-freedom of a MIMO channel is concurrently impacted by the LOS component and the antenna array size. In Fig. 3.6 (a) and (b), we present the performance of the algorithms with K-factor of values $0dB$, $10dB$ and $20dB$, and the variance of antenna array size is 1; and in (c) and (d), the variance is changed to 2. The results are consistent with the study of impacts of LOS component and variance of antenna array size independently. With a stronger LOS component at $20dB$ and a larger variance at 2, the distributed algorithm is shown to have more significant performance improvement over both scheme I and II, with 56% higher data rate and 10% lower delay compared to scheme I, and 2.4 times the data rate and 41% lower delay compared to scheme II. The results demonstrate our proposed algorithm is very effective in handling the heterogeneity of network nodes and the variation of channel conditions, and can efficiently exploit the multi-user diversity and antenna selection diversity in a distributed network environment to achieve significant performance gain.

(4) Impact of Node Density.

The impact of node density is shown in Fig. 3.7. Irrespective of the density, the distributed algorithm has the closest performance to that of the centralized algorithm in terms of both aggregate data rate and normalized delay. As the node density increases, the aggregate data rates of all schemes increase as they can better take advantage of the multiuser diversity.

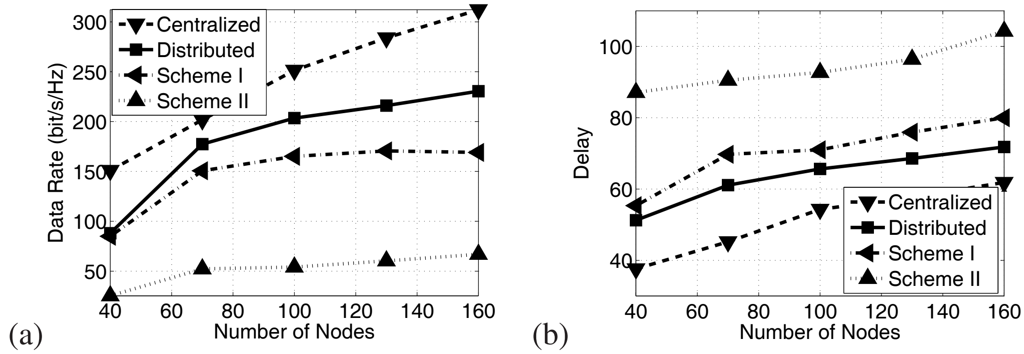


Figure 3.7: Impact of node density: (a) data rate; (b) delay.

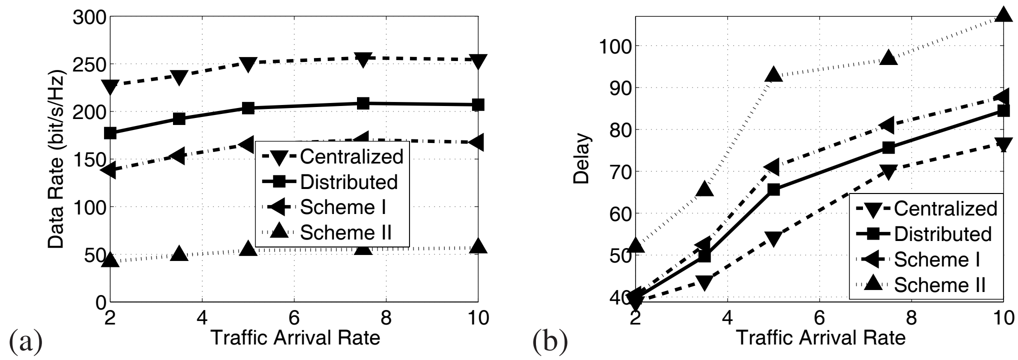
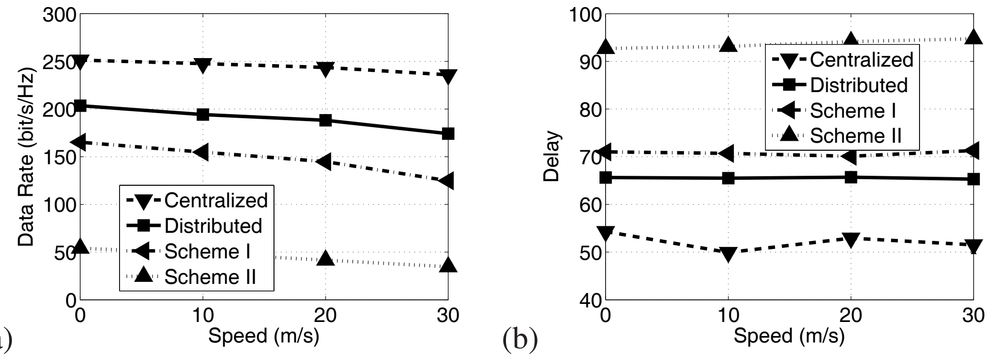


Figure 3.8: Impact of traffic arrival rate: (a) data rate; (b) delay.

Compared with scheme I, with adaptive selection of transmission strategy based on node types and channel conditions, our distributed algorithm is shown to have up to 36% higher aggregate data rate and 14% lower delay. Compared to scheme II, the distributed algorithm achieves up to 3.6 times the data rate and 31% lower delay. As expected, with only single-user to single-user links, scheme II cannot exploit the transmission potential of nodes and has the lowest data rate and the highest delay.

(5) Impact of the Traffic Arrival Rate. The traffic arrival rate is denoted by the parameter λ, which is the mean value of the number of packets arrived at the queues of each nodes in each TD, with each queue corresponds to a specific receiver. The value of λ impacts the network performance. If the value is too low, the network is not fully utilized for packet transmission and some of the transmission capacity is wasted. In Fig. 3.8, when $\lambda = 2$, the data rate is relatively low. If the value of λ is too high, the network may be overloaded which results in an excessive queuing delay of packets. As in Fig. 3.8, the data rate of each many-to-many scheme initially increases with the increase of traffic and keep almost constant beyond $\lambda \geq 5$,



(a) (b)

Figure 3.9: Impact of mobility: (a) data rate; (b) delay.

as the network throughput is saturated and cannot accommodate more stream transmissions, while the delay of each scheme is observed to increase due to the longer queuing delay. The reference scheme II has the throughput saturated at a lower traffic arrival rate $\lambda = 3.5$. In order to keep the network in a balanced state, we set $\lambda = 5$ as the default setting. We can also see from the figure that the proposed distributed algorithm outperforms the reference schemes under different traffic arrival rates.

(6) Impact of Mobility. The mobility is implemented using improved random way-point model [41], with maximum moving speeds varied. As in Fig. 3.9, the aggregate data rate and delay of all the schemes are not significantly impacted by the mobility, as all of them consider the channel conditions which are impacted by network topology changes and take advantage of the mobility to give transmission preference to the nodes that are closer to the receivers. The proposed distributed algorithm significantly outperforms the reference schemes under all the speeds studied, with up to 36% higher aggregate data rate and 10% lower delay compared to scheme I. The result shows that our algorithm is robust to mobility in the network, as it is always able to coordinate the transmissions based on traffic demand and schedule high-quality streams at any topology. This indicates that our scheme will perform well in a mobile ad hoc network with dynamic topology change.

3.7 Conclusions

It is important and challenging to coordinate transmissions in a heterogeneous MIMO-based distributed system with mobile devices having different number of antennas, in presence of channel dynamics and network topology changes. In this chapter, we first formulate the problem to maximize the weighted network data rate and propose a centralized scheduling algorithm with a provable approximation ratio as the performance reference. We then propose an effective distributed scheduling algorithm in MIMO-based ad hoc networks by concurrently considering node heterogeneity, impact of channel condition on the degree of freedom and transmission reliability, traffic demand and network load, and taking advantage of multiuser diversity and spatial diversity. Our algorithm adaptively assumes different transmission strategies based on the decoding capacity of receivers to alleviate the bottlenecks caused by nodes with smaller antenna arrays, and avoid transmission failure due to channel degree of freedom constraint. Our scheduling algorithm also exploits both multiplexing and diversity to opportunistically select transmitter nodes and antennas to improve the transmission rate and reliability, while supporting QoS and fairness. Nodes in a neighborhood can cooperate in transmission and form a many-to-many virtual MIMO array. We form a concrete channel model, and apply the channel model in our algorithm design to efficiently optimize network performance. The performance results demonstrate that our proposed scheduling algorithm is very efficient in coordinating transmissions in a MIMO-based ad hoc network, achieving up to 3.6 times the data rate and reducing the transmission delay up to 31% compared with the scheme of selecting only one user pair at a time as often used in conventional MIMO schemes. Compared with the scheme not considering node heterogeneity and channel constraint, our scheduling algorithm can achieve about 36% higher data rate and 14% lower delay.

Chapter 4

Adaptive Exploitation of Cooperative Relay

Although various MAC schemes have been designed to exploit the intrinsic features of MIMO to improve the throughput and reliability, they may not be able to handle consecutive packet loss due to severe path loss, continuous deep fading or temporary topology changes and link breakages. Continuous packet retransmissions would lead to significant throughout reduction. The severe transmission conditions pose a big threat to the growth of wireless applications. Although beamforming can help improve the transmission reliability, it compromises the potential multiplexing gain and hence reduces the transmission rate. In addition, when the channel condition is extremely weak or the distance between the transmitter and receiver is temporarily very long, even beaming-forming may not be able to ensure the transmission reliability for the direct link. Moreover, the design of MAC scheme to coordinate beamforming transmissions in a multi-hop network is very difficult. As an alternative to MIMO technique, recent efforts have been made to enable cooperative relay transmission to cope with channel degradation, with the assumption that network nodes have single antenna [42–44]. One question to raise is: is it beneficial to adopt cooperative relay to facilitate transmission in a MIMO-based ad hoc network?

The introduction of cooperative relay transmission into a network where nodes are equipped with multiple antennas could bring in benefits far beyond that of simply combining the two techniques together. It would not only allow joint exploitation of multiplexing gain of MIMO and cooperative diversity gain of relay transmission, but would also help mitigate many issues presenting in conventional relay transmissions. First, with the support of relay nodes, transmissions on MIMO links with harsh conditions or temporary breakages can possibly be bridged through relay links over source-relay-destination paths. Without being impacted by a poor link for a continuous time period, traffic can be scheduled more efficiently to avoid a significant transmission delay and extra consumption of precious network resources. Second, with a careful relay selection, the channel quality of a relay link would be generally better thus allow for a higher rate, which reduces the cost of using relay transmission. Third, taking advantage of multi-packet transmission/reception capability enabled by MIMO technique, a relay node which has multiple antennas can overhear the transmission from a source while receiving its own packets, which avoids the need for the source to forward the packet explicitly to the relay node as in conventional cooperative transmission. Meanwhile, a relay node can simultaneously forward packet for others while transmitting its own packets.

Although the benefits of using relay transmission in a MIMO ad hoc network are significant, there are also big challenges in efficiently selecting and triggering cooperative relay transmissions, especially in concert with multi-user-based MIMO transmissions in an ad hoc network environment. Without a properly designed strategy, the use of relay would cost much more transmission time and bandwidth instead of supplementing the spatial multiplexing transmission.

In this chapter, our focus is to design algorithms along with a MAC scheme that *opportunistically* use cooperative relay in MIMO-based ad hoc networks to further improve the transmission reliability and throughput when the transmissions between two nodes encounter difficulty. Our proposed strategy is named as Cooperative Relayed Spatial Multiplexing (CRSM). The main contributions of this chapter are as follows.

- We mathematically model the problem and provide a centralized algorithm with proved approximation ratio to serve as the performance reference of the distributed algorithm;
- We practically divide the problem into two phases and provide simple but effective distributed scheduling algorithms that seamlessly incorporate the use of cooperative relay into MIMO transmission, which can guide the practical protocol design;
- We propose a simple relay scheme to formulate relay set and invoke relay transmission without extra signaling overhead;
- We design an efficient MAC protocol to support our distributed algorithm.

The rest of the chapter is organized as follows. We introduce the motivation of our work in Section 4.2. We formulate the problem and propose a centralized algorithm with proved approximation ratio in Section 4.3. We then present our scheduling algorithms to support seamless use of cooperative relay with multi-user-based MIMO transmission in an ad hoc network in Section 4.4, and provide more details about relay operation and MAC protocol design in Section 4.5. The performance of the proposed algorithms is studied through simulations in Section 4.6. Finally, we discuss the related work in Section 4.1 and conclude the chapter in Section 4.7.

4.1 Related Work

Though cooperative diversity has been extensively studied theoretically [42], there are limited work that investigate the solution of scheduling in practical network implementations. In [43], the authors proposed relaying strategies to increase the system reliability and the work in [45] tries to emulate the function and achieve the transmit diversity gain of using space-time codes in a distributed manner through node cooperation without the use of multi-antenna arrays. A multi-layer approach for exploiting virtual MISO links in ad hoc networks is presented in [46] and an optimal relay assignment is discussed in [47]. A relay selection scheme is proposed in [48] for multi-node decode-and-forward cooperative scenarios via the available partial channel state information (CSI) at the source and the relays, and a distributed relay selection scheme is proposed in [49] using finite-state Markov channels. However, the scale of network considered in these studies is relatively small, and they do not provide MAC protocols to implement in a wireless multi hop wireless mesh network. The utilization of cooperative relay in wireless cognitive radio networks is investigated in [44] and a new MAC protocol is proposed. In this work, cooperative relay is only considered for networks with single antenna nodes, while it requires specific strategies to leverage the benefits of cooperative relay in a MIMO-based network. In [50], the authors analytically considers a general multiple-antenna network with multiple relays in terms of the diversity-multiplexing tradeoff. In [51], retransmission diversity through node cooperation is investigated in specific homogeneous omni-directional and smart antenna networks. Cooperative spatial multiplexing is systematically implemented with hybrid ARQ in [52], however, it lacks a detailed algorithm and protocol to specifically enable cooperative transmission which is generally very challenging to achieve in a dynamic network.

Our work distinguishes itself from the aforementioned work in that it adaptively adopts relay forwarding with cooperative MIMO multiplexing to significantly improve the throughput while supporting transmission reliability. The initial results have been presented in [23]. In this chapter, we present more details of our design and perform more extensive simulations to demonstrate the functionality of the proposed algorithms.

4.2 Background and Motivation

In an ad hoc network where nodes are equipped with multiple antennas, there are generally two types of gain achieved by MIMO transmission. *Multiplexing gain* refers to the increase in raw data rate by concurrent transmission of multiple data streams between a node pair, and *diversity gain* is achieved by space time coding or antenna selection which may be exploited to improve the transmission reliability. In this work, we make an effort to leverage the multiplexing gain and diversity gain brought by MIMO transmission along with multi user diversity in a network with mesh topology. Instead of only allowing multiplexed transmission between a pair of nodes as in traditional MIMO scheme, we consider cooperative MIMO multiplexed transmission in which multiple nodes can simultaneously transmit to a receiver that has multiple antennas, i.e. forming a virtual MIMO array [32], and a sender with multiple antennas can also transmit multiple streams to a set of nodes. In this way, many-to-many transmissions are allowed between node pairs to better exploit multiplexing gain. Moreover, among the transmission links between node pairs, those whose channel qualities are higher can be selected for transmission to exploit multiuser diversity gain. When the information of channel coefficients is available for a node pair, a subset of antennas that transmit signals at better quality can be opportunistically selected for transmissions, such a scheme takes advantage of selection diversity and is shown to outperform space-time coding [31]. This framework, named Opportunistic and Cooperative Spatial Multiplexing (OCSM), is illustrated in Fig. 4.1. Empowered with the opportunistic and cooperative transmission capability, node 3 transmits to node 2 and 4 simultaneously with selected antennas, and node 2 is able to receives two data streams from node 1 and one data stream as well as one interference stream from node 3.

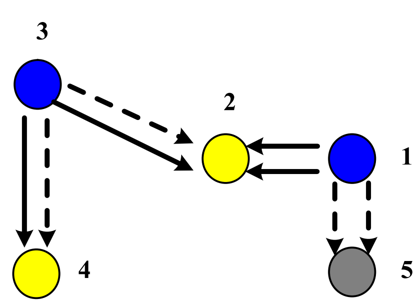


Figure 4.1: An illustration of Opportunistic and Cooperative Spatial Multiplexing (OCSM).

The OCSM framework allows the exploration of multi-user diversity and antenna selection diversity to further improve the capacity and reliability of the network [19]. These diversity techniques, however, are insufficient when the channel condition is extremely weak, the existence of correlated fading between a sender and receiver pair, or the distance between a node pair changes as a result of temporary topology change. If the channel degradation is short-term, it would be inefficient to change the transmission path immediately. Although schemes such as beamforming could be used between the transmission pair which has severe channel condition, it may prevent concurrent transmissions from the same or other nodes and compromise the potential throughput gain of the network that could be achieved with multiplexed transmissions. Also, sometimes even beamforming is hard to handle a weak transmission between two nodes when their distance is large enough or the channel is very weak, although the two nodes are within two-hop transmission distance.

In order to alleviate the problem of data rate reduction and excessive queuing delay caused by severe channel condition and/or link breakage as a result of temporary network topology change, in this chapter, we propose to adaptively invoke *cooperative relay* in conjunction with *cooperative multiplexing MIMO communications* when direct transmission cannot be successfully pursued. There are some unique benefits by taking advantage of both techniques.

- Different from the literature work which exploits cooperative diversity in a single antenna case only to improve the transmission quality, in the proposed work, the relay transmissions coordinate with the transmissions in a neighborhood and take advantage of *cooperative multiplexing* to improve the overall network throughput.

- With multi-packet reception capability brought by multiple antennas, a relay node can obtain the packet to be relayed through overhearing during its own data receiving when the sender attempts for initial direct transmission. As an example, in Fig. 4.2(a), R receives the relay packet as an interference stream while it is receiving data stream from Q. Assume R has 2 antennas, it is therefore able to decode the packet from Q as well as the relay packets from S.

- Instead of simply postponing the transmissions of packets with relay nodes as the direct sender, which is often the case in the conventional cooperative diversity study, a relay node can transmit a relay packet concurrently with its own packets, therefore avoid excessive delay for its own packets. As shown in Fig. 4.2(b), node R can simultaneously transmit to Q when it serves as a relay node to transmit the relay packet to D. A relay node can even have a higher transmission probability driven by our priority based scheduling, as the priority of a relay node increases when its packets experience longer delay due to relay transmissions.

- The direct transmissions and relayed transmissions are performed independently, and a receiver node takes advantage of multiple antennas to decode transmissions from multiple streams without requiring synchronization at the symbol level between neighboring nodes as in conventional cooperative diversity schemes;

With use of coded cooperation, the network performance can be further improved. As our focus is to investigate the benefit and strategy of incorporating relay into multiplexed MIMO transmission, we consider decode and forward cooperative strategy here for simplicity.

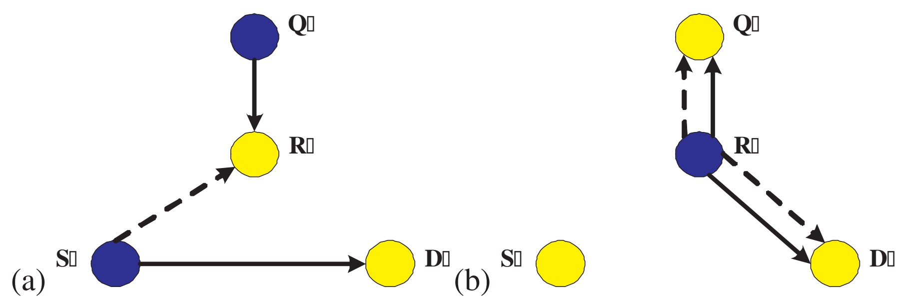


Figure 4.2: An illustration of cooperative relay transmission.

4.3 Problem Formulation and A Centralized Solution

In this section, we first describe the system model and introduce some notations to use in the chapter. We then provide a mathematical formulation of the problem to guide the design of scheduling algorithms. The modeling of transmission opportunities and constraints to enable cooperative MIMO transmissions in multi hop wireless mesh network involves a big challenge, while the need of incorporating relay transmissions makes the problem even harder. Finally, we provide a centralized algorithm with provable approximation ratio to serve as the performance reference of the distributed algorithm to be introduced in the next section.

4.3.1 Problem Formulation

To enable concurrent many-to-many stream transmission, our MAC design is TDMA-based, in which the time domain is divided into transmission durations (TD). A TD covers one round of control signal exchange and data frame transmission and consists of a fixed sequence of phases each with a fixed length. Channel conditions are supposed to be quasi-static during a TD. The data transmission rate within a TD can vary for different links based on the their channel conditions, i.e. more efficient coding can be used to encode the symbols at a higher rate for a channel with higher quality. As the total transmit power of each node is generally fixed, the transmit power of each antenna is different when a node uses a different number of antennas for transmission.

As the complete information about future traffic is unavailable, it is a practical option to schedule the transmission of packets in each TD considering the existing traffic and queueing delay, and the scheduling scheme is consecutively executed during the lifetime of the network. In a TD, suppose there is a set of N_n nodes $N = \{1, 2, \ldots, N_n\}$ in the network, and there are N_p packets waiting for transmission which are contained in the set $P_{pkt} = \{1, 2, \ldots, N_p\}$. A node j has an antenna array of size N_j^{ant}. There is a buffer queue at each node where data packets are stored. For a packet i, a parameter called *priority* $\mathscr{P}(i)$ is used to capture both its service type and queuing delay. For the convenience of calculation, $\mathscr{P}(i)$ is measured in the unit of TD. A possible way to integrate both factors into the priority calculation is to equate the service priority of i to an initial value of $\mathscr{P}(i)$ in terms of TD, and $\mathscr{P}(i)$ increases as the queuing time of p_i increases. A higher value of $\mathscr{P}(i)$ indicates that the packet i has a higher priority.

The transmissions of packets are organized as *streams*. For spatial multiplexed transmission, a stream s is defined to be an independent data flow transmitted from an antenna of a transmitter node to a receiver node and identified by a triplet $s = (I_t, I_r, I_{ant})$, where $I_t/I_r/I_{ant}$ is the index of the transmitter/receiver/antenna that involves in the transmission of the stream. Suppose the signal to noise and interference ratio (SINR) at the receiver node is $\rho_{I_r}(s)$ for stream s, the data rate of s can be calculated as $\mathscr{R}(s) = \log(1 + \rho_{I_r}(s))$. In a practical system, a receiver can include its estimated $\rho_{I_r}(s)$ in its feedback message, and a transmitter can then decide the actual data rate based on the SINR information, i.e. by looking up a pre-set table. The transmissions in the network are half-duplex, so a node cannot be a transmitter and receiver at the same time. In a TD, a subset of nodes, denoted as T, are selected as transmitter nodes.

The notations used in the problem formulation are summarized in Table 1. Denote the set of neighboring nodes of node j as $\mathscr{V}_j$. Suppose the transmission of a packet i is through stream $s(i)$, and the reception is successful when the receiving SINR $\rho_{I_r}(s(i))$ is above a certain threshold Γ. After a direct transmission of a packet i from s_i to d_i, nodes that successfully

Table 1: List of notations used in problem formulation.

Notation	Definition
$i = 1, \ldots, N_p$	Index of packets
$j = 1, \ldots, N_n$	Index of nodes
$h_j \in \{0, 1\}$	$h_j = 1$ if and only if node j is selected as a receiver
$t_j \in \{0, 1\}$	$t_j = 1$ if and only if node j is selected as a transmitter
$y_{ij} \in \{0, 1\}$	$y_{ij} = 1$ if and only if packet i is assigned to be transmitted from node j
$a_{ijk} \in \{0, 1\}$	$a_{ijk} = 1$ if and only if packet i is assigned to be transmitted from the k-th antenna of node j
$s = (I_t, I_r, I_{ant})$	Stream from the I_{ant}-th antenna of transmitter I_t to receiver I_r
R_i	The set of candidate relay nodes for packet i
$\mathscr{P}(i)$	Priority of packet i
$\mathscr{R}(s)$	Data rate of stream s
$\mathscr{I}(d_i)$	Interference at receiver node d_i when receiving packet i

overhear the packet while are in the transmission range of s_i and the receiving range of d_i, i.e. those in the set $R_i = \{r | \forall r \in N \setminus T, s.t.\ s_i \in \mathscr{V}_r, d_i \in \mathscr{V}_r, \rho_{I_r}(s(i)) \geq \Gamma\}$, store the packet in their own buffers. These nodes become candidate relay nodes for packet i. The packet i becomes available to nodes in $R_i \bigcup \{s_i\}$, which store the packet with the consistent priority. R_i is updated to include more qualified relay nodes whenever there is any direct transmission of i. When $y_{ij} = 1$, it implicitly indicates that $j \in R_i \bigcup \{s_i\}$. Note that if $a_{ijk} = 1$, the transmission rate of packet i depends on the channel condition of the stream $s(i) = (j, d_i, k)$ and the interference at node d_i when receiving the stream, denoted as $\mathscr{I}(d_i)$. Therefore, the rate of stream $s(i)$ is denoted as $\mathscr{R}(s(i), \mathscr{I}(d_i))$.

We now can formulate the constraints for the problem of cooperative relayed spatial multiplexing in a MIMO ad hoc network to capture the features of MIMO transmissions and conditions of relay transmissions. Firstly, it is necessary to ensure that a packet i is assigned to at most one transmitter node among all the candidate ones (including the source node s_i and candidate relay nodes in R_i) to avoid redundant transmission,

$$\sum_{j \in R_i \bigcup \{s_i\}} y_{ij} \leq 1, i \in P_{pkt}. \tag{4.1}$$

As the *transmitting constraint*, an antenna k at a transmitter j can only accommodate the transmission of at most one stream in a TD,

$$\sum_{i \in P_{pkt}} a_{ijk} \leq 1 + (1 - t_j)M, j \in N, k = 1, \ldots, N_j^{ant}; \tag{4.2}$$

where M is a sufficiently large number introduced to relax the constraint when node j is not selected as the transmitter, i.e., $t_j = 0$. Similarly, the *receiving constraint* is used to model the impact of interference at the receiver end of a MIMO link, where the total number of receiving streams (data streams plus interference streams) at a receiver node j is restricted to be no more than its number of antennas in order to decode the receiving data packet,

$$\sum_{i \in P_{pkt}} \sum_{m \in \mathscr{V}_j} \sum_{k=1}^{N_m^{ant}} a_{imk} \leq N_j^{ant} + (1 - h_j)M, j \in N. \tag{4.3}$$

To ensure that the transmission is half-duplex, t_j and h_j for each j have to satisfy

$$t_j + h_j \leq 1, j \in N. \tag{4.4}$$

It is also important to constrain the relation between the parameters,

$$\begin{aligned} & a_{ijk} \leq y_{ij} \leq t_j, a_{ijk} \leq y_{ij} \leq h_{d_i}, \\ & i \in P_{pkt}, j \in N, k = 1, \ldots, N_j^{ant}. \end{aligned} \tag{4.5}$$

Finally, following the scheduling framework in [53], our scheduling aims to maximize the sum of priority-weighted capacity so that both data rate and priority can be jointly optimized. The objective function is:

$$\max \sum_{i \in P_{pkt}} \sum_{j \in R_i \bigcup \{s_i\}} \sum_{k=1}^{N_j^{ant}} a_{ijk} \mathscr{R}(s(i), \mathscr{I}(d_i)) \mathscr{P}(i). \tag{4.6}$$

With this formulation, the nodes without packets will have the priority set to 0 and not be scheduled to transmit, while the packets associated with worse quality links will still get chance to transmit as their priority increases.

So far, we formulate the problem of cooperative transmission with relays in a MIMO ad hoc network as an integer linear programming (ILP) problem with objective function in

(4.6) subject to constraints (4.1)(4.2)(4.3)(4.4)(4.5). As an ILP problem is NP-hard in general and needs exponential time complexity to find a solution, an efficient heuristic algorithm is required for the practical implementation.

4.3.2 A Centralized Algorithm

In Algorithm 8, we propose a centralized scheme to schedule packet transmissions in a single TD. As the interference streams which can transmit simultaneously with stream i are unknown before the scheduling is finalized, it makes the accurate determination of $\mathscr{R}(s(i), \mathscr{I}(d_i))$ difficult. On the other hand, as the transmission rate is only used as a guidance to select the streams that potentially support higher throughput for transmissions, it is not necessary to know the accurate transmission rate at scheduling time. Therefore, we consider the maximum possible receiving interference and use it for the conservative estimation of rate for each candidate stream. Specifically, as the number of interference and data stream could not exceed $N_{d_i}^{ant}$ for correct decoding, $N_{d_i}^{ant} - 1$ strongest candidate streams around d_i are considered to calculate the interference strength. The estimated value of $\mathscr{R}(s(i), \mathscr{I}(d_i))$ is then calculated based on the channel condition of the stream and the interference strength, and is then used in the centralized algorithm. Note that our algorithm does not prevent using other model for stream rate determination. When channel conditions from all the potential transmitters are estimated in advance, more sophisticated techniques could be used to cancel the majority of interference, and thus further improve the transmission rate.

The algorithm is to be executed by a central controller of the network which has the complete information of packets and channels. To facilitate scheduling, a parameter $w(ijk)$ is introduced to represent the priority weighted data rate achieved with the transmission of packet i from transmitter j using antenna k as in (4.6), and the set W consists of the weighted rates of all candidate streams, as in lines 2-7. The algorithm greedily schedules a packet i^* to transmit from antenna k^* of transmitter node j^*, which has the highest weighted rate among all the candidate ones and guarantees the constraints (4.2)-(4.3). P is the set of scheduled packets and T contains all selected transmitters. In line 12, all the candidate streams that have transmission conflict with the scheduled stream $s = (j^*, d_{i^*}, k^*)$ are removed from the set W, including the ones that have the node j^* as the receiver, have d_{i^*} as the transmitter, or have node j^* as the transmitter but are associated with the antenna k^*. A packet may be queued at multiple candidate transmitting nodes, i.e. source and candidate relay nodes. To avoid repetitive transmission of a packet and satisfy constraint (4.1), all other candidate streams for the selected packet i^* are also removed from W after i^* is successfully scheduled in the current TD. The algorithm then checks if packets are correctly received at destinations in lines 18-19, and successfully received packets are removed from the packet set P_{pkt}. For any incorrectly received packet i, its candidate relay list R_i is updated to add in nodes that are within the range of both the source and destination of i and have correctly overheard the direct transmission, as i n lines 21-23, so that nodes in R_i would assist in the transmission of i in the following TDs.

The numbers of cycles for the Initialization phase from line 0 to 7 and the Relay Set Update phase from line 17 to 27 are both in $O(N_p N_n)$. In the Scheduling phase from line 8 to 16, at least one candidate stream is removed from set W in each iteration. As the size of antenna array is a constant for each node, the number of candidate streams between each node pair does not exceed a constant $A = \{\max_i\{N_i^{ant}\}\}^2$, and there are thus no more than $A(N_n)^2$ candidate streams in W. Suppose the elements in the set W are sorted in descending order by their values, and it requires $O(N_n)$ to suppress streams in line 12. Therefore, the complexity of the algorithm is $O((N_n)^3)$.

Similar as in Chapter 3, it can be proved that the centralized scheduling algorithm can achieve an approximation ratio of $1/\left((2+\mathcal{D})\max_i\{N_i^{ant}\}+2\right)$, where $\mathcal{D}$ is the maximum node degree in the network. Note that the approximation ratio represents the worst case that can be achieved for the centralized algorithm and is rather conservative. In general, there are not many idle nodes in the network and nodes are either transmitters or receivers, especially when many-to-many communication is enabled. In that case, it is unnecessary to consider the suppression of a potential idle node when a stream is selected, and the approximation ratio can be improved to $1/\left(2\max_i\{N_i^{ant}\}+2\right)$ when all the nodes are active transmitter or receivers.

4.4 Packet Scheduling with Relay Transmission

In order to achieve the optimum system performance, it is essential for a scheduling scheme to determine the set of nodes that serve as the transmitters and the packets to be transmitted in a transmission duration, and assign them to the appropriate antennas for transmissions. The coordination among nodes and the selection of antennas to complete these procedures in a distributed manner are highly nontrivial. The need of invoking relay transmissions upon severe channel conditions adds in significantly more challenges. In this section, we design a distributed scheduling algorithm to fully exploit the multiplexing gain enabled by cooperative MIMO transmission and diversity gain enabled by cooperative relay transmission for overall higher system performance. Specifically, our scheduling has the following features

Algorithm 8 Centralized Scheduling

Initialization:
$W \Leftarrow \oslash, T \Leftarrow \oslash, P \Leftarrow \oslash, y_i \Leftarrow 0, x_j \Leftarrow 0, a_{ijk} \Leftarrow 0, \forall i, j, k$, update P_{pkt} to include new packets
for $\forall i \in P_{pkt}$ **do**
  for $\forall j \in R_i \bigcup \{s_i\}$ **do**
    $w(ijk) \Leftarrow \mathscr{R}(s(i), \mathscr{I}(d_i))\mathscr{P}(i), \forall k \in \{1, \ldots, N_j^{ant}\}$
    $W \Leftarrow W \bigcup \{w(ijk)\}$
  end for
end for
Scheduling:
while $W \neq \oslash$ **do**
  $(i^*, j^*, k^*) = \arg\max_{\{i,j,k\}} W$, the corresponding destination node is d_{i^*}
  if Selecting stream (j^*, d_{i^*}, k^*) satisfies (4.2) for j^* and (4.3) for all nodes in $\mathscr{V}_{j^*}$ **then**
    Schedule the stream (j^*, d_{i^*}, k^*), $y_{i^*} \Leftarrow 1$, $x_{j^*} \Leftarrow 1$, $a_{i^*j^*k^*} \Leftarrow 1$, $P \Leftarrow P \bigcup \{i^*\}$, $T \Leftarrow T \bigcup \{j^*\}$
    $W \Leftarrow W \setminus$
    $\{w(ijk) | \forall i \; s.t. d_i = j^*, \forall j \in R_i \bigcup \{s_i\}, \forall k\} \bigcup$
    $\{w(ijk) | \forall i, j = d_{i^*}, \forall k\} \bigcup$
    $\{w(ijk) | j = j^*, k = k^*, \forall i\} \bigcup$
    $\{w(ijk) | i = i^*, \forall j \in R_{i^*} \bigcup \{s_{i^*}\}, \forall k\}$
  else
    $W \Leftarrow W \setminus w(i^*j^*k^*)$
  end if
end while
Relay Set Update:
for $\forall i \in P$ **do**
  if i is correctly decoded at d_i **then**
    $P_{pkt} \Leftarrow P_{pkt} \setminus \{i\}$
  else
    for $\forall m \in \{r | r \in \mathscr{V}_{s_i} \bigcap \mathscr{V}_{d_i}, \sum_k a_{is_ik} \geq 1, r \in N \setminus T\}$ **do**
      if i is correctly decoded at m **then**
        $R_i \Leftarrow R_i \bigcup \{m\}$
      end if
    end for
  end if
end for

for relay handling.

- *Simple formulation of a candidate relay set for a packet.* The nodes in a neighborhood collaboratively determine if a relay transmission is needed without sophisticated signaling.

- *Simple priority-based relay selection without extra signaling.* A candidate relay node schedules the transmissions of relay packets with its own packets based on their relevant priorities. As the relevant priority of relay packets to existing packets in different candidate relay nodes are different, our scheduling naturally selects the relay transmission among a group of candidate relay nodes.

- *Support of load balancing and reduction of delay impact on relay nodes.* By incorporating delay into priority in our scheduling, a packet that experiences a longer delay as a result of repeated transmission failures of its source node has its priority increased, which may be higher than some packets at a candidate relay node (especially when the relay node has a lower load). It is therefore more likely for a relay node with lower traffic to forward the relay packets, which would balance the load of nodes in a neighborhood and the relay transmission would not significantly impact the transmission of an overloaded candidate relay node. In addition, with extra packets buffered to forward for other nodes, a candidate relay node could have a higher priority of being scheduled for transmission.

- *Receiver-facilitated reduction of redundant relay transmission.* As a node self-determines if it can be a relay in a time slot based on the priority of the cached packet to avoid signaling overhead, there is a likelihood that multiple nodes may attempt to perform relay transmission. Our MAC scheme let the receiver to select the relay as discussed at the end of Section 4.4.1.

From the problem formulation in Section 4.3, it is clear that the scheduling problem has to determine the values of the four parameter set: $\{t_j\}$, $\{h_j\}$, $\{y_{ij}\}$, and $\{a_{ijk}\}$ to assign a packet to an appropriate transmitter antenna in order to maximize the total weighted rate of the network. In a practical half-duplex network, it is reasonable to divide the problem into two parts: transmitter selection and stream allocation. In the first phase, a set of nodes are selected as transmitter nodes, and for each selected node, it needs to determine the number of packets to transmit in the current transmission duration. Thus the values of $\{t_j\}$, $\{h_j\}$and $\{y_{ij}\}$ are determined. The decision in our scheduling is made based on the transmission priority of the packets in queue, and the antenna constraints of the transmitter nodes and receiver nodes. In the second phase, each selected transmitter node needs to assign its packets to appropriate antennas for transmission based on the number of streams it is allowed to transmit, the priority of the packets, and the channel conditions. Thus, the value of $\{a_{ijk}\}$ is determined. In the next two subsections, we introduce the problem and algorithm for each scheduling phase.

4.4.1 Determination of Transmitter Nodes and the Number of Transmission Streams

Instead of randomly selecting the transmitter nodes in a TD, in this phase, we propose a *priority-based self-selection* strategy with which an active node self-determines if it can serve as the transmitter and the number of streams to transmit based on the priority of its packets, its transmitter constraint and the decoding constraints of its neighbors. A candidate relay node incorporates the relay packet with its own transmission and participates in the transmitter selection process.

As the selection is performed at the beginning of each TD before any transmissions, the rate information for candidate streams is unavailable. The transmitter node assignment and the number of streams are thus determined with the goal of optimizing the overall priority performance, and the goal of rate optimization is addressed later in the stream allocation phase. The problem in equations (4.1)-(4.6) is then reduced to the subproblem formulated as follows:

$$\max \sum_{i \in P_{pkt}} \sum_{j \in R_i \cup \{s_i\}} y_{ij} \mathscr{P}(i); \tag{4.7}$$

$$\sum_{j \in R_i \bigcup \{s_i\}} y_{ij} \leq 1, i \in P_{pkt}; \tag{4.8}$$

$$\sum_{i \in P_{pkt}} y_{ij} \leq N_j^{ant} + (1 - t_j)M, j \in N; \tag{4.9}$$

$$\sum_{m \in \mathscr{V}_j} \sum_{i \in P_{pkt}} y_{im} \leq N_j^{ant} + (1 - h_j)M, j \in N; \tag{4.10}$$

$$t_j + h_j \leq 1, j \in N; \tag{4.11}$$

$$y_{ij} \leq t_j, y_{ij} \leq h_{d_i}, t_j, h_j, y_{ij} \in \{0, 1\}, \quad i \in P_{pkt}, j \in N; \tag{4.12}$$

where M is a sufficiently large number as defined in section 4.3. Corresponding to constraints (4.1)-(4.3), (4.8) limits a packet to only one transmitter to avoid simultaneous transmissions of a packet from multiple relay nodes for improved transmission throughput, (4.9) and (4.10) represent degree constraints at a transmitter and a receiver respectively. Note that the set P_{pkt} is updated at the beginning of each TD so that the packets that arrive during the previous TD can be included.

Distributed Transmitter Node Selection

A distributed solution for the problem aims at maximizing the objective in (4.7) while probabilistically satisfying constraints (4.8)-(4.12). Let Q_j denote the packet queue at node j, where original packets and relay packets are sorted in a descending order of their priorities. Let N_j^0 be the proposed number of transmission streams, obviously $N_j^0 = \min\{N_j^{ant}, |Q_j|\}$. Denote the l-th packet of node j as $p(j,l)$. Parameter U_j is defined to be the priority of the head-of-the-line packets in node j's queue, i.e., $U_j = \sum_{l=1}^{N_j^0} \mathscr{P}_{p(j,l)}$, which is used as the priority of j for scheduling.

In order to avoid unnecessary channel measurement and message processing at a receiver, our algorithm first selects a candidate set of transmitters. To guide the transmitter selection, we introduce a probability P_j^{TX}, below which an active node j can be selected as a transmitter node. Suppose m, a neighboring node of j, has N_m^{active} neighboring nodes and can decode N_m^{dec} concurrent streams, which can be obtained from periodic Hello messages sent in the two-hop neighborhood of j at the network layer. If the average number of streams from a single transmitter node around a receiver m is known and denoted as $\bar{N}_{\mathscr{V}_m}^{allo}$, in order to not exceed its decoding capacity, m generally only allows $\widetilde{N}_m = N_m^{dec}/\bar{N}_{\mathscr{V}_m}^{allo}$ nodes among its N_m^{active} neighbors to transmit in a TD. That is, each of the nodes around m is allowed to have a probability of $N_m^{dec}/(\bar{N}_{\mathscr{V}_m}^{allo} N_m^{active})$ to serve as the transmitter. As $\bar{N}_{\mathscr{V}_m}^{allo}$ is hard to know before scheduling is performed, a node can at most have a probability of N_m^{dec}/N_m^{active} to serve as the transmitter. The parameter P_j^{TX} of j can then be calculated as follows to consider the decoding capability of all its neighboring receiver nodes:

$$P_j^{TX} = \min_{m \in \mathscr{V}_j} \left(N_m^{dec}/N_m^{active} \right). \tag{4.13}$$

Instead of only considering the decoding capability of the selected receiver nodes which is not available at the selection time, our selection considers the decoding capability of all the neighboring nodes and is more conservative.

With this calculation, when there is only a small number of nodes around each receiver, there is a possibility that all the nodes within a neighborhood are selected as the transmitters. For example, if the network has only two nodes and each node can decode up to four streams, both nodes may be selected as transmitters and it is not possible to complete the transmission. To avoid this problem, when $P_j^{TX} \geq 1$, the value of P_j^{TX} is replaced with $P_j^{TX} = \max_{m \in \mathscr{V}_j} (N_m^{active}/(N_m^{active}+1))$, so that at least one node will be kept as the receiver.

The priority of a node can be attached with periodic Hello messages sent at the network layer, and updated with the data packets sent. The priority of the active nodes not having packets sent in a TD can be predicted as the time moves forward. A node j can then record the maximum priority U_j^{max} and the minimum priority U_j^{min} of all the N_j^{active} active nodes in its neighborhood and itself, and also calculate the average priority $\bar{U}_j$ as $\bar{U}_j = (\sum_{m=1}^{N_j^{active}} U_m + U_j)/(N_j^{active}+1)$.

To avoid extra signaling and control overhead, an active node j *self-decides* if it should be selected as a transmitter node by calculating an index number r_j^{TX} as follows:

$$r_j^{TX} = \begin{cases} (\bar{U}_j - U_j)/(U_j^{max} - U_j^{min}) + \gamma_j & \text{if } U_j^{max} \neq U_j^{min} \\ \gamma_j & \text{if } U_j^{max} = U_j^{min} \end{cases} \tag{4.14}$$

where the parameter γ_j is uniformly distributed in the range $[0,1]$ and randomly generated by a node j in each transmission duration (TD) to provide some fairness among nodes. The factor $(\bar{U}_j - U_j)/(U_j^{max} - U_j^{min})$ is used to give the higher priority node a larger probability for transmission. In a TD, if $r_j^{TX} < P_j^{TX}$, node j is selected as a transmitter node; otherwise, it has no right of transmission. Our transmitter selection algorithm gives preference to a node with a higher service priority and/or a larger load and hence longer delay, and thus supports load balancing. Moreover, as the priority parameter dynamically reflects the queuing status of nodes so a node does not always have higher priority than its neighbors, it helps ensure fairness over the network.

Note that in this phase relay packets and original packets are treated equally, and the value of $\{x_j\}$ is determined.

Distributed Determination of the Number of Streams

Through the procedure described next in Section 4.5, a receiver node estimates the total number of candidate streams it may receive N_j^{rec} and broadcasts it together with the number of streams it is able to decode N_j^{dec}. These two parameters are used at a transmitter node to determine the actual number of transmission streams it is allowed to transmit.

Denote the set of receiver nodes within the transmission range of a transmitter node j as X_j^{rc}. In order to ensure all the receiver nodes in its neighborhood to have high probability of meeting their degree constraints, j constrains its number of sending streams to a number N_j^{allo} as follows:

$$N_j^{allo} = N_j^0 \min_{m \in X_j^{rc}} \left(\frac{N_m^{dec}}{N_m^{rec}} \right). \tag{4.15}$$

Note that the value N_j^{allo} may be a fractional number. To achieve a higher accuracy in calculating N_j^{allo} than using simple rounding, let $N_{j,0}^{allo} = N_j^{allo} - \lfloor N_j^{allo} \rfloor$. If $N_{j,0}^{allo} > 0$, generate a random variable β_j uniformly distributed in $[0, 1]$. If $\beta_j \leq N_{j,0}^{allo}$, $N_j^{allo} = \lfloor N_j^{allo} \rfloor + 1$; otherwise, $N_j^{allo} = \lfloor N_j^{allo} \rfloor$. So far, the number of streams to be transmitted is determined.

4.4.2 Allocation to Antennas

In this phase, N_j^{allo} data packets of node j are allocated to N_j^{allo} out of N_j^{ant} antennas for transmission. For a node that does not serve as a relay, it simply considers the first N_j^{allo} data packets in the queue. For a potential relay node, it would waste network resource if it forwards the same packet concurrently with other relay nodes. Our scheduling scheme naturally selects the forwarding nodes based on the relevant priority of the to-be relayed packet and the priorities of the other packets of a relay node. After this self selection process, there are still the possibility that some relay nodes choose the same TD to forward i. To further reduce the chance of unnecessary relay forwarding, when the destination receiver receives multiple relay transmission requests, it selects the relay node with the best channel condition to forward the packet. The rest of the requesting relay nodes can use the slot to send other packets. More details of the relay selection operation are presented in section 4.5.

The packets may have different destination nodes thus varied link loss, and the spatial channels from different elements of the antenna array undergo different fading. As discussed in [19], the data rate can be improved by opportunistically allocating the packets to transmitted antennas. Moreover, with channel information available at transmitters' side, selection diversity is shown to outperform space-time coding in improving the link reliability [31]. With the goal of maximizing transmission rate, the stream allocation problem is essentially a bipartite maximum matching problem.

Construct a graph $G = (V_1 \cup V_2, E)$ for a transmitter node j. V_1 denotes the set of packets to be allocated to antennas and V_2 denotes the set of transmitting antennas of j. Thus $|V_1| = N_j^{allo}$

and $|V_2| = N_j^{ant}$. Form an edge (v, u) between v and u where $v \in V_1$ and $u \in V_2$, and the weight of the edge is $w_{vu} = \mathscr{R}(v, u)$. Here $\mathscr{R}(v, u)$ is the rate of the stream to transmit a packet represented by node v to its destination node through the antenna represented by node u, which is estimated through signal exchange as discussed in Section 4.5. If $|V_1| \neq |V_2|$, add dummy nodes to make $|V_1| = |V_2|$ and the edges connected to a dummy node has weight 0.

By solving the maximum weight matching problem formulated above (i.e. using successive shortest path algorithm [54]) and then deleting the dummy nodes and edges connected to them, the optimum solution of the allocation is derived. Let $|V| = |V_1| + |V_2|$, the complexity of the algorithm is bounded by $O(|V| \log |V|)$.

4.5 Protocol Design

In the previous section, the scheduling is performed in each transmission duration to determine the transmission schedule of the packets, including original packets and relay packets, in the queue of each node. However, the details about cooperative relay transmission, i.e. how to maintain the queue to store relay packets, how to trigger and enable a relay node to transmit relay packets have not been addressed yet. In this section, we propose the protocol to facilitate cooperative relay transmission in a MIMO-based ad hoc network and implement the distributed scheduling algorithm described in Section 4.4. We first give an overview of the relay operations in Section 4.5.1, and then describe the details of the protocol in Section 4.5.2. An example is presented in Section 4.5.3.

4.5.1 Relay Operations

There are several challenges arising in integrating the cooperative relay transmission with the cooperative MIMO multiplexing transmission scheme. We propose a few strategies to address the issues, some of which are also mentioned in previous sections, and we summarize them here for the protocol design.

Finding Candidate Relay Nodes

In a conventional relay strategy, a source often broadcasts a relay request explicitly, and waits for replies from the potential relay nodes. This process not only introduces extra signaling overhead, but also adds in delay for relay transmission. Instead, the process of finding candidate relays in our scheme is automatically performed at qualified nodes without involving the source and destination of a packet. Specifically, a node r_i identifies its potential of being a candidate relay node of a packet i which is targeted to d_i when successfully receiving the packet from its sender s_i, either because r_i is idle or because r_i could decode i when receiving its own packet with its multi-packet reception capability. If the destination of the data packet i is also in r_i's neighbor list, r_i temporarily stores i in its buffer with the current priority of i. If i is successfully received by d_i, r_i removes i from its buffer; otherwise, the priority of i is updated as its buffering time in r_i increases. In a dense network, to avoid excessive buffering, a node may only buffer a packet with certain probability, or a sender could tag the packets that may need relay.

Triggering of Relay Transmission

Instead of explicitly invoking relay transmission, in our scheme, triggering of relay transmission and selection of relay node is incorporated with normal packet scheduling. If a failed direct transmission is detected, i.e. a candidate relay r_i receives packet i from s_i but does not receive the successful reception acknowledgement for packet i (either through ACK-I or ACK-II as described in section 4.5.1) in the same TD, r_i immediately moves the relay packet i from the buffer to its MAC queue, and treats it as a normal packet waiting for transmission. The node then serves as a relay node in the following TDs. There may be multiple candidate relay nodes for a packet, and the packet to relay is generally placed in different positions of the packet queues in different candidate relay nodes depending on the relative priority of the packets. In a TD, a candidate relay node that has the relay packet scheduled to transmit is implicitly selected as the relay node of the packet. With multiple candidate relay nodes, as long as a subset of the nodes receive a packet from the source, the packet can be relayed to the receiver. Multiple relay nodes and maybe also the source node of i may intend to transmit it in the same TD, if i happens to be a head-of-the-line packet in all of their queues. In order to reduce the chance of unnecessary concurrent transmission, the targeted receiver node counts the number of successful transmission requests for the same packet. The node with the best channel condition is selected to serve as the packet sender and the selection is broadcast by the receiver. In summary, our scheduling strategy triggers relay transmission through the implicit self-selection by candidate relay nodes and explicit selection by the destination receiver to reduce the signaling overhead as well as to avoid redundant transmission.

Constraining the Delay of Relay Transmission

To avoid excessive traffic increase and occupation of network resource, a retransmission threshold F is introduced that a packet is dropped if its reception fails after F TDs has elapsed since its first direct transmission. To ensure that the source node and all candidate relay nodes have a consensus on the packet transmission status, a packet transmitted from its source node is attached with a time-stamp indicating the current elapsed time since its initial transmission, so that candidate relays can record this stamp and update it as the queuing time increases. If the transmission fails continuously over a period of time, e.g. longer than $3F$ TDs since the first direct transmission, a source node may even give up its transmission towards a particular receiver as the continuous failure indicates a long-term brokage of the link, e.g. topology change due to mobility. It may then look for an alternative path to the destination, e.g. through multi-hop relays.

Broadcast of Packet Reception Status

The information about successful or failed reception of a packet is usually broadcast through ACKs. However, as all receivers in a TD send ACK simultaneously as described in Sections 4.5.2, only nodes that are not receivers in the current TD can receive the ACKs. As a candidate relay node may either serve as a transmitter or a receiver in a TD, it is necessary to inform all of them about the updated reception status, so that successfully received packets can be removed while unsuccessfully received packets can have their priority increased. In addition, a source may not be able to get the ACK if the channel condition from the destination to it is very poor, and a potential relay node also needs the reception status to determine whether the packet should be moved from the buffer to the MAC queue. To address those issues, an extra ACK phase is introduced into the protocol, during which the information included in the first ACK is rebroadcast by nodes that receive it in the current TD. Through the two phases of ACK from multiple nodes, extra diversity is provided to guarantee the correct update of the packet reception status for all the nodes in concern. To differentiate between the two ACK messages, they are named ACK-I and ACK-II respectively. In the proposed MAC scheme, the data transmission can be in burst, so the overhead of ACK signaling is relatively small.

Rate Determination

As described in the protocol, both transmitter nodes and receiver nodes are able to estimate the full channel condition matrix through training sequences. Also, a receiver node can estimate the interference and noise around it, and announce this information to the corresponding senders. With the channel matrix and the interference and noise at the receiver, a transmitter can determine the rate to use for transmission. If a packet is scheduled for its first direct transmission and the link to its destination is estimated to be severe, the source node uses a default moderate transmission rate for its transmission, so as to increase the chance of having some relay node successfully receive the packet as well as avoid wasting the transmission opportunity in the current TD. Note that the transmission of a specific packet is canceled for the current TD if a sender node could not receive response, i.e. CTS, from the corresponding receiver after sending an initial handshaking signal, as it can be expected that the requested receiver is currently a transmitter or the link condition is temporarily poor. However, if the response from a receiver is consecutively missing, e.g. for more than F of transmission requests, it is indicated that the link between the source and destination undergoes relatively long term degradation. In such a case, the source node may still initiate transmission in the following $2F$ TDs and send out the packet using the default moderate transmission rate, in the hope that it can be received and forwarded by some relay nodes in the neighborhood.

4.5.2 Protocol Details

Based on the above operations, we propose a TDMA-based MAC protocol to support the cooperative relay transmission in a MIMO-based ad hoc network. A time frame is divided into five phases with different transmission duration, namely RTS, CTS, DATA, ACK-I, and ACK-II. Note that slot synchronization is currently achievable in the IEEE 802.11 family of protocols. By taking advantage of various diversity techniques, our scheme effectively increases the SINR of received signals, which helps improve the accuracy of synchronization as well as mitigate the impact of asynchronicity in a distributed scenario. A group of random access codes, called ID code, which are almost orthogonal for different nodes and assigned similarly to that in [36] , are used to mask and differentiate simultaneously transmitted control signals from selected nodes, and used for transmission coordination and channel estimation.

RTS In RTS transmission phase, nodes that determine themselves to be transmitter nodes (using algorithm in Section 4.4.1) broadcast RTSs. For a transmitter node j, the RTS message contains the number of streams it plans to transmit N_j^0, its node ID and the IDs of the destination nodes. The preamble of a packet is used as the training sequence (without incurring extra overhead for adding in pilot signal) for channel estimation purpose. The preamble of an RTS message is transmitted rotationally from each antenna so the full channel condition matrix can be estimated at receiver nodes. RTS messages sent from different transmitters are masked by different ID code to allow a receiver to differentiate the messages. As the number of antennas is generally small and only the preamble of the RTS message is transmitted through all antennas, the total transmission delay for channel estimation purpose is small. The full knowledge of the channel as a result of the estimation, however, could enable simultaneous transmission of multiple spatial streams and bring in multi-fold capacity gain [19] and thus delay reduction.

CTS The RTSs are received at receiver nodes, where channel matrices are estimated by extracting the preambles. A receiver node m also estimates the number of streams it may receive $N_m^{rec} = \sum_{j \in \mathscr{V}_m, x_j=1} N_j^0$. Constrained by its degree of freedom, m can decode at most N_m^{dec} streams simultaneously. If m receives multiple RTSs (from the source and/or candidate relay nodes) on the transmission of i in current TD and is the target receiver of i, it then selects the node r_i which has the best channel condition between r_i and m to forward the packet. Based on the decoding capability and the signal strength received, m estimates the interference plus noise level (SINR) for candidate transmission nodes. In general, SINR can be quantitized into different levels and only the index of level is needed in feedback instead of its absolute value, which can effectively reduce the amount of overhead. Finally, m broadcast a CTS message including SINR, N_m^{rec}, N_m^{dec} and r_i. Note that CTS message is also masked by ID code and the preamble is transmitted rotationally from each antenna of m for transmitter nodes to estimate the full channel condition matrix.

DATA In the DATA phase, a sender first determines the number of streams it is allowed to transmit using the algorithm in Section 4.4.1, based on the information received from CTSs sent by neighboring receivers. It should also select the packets to be transmitted based on the receivers' confirmation for the initial handshaking messages. Specifically, a node should check if it has been selected as the sole forwarder by the receiver if a request for relay transmission is sent earlier. If a node is the source for a packet and the CTS has been missing for more than F times, it would also send out this packet for relay purpose. The transmitter then estimates the transmission rate from each antenna based on the estimated channel condition and interference at a destined receiver, and transmits the packets from the antennas selected using the maximum weight matching algorithm in Section 4.4.2. A receiver node then differentiates all streams it receives and extracts the data packets targeted for it. Instead of discarding packets transmitted through interference streams, a receiver buffers an overheard packet if it is within the transmission range of the packet destination for potential relay transmission.

ACK-I Receiver nodes broadcast ACKs about those successfully received packets, which include the original sources of the packets. These messages are received by nodes that are not receivers in the current TD.

ACK-II If a relayed packet is received successfully, the source node as well as all the potential relay nodes should remove it from their buffers and queues in order to avoid redundant transmissions. Some of these nodes may not be able to receive the ACKs as they are also in transmitting states during the transmission of ACKs. After the transmission of ACK-I, ACK-II is rebroadcast by non-receiver nodes in the current TD. With the transmission of ACKs in consecutive phases, it not only ensures all candidate relay nodes to learn the packet transmission status, but also guarantees that the original packet sender is informed about the successful transmission of the relay packet. In the case that the channel condition between the source and the destination is poor and ACK-I message from the destination cannot be received by the source node, the rebroadcast of ACK-II messages from intermediate nodes plays an important role to avoid the continuous redundant retransmissions and thus more waste of wireless resources. In this way, a potential relay node that successfully overhears a packet but does not have a functional link towards the destination will also be informed by the sender through ACK-II, so that it will not vainly consider relaying the packet.

4.5.3 An Example

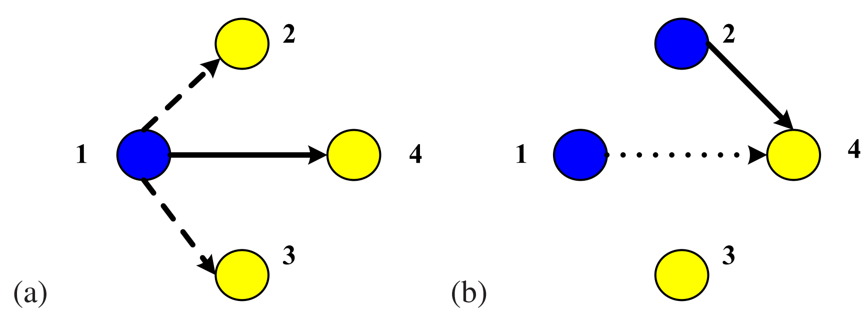


Figure 4.3: An example of cooperative relay transmission.

In this section, we give a brief example to explain the process of cooperative relay transmission. In the simple topology shown in Fig. 4.3, node 1 has a packet to transmit to node 4, node 2 and 3 are in the neighborhood of both node 1 and node 4 but are not in each other's neighborhood. Assume the channels are with good quality between node 1 and node 2/3 and between node 2/3 and node 4, but the channel between nodes 1 and 4 experiences severe fading. In the transmission duration shown in Fig. 4.3(a), node 1 initiates a direct transmission towards node 4. As node 2 and 3 are both in the receiving mode, they overhear the packet, as indicated by the dashed edges. Perceiving that there is no ACK for the packet from node

4 due to the link failure, node 2 and 3 both store the packet into their own MAC queue and treat it equally with their own direct packet so that they are potential relays for the packet. In a following transmission duration shown in Fig. 4.3(b), according to the transmitter selection criterion, node 1 and 2 could both be selected as transmitters and node 3 and 4 are still in the receiving mode. Suppose that the priority of the packet from node 1 to node 4 is relatively high, and both node 1 and 2 indicate their preference to send it to node 4 in the RTSs. By receiving the RTSs and completing channel estimation, node 4 selects node 2 as the transmitter for the packet as the channel condition from node 2 to node 4 is better than that from node 1 to node 4, in order to avoid redundant transmission. Therefore, node 1 withholds the transmission, as indicated by the dotted edge, and node 2 successfully relays the packet to node 4. In the ACK-I phase, node 4 feeds back the information about the successful reception. After receiving the ACK, potential nodes (original source or relays) that are currently transmitters, i.e. node 1 and 2, remove the packet from their queues. In order to make sure the packet is also removed from the queues of candidate relay nodes that currently serve as receivers (which are also in the process of sending out ACK to their corresponding transmitters) and are not able to receive ACK-I, e.g. node 3, transmitter nodes that have received the ACK-I, i.e. node 1 and 2, send out ACK-II to further rebroadcast the successful reception information. To this end, the packet is successfully transmitted through the cooperative relay transmission and removed from all queues.

4.6 Performance Evaluation

In this section, we evaluate the performance of our proposed algorithms through simulations based on a detailed MATLAB simulator we have built. We consider an ad hoc network with random topology where nodes are distributed uniformly over a $1250m \times 1250m$ area. Each node is equipped with an array of 4 antennas and has a reference transmission range of $250m$ as in a standard IEEE 802.11 wireless network. Both path loss and independent Rayleigh fading are incorporated for each wireless link between an antenna pair. For each node, the number of incoming data packets is Poisson distributed with a given mean value λ and the destination of each packet is chosen at random. The size of a packet is 200 bytes. A simulation result is obtained by averaging over ten runs of simulations with different seeds.

The two-phase scheduling algorithm proposed in Section 4.4 is implemented based on the MAC protocol described in Section 4.5. The Cooperative Relayed Spatial Multiplexing schemes proposed in this chapter are named as CRSM-C or CRSM-D respectively, depend-

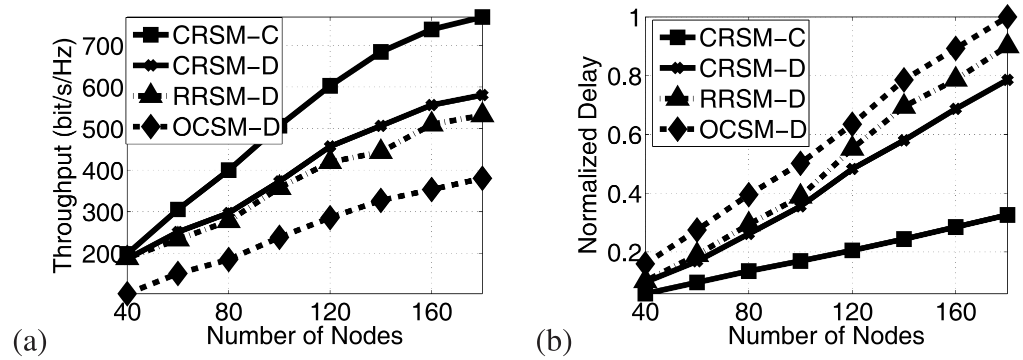


Figure 4.4: Impact of node density: (a) Throughput; (b) Normalized delay.

ing on whether a centralized scheme or a distributed scheme is used for the determination of transmitter nodes and the number of transmission streams. Correspondingly, we implemented two reference TDMA-based schemes in the distributed manner for performance comparison. One scheme is the Distributed Opportunistic and Cooperative Spatial Multiplexing (OCSM-D) scheme proposed in [19] which does not involve a relay transmission, the other scheme is also based on OCSM-D but have random relay selection enabled for performance enhancement, which is denoted as Distributed Random Relayed Spatial Multiplexing (RRSM-D). The metrics we use are throughput and normalized delay. Throughput is the total effective data rate of the network averaged over the number of transmission durations. Delay time is defined as the number of transmission durations a packet waits in the queue before it is removed from the MAC queue. The transmission delay includes the time for transmission of control packets. For the convenience of comparison, the results of delay are normalized to the maximum value in each figure. We investigate the impact on network performance due to four factors, namely node density, link failure ratio, packet arrival rate and retransmission threshold. The retransmission threshold defined in Section 4.5.1 is in the unit of TD, and a packet is dropped from both the source queue and queues of candidate relay nodes when the time lasted from the initial packet transmission exceeds the threshold. If not otherwise specified, the number of nodes in the network is 100, the link failure ratio is 0.3, the average packet arrival rate λ is 0.5 and the retransmission threshold is 8.

4.6.1 Impact of Node Density

The impact of node density is shown in Fig. 4.4. Increased node density leads to heavier traffic and also provides more links among nodes in a network. In case of severe links, the two CRSM schemes have a higher possibility of finding candidate relay nodes to assist in transmission by taking advantage of the improved connectivity. In Fig. 4.4 (a), CRSM-D is observed to improve the throughput up to 53% compared to OCSM-D. Effective scheduling of packets with relay also reduces the queuing delay as seen in Fig. 4.4 (b). Compared with RRSM-D which uses a preselected relay, CRSM-D implicitly and adaptively selects the node scheduled to transmit the first as the relay, which not only helps to speed up relay forwarding but also helps to balance load among nodes. These benefits are reflected in the up to 14% improvement in throughput and 13% reduction in delay.

4.6.2 Impact of Link Failure Ratio

A link is considered to be failed if a packet transmitted through it can not be received successfully by its receiver. Link failure can be a result of path loss, deep fading of channels, mobility of nodes, etc. We use link failure ratio (LFR) to model the percentage of failed links over all direct data transmission links between each pair of source and destination in the network. The failed links are randomly selected based on the link failure ratio and they are disconnected throughout the current run of simulation. The two CRSM schemes are shown to have a robust performance under different link failure ratios, as in Fig. 4.5. In Fig. 4.5 (a), while the throughput of OCSM-D degrades tremendously with increasing LFR, only a slight throughput degradation is observed with both CRSM schemes. As the CRSM schemes can smartly leverage the functional relay links to send packets out, it helps maintain the throughput performance. The throughput of CRSM-D is three times that of OCSM-D when a frequent link breakage occurs at $LPF = 0.6$, and the delay reduction is up to 50%. A higher link breakage ratio would lead to increased delay. The significant performance improvement demonstrates the effectiveness of adaptively using relay in MIMO transmissions to improve reliability in a harsh transmission environment. Although RRSM-D also supports the use of relay, the random relay selection which does not take advantage of the channel conditions to select node for more reliable relay transmission is observed to be less effective than the adaptive scheme of cooperative relay proposed in this chapter, as the throughput drops faster with increasing LFR compared with CRSM-D. RRSM-D has up to 26% lower throughput and 25% higher delay compared with CRSM-D.

4.6.3 Impact of Packet Arrival Rate

The mean packet arrival rate λ captures the traffic load in a network. By adaptively using cooperative relay transmissions, high rate links are more efficiently utilized to schedule heavier

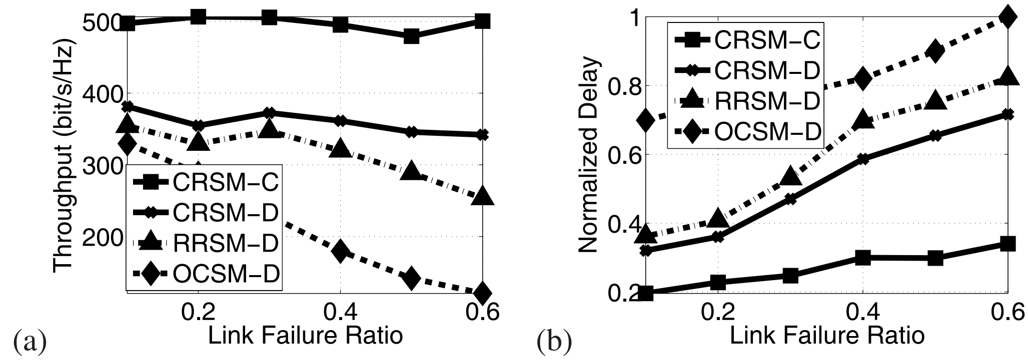


Figure 4.5: Impact of link failure ratio: (a) Throughput; (b) Normalized delay.

traffic load. In Fig. 4.6 (a), even with the heaviest traffic load, CRSM-D still achieves 35.7% higher throughput than OCSM-D. Although higher traffic increases queuing delay of packets due to limited network capacity, the delay of CRSM-D scheme is about 30% lower than that of OCSM-D. This demonstrates that even in the heavy traffic load condition, the relay can effectively improve performance. The node with the lowest load will be naturally selected as the relay. Meanwhile, CRSM-D consistently outperforms RRSM-D by up to 20% higher throughput and 19% lower delay, which further demonstrates the advantages of using adaptive cooperative relay instead of conventional relay schemes. In the case of heavy load, the packets are backlogged in the queue of nodes, and the delay increases significantly.

4.6.4 Impact of Retransmission Threshold

Retransmission is a common strategy used to deal with temporary transmission failure. The performances of CRSM and OCSM are compared in Fig. 4.7 under different values of the retransmission threshold F, as introduced in Section 4.5. In CRSM schemes, packets experienced direct transmission failure can be forwarded through relay links which may have better link conditions than the direct link. With increased value of F, both CRSM schemes keep a nearly constant throughput values, while OCSM-D undergoes 33.5% throughput reduction from $F = 2$ to $F = 14$. Even though more retransmissions help to increase the probability of successful packet reception, transmissions over poor links for a longer period of time would consume more network resources. On the contrary, both CRSM schemes actually take advantage of a larger F to conduct relay transmissions through adaptive scheduling. The delays of two OCSM schemes and CRSM-D scheme all increase with F with the increase of time to keep the packets in buffers, while CRSM-D remains to have much lower delay (up to 40%)

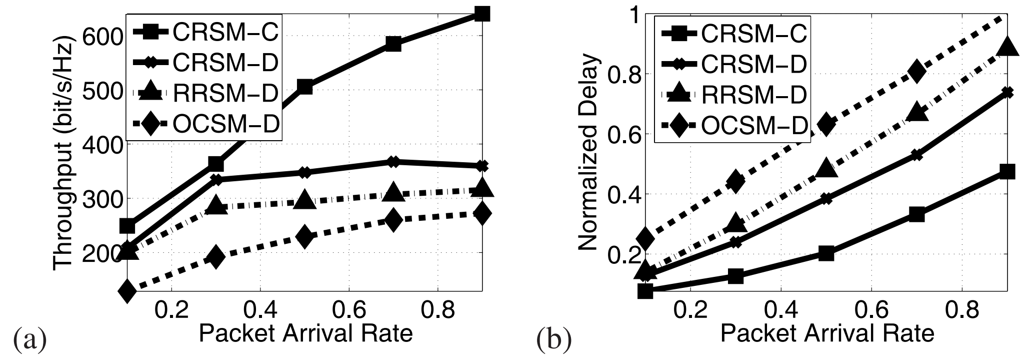


Figure 4.6: Impact of packet arrival rate: (a) Throughput; (b) Normalized delay.

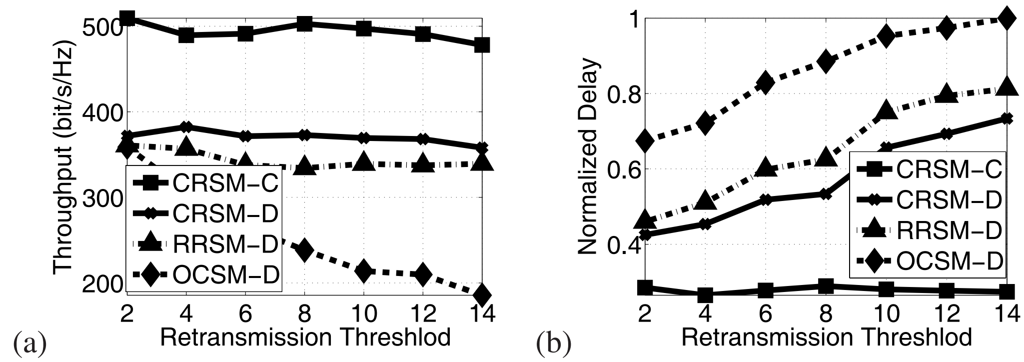


Figure 4.7: Impact of retransmission threshold: (a) Throughput; (b) Normalized delay.

than OCSM-D under all values of F. With varied valued of F, CRSM-D still takes advantage of the adaptive relay selection to achieve both higher throughput and lower delay than RRSM-D.

4.7 Conclusions

Ad hoc networks are popularly used in military and emergency rescue environments. In addition, there are increasing interests in applying ad hoc networks to connect various wireless devices to enable more powerful wireless applications and mobile computing capabilities. All these applications require higher network throughput and reliability. In this chapter, we design scheduling algorithms and MAC protocol to enable cooperative relay transmission in MIMO-based ad hoc networks, in order to jointly exploit the cooperative multiplexing gain and cooperative diversity gain to achieve overall higher data rate and lower delay under harsh channel conditions. We formulate the problem of packet scheduling with cooperative relay in MIMO ad hoc networks as an integer programming problem, and propose both centralized and distributed solutions to support relay transmissions. We also design an effective MAC protocol to facilitate the implementation of the distributed scheduling algorithm. Through extensive simulations, our scheme is shown to outperform the reference MIMO scheme which does not use relay or employs random relay selection, with significantly higher throughput and reduced average delay. This demonstrates the importance of incorporating relay transmissions in MIMO-based ad hoc networks and the effectiveness of the proposed algorithm in enabling concurrent MIMO and relay transmissions.

Chapter 5

Distributed Interference Management

Interference cancellation for MIMO links has been studied typically in downlink multiuser MIMO scenarios [55][56], where the common assumption is that the base station with multiple antennas takes the initiative to transmit to multiple users. However, it is difficult to apply these existing schemes to a wireless mesh network, where nodes are peer to each other and MIMO links exist between any neighboring node pairs. Moreover, in the downlink multiuser MIMO case, a user is supposed to receive from only one base station at a given time, and an optimal multiuser beamforming strategy at the transmitter side can completely eliminate the interference at targeted receivers and yield the maximum throughput. In a wireless mesh network, a node may simultaneously receive from several transmitters, either in terms of data streams or interference streams resulting from the data streams targeted to other receivers. The capability for a node to transmit and receive multiple streams is commonly modeled by the metric degree-of-freedom (DoF), which generally depends on the antenna array size. In a dense network where a transmitter has multiple concurrent receivers and/or a receiver has multiple concurrent transmitters, it would be quite difficult to suppress all the interference. Due to the limited DoF of nodes, there is also a tradeoff between MIMO multiplexing and interference cancellation. From the perspective of improving the network throughput, multiplexing can be utilized to increase the data rate of point-to-point link between a pair of nodes by simultaneously transmitting multiple streams on the link, while interference cancellation can take advantage of the DoF of either the transmitter or the receiver to eliminate the effect of interference streams and can thus enable ambient spatial reuse. To achieve the maximum network throughput, it would be necessary to flexibly allocate the DoF of nodes for multiplexing and interference cancellation according to the network topology, channel conditions and traffic requirements. In such a case, how to adaptively enable the interference cancelation to improve the throughput at a network level is a very challenging problem.

With the proliferation of mobile network enabled wireless devices and the fast progress of computing techniques and wireless networking techniques, it becomes more demandable to exploit MIMO advantages in a distributed way, where each node only have limited information of the network. Although MIMO techniques have been widely studied in a more centralized and infrastructure-based cellular system, there are very limited work and great challenges in extending MIMO technique into a fully distributed system over an infrastructure-free wireless ad hoc network. Some of the existing work have considered the problem of MAC layer algorithm design for MIMO network with MIMO transmission [12, 16, 19, 29], but they did not fully take advantage of interference cancellation to further leverage the MIMO benefits. Although some efforts have been made to consider the interference from the theoretical perspective [37, 38, 57–59], the physical model is simplified and the interference is considered to be ideally canceled. None of them provides a feasible solution to be implemented in a practical distributed scenario, which is a big design challenge. A distributed algorithm is also necessary for guiding the protocol design that could harvest MIMO gain to boost up the network performance. Moreover, the simplifications of physical models assumed in these work may significantly compromise the network performance.

In this chapter, we thoroughly investigate the physical layer characteristics of MIMO transmission to identify the opportunities and constraints for the design of adaptive MIMO multiplexing with interference management, and propose a distributed scheduling algorithm which considers the tradeoff of multiplexed transmission and interference cancellation and adaptively selects different strategies to maximize the capacity of MIMO-based meshed networks by fully exploiting the spatial DoF of both transmitters and receivers. To the best of our knowledge, it is the first distributed scheduling algorithm for MIMO-based mesh network with interference cancellation both at the transmitters and receivers and also consider the physical channel constraints.

The rest of this chapter is organized as follows. We discuss related work in Section 5.1. We describe the system model, investigate the physical layer details and identify the transmission constraints of interference management in Section 5.2. The proposed distributed scheduling algorithm is presented in Section 5.3 along with a centralized algorithm as the performance benchmark. Results for performance evaluation through simulations are presented in Section 5.4. The chapter is concluded in Section 5.5.

5.1 Related Work

As an important aspect of MIMO transmissions, interference cancellation has drawn interest from some research efforts. Sundaresan et al. [57] exploited the interference cancellation at receivers only. In [37], interference cancellation is supposed to be done by costing DoF of both transmitters and receivers. Hamdaoui and Shin in [38] considered that the interference between two links could be canceled by either a transmitter or a receiver, but not both. In [58], a node-level ordering scheme was proposed to identify the role of each node in performing interference cancellation. Li et al. [60] concurrently considered interference cancellation and alignment, and propose a convex programming based algorithm for the problem. In [59], Shi et al. re-visited the problem of MIMO modeling and developed a simple link layer model for multi-hop MIMO networks based on accurate accounting of how DoFs are consumed. In [61], optimal stream scheduling for MIMO links was studied for a single collision domain, and it was shown that optimum throughput is achieved when the task of interference cancellation is shared equally between every transmitter and every receiver. However, the aforementioned work [37, 38, 57–60] have assumed simplified physical model and overlooked the impact of channel condition by assuming streams have homogeneous data rate. In fact, the cancellation of interference is dependent on the network topology and channel condition, and interference cannot always be ideally canceled as assumed in these work. These simplifications may not only significantly compromise the network performance, but also make the optimal model formulated far from the practical network condition. In addition, none of them provides a feasible solution to be implemented in a practical distributed scenario and meet the physical channel constraints. In [13], the newly active transmitter and receiver were supposed to cancel interference when a newly active transmission joins other ongoing transmissions. Although it is distributed, this scheme cannot take advantage of MIMO to support more concurrent transmission, and results in suboptimal performance. In [62], Gelal et. al studied topology control issues of using successive interference cancellation in multi-user MIMO networks and proposed both centralized and distributed frameworks. However, transmitter side interference cancellation was not considered, which may cause potential DoF loss.

Recently, there have also been some efforts in applying MIMO and interference cancellation into wireless networks through testbed study. Experimental study was performed in [63], [64] and [65] for MIMO application in wireless LANs. Specifically, Gollakota et. al [63] proposed a new approach named interference alignment and cancellation (IAC) for decoding concurrent sender-receiver pairs in MIMO LANs. Aryafar et. al [64] presented the design and implementation of a multiuser beamforming system and experimental framework for wireless LANs. More recently, a distributed random access protocol for MIMO networks is proposed in [65] using a combination of interference nulling and interference alignment. However, wireless LANs are different from wireless mesh and ad hoc networks in the network scale and the algorithm and framework for wireless LANs cannot be simply applied to wireless ad hoc networks, especially in the challenging distributed case.

5.2 Fundamentals for Spatial Multiplexing with Interference Cancellation

To design an effective scheduling scheme with consideration of MIMO multiplexing transmission with interference cancellation, it is important to accurately understand the principles and model the techniques of MIMO physical layer. In this section, we present the physical layer fundamentals of the MIMO node operations for maximizing the aggregate data rate in the network.

5.2.1 Notations and Concepts

Denote the nodes in the network as a set V. For a node i, its neighboring nodes are denoted as a set V_i. In a specific transmission time slot, the neighboring transmitter nodes and receiver nodes are referred to as sets T_i and R_i respectively. The size of the antenna array of node i is denoted as N_i^{ant} and the total transmit power of node i is denoted as P_i.

For the spatial multiplexing transmission between a transmitter node i and a receiver node j, suppose m_{ij} data streams can be sent, which is denoted as a data vector $\mathbf{d}_{ij}$ of dimension m_{ij}. While some conventional spatial multiplexing scheme, e.g. the BLAST approach [4] does not use any channel precoding and essentially leaves the task of interference cancellation to the receiver, interference can also be canceled at the transmitter side, using the so called beamforming scheme, such as zero-forcing beamforming [55]. Considering transmitter beamforming, the transmission towards node j is associated with a specific precoding matrix $\mathbf{W}_{ij}$ of dimension $N_i^{ant} \times m_{ij}$. The aggregate transmitting signal from node i can be represented as

$$\mathbf{s}_i = \sum_{j \in R_i} \mathbf{W}_{ij} \mathbf{d}_{ij}. \tag{5.1}$$

Note that $\mathbf{W}_{ij}$ is subject to the power constraint of node i, i.e., $\| \mathbf{W}_{ij}\mathbf{d}_{ij} \| = P_i$.

Denote the channel matrix from the transmitter node i to the receiver node j as an $N_j^{ant} \times N_i^{ant}$ matrix

$$\mathbf{H}_{ij} = \left[\mathbf{h}_{ij1}, \mathbf{h}_{ij2}, \ldots, \mathbf{h}_{ijN_i^{ant}}\right], \tag{5.2}$$

where $\mathbf{h}_{ijk}$ ($k = 1, \ldots, N_i^{ant}$) is an $N_j^{ant} \times 1$ vector representing the channel between the k-th antenna of node i towards the receiver node j. The received signal at a receiver node j is therefore an $N_j^{ant} \times 1$ vector $\mathbf{y}_j$:

$$\begin{aligned} \mathbf{y}_j &= \sum_{i \in T_j} \mathbf{H}_{ij}\mathbf{s}_i + n_j \\ &= \sum_{i \in T_j} \sum_{r \in R_i} \mathbf{H}_{ij}\mathbf{W}_{ir}\mathbf{d}_{ir} + n_j \\ &= \sum_{i \in T_j} \mathbf{H}_{ij}\mathbf{W}_{ij}\mathbf{d}_{ij} + \sum_{i \in T_j} \sum_{r \in R_i, r \neq j} \mathbf{H}_{ij}\mathbf{W}_{ir}\mathbf{d}_{ir} + n_j. \end{aligned} \tag{5.3}$$

In equation (5.3), the first term is the aggregation of the desired data streams, the second term represents the multi-user interference and the third term is the receiving noise. Due to the broadcast nature of transmissions in a wireless mesh network, the data stream towards a receiver is also received at other neighboring receiver nodes of the transmitter as an interference stream if it is not intentionally eliminated, which induces the second term in equation (5.3) and reduces the receiving signal to interference and noise ratio (SINR) thus lowers the aggregate data rate.

Considering the receiver of an interference stream, the interference in a network can be categorized into two types. One is the *inter-user interference*, referring to the interference caused at receivers that are not the target receiver of a stream, and *inter-stream interference*, referring to the interference between data streams that are received simultaneously at a same receiver. The inter-stream interference can only be canceled when the transmitter and receiver cooperate, i.e. employ both pre-coding and post-coding at transmitter and receiver respectively with the coding weights calculated through SVD-decomposition [32], so that the streams are transmitted equivalently over orthogonalized channels. Different from inter-stream interference, the inter-user interference can be canceled either at the transmitter or the receiver, by using specific pre-coding or post-coding weight. For example, when all the interfering transmitters apply transmitter side interference cancellation towards node j by appropriately selecting the values of $\mathbf{W}_{ij}$, the second term in equation (5.3) is forced to be zero. Both inter-user and inter-stream interference can also be canceled at the receiver side through successive interference cancellation (SIC) technique. Note that SIC is a supplement to the receiver architecture to improve the receiving quality, and a receiver can use simple zero forcing method to decode streams. The algorithms proposed in this chapter can be adapted to work without SIC at the receiver.

Due to the limited DoF of a transmitter node, it is not able to cancel its interference towards all the neighboring receiver nodes, which will be further explained in details later. In order to denote a transmitter's role in interference cancellation, the portion of interference towards an interfering receiver that is intentionally canceled by a transmitter is called *tended interference*, which does not cost DoF at the specific receiver's side for cancellation. The other received interference streams are not canceled by the transmitter, which we call *untended interference*, and these streams could take up DoF of receiver nodes for appropriate cancelation, i.e. through SIC decoding.

Illustrations are given in Figure 5.1 to show the tradeoff between multiplexing transmission and interference cancellation in a MIMO network. The 4 nodes in the figures are within the transmission range of each other, and each of them is equipped with four antennas. In Figure 5.1 (a), node 1 transmits simultaneously towards node 2 and 4 with a data stream each, and transmitter side interference cancellation is employed to eliminate the inter-user interference. In such a case, node 4 is able to receive three additional streams from node 3. Note that without the transmitter side interference cancellation, node 4 is interfered by the transmission between node 1 and 2 and can receive at most two streams from node 3. This example demonstrates that employing the transmitter side interference cancellation is beneficial in increasing the total number of concurrent streams in a network. However, interference cancellation can only be performed under certain constraint. In Figure 5.1 (b), node 3 as a transmitter can transmit three data streams towards node 4 and one data stream towards node 2 simultaneously, with inter-user interference canceled. It is then unable to cancel the interference towards node 1 as its DoF has been taken up. Node 1 as a receiver then needs to perform receiver side interference cancellation. Suppose its DoF is 4, it will not able to receive any data stream in this case. The example indicates that a transmitter can cancel interference towards only a limited number of receivers, and needs to select properly among all its neighboring receivers.

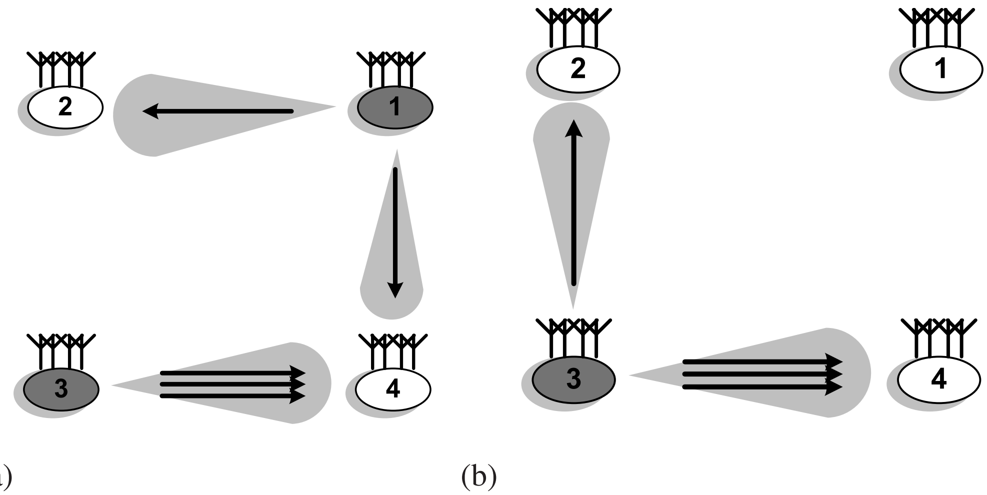


Figure 5.1: Illustrations of multi-user multiplexing and interference cancellation.

5.2.2 The Transmission Mechanism

According to equation (5.3), a transmitter may need to maximize the strength of data streams, i.e. the magnitude of the first term in (5.3), while having the impact of interference to be as low as possible, i.e. the magnitude of the second term in (5.3).

To eliminate the inter-user interference from the transmitter node i, a constraint is imposed for the selection of modulation matrix that $\mathbf{H}_{ij}\mathbf{W}_{ir} = \mathbf{0}$ for $j \neq r$, known as block diagonalization constraint [55]. Define R_i^t as the set of *tended* receivers that the inter-user interference from node i to nodes in R_i^t is eliminated. A matrix $\widetilde{\mathbf{H}}_{ij}$ is defined as

$$\widetilde{\mathbf{H}}_{ij} = \left[\mathbf{H}_{i1}^T \ldots \mathbf{H}_{i(j-1)}^T \; \mathbf{H}_{i(j+1)}^T \ldots \mathbf{H}_{i|R_i^t|}^T\right]^T. \tag{5.4}$$

The value of $\mathbf{W}_{ij}$ is forced to lie in the null space of $\widetilde{\mathbf{H}}_{ij}$. In other words, a data stream can be transmitted from node i to node j without causing the interference to nodes in $R_i^t \setminus \{j\}$ only if the null space of $\widetilde{\mathbf{H}}_{ij}$ has a dimension greater than 0, which is satisfied when $N_i^{ant} >$ rank$(\widetilde{\mathbf{H}}_{ij})$. Let $\widetilde{\mathbf{L}}_{ij}$ be the rank of $\widetilde{\mathbf{H}}_{ij}$, and the singular value decomposition (SVD) of $\widetilde{\mathbf{H}}_{ij}$ is represented as

$$\widetilde{\mathbf{H}}_{ij} = \widetilde{\mathbf{U}}_{ij}\widetilde{\mathbf{\Sigma}}_{ij}\widetilde{\mathbf{V}}_{ij}^*, \tag{5.5}$$

where * indicates the Hermitian transpose. $\widetilde{\mathbf{U}}_{ij}$ and $\widetilde{\mathbf{V}}_{ij}$ are both unitary matrix, and $\widetilde{\mathbf{\Sigma}}_{ij}$ is a diagonal matrix with singular values of $\widetilde{\mathbf{H}}_{ij}$. The last $N_i^{ant} - \widetilde{\mathbf{L}}_{ij}$ right singular vectors in $\widetilde{\mathbf{V}}_{ij}$ form an orthogonal basis for the null space of $\widetilde{\mathbf{H}}_{ij}$ and thus are the candidates for the modulation matrix $\mathbf{W}_{ij}$ for the receiver j. However, this is a relatively strong condition especially in a rich scattering environment where the channel matrix for each link is prone to have full rank, i.e. rank$(\mathbf{H}_{ij}) = \max\{N_i^{ant}, N_j^{ant}\}$.

To relax the block diagonalization constraint for precoding of transmitters, it is necessary to consider the postcoding matrix of receivers. Let $\mathbf{M}_{ij}$ be an $m_{ij} \times N_j^{ant}$ matrix consisting of the m_{ij} beamformers the receiver j employs in receiving the m_{ij} data streams from node i, each matrix $\mathbf{H}_{ir}$ ($r = 1, \ldots, j-1, j+1, \ldots, |R_i^t|$) in (5.4) is now replaced by $\mathbf{M}_{ir}^* \mathbf{H}_{ir}$. By using the resulting $\widetilde{\mathbf{H}}_{ij}$ to find the precoding matrix, $\sum_j m_{ij} < N_i^{ant}$ streams can be transmitted concurrently. Although some inter-user interference is allowed to be transmitted here, it is later eliminated at the output of the receiver beamformers with which a stream is steered to the nulls of the $\mathbf{M}_{ij}$ beam patterns. An appropriate candidate for $\mathbf{M}_{ij}$ is the m_{ij} dominate left singular vectors of $\mathbf{H}_{ij}$ [55], which will be used in our algorithm.

Recall that for a transmitter node i, the degree-of-freedom (DoF) is constrained by its antenna array size N_i^{ant}. The DoF can be utilized for multiplexed data stream transmission or interference cancellation and results in the tradeoff between the two. To be specific, if the transmitter i intends to cancel interference towards node r which is not a target receiver for any of its data streams, the calculation of $\mathbf{W}_{ij}$'s with consideration of receiver r takes up 1 DoF and reduces the number of data streams. Moreover, as interference cancellation imposes extra constraints for the calculation of $\mathbf{W}_{ij}$'s, the strength of the data streams may not be as high as that without the interference cancellation.

The DoF of a receiver node i is also constrained by its antenna array size N_i^{ant}. In order to facilitate better decoding performance, the receiver side postcoding for interference cancellation can work concurrently with successive interference cancellation (SIC). Moreover, SIC is supposed to cancel the part of interference from transmitters which do not apply transmitter side interference cancellation for this receiver, i.e. transmitter nodes that are in the set $T_j \setminus T_j^t$ where T_j^t is the set of transmitters that have their tended interference (thus the interference will be canceled) towards j. We apply receiver postcoding only to cancel intra-stream interference when a transmitter sends multiple concurrent streams to a specific node. The structure of the transmitter and receiver are illustrated in Figure 5.2(a) and (b) respectively. Considering both tended and untended interference, we can see that the sum of the DoF for decoding data streams and for canceling untended interference at a receiver node i should not exceed the antenna array size N_i^{ant}.

In summary, a data stream $\mathbf{d}_{ij}$ can cause inter-user interference to all the receiver nodes other than j in the set R_i, which can be canceled in three ways.

- The inter-user interference is canceled solely by the transmitter. In this case, $|R_i| \leq$

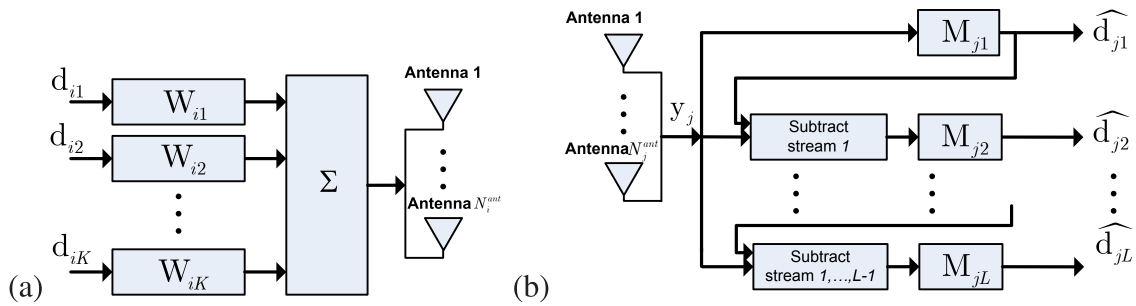


Figure 5.2: Illustrations (a) the transmitter structure; (b) the receiver structure.

N_i^{ant} should be satisfied, so that the DoF of the transmitter is sufficient to eliminate the interference towards all the neighboring receiver nodes.

- The inter-user interference is canceled solely by the receiver. This case exists only when a transmitter uses all its DOF for multiplexing transmission towards the receiver j, and does not have DoF available to eliminate the interference towards any interfering neighbor nodes.
- The inter-user interference is canceled by both the transmitter and receiver. Typically, we may have $|R_i| > N_i^{ant}$, in such a case, node i needs to select no more than N_i^{ant} receivers as tended receivers in R_i^t and cancel the interference towards them using precoding; for the other interfered receivers in R_i, the interference is canceled at the receivers' side using successive interference cancellation.

5.2.3 Formulation of Constraints

Based on the above observations, we can formulate the constraints for DoF cost. Denote the set of transmitters nodes as T and the set of receiver nodes as R. Parameter x_i denotes the number of streams sent out by transmitter i, parameter y_{ij} indicates if the interference from the transmitter i to the receiver j is specifically canceled by i, i.e. $y_{ij} = 1$ if the interference from i towards j is tended and canceled using a DoF of i and $y_{ij} = 0$ otherwise. Note that when the multiplexed transmission is used in combination with the interference cancellation for a transmitter to simultaneously send concurrent streams towards a set of receivers, it does not cost extra DoF for interference cancellation. Therefore, we enforce that $y_{ij} = 0$ when there is at least one data stream from node i to node j.

The constraints for spatial multiplexing transmission with interference cancellation can then be formulated as follows:

$$x_i + \sum_{j \in R_i} y_{ij} \leq N_i^{ant}, \forall i \in T; \tag{5.6}$$

$$\sum_{i \in T_j} x_i(1 - y_{ij}) \leq N_j^{ant}, \forall j \in R; \tag{5.7}$$

$$x_{ij} = 0, 1, 2, \ldots; y_{ij} = \{0, 1\}. \tag{5.8}$$

The formulation is based on the DoF cost of nodes. Equation (5.6) and (5.7) are the constraints for transmitter and receivers respectively.

So far, we have described the physical model of MIMO multiplexing transmission with interference cancellation. Obviously, it is important to consider the tradeoff of the DoF for multiplexing and interference cancelation. In summary, the **problem** aims at maximizing the aggregate throughput of the network and involves the following two steps:

- Schedule the transmissions for multiplexing and interference cancellation subject to constraints (5.6)-(5.8).
- Calculate the precoding and postcoding weights.

Although earlier work [37, 38, 58, 59] attempt to solve the problem using standard method, as we clarified in section 5.1, the physical models assumed in these studies are often very simple, e.g. interference is assumed to be canceled ideally and all the streams are assumed to have homogeneous data rate. These simplifications may not only significantly compromise the network performance, but also make the optimal model formulated far from the physical network condition. Moreover, the assumption on a priori knowledge of the complete traffic information is impossible to obtain in a dynamic network scenario, especially when the channel condition and topology are subject to variations. As a result, there is a lack of practical algorithms to give a guideline for the implementation of these proposed models.

To take into account the physical channel opportunities and constraints, our problem is complex and hard to solve using a standard optimization solution, especially in a large-scale network. In addition, the goal of this work is to design *feasible* scheduling algorithms to maximize aggregate network throughput under practical channel constraints. Therefore, we propose a distributed scheduling algorithm in Section 5.3.2 to achieve our design goal by trading-off between the multiplexing and interference and considering channel conditions. For performance comparison, we also propose a centralized scheduling algorithm in Section 5.3.3.

5.3 Scheduling Algorithms for Optimal Spatial Multiplexing with Interference Cancellation

In this section, we present our algorithms for optimal spatial multiplexing with interference cancellation based on the model described in section 5.2.

5.3.1 Design Guidelines

Our scheduling algorithm attempts to schedule the streams in a network for multiplexing transmission and interference cancellation based on the topology and channel conditions. Specifically, there are two tasks that need to be fulfilled. One is to guarantee the transmitting and receiving DoF constraints and avoid the possibility of collisions at the receiver, and the other is to enable higher rate data streams so that the aggregate data rate of the network can be improved.

In a distributed scenario, as there is no centralized control mechanism, the scheduling decision can be made either at the transmitter nodes or at the receiver nodes. However, there is a tradeoff for taking either of the options. If the decisions are made at the transmitter nodes, channel information should be made available at the transmitter side first. A transmitter node can properly allocate streams to transmit antennas through pre-coding and cancel the interference partially. However, if all the transmitter nodes make the scheduling decision independently, it is very likely that the total number of streams (including data streams and interference streams) received at a receiver node exceeds the node's decoding capability. If the decisions are made at the receivers, as a receiver node has full knowledge of all data and interference streams it will receive, it can better select the set of streams to turn off so as to maximize its own receiving data rate. The disadvantage is that different receivers may decide to turn off different streams and lead to conflicting decisions, so extra coordination is still needed at transmitter nodes to finalize the decision.

As a result, the determination of the appropriate scheduling scheme in our algorithm involves the *cooperative decision* of both the transmitter and the receiver. The guidelines for the algorithm design is summarized as follows:

- For interference cancellation at the transmitter side, receiver nodes that perceive the transmission from the transmitter as strong interference and have lower decoding capability yet more potential neighboring transmitters are first selected, as these nodes are prone to have a lower possibility of receiver side interference cancellation and may suffer more from the interference caused by the transmission.

- For the interference cancellation at the receiver side, only residual interference due to the limited interference cancellation capability of transmitters is canceled.

- For the receiver side multiplexed stream selection, a receiver would prefer streams that result in a higher aggregate data rate. Therefore, streams that have higher stream quality are preferred. A receiver node should also notify the transmitters about strong interference streams, as canceling these streams using SIC at the receiver may possibly reduce the receiving data rate.

- For the transmitter side multiplexed stream selection, a transmitter node first needs to determine the number of streams it can transmit, to take full advantage of the spatial multiplexing and guarantee to meet the decoding constraint of receivers. It then has to determine the set of receivers for its transmission. Specifically, the set consists of receivers as the target for the data streams and receivers for tended interference to cancel. The selection of the receiver set is based on the channel conditions jointly with the receiver preference feedback.

Based on the guidelines, we propose a distributed scheduling algorithm in the following section.

5.3.2 A Distributed Scheduling Algorithm

In this section, we propose a distributed scheduling algorithm to adaptively allocate the DoF of transmitters and receivers for either multiplexing or interference cancellation purpose. The following steps are performed repeatedly in a time slot basis, and the whole network is operated in a TDMA fashion. Each transmission duration consists of a few control message exchanges and data transmission using different sub-duration. During each sub-duration for control message, the same type of messages are transmitted simultaneously and each of them is masked by a unique random code of its sender [36]. The random code is assigned to each node according to its node ID, and are almost orthogonal for different nodes, i.e. the cross-correlation of different nodes' codes is close to zero. Each node keeps a set of random codes, where the size of the set is large enough to cover the maximum number of nodes in its neighborhood. Taking advantage of the random codes, concurrently transmitted messages can be identified when received. By de-masking the message, original control messages can be recovered and the preambles in them can be used for channel estimation. As described below, the proposed algorithm only requires a very limited feedback and thus causes little overhead.

Announcement of Transmitters

Before the transmission starts, it is necessary to pre-select the candidate data streams and notify the potential receivers. As the topology and channel information is not available at this stage, the selection is solely based on the queuing packets of the transmitter.

In this step, each transmitter node i examines its packet queue, and picks up to N_i^{ant} candidate receiver nodes (denoted as a set R_i^{ini}) that the priority of the packets towards them are among the highest. The number of potential streams for node j is denoted as N_{ij}^{ini}. The transmitter then broadcasts the candidate receiver list along with a training sequence for channel estimation purpose.

Information Gathering at Receivers

Based on the information received from the potential transmitters, a receiver node needs to collect the local information including the current topology and channel conditions. Although a receiver is able to predict which transmitter and streams could contribute to its highest receiving data rate, explicit decisions at receivers may conflict with each other at a shared transmitter node, and is subject to further change. In order to avoid unnecessary operations and reduce the amount of signaling overhead, in this step, a receiver node does not particularly select preferred streams but attempts to gather sufficient local information to form the limited feedback for the transmitters.

In the slot of transmitter announcement, a receiver j gets the broadcast information from all its neighboring transmitters. Specifically, following the framework in [19], it is able to estimate the channel between each neighboring transmitter i (i.e. $\forall i \in T_j$) and itself, denoted as a channel matrix H_{ij} with dimension $N_j^{ant} \times N_i^{ant}$. The node j also collects a list of transmitters that intend to send to itself, denoted as a set T_j^d as well as a list of interfering transmitters, denoted as a set T_j^{int}.

In order to facilitate transmissions that guarantee the receiver DoF constraint (5.7), a receiver node would prefer to constrain its total number of received streams, including both data streams and interference streams. Instead of explicitly pointing out which streams to allow or turn off, the receiver could assign a possibility for each potential stream and have the transmitter to determine the stream selection later, so as to avoid the decision confliction at a transmitter shared by multiple receivers. Note that the total number of announced streams that node j could receive is counted as $N_j^{total} = \sum_{i \in T_j^d \bigcup T_j^{int}} \sum_{p \in R_i^{ini}} N_{ip}^{ini}$. If each of the stream is enabled with the probability of $\rho_j = \frac{N_j^{ant}}{N_j^{total}}$, the receiver DoF constraint can be guaranteed statistically. Therefore, the value of ρ_j is sufficient as feedback information for ensuring receiver DoF constraint while providing the transmitter with the stream selection flexibility.

To achieve a higher receiving data rate, a receiver should also select its preferred data streams as well as interference streams that impact its receiving the most. As the scheduling is not finalized yet so the complete information of signal and interference is unavailable, a receiver can only make *rough* estimation of the data rates achieved for the candidate data streams. The estimation of rate achieved by the data streams is based on the fact that the channel capacity of an orthogonalized channel depends on the magnitude of the eigen values, e.g. the rate of the k-th eigenchannel between node i and j can be calculated as $\mathscr{R}_{ijk} = \log \lambda_{ijk}$. A receiver can also select to use successive interference cancellation (SIC) for decoding and strong interference streams could severely impact the decoding performance. As a metric for measuring the spatial channel strength, the average norm of the channel vector is calculated. The potential interference streams are then sorted in the sequence of their strength to denote their impact to node j. The operations at a receiver node j is summarized in algorithm 9.

The receiver then broadcasts a message, including the probability ρ_j, the selected transmitter list in T_j^{sel} and the ordered interference list $T_j^{sel-int}$. A training sequence is also included for the channel estimation at transmitters.

Reselection at Transmitters

In a distributed scenario, a node can only get access to its local information, e.g. a receiver node is informed about the potential transmission of its neighboring transmitter nodes but does not have any knowledge about the local situation of its neighboring receiver nodes. As a result, a reselection phase has to be performed at the transmitter's side. After receiving the messages from all its neighboring receivers, a transmitter node i needs to determine the number of streams it transmits as well as the target of the streams. It may also have to select

Algorithm 9 Operations at Receiver Node j

Input: $H_{ij}, \forall i \in T_j^d \bigcup T_j^{int}$, $N_{ij}^{ini}, \forall i \in T_j^d$
Output: ρ_j, T_j^{sel}, $T_j^{sel-int}$
Initialize: $T_j^{sel} = \emptyset$, $T_j^{sel-int} = \emptyset$
$\rho_j = \frac{N_j^{ant}}{\sum_{i \in T_j^d \bigcup T_j^{int}} \sum_{p \in R_i^{ini}} N_{ip}^{ini}}$
for $\forall i \in T_j^d$ **do**
    Perform SVD decomposition for matrix H_{ij}, get a number of eigenvalues $\{\lambda_{ijk}\}$ sorted in the order of magnitude
    Calculate rate estimation using beamforming $\hat{\mathscr{R}}_{ij} = \sum_{k=1}^{N_{ij}^{ini}} \log \lambda_{ijk}$
end for
T_j^{sel} contains the transmitters that correspond to the N_j^{ant} largest $\hat{\mathscr{R}}_{ij}$, to guarantee satisfying Equ. (5.7)
for $\forall i \in T_j^{int}$ **do**
    Estimate the channel strength $\gamma_{ij} = ||\mathbf{h}_{ijk}|| / N_i^{ant}$ for the channel vector from the k-th transmitter antenna to the receiver j
end for
$T_j^{sel-int}$ contains the interfering transmitters in the descending order of γ_{ij}

among all its neighboring receivers for appropriate transmitter side interference cancellation if needed.

Firstly, for a transmitter i, it estimates the channel and obtains the channel matrix between any target receiver nodes (denoted as a set R_i) and itself, denoted as H_{ij}. In order to guarantee that the data streams can be decoded correctly at the receiver side, the DoF constrain (5.7) needs to be satisfied. As a result, a transmitter node i has to constrain its number of transmitted streams based on the topology information fed back by its neighboring receivers. In order to ensure all the receiver nodes in its neighborhood to have a high probability of meeting the degree constraint, node i constrains the number of independent data streams it sends to a number as $N_i^{allo} = \min\{N_i^{ant}, \alpha \min_{j \in R_i} N_{ij}^{ini} \rho_j\}$, where $\alpha \geq 1$ is a compensation factor. As constraining the number of streams to be the minimal allowable number in the neighborhood is quite conservative and may cause the waste of receiver DoF, the value of N_i^{allo} is proportionally scaled by α to potentially increase the total number of streams in the network.

As the next step, transmitter i needs to determine the specific receivers for the transmission of the independent data streams. The reselection process is described in algorithm 10. In lines 4-10, the number of streams is adjusted from N_i^{allo}, which could be non-integer, to an integer value N in a randomized way. In lines 11-15, the strength of the eigenchannels between i and its candidate receivers is examined through SVD decomposition. In lines 16-27, no more than N data streams are selected based on the channel strength and the receiver preference. Finally, if a transmitter has residual DoF, it can be used for transmitter side interference cancellation as in lines 28-32. The selection is based on the preference list fed back by the receivers, i.e. the transmitter selects the receivers that perceives it as the most significant interfering transmitter first. If multiple receivers regards node i as the interfering transmitter of the same level, further selection is made based on the receiver's antenna array size and number of potential streams. As an interfered receiver is prone to have less capability of receiver side interference cancellation when it has relatively small size of antenna array but larger number of incoming streams, it is prioritized to be selected as tended interfering receiver in R_j^t. The transmitter should ensure the constraint (5.6) is met during this process. Finally, the precoding weight $\mathbf{W}_{ij}$ is calculated based on the singular vectors as in Equ. (5.5).

Data Transmission

After the reselection decisions are made, transmitters start transmission in the data transmission slot. With channel coefficients estimated in the first step, streams are decoded as described in Section 5.2. If a data stream is decoded correctly, the receiver node confirms with the transmitter node through ACK broadcast.

5.3.3 A Centralized Scheduling Algorithm

As discussed earlier, with the consideration of practical physical transmission channel and coding schemes, it is extremely difficult to find optimal solution using standard optimization methods. For providing a performance reference, in this section, we propose a centralized algorithm to gradually schedule the data streams in the network, while considering multiplexing transmission and interference cancellation at both the transmitter and the receiver.

The algorithm 11 proceeds in a sequential manner, and the receiver nodes that have so far been scheduled are saved in a set R_{sel}. When determining the optimal set of streams from transmitter nodes towards a receiver j as in lines 5-7, two parts of effect should be taken into consideration. For a receiver node j, the positive impact of a stream set Γ_j^l, referred to as $\mathcal{R}^+_{\Gamma_j^l}$ is the increase in the aggregate data rate perceived at receiver j, and the negative impact $\mathcal{R}^-_{\Gamma_j^l}$ is the decrease in aggregate data rates perceived by other receiver nodes that are already in R_{sel}, which is caused by the interference of the newly added streams in Γ_j^l. The streams considered here also include *sole IC streams*, i.e. streams that are solely for the purpose of transmitter

Algorithm 10 Reselection at Transmitter Node i

Input: $H_{ij}, \forall j \in R_i$
Output: R_i^{sel},R_i^t
Initialize: $R_i^{sel} = \emptyset$, $R_i^t = \emptyset$, $\Lambda = \emptyset$, $N_0 = 0$
$\bar{N}_i^{allo} = \lfloor N_i^{allo} \rfloor$
Generate a random number γ that $0 \leq \gamma \leq 1$
if $\gamma \leq N_i^{allo} - \bar{N}_i^{allo}$ **then**
$N = \bar{N}_i^{allo} + 1$
else
$N = \bar{N}_i^{allo}$
end if
for $\forall j \in R_i^{ini}$ **do**
Perform SVD decomposition for matrix H_{ij}, get a number of eigenvalues $\{\lambda_{ijk}\}$
$\Lambda = \Lambda \bigcup \{\lambda_{ijk}\}$
end for
Sort the elements in Λ in the descending order according to their magnitude
$k = 1$ as the index indicator for elements in Λ
while $N_0 < N$ **do**
Denote the receiver of the k-th element in Λ as $R(k)$
if $i \in T_{R(k)}^{sel}$ **then**
$R_i^{sel} \leftarrow R_i^{sel} \bigcup \{R(k)\}$
$N_0 \leftarrow N_0 + 1$, $T_{R(k)}^{sel} \leftarrow T_{R(k)}^{sel} \setminus \{R(k)\}$
end if
$k \leftarrow k + 1$
if $k > |\Lambda|$ **then**
Break
end if
end while
if Equ. (5.6) is satisfied **then**
Find receivers that regard i as strong interferer, denoted as R_i^{int}
Sort the receivers in R_i^{int} in the ascending order according to their value of N_j^{ant}/N_j^{total}
R_i^t contains the receivers that correspond to the first $N_i^{ant} - N$ elements in R_i^{int}
end if
Calculate the precoding weight $\mathbf{W}_{ij}, \forall j \in R_i^{sel} \bigcup R_i^t$

side interference cancellation. The net gain of Γ_j^l, i.e. an estimate of the its contribution to the aggregate data rate of the network, is therefore $\mathcal{R}^+_{\Gamma_j^l} - \mathcal{R}^-_{\Gamma_j^l}$. Each receiver node searches over all combinations of its candidate streams with the constraints defined in equations (5.6)-(5.8), and select the feasible set of streams which achieves the highest net gain. Note that the power of a stream s_{ijk} is assumed to be P_i/N_i^{ant} in this phase, and the actual power allocation is updated in line 29. After the net gain is calculated for each receiver, we then find the *nonconflicting* set of receivers that achieves the maximum aggregate data rate gain. In this way, we can avoid the zig-zag effect when different selection of receivers cause confliction at transmitters. Finally in line 29, after the selection for scheduling is made, we update the power allocation according to the actual number of streams scheduled for each transmitter, and calculate the precoding weight according to equations (5.5).

The complexity of the centralized algorithm is dominated by the complexity of searching for the stream set with the maximum net gain, which is $O(N^{N_j^{ant}})$, where N is the total number of candidate streams for j and N_j^{ant} is the antenna array size of node j. By selecting receiver with maximum non-negative net gain, it is guaranteed that the aggregate data rate of the network is improved. In each round of the cycle, at least one receiver is saturated and removed from R_{res}, so the algorithm is guaranteed to converge.

5.4 Performance Evaluation

In this section, the performance of our proposed algorithms is evaluated through simulations to verify their effectiveness. We have implemented the distributed algorithm proposed in Section 5.3.2, and the centralized algorithm proposed in Section 5.3.3 as a performance benchmark. As there is no existing algorithm for distributed scheduling with interference cancellation available, for the sake of comparison, we also implement a reference scheme named WOTXIC, which is a TDMA-based distributed algorithm share the same framework as proposed in this chapter. Different from our proposed algorithm, WOTXIC simply have the streams transmitted out without transmitter side interference cancellation, and solely relies on receivers for successive interference cancellation. Comparison with WOTXIC helps to demonstrate the benefits of adaptively employing transmitter interference cancellation.

We consider an ad hoc network with random topology. Nodes are distributed uniformly over a 1250m $\times$ 1250m area and the transmission range is 250m. Each node is equipped with an antenna array. The spatial channel between each neighboring node pair is subject to independent Rayleigh fading, and the path-loss index is 3.5. In each time slot, nodes are ran-

Algorithm 11 Centralized Scheduling

Input: $H_{ij}, \forall i \in T, j \in R$
Output: $\mathbf{s}_i, \forall i \in T$
Initialize: $T_{res} = T$, $R_{res} = R$, $T_{sel} = \emptyset$, $R_{sel} = \emptyset$,$STOP = 0$
while $STOP == 0$ **do**
  for $j \in R_{res}$ **do**
    Determine the *optimal* set of streams from transmitter nodes $i \in T_{res} \bigcap T_j$ to saturate j, i.e. satisfying Equ. (5.7)
  end for
  if None of $j \in R_{res}$ can find streams with positive aggregate data rate gain **then**
    BREAK
  end if
  Find the nonconflicting set of receivers in R_{res}, R^*, that achieves the maximum aggregate data rate
  for $j \in R^*$ **do**
    Calculate $\mathbf{s}_i$ for a selected transmitter i for j
  end for
  for $j \in R_{res}$ **do**
    if $j \in R^*$ or $V_j^{tx} = \emptyset$ **then**
      $R_{res} = R_{res} \setminus \{j\}$
    end if
  end for
  for $i \in T_{res}$ **do**
    if Equ. (5.6) is not satisfied for node i or $V_i^{rc} = \emptyset$ **then**
      $T_{res} = T_{res} \setminus \{i\}$
    end if
  end for
  if $T_{res} = \emptyset$ or $R_{res} = \emptyset$ **then**
    $STOP = 1$
  end if
end while
UPDATE $\mathbf{W}_{ij}, \forall i \in T, j \in R$

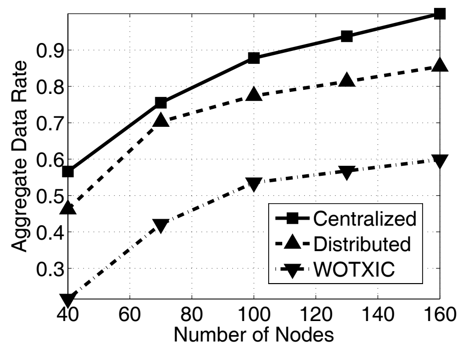


Figure 5.3: Aggregate data rate with different node density of the network.

domly selected as transmitters with a possibility of tx_ratio. A simulation result is obtained by averaging over 100 runs of simulations with different seeds. If not otherwise specified, the number of nodes in the network is 100, the size of antenna array at each node is 4, and the ratio tx_ratio for transmitter node selection is 0.4. The results in the figures are normalized with the maximum value in each figure for better comparison.

In Figure 5.3, we show the performance of our algorithm under different node density of the network. Regardless of the node density, the aggregate data rate achieved by our distributed algorithm is reasonably close to that achieved by the centralized algorithm, with a difference lower than 20%, which justifies that our distributed algorithm is capable of utilizing local information for making effective scheduling decisions. The reference scheme WOTXIC is observed to have up to 53% degradation in aggregate data rate compared with our distributed algorithm, as it does not take advantage of the transmitter side interference cancellation to allow more data streams in the network as well as improve the receiving signal to noise and interference ratio (SINR). Note that the data rate generally increases as more nodes are populated in the network. When the node density is sufficiently high, e.g. larger than 100, the increase in data rate slows down, as the number of data streams thus the data rate is further constrained by the DoF of nodes.

Figure 5.5 illustrates the changing of data rate with varied antenna array size. The data rate achieved by the three algorithms all increase almost linearly, as larger antenna array size generally provides higher DoF which can be utilized to accommodate the transmission of more data streams in the network. The difference between the centralized algorithm and the distributed algorithm is still less than 20% and the distributed algorithm can improve the data rate for over 45% compared with the WOTXIC scheme. The difference increases as the number of

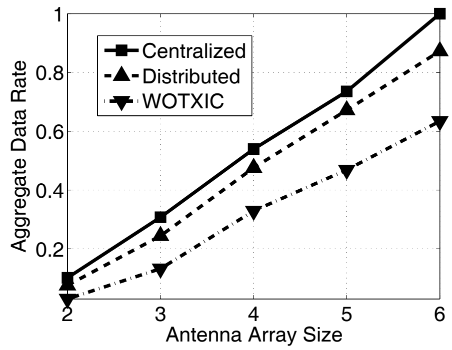


Figure 5.4: Aggregate data rate with different node antenna array size.

antennas increases, which indicates that our distributed algorithm can effectively exploit the available number of antennas for throughput improvement. It also demonstrate that, to better take advantage of MIMO transmission, it is important to adaptively employ the interference cancellation strategy.

Finally, we investigate the effect of the parameter tx_ratio, with which a portion of nodes in the network are randomly selected as transmitters. The receivers that they send streams to are determined by the scheduling algorithm as in 5.3.2. As our algorithm adaptively assumes interference cancellation in concert with multiplexing transmission, it consistently outperforms WOTXIC scheme under different values of tx_ratio. For all the three algorithms implemented, the maximum aggregate data rate is achieved when tx_ratio is around 0.4. On one hand, when the value of tx_ratio is too low, the transmitter nodes selected are not sufficient to saturate the receiving constraints of receiver nodes, i.e. receiver nodes have residual DoF to decode more data streams. On the other hand, when the value of tx_ratio is too high, the number of concurrent transmission is greatly limited by the DoF of receiver nodes and the aggregate data rate drops as a result. This indicates that, to achieve the optimum aggregate data rate, it is necessary to select an appropriate portion of nodes as transmitters. The result provides some guidance for transmitter selection, which could be part of our future work.

5.5 Conclusions

It is important to adaptively select transmission strategies in a MIMO-based wireless mesh networks to support multiple concurrent data transmissions and eliminate the impact of interference. Existing work in modeling and algorithm design has not provided a feasible solution

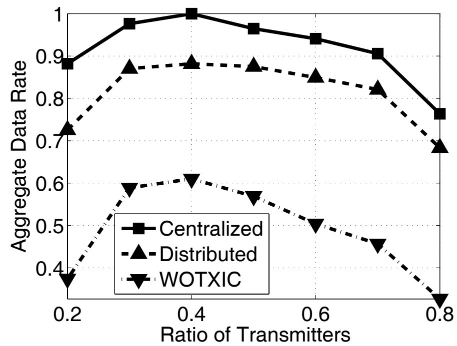


Figure 5.5: Aggregate data rate with different transmitter nodes ratio.

for MIMO multiplexed transmission with interference cancellation to be implemented in a practical distributed scenario. Moreover, a simplified physical model is generally assumed and the impact of channel conditions has been overlooked. In this chapter, we thoroughly investigate the physical model and make the first effort to design a distributed mechanism for MIMO transmission with adaptive interference cancellation at both the transmitter and receiver. Specifically, we allow transmitters to allocate their DoF to transmit multiple data streams and cancel interference towards a selected set of receivers, and receivers could use their antenna array for receiving and cancel the residual interference. Our performance results demonstrate that our proposed algorithms are very efficient in coordinating multiplexing transmissions and interference cancellation in a MIMO-based network, and achieve up to 52% improvement in data rate compared with the reference scheme that does not enable interference cancellation.

Chapter 6

MIMO-aware Routing

Most existing studies on applying MIMO technique in ad hoc and mesh networks focus on the physical and MAC layers [12, 14–19, 21, 66]. In wireless ad hoc and mesh networks, routing is an important factor that affects the system performance. From a network layer's perspective, MIMO nodes provide different transmission/receiving capabilities from conventional single-antenna nodes. A node equipped with multiple antennas could possibly transmit/receive more downlink/uplink streams, which can significantly impact the determination of optimal routes for traffic transmission. Moreover, the option of different MIMO strategies, i.e. spatial multiplexing or diversity with different levels of degree of freedom, could further increase the flexibility of routing decisions in a network with MIMO nodes. Therefore, it is of paramount importance to have the routing scheme to be *MIMO-aware* in order to fully leverage the benefits brought by MIMO into wireless networks. Some earlier work, i.e. [67, 68], has made an effort in designing heuristic routing algorithms and protocols. However, to the best of our knowledge, there is very limited work that has studied the problem of routing in MIMO-enabled networks from an optimization perspective and it is still not clear theoretically how much benefit can be achieved by taking advantage of the opportunities and addressing the constraints imposed by the incorporation of MIMO.

In this chapter, we study the problem of MIMO-aware routing in wireless mesh networks to leverage the benefits brought by MIMO. Our focus is to show that how much benefit can be achieved theoretically by having routing schemes be aware of MIMO capabilities and support the flexibility of MIMO mode selection. Different from previous work, we formally formulate the multi-source multi-destination multi-hop routing problem in MIMO-based wireless mesh networks as a multi-commodity flow problem to model the end-to-end traffic, subject to constraints that model the specific features of MIMO transmissions. To support the powerful features of MIMO transmission, we allow more flexible cooperation among nodes. Specifically, nodes in the network can perform many-to-many transmissions, in which a transmitter node can simultaneously transmit to multiple downstream nodes and a receiver node can simultaneously receive from multiple upstream nodes, and a transmission path can be established in reference to different MIMO channel modes based on the statistics of the channel conditions and different traffic demands. Based on the solid formulation, we make an effort to provide a sound theoretical upper bound which could serve as the reference for a practical system design. Specifically, we study two important network performance metrics, namely network flow and measure of congestion. We propose an approximate algorithm with polynomial time complexity to maximize the network flow, and also a distributed algorithm with provable efficiency to balance the traffic over the network and control the network congestion. The proposed algorithms provides important insights to serve as the reference for the design of practical routing strategy.

The rest of the chapter is organized as follows. Section 6.1 discusses the related work and we introduce the system model in Section 6.2. We define the constraints for MIMO-aware routing and formally formulate the problem in Section 6.3. Section 6.4 presents an alternative formulation of the problem to facilitate a centralized polynomial time approximation solution. Then in Section 6.5, we propose a distributed approximate solution to solve the joint routing and MIMO channel assignment problem. Finally, we provide the simulation results in Section 6.6 and conclude the chapter in Section 6.7.

6.1 Related works

The literature studies on MIMO communications are mostly constrained to physical layer and MAC layer, and there is very limited research on routing design. Two heuristic studies have been made in [67, 68]. The authors in [67] propose a routing protocol to switch between multiplexing and diversity at route maintenance stage, and routing with QoS provisioning is presented in [68] to exploit the multiplexing gain and interference cancelation properties of MIMO antennas. In [37], routing is briefly considered in solving the problem of joint control of routing, scheduling and stream subject to fairness constraints. In [38], the authors attempt to maximize the achievable throughput considering spatial multiplexing and spatial reuse and solve the problem under relaxed LP constraints, while in [39] the authors jointly optimize power and bandwidth allocation at each node. In [37–39], MIMO is modeled as a simplified constraint by counting the number of streams without considering the opportunities and constraints brought by different MIMO operational modes. Recently, there are some work on interference alignment in WLAN, [63, 69].

Meanwhile, multi-channel multi-radio ad hoc networks have also drawn great research interests. In [70], the authors provide necessary conditions for the feasibility of rate vectors in multi-channel wireless networks with multiple interfaces, and use them to find upper bounds on throughput. A solution for routing in multi-channel, multi-interface wireless mesh networks that maximizes the overall throughput of the network subject to fairness and interference constraints is developed in [71]. In [72], an online distributed algorithm that jointly solves the channel-assignment, scheduling and routing problem is proposed. As MIMO system shares some similarity as multi-channel multi-radio system, those works provide a good reference for our design. However, different from fixed frequency channels, MIMO transmission has more modes and there are more complicated constraints to be considered.

We have made an effort to model the multi-source multi-destination multi-hop routing problem in MIMO-based wireless mesh networks in [26], and provided a centralized and a distributed approximate solutions. In this chapter, we make the further attempt to provide insights on practical routing design based on the theoretical study. In addition, we present more details of our design and perform more extensive simulations to demonstrate the functionality of the proposed algorithms.

6.2 System model

We consider a fixed wireless mesh network where nodes are peers to each other. Nodes in the network are equipped with antenna arrays to facilitate MIMO communications.

6.2.1 Fundamentals of MIMO Transmission for Routing

In an ad hoc network where nodes are equipped with multiple antennas, MIMO transmission can be conducted over the data link between a pair of nodes. In general, there are generally two types of gain achieved by MIMO transmission. With multiple antennas at the transmitter and/or receiver, *spatial multiplexing* can be used to transmit multiple independent data streams between a node pair. At the receiver, each antenna receives a superposition of all of the data streams. In a rich scattering environment where the transmission channels for different streams are differentiable and independent, i.e., *orthogonal*, an intended receiver node

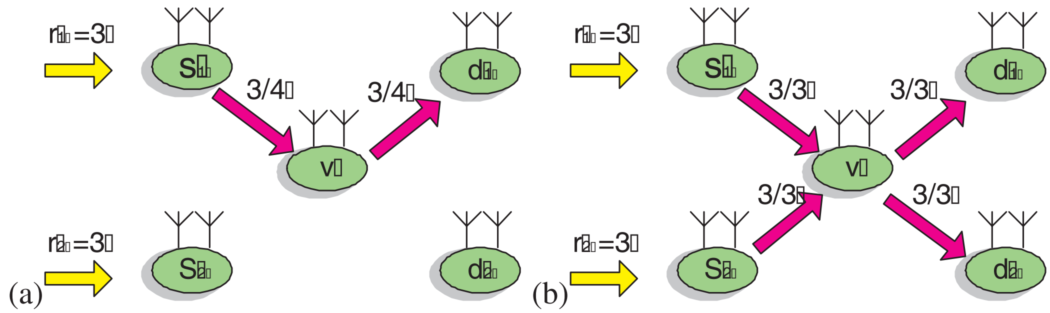


Figure 6.1: Illustration of routing in a MIMO-based network.

can separate and decode its received data streams based on their unique spatial signatures. This achieves the multiplexing gain that can provide a linear increase (in the number of antenna elements) in the asymptotic link capacity. As the transmission quality could be very different for multiple spatial paths, *spatial diversity* may be exploited to improve transmission reliability. There are different types of diversity techniques. When channel information is unavailable, dependent streams can be transmitted on different antenna elements over multiple time slots and improve transmission quality through *space time coding*. With adequate channel information, a subset of antennas that can transmit signals at better quality can be selected for transmissions through *selection diversity*, which is shown to outperform space-time coding [31].

Although some recent studies have been performed at the physical and MAC layer to address the challenges of leveraging MIMO advantages in networking, we believe that it is very important for the network layer to be aware of the specific characteristics of MIMO nodes and make more intelligent routing decisions. Based on the features of MIMO strategies, the array of antennas in each node can be grouped to form different MIMO channels, with different number of antennas and different achievable channel capacity. It is well known that MIMO link capacities vary significantly not just across strategies but also across the coherence time of the channel which is in the order of several millisecs.

Figure 6.1 illustrates the advantage of using MIMO-aware routing. In a network of five nodes, node s_1 and s_2 have 3 units of traffic demand for d_1 and d_2 respectively. The end-to-end paths of the two traffic flows both have to go via node v. As each node is equipped with two antennas in this network, we assume that the channel using two antennas has the capacity 4 and the channel using one antenna has the capacity 3 for each link. As node v can receive two independent data streams at most, a conventional routing strategy can route at most 3 units of traffic for either s_1 or s_2 as shown in Figure 6.1 (a), using the 2-antenna channel. As a better alternative, MIMO-aware routing can adaptively select the set of MIMO-channels to route the traffic, so that all the 6 units of traffic demand can be satisfied as in Figure 6.1 (b), by having each link use the 1-antenna channel.

6.2.2 Challenges

Due to the specific features of MIMO-based network, we have the following issues to consider in the design.

- **Link capacity.** Although in conventional networks the link capacity generally depends on network topology and channel conditions, in MIMO-based networks, it also depends on the size of the antenna arrays of nodes. For a transmission link between a node pair, the link capacity can be chosen from a set of varied capacities of different antenna combinations and strategies. Moreover, more than one combination may be used simultaneously to form several MIMO channels. The actual capacity of each MIMO channel can be estimated on a periodic basis and the statistics is used in routing decision.

- **Link channel assignment.** As an antenna array has limited size, the number of simultaneously used antenna combinations should not exceed the available number of antennas of the node, which is known as *transmitter degree constraint* in scheduling. Meanwhile, for simultaneous transmissions from multiple spatial channels, the set of antennas used by different spatial channels should not overlap, which we name it as *antenna compatibility constraint*. Also, as different antenna combinations have different capacities, it is important to determine which antenna combination to use when a route is determined.

- **Interference consideration.** With multiple antennas, a node can receive data streams while canceling interference streams concurrently, and the total number of streams received depends on the antenna size, which is described as *receiver degree constraint* in scheduling. From the perspective of routing, the antenna size can be also regarded as a measure of a node's capability of concurrent data receiving and interference cancellation.

- **Multi-path routing.** As nodes are endowed with many-to-many transmission capability by multiple antennas, it is beneficial to incorporate multi-path routing for end-to-end flows in order to better exploit multi-path diversity and maximize throughput. While using multi-path may lead to the problems regarding packet re-ordering and loss recovery, these issues have been studied in literature work on multi-path routing and are beyond the scope of this chapter.

In this chapter, we focus on routing traffic between different source/destination pairs and the corresponding MIMO mode selection. The problem of scheduling the flow in a specific time slot is beyond the scope of this chapter.

6.3 Problem Formulation

Based on the system model described in Section 6.2, we formally formulate the MIMO-aware routing problem in wireless mesh networks as an optimization problem. In order to model end-to-end traffic, we use a multi-commodity flow model for the routing of data packets across the network. That is, source nodes may send different data to their intended destination nodes through multi-path and multi-hop routing.

6.3.1 Graph representation

We represent the multi-hop wireless network via node topology graph $G = (V, E \bigcup E^I, F)$, where V is the set of nodes in the network, F is the set of data flows to be routed, E is the set of directed edges between nodes that can transmit data from one to the other and E^I is the set of directed edges which indicate the interference from a transmitter to nodes within its interference range during data transmission. To be more specific, given a data link $e \in E$, $t(e)/h(e)$ is used to represent the transmission/receiving end of the link e, and there is a directed edge from $t(e)$ to $h(e)$. In the network, there is a set of sources **s**, which send data to a set of destinations **d**, with the end-to-end rate demand vector **r**. Assume the rate vector has $F < |V|(|V| - 1)$ components. Each source-destination pair between which there is a traffic request is termed as a commodity. Let $s(f)/d(f)$ represent the source/destination node for commodity f, and $r(f)$ represent the flow that has to be routed from $s(f)$ to $d(f)$.

6.3.2 Notations and Example

Define the concept *MIMO channel* ($\mathcal{MC}$) as the MIMO spatial channel over a link that uses a designated set of antennas and corresponds to a specific MIMO operation mode. Denote the set of MIMO channels over link e as $\mathcal{MC}(e)$, and each element $\mathcal{MC}_i(e) \in \mathcal{MC}(e)$ has a *size*, denoted as $m_i^t(e)$ for the transmitter node and $m_i^h(e)$ for the receiver node, which is

Table 1: List of notations used in problem formulation.

Notation	Definition
f	Index of flows
i	Index of channels
e	Index of links
$s(f)$	Source of flow f
$d(f)$	Destination of flow f
$x_i^f(e)$	Flow on MIMO channel i over data link e that carries the data of flow session f
$c_i(e)$	Capacity of data link e on MIMO channel i
$m_i^t(e)$	Number of antennas used for constructing MIMO channel $\mathcal{MC}_i(e)$ at the transmitter end
$m_i^h(e)$	Number of antennas used for constructing MIMO channel $\mathcal{MC}_i(e)$ at the receiver end
$u_{i,a,e}$	Indicator of channel antenna association. $u_{i,a,e} = 1$ if and only if the MIMO channel i over link e uses antenna a of node $t(e)$ for transmission ($a = 1, \ldots, N_{t(e)}^{ant}$).

the number of antennas used for constructing this MIMO channel at the two ends of the link e. By using different sizes, a set of MIMO channels can be constructed to take advantage of spatial multiplexing and/or spatial diversity. Note that, with the calculation of antenna weights at transmitters and transmit over eigen-modes of the channel, MIMO channels can be considered orthogonal [32]. Suppose node $v \in V$ has N_v^{ant} antennas. Each MIMO channel (e, i) is associated with a set of antennas of node $t(e)$, which is indicated using the parameter $u_{i,a,e}$ ($a = 1, \ldots, N_{t(e)}^{ant}$). $u_{i,a,e} = 1$ if and only if the MIMO channel i over link e uses antenna a of node $t(e)$ for transmission.

For example, as in Figure 6.1, node v has two antennas a_0 and a_1, which can be used to compose different MIMO channels for the link $e = v \rightarrow d_1$. Channels 0 and 1, i.e. $(e, 0)$ and $(e, 1)$, both use one antenna, so $m_0^t(e) = m_1^t(e) = m_0^h(e) = m_1^h(e) = 1$, and thus $u_{0,0,e} = u_{1,1,e} = 1$ and $u_{0,1,e} = u_{1,0,e} = 0$. Channels 2 and 3, i.e. $(e, 2)$ and $(e, 3)$, are constructed by transmitting simultaneously over antennas a_0 and a_1, so $u_{2,0,e} = u_{2,1,e} = u_{3,0,e} = u_{3,1,e} = 1$ and $m_2^t(e) = m_3^t(e) = 2$. If the MIMO transmission strategy used for channel 2 is spatial multiplexing, i.e., independent data streams are transmitted simultaneously from the two antennas, the receiver has to use at least two antennas for successful decoding, therefore $m_2^h(e) = 2$. Alternatively, if the space-time coding, i.e. Alamouti code [6], is used for the transmission over channel 3, the receiver only needs one antenna for decoding so $m_3^h(e) = 1$. Note that the values of $m_i^h(e)$, $m_i^t(e)$ and $u_{i,a,e}$ are easy to obtain off-line and are static for each node.

All the notations are summarized in Table 1.

6.3.3 Flow Constraints

Each data link e has capacity $c_i(e)$ on MIMO channel i, and there is an estimated capacity for a given MIMO channel over a link for an estimation period. The set of $\mathcal{MC}s$ and the values of $c_i(e)$ can be saved as a look-up table and updated in each estimation period according to the topology/channel condition variations. The length of the period should be properly determined so that the value of $c_i(e)$ can correctly reflect the actual link condition. We use $x_i^f(e)$ to denote the flow on channel i over data link e that carries the data of the end-to-end flow session f, and define $g_i(e) = \sum_f x_i^f(e)/c_i(e)$ as the *utilization* of MIMO channel i over link e for all flows.

A necessary condition for rate vector **r** to be achievable is the existence of a link flow $x_i^f(e)$ that satisfies the following *flow conservation constraints*:

$$\sum_{e:t(e)=s(f)} \sum_{i \in \mathcal{MC}(e)} x_i^f(e) = r(f), \forall f; \tag{6.1}$$

$$\sum_{e:h(e)=d(f)} \sum_{i \in \mathcal{MC}(e)} x_i^f(e) = r(f), \forall f; \tag{6.2}$$

$$\sum_{e \in E_{in}(v)} \sum_{i \in \mathcal{MC}(e)} x_i^f(e) = \sum_{e \in E_{out}(v)} \sum_{i \in \mathcal{MC}(e)} x_i^f(e), \quad \forall f, \forall v \neq s(f), d(f); \tag{6.3}$$

where $E_{in}(v)$ and $E_{out}(v)$ are incoming and outgoing edges of node v in the set E. As the *flow capacity constraint*, each link should satisfy:

$$\sum_f x_i^f(e) \leq c_i(e), \forall e, \forall i \in \mathcal{MC}(e), \tag{6.4}$$

which can be simplified as $g_i(e) \leq 1, \forall e, i \in \mathcal{MC}(e)$.

6.3.4 MIMO Specific Constraints

While constraints (6.1)-(6.4) are conventional for flow problems, the use of MIMO technique imposes new constraints. Even though the degree constraints introduced in Section 6.2 are generally formulated in MAC layer, they actually have a significant impact over routing in the network layer. In order to address these constraints, we first present them with link-flow variables in each time slot, and then translate them into end-to-end rate variables for routing purpose.

Let $I_{e,i,\tau}$ be the indicator variable that has value 1 if and only if channel i is *active* over link e at time slot τ. Note that the channels over outgoing edges of v in E are considered active if there are data transmissions from node v, and the channels over incoming edges of v in the set E and E^I are considered active if there are data transmissions and interference transmissions to v respectively. To satisfy the *degree constraint* at the transmitter side, the number of antennas used by the active outgoing edges of a node v must be no larger than its number of antennas N_v^{ant} in each time slot τ:

$$\sum_{e \in E_{out}(v)} \sum_{i \in \mathcal{MC}(e)} m_i^t(e) I_{e,i,\tau} \leq N_v^{ant}, \forall v. \tag{6.5}$$

Similarly, corresponding to the receiver's degree constraint, the total number of antennas that is required to decode the receiving transmissions, including data and interference transmissions, should not exceed the receiving capability of the node. Therefore, we have:

$$\sum_{e \in E_{in}(v) \bigcup E_{in}^I(v)} \sum_{i \in \mathcal{MC}(e)} m_i^h(e) I_{e,i,\tau} \leq N_v^{ant}, \forall v. \tag{6.6}$$

Suppose routing is performed for each T time slots. Adding these sets of equations for all the T time slots and dividing by T results in the constraints:

$$\sum_{e \in E_{out}(v)} \sum_{i \in \mathcal{MC}(e)} m_i^t(e) g_i(e) \leq N_v^{ant}, \forall v; \tag{6.7}$$

$$\sum_{e \in E_{in}(v) \bigcup E_{in}^I(v)} \sum_{i \in \mathcal{MC}(e)} m_i^h(e) g_i(e) \leq N_v^{ant}, \forall v. \tag{6.8}$$

where $g_i(e)$ is the fractional link utilization for channel i over link e. Specifically, $g_i(e) = \sum_f \frac{x_i^f(e)}{c_i(e)} = \frac{1}{T} \sum_{1 \leq \tau \leq T} I_{e,i,\tau}$ for all e and i.

In addition, each node only has a limited number of antennas, and an antenna cannot be used for transmission over different MIMO channels simultaneously. To address this *antenna compatibility constraint*, we use the indicator variable $u_{i,a,e}$ introduced earlier to represent the constraint as follows:

$$\sum_{e \in E_{out}(v)} \sum_{i \in \mathcal{MC}(e)} u_{i,a,e} I_{e,i,\tau} \leq 1, \forall \tau, v, a. \tag{6.9}$$

Similarly as in (6.7)(6.8), adding these sets of equations for all the T time slots and dividing by T, we have:

$$\sum_{e \in E_{out}(v)} \sum_{i \in \mathcal{MC}(e)} u_{i,a,e} g_i(e) \leq 1, \forall v, a. \tag{6.10}$$

6.3.5 Optimization formulation

So far, we have derived the set of constraints for a feasible flow for routing data packets in a MIMO-based mesh network. There are many different objectives of interest that can be solved using an optimization framework. Based on the constraints, we formulate the routing problem in the form of a concurrent flow problem, where the desired rate vector is scaled and the objective is to determine the maximum scaling factor λ that satisfies the necessary conditions. In this way, the fairness in the resource allocation over flows can be ensured. The resulting linear program (LP) is given below:

$$\max \lambda, \tag{6.11}$$

Subject to:

$$\sum_{e:t(e)=s(f)} \sum_{i\in\mathcal{MC}(e)} x_i^f(e) = \lambda r(f), \forall f; \tag{6.12}$$

$$\sum_{e:h(e)=d(f)} \sum_{i\in\mathcal{MC}(e)} x_i^f(e) = \lambda r(f), \forall f; \tag{6.13}$$

$$\sum_{e\in E_{in}(v)} \sum_{i\in\mathcal{MC}(e)} x_i^f(e) = \sum_{e\in E_{out}(v)} \sum_{i\in\mathcal{MC}(e)} x_i^f(e), \quad \forall f, \forall v \neq s(f), d(f); \tag{6.14}$$

$$\sum_{e\in E_{out}(v)} \sum_{i\in\mathcal{MC}(e)} m_i^t(e) g_i(e) \leq N_v^{ant}, \forall v; \tag{6.15}$$

$$\sum_{e\in E_{in}(v)\bigcup E_{in}^I(v)} \sum_{i\in\mathcal{MC}(e)} m_i^h(e) g_i(e) \leq N_v^{ant}, \forall v; \tag{6.16}$$

$$\sum_{e\in E_{out}(v)} \sum_{i\in\mathcal{MC}(e)} u_{i,a,e} g_i(e) \leq 1, \forall v, a; \tag{6.17}$$

$$0 \leq g_i(e) \leq 1, x_i^f(e) \geq 0, \forall e, i \in \mathcal{MC}(e). \tag{6.18}$$

where equations (6.12)-(6.14) are flow conservation constraints, equations (6.15)-(6.16) stand for the routing constraints as the result of using MIMO antenna arrays, and equation (6.17) is used to meet the antenna compatibility constraint.

So far, we have presented a formulation for routing in MIMO-based wireless networks with flow variables.

6.4 The Centralized Algorithm for Throughput Maximization

The optimization problem formulated in section 6.3 is linear and can generally be solved by linear optimization algorithms, i.e. simplex method. However, the complexity is still an important concern. In this section, we follow the work in [73] and [70], and develop a fully polynomial time approximation algorithm using primal-dual algorithm, which is simple to implement and therefore can be potentially applied in a practical wireless network. In order to facilitate the solution, we first reformulate the problem using edge-path formulation and generalize the constraints, which is amenable to the development of the algorithm, then we describe the primal-dual algorithm to solve the optimization problem and obtain the maximum scaling factor λ.

6.4.1 Edge-path reformulation

First, note that the set of constraints (6.15)-(6.17) share a similar format in that each of them concerns a specific set of link/MIMO-channel pairs, so it is possible to generalize them into a simpler form for an easier reformulation. Suppose there are $\mathcal{L}$ sets $\{Q_j\}$ composed of link/MIMO-channel pairs that are as defined in constraints (6.15)-(6.17), then each of these constraints can be stated in the form as follows:

$$\sum_{(e,i)\in Q_j} \alpha_i(e) g_i(e) \leq \beta(Q_j), j = 1, 2, \ldots, \mathcal{L}, \tag{6.19}$$

where $\alpha_i(e)$ and $\beta(Q_j)$ are constants associated with the above constraints. For example, for a node v^*, constraint (6.15) concerns the set $Q_j = \{(e,i)|e \in E_{out}(v^*)\&i \in \mathcal{MC}(e)\}$, the corresponding constants are then $\alpha_i(e) = \{m_i^t(e)|e \in E_{out}(v^*)\&i \in \mathcal{MC}(e)\}$ and $\beta(Q_j) = N_{v^*}^{ant}$. In this way, although the number of constraints as described in (6.15)-(6.17) remains the same, they are generalized into a single formula (6.19).

In order to have an approximate solution, we first reformulate the problem into an edge-path formulation, so that the multi-commodity flows are represented as positive LPs. Let $\mathcal{P}_f$ represent the set of all possible simple paths composed of link/MIMO-channel pairs for the commodity f. For a path $P \in \mathcal{P}_f$ that is from $s(f)$ to $d(f)$, let $x(P)$ be the amount of flow on this path, constraints (6.12)-(6.14) are then translated to:

$$\sum_{P\in\mathcal{P}_f} x(P) = \lambda r(f), \forall f. \tag{6.20}$$

Furthermore, $x_i(e)$, the total amount of flow on channel i over link e is given by:

$$x_i(e) = \sum_f \sum_{P\in\mathcal{P}_f,(e,i)\in P} x(P), \forall(e,i). \tag{6.21}$$

As $g_i(e) = x_i(e)/c_i(e)$, equation (6.19) becomes:

$$\sum_{(e,i)\in Q_j} \alpha_i(e) \frac{\sum_f \sum_{P\in\mathcal{P}_f,(e,i)\in P} x(P)}{c_i(e)} \leq \beta(Q_j), j = 1, \ldots, \mathcal{L}. \tag{6.22}$$

In this constraint, link/MIMO-channel pairs that are both on path P and in set Q_j, i.e. $(e,i) \in P \cap Q_j$, are examined. Consider a single path P, from (6.22), we have $\sum_{(e,i)\in P\cap Q_j} \frac{x(P)}{c_i(e)/\alpha_i(e)} \leq \sum_{(e,i)\in Q_j} \alpha_i(e) \frac{\sum_f \sum_{P\in\mathcal{P}_f,(e,i)\in P} x(P)}{c_i(e)} \leq \beta(Q_j)$. Therefore, $x(P,j) = \beta(Q_j)(\sum_{(e,i)\in P\cap Q_j} \frac{1}{c_i(e)/\alpha_i(e)})^{-1}$ is the maximum amount of flow on path P allowed by $Q(j)$.

In summary, the edge-path formulation of the constraints in the original optimization problem is restated as follows:

$$\sum_{(e,i)\in Q_j} \frac{\sum_f \sum_{P\in\mathcal{P}_f,(e,i)\in P} x(P)}{\beta(Q_j)c_i(e)/\alpha_i(e)} \leq 1, \quad j = 1, 2, \ldots, \mathcal{L}; \tag{6.23}$$

$$\sum_{P\in\mathcal{P}_f} x(P) = \lambda r(f), \forall f; \tag{6.24}$$

$$x(P) \geq 0, \forall P \in \mathcal{P}_f, \forall f. \tag{6.25}$$

6.4.2 Primal-dual solution

According to the *weak duality property*, the objective value of any feasible solution of the minimization problem gives an upper bound on the optimal objective of the dual maximization problem. Following [73] and [70], we formulate the dual of the LP problem and develop a fully polynomial time approximation algorithm using a primal-dual algorithm. Let $y(j)$ be the dual variables for each set Q_j, and $z(f)$ be the dual variable for the rate scaling constraints in (6.25). The dual of the LP problem is then as follows:

$$\min \sum_j y(j), \tag{6.26}$$

Subject to:

$$\sum_{(e,i)\in P} \frac{\sum_{j:(e,i)\in Q_j} \frac{\alpha_i(e)y(j)}{\beta(Q_j)}}{c_i(e)} \geq z(f), \forall P \in \mathcal{P}_f, \forall f; \tag{6.27}$$

$$\sum_f r(f)z(f) \geq 1; \tag{6.28}$$

$$y(j) \geq 0, j = 1, \ldots, \mathcal{L}. \tag{6.29}$$

The dual problem is essentially an assignment of lengths to link/MIMO-channel pairs, such that $\sum_j y(j)$ is minimized. The proposed primal-dual algorithm is given in algorithm 12.

Algorithm 12 Centralized Routing

0: **Initialize:**
1: $y(j) = \delta, \forall j \in \{1, \dots, \mathcal{L}\}$ and $b = 0$
2: **while** $\sum_j y(j) < 1$ **do**
3:     **for** $f = 1, 2, \dots, F$ **do**
4:         $r = r(f)$
5:         **while** $r > 0$ **do**
6:             Assign each pair (e, i) with length $l_i(e) = \frac{\sum_{j:(e,i)\in Q_j} \alpha_i(e) y(j)/\beta(Q_j)}{c_i(e)}$
7:             Find the shortest length for each edge: $l(e) = \min_{i \in \mathcal{MC}(e)} l_i(e)$
8:             Compute the shortest path P^* from $s(f)$ to $d(f)$ based on $\{l(e)\}$
9:             Find the bottleneck capacity
            $u = \min_{j:(e,i)\in P^* \& (e,i)\in Q_j} x(P^*, j)$
10:             $\delta = \min\{r, u\}, r \Leftarrow r - \delta$
11:             $x_i(e) \Leftarrow x_i(e) + \delta, \forall (e, i) \in P^*$
12:             $y(j) \Leftarrow y(j)(1 + \frac{\epsilon\delta}{x(P^*, j)}), \forall j : \exists (e,i)^* \in P^* \& (e,i)^* \in Q_j$
13:         **end while**
14:     **end for**
15:     $b \Leftarrow b + 1$
16: **end while**
17: $\rho = \max_j \sum_{(e,i)\in Q_j} \frac{x_i(e)}{c_i(e)}$
18: Output $\lambda^* = \frac{b}{\rho}$

The algorithm initially assigns a weight of δ to all sets Q_j, and then proceeds in phases. In each phase we route $r(f)$ units of flow from $s(f)$ to $d(f)$ for each commodity f, and a phase ends when all the F commodities are routed. For each commodity f, The $r(f)$ units of flow from $s(f)$ to $d(f)$ are sent via multiple iterations, as in lines 5-13. In each iteration, each pair (e, i) is assigned with a length $\frac{\sum_{j:(e,i)\in Q_j} y(j)/\beta(Q_j)}{c_i(e)/\alpha_i(e)}$, and a corresponding shortest path path P^* from $s(f)$ to $d(f)$ that minimizes the sum of the length is determined by a shortest path algorithm, i.e., the Dijkstra's algorithm. Among the sets in $\{Q_j\}$ that have the intersection with P^*, we compare their values of maximum allowable flow $x(P^*, j)$, and the one with the minimum value $u = \min_{j:(e,i)\in P^* \& (e,i)\in Q_j} x(P^*, j)$ is the amount of the flow that can be sent on P^* in this iteration. Moreover, since $r(f)$ units of flow have to be sent for commodity f in each phase, the actual amount of flow sent is the lesser of u and the remaining amount of flow r to make up $r(f)$ in this phase. Once a flow is sent via a path, the weights of the sets $\{Q_j\}$ associated with the link/MIMO-channel pairs that carry the flow is updated, as in line 12. The algorithm then alternates between sending flow along shortest paths and adjusting the length of the link/MIMO-channel pairs along which flow has been sent until an optimal solution is reached.

The complexity of the primal dual algorithm mainly lies in solving a sequence of shortest path problems. Following [73], it can be shown that by choosing δ and ϵ appropriately, the solution can get as close to the optimum solution as desired at the expense of increasing running time, as in the following remark.

Remark I: The algorithm 12 computes a $(1-\epsilon)^{-3}$ optimal solution to the scaling factor of the maximum concurrent flow problem in time polynomial in F, $\mathcal{L}$, $|V|$ and $1/\epsilon$, where F is the number of commodities, $\mathcal{L}$ is the number of constraining sets, and $|V|$ is the number of nodes.

6.5 The Distributed Algorithm for Congestion Control

The primal-dual algorithm in the previous section gives an upper bound on the achievable maximum concurrent throughput. In many practical wireless mesh networks, it is important to develop a distributed algorithm, where the computing of routes is performed in a distributed manner to approach a global optimization objective. It is therefore more practical to use an alternative objective function, i.e. to optimally distribute the end-to-end traffic into different paths and link/channel pairs thus balance the load and control the congestion of the network.

In this section, we follow [74] and derive a distributed version of the MIMO-aware routing algorithm in wireless mesh networks that can achieve fast convergence to the near-optimum solution. We assume that each commodity is associated with an agent. The multiple agents make parallel routing decisions without coordination with each other. The only accessible global information for each agent is a common clock and the utilization level of the network edges. The objective is to route $r(f)$ amount of flow for flow f from $s(f)$ to $d(f)$ for all $f \in F$, possibly along several paths, such that the maximum ratio of the total flow routed along an link/MIMO-channel pair (e, i) to its capacity is minimized. In other words, we aim to distribute the traffic evenly in the network and hopefully no (e, i) would be congested or overloaded. Recall that the *utilization* of a MIMO-channel over an edge e is previously defined as $g_i(e) = \sum_f \frac{x_i^f(e)}{c_i(e)}$, and the objective is to minimize $\max_{(e,i)} g_i(e)$. The distributed scheduling scheme is described in Algorithm 13. Throughout the algorithm, $x_i^f(e)$ and $l_i^f(e)$ are the current flow value and length of (e, i) for the agent of commodity f respectively, and x^f is the amount of flow that has been routed in the current phase for commodity f.

Algorithm 13 Distributed Routing

Initialize:
Set $x_i^f(e) \Leftarrow \epsilon c_i(e)/F$, $x^f = \epsilon c_i(e)/F$ and $l_i^f(e) = \frac{1}{c_i(e)} k^{\sum_f x_i^f(e)/(\epsilon c_i(e))}$ for each link/MIMO-channel pair (e,i) and commodity f
for $N_p = (\log k)/\epsilon^2$ phases **do**
    For each commodity f, do in *parallel*:
    while $x^f < r(f)/N_p$ **do**
        1. *subroutine:* PRE-CHECK
        2. Define the capacities $c_i^f(e) = \epsilon^2 x_i^f(e)/\log k$ if the channel pair (e,i) is not yet tagged
        3. Find the shortest path P^* from $s(f)$ to $d(f)$ under the current length function $\{l_i^f(e)\}$
        4. Compute a blocking flow $x(P^*)$ under capacities $\{c_i^f(e)\}$ along the shortest path P^*
        5. $\Delta x^f = \min\{x(P^*), r(f)/N_p - x^f\}$
        6. $x_i^f(e) \Leftarrow x_i^f(e) + \Delta x^f$ and
        $l_i^f(e) = \frac{1}{c_i(e)} k^{\sum_f x_i^f(e)/(\epsilon c_i(e))}, \forall (e,i) \in P^*$,
        $x^f \Leftarrow x^f + \Delta x^f$
    end while
    $x^f = 0, \forall f$
end for

Algorithm 14 Subroutine: PRE-CHECK

for Nodes $v^* \in \{v\}$ **do**
    if Constraint (6.15) is not satisfied **then**
        Tag $\forall (e,i) \in \{(e,i) | \forall e \in E_{out}(v^*), \forall i \in \mathcal{MC}(e)\}$
    end if
    if Constraint (6.16) is not satisfied **then**
        Tag $\forall (e,i) \in \{(e,i) | \forall e \in E_{in}(v^*) \bigcup E_{in}^I(v^*), \forall i \in \mathcal{MC}(e)\}$
    end if
    if Constraint (6.17) is not satisfied for antenna(s) $\{a\}$ of node v^* **then**
        Tag $\forall (e,i) \in \{(e,i) | \forall a^* \in \{a\}, u_{i,a^*,e} = 1\}$
    end if
end for
for Nodes v^* that is connected to any node in $\{v\}$ by any edge $e \in E \bigcup E^I$ **do**
    if Constraint (6.16) is not satisfied **then**
        Tag $\forall (e,i) \in \{(e,i) | \forall e \in E_{in}(v^*) \bigcup E_{in}^I(v^*), \forall i \in \mathcal{MC}(e)\}$
    end if
end for

Similar to the centralized algorithm, the distributed algorithm is also based on the steepest descent framework as in [73]. Let k be the number of (e, i) pairs in the network, obviously $k \sim O(|E|)$. The algorithm goes though $N_p = (\log k)/\epsilon^2$ phases. A flow of amount $r(f)/N_p$ is routed for each commodity f in each phase and a feasible solution is derived at the end. For each phase, the process is further divided into steps, as in the *while* loop in line 4-11. In each step, each commodity performs in parallel to route a fraction of its own flow. Different from the case in the centralized algorithm, we have the additional problem of how to efficiently perform concurrent routing in a distributed scenario, as concurrent attempts to route on a shortest path may lead the path to be no longer shortest and result in unpredictable oscillations. To handle this problem, a special approach is to guarantee the so-called *step-size constraint*, that the length increase of any link/channel pair (e, i) can be no larger than an ϵ fraction. Throughout the algorithm, we initially route a tiny amount of flow of all commodities on all link/MIMO-channel pairs, and later increase the flow multiplicatively. Although the initial pre-flow may not even satisfy the flow conservation constraints, as its total capacity is ϵ of the actual capacities, it has effect on the optimality only to the extent ϵ. In each step, the algorithm computes the shortest path based on the current length function, and determines the blocking flow along the path, which is the maximum amount of flow that can be routed in the path under the capacity constraint $c_i^f(e)$ for each (e, i). By computing the blocking flow, it saturates at least one edge on the path, which effectively reduces the number of steps.

In order to make the solution feasible, especially for MIMO-based networks, we revisite the constraints in equations (6.15)-(6.17). Denote $\{v\}$ as the set of nodes that are in the augmenting paths of the previous step. Recall that $g_i(e) = \sum_f \frac{x_i^f(e)}{c_i(e)}$ can be regarded as a measure of congestion of channel i over link e. Therefore, a PRE-CHECK step is added at the beginning of each step, as in algorithm 14, so that each node that is included in the augmenting paths of the last step examines if it still satisfies constraints (6.15)-(6.17). If either (6.15) or (6.16) is not satisfied, the node can no longer accept extra load, so its incident edges are set to have capacity 0 for all the possible MIMO channels. If (6.17) is not satisfied, it indicates that some of the antennas, say a, of the node is fully-loaded, so MIMO channels that have $u_{i,a,e} = 1$ are set to have capacity 0. We also check the nodes that are connected to nodes in the augmenting paths by edges in $E \cup E^I$, in order to account for the interference from the flows in the augmenting paths.

We first prove that the algorithm can achieve a $(1 + O(\epsilon))$ approximation. The analysis proceeds as in [74], but is slightly different since a link can be associated with several MIMO channels in our algorithm. Denote Φ as the potential of the network:

$$\Phi = \sum_{(e,i)} (k^{1/\epsilon})^{g_i(e)}. \tag{6.30}$$

Assume the optimum value of $\max_{(e,i)} g_i(e)$ is 1, so the optimum value of Φ satisfies $\Phi^* \geq k^{1/\epsilon}$.

Consider phase p and step t, let $l^f(t)$ and $l^f(t)'$ be the length of the shortest path at the beginning and the end of the step for commodity f. In each step, each commodity simultaneously augments its flow along certain paths. Suppose a commodity f augments flow $\Delta x_i^f(e)$ along (e, i), the total additional flow is $\Delta x_i(e) = \sum_f \Delta x_i^f(e)$. We first calculate the overall increase in Φ due to the augmentation along (e, i):

$$\Delta\Phi(e) = k^{x_i(e)/\epsilon c_i(e)}(k^{\Delta x_i(e)/\epsilon c_i(e)} - 1) \tag{6.31}$$

$$\leq k^{\frac{x_i(e)+\Delta x_i(e)}{\epsilon c_i(e)}} \cdot \frac{\Delta x_i(e) \log k}{\epsilon c_i(e)} \tag{6.32}$$

$$= \sum_f \Delta x_i^f(e) \frac{\log k}{\epsilon} \frac{k^{g_i(e)'/\epsilon}}{c_i(e)}. \tag{6.33}$$

where $g_i(e)'$ is the utilization factor after the augmentation step. The inequation (6.32) is derived from the inequality $e^a - 1 \leq ae^a$ by letting $a = \frac{\Delta x_i(e) \log k}{\epsilon c_i(e)}$. Note that $\frac{1}{c_i(e)} k^{g_i(e)'/\epsilon}$ is the length of (e, i) in the next step, denoted as $l_i^f(e)'$. Based on the above inequation, the total increase in Φ at the end of this step is:

$$\Delta\Phi \leq \sum_{(e,i)\in P^*} \sum_f \Delta x_i^f(e) \frac{\log k}{\epsilon} l_i^f(e)' \tag{6.34}$$

$$= \frac{\log k}{\epsilon} \sum_f \Delta x^f l^f(t)' \tag{6.35}$$

$$\leq \frac{\log k}{\epsilon} \sum_f \Delta x^f (1+\epsilon) l^f(t), \tag{6.36}$$

where P^* is the shortest path found on line 7 of Algorithm 13, Inequation (6.34) is derived directly from (6.33), equation (6.35) is derived from the fact that the blocking flow values on all edges of the path P^* are the same and equal to Δx^f, and $l^f(t)' = \sum_{(e,i)\in P^*} l_i^f(e)'$ is the length of the shortest path. Inequation (6.36) is from the step-size constraint which ensures that the length increase of each edge can be at most an ϵ fraction of the original length, i.e. $l_i^f(e)' \leq (1+\epsilon)l_i^f(e)$. Note that $l^f(t) \leq l^f(p)$ where $l^f(p)$ is the length of the shortest path at the end of phase p. As we route $r(f)/N_p = \epsilon^2 r(f)/\log k$ amount of flow for each commodity f in each phase, we can estimate the change in the potential during phase p as follows:

$$\Phi(p) - \Phi(p-1) \leq \frac{\log k}{\epsilon} \sum_f \sum_t \Delta x^f(t)(1+\epsilon)l^f(t) \tag{6.37}$$

$$\leq \frac{\log k}{\epsilon} \sum_f \frac{\epsilon^2 r(f)}{\log k}(1+\epsilon)l^f(p) \tag{6.38}$$

$$\leq \epsilon(1+\epsilon) \sum_f r(f) l^f(p). \tag{6.39}$$

Denote the optimum solution to the problem as $\{x_i^f(e)^*\}$. Note that $l_i^f(e)$ for all f is the same at the end of phase p, denoted as $l_i(e)|_p$. For each (e,i), since $\sum_f x_i^f(e)^* \leq c_i(e)$, we have:

$$\Phi(p) = \sum_{(e,i)} c_i(e) l_i(e)|_p \geq \sum_f \sum_{(e,i)} x_i^f(e)^* l_i(e)|_p \geq \sum_f r(f) l^f(p), \tag{6.40}$$

as $l^f(p)$ is the shortest path length from $s(f)$ to $d(f)$ and the total flow amount is $r(f)$. Combining (6.39) and (6.40), we have $\Phi(p) \leq \Phi(p-1)/(1-\epsilon(1+\epsilon))$. Initially, $g_i(e) = 0$ for all (e,i), so $\Phi(0) = k$. As $N_p = \log k/\epsilon^2$, we have $\Phi(N_p) \leq k(\frac{1}{1-\epsilon+\epsilon^2})^{\log k/\epsilon^2} \leq k^{O(1)} k^{1/\epsilon} \leq k^{O(1)}\Phi^*$. It can be proved that an $k^{O(1)}$-approximation of Φ yields a $(1+O(\epsilon))$ approximation of $\max_{(e,i)} g_i(e)$. We then arrive at the following proposition.

Proposition I: The distributed algorithm achieves an $1+O(\epsilon)$-approximation to the optimum solution.

Once the algorithm runs to the end, we can get the solution with $\{x_i^f(e)\}$, which actually includes the end-to-end routes and the corresponding link/MIMO-channel pairs for each flow commodity. From [74], we have the following proposition which shows that the convergence time of the proposed algorithm is bounded and essentially linear in the maximum path length of the network.

Proposition II: The while loop can ends in $O(L(\log^2 k \log(F/\epsilon))/\epsilon^4)$ steps, where L is the largest number of edges in a path.

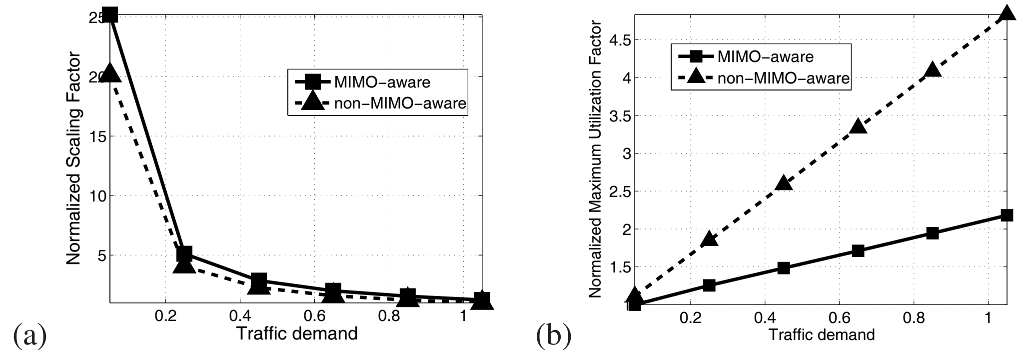


Figure 6.2: Grid topology: impact of traffic demand.

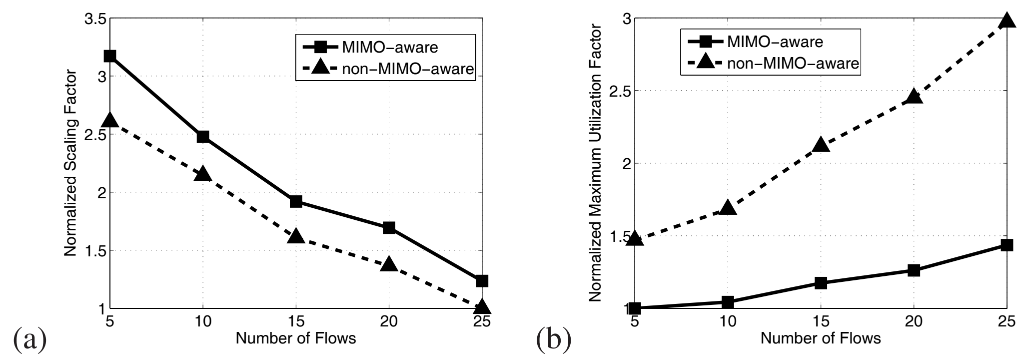


Figure 6.3: Grid topology: impact of the number of flows.

6.6 Performance evaluation

In this section, the performance of our proposed algorithms is evaluated through simulations. Our goal is to verify that by adaptively selecting a set of MIMO channels for each link subject to MIMO constraints, the MIMO-aware routing can achieve better performance under different network settings, compared with the reference non-MIMO-aware routing strategy which does not have the flexibility to switch between channels and always uses the MIMO-channel with the highest capacity for each link. The evaluated performance metrics are the objectives of the proposed algorithms, namely the scaling factor λ (which is also a measure of achievable throughput) for the centralized algorithm and the maximum utilization factor $g^* = \max_{(e,i)} g_i(e)$ for the distributed algorithm.

We generate both grid and random topologies, and run simulations with different parameter settings. In each evaluated network, a node is equipped with an array of antennas to facilitate MIMO transmission. For a link with N_t^{ant}/N_h^{ant} antennas at transmitter/receiver ends, we consider up to $\widetilde{N} = \min\{N_t^{ant}, N_h^{ant}\}$ MIMO channels are available to the link, with

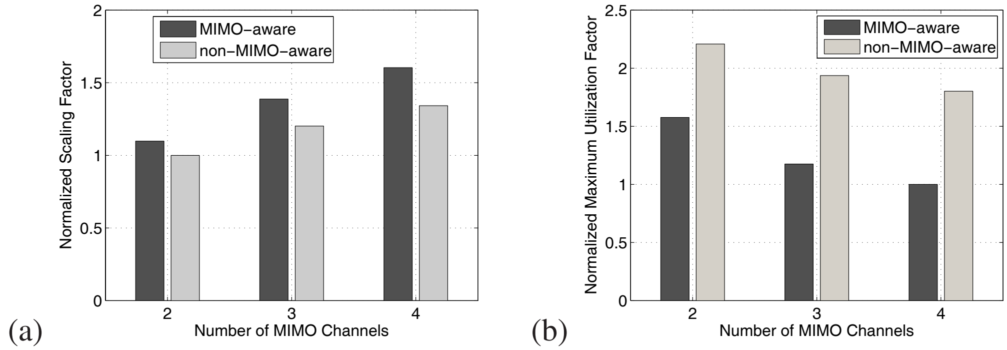


Figure 6.4: Grid topology: impact of the number of MIMO channels.

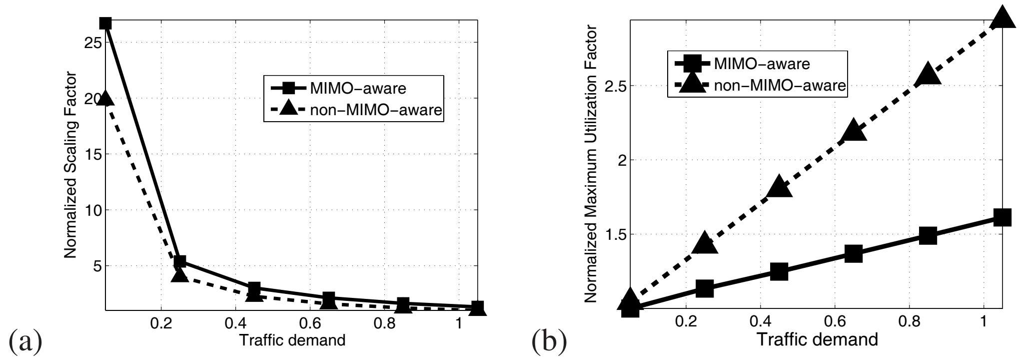


Figure 6.5: Random topology: impact of traffic demand.

each MIMO channel corresponding to one MIMO operational mode and having the degree-of-freedom value ranging from 1 to $\widetilde{N}$. The channel capacity value is estimated by averaging over a sequence of fading coefficients and set as an empirical parameter for each topology setting. We generally assume all the nodes have the same number of antennas N^{ant} to show the performance improvement by MIMO-aware routing, and we also present the performance under different values of N^{ant} and in the case that nodes have heterogeneous antenna array sizes. The traffic in the network is modeled by two parameters: the number of flows F and the demand of each flow $r(f)$. For simplicity, all flows are assumed to have the same demand, whose value is normalized to the capacity of MIMO-channel with the degree-of-freedom 1. The constants δ and ϵ used in the algorithms are set as empirically derived values. The default values of N^{ant}, F and $r(f)$ are 4, 15 and 0.5 respectively, and the network has 30 nodes if not otherwise specified. For the clarity of comparison, results are normalized with regard to the minimum value in each figure.

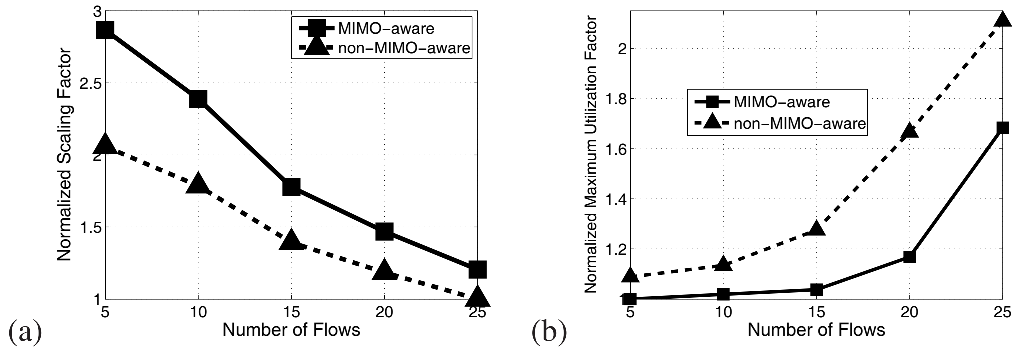


Figure 6.6: Random topology: impact of the number of flows.

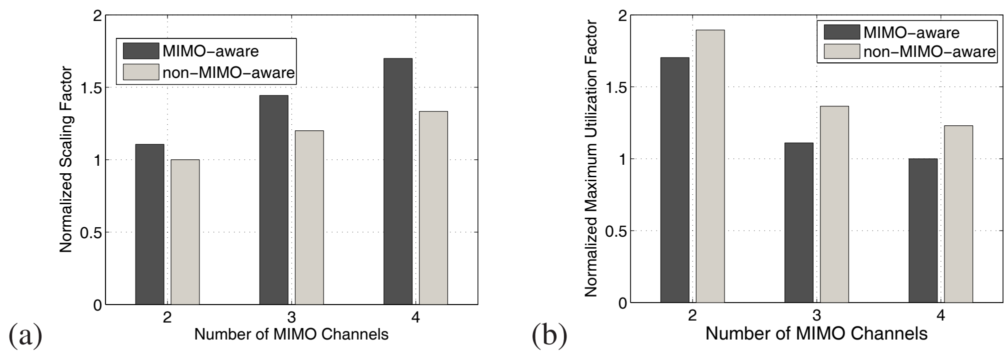


Figure 6.7: Random topology: impact of the number of MIMO channels.

6.6.1 Performance in Grid Topology

We first study the performance in a grid topology. Consider a 5×6 grid topology with 30 nodes and each node has at most 4 neighbors. Dividing the grid into four quadrants and the four nodes centered in each quadrant are set as sinks for flows. The destination of a node is the sink node that is closest to it.

As the traffic demand increases in figure 6.2, MIMO-aware routing consistently obtains a larger value of scaling factor λ (up to 25% higher) and a smaller maximum utilization factor g^* (up to 55% lower) than that for non-MIMO-aware routing. With an increased number of flows, MIMO-aware routing improves λ up to 25% and reduces g^* up to 50% as in figure 6.3. The results show that by being aware of the MIMO constraints and adaptively selecting MIMO channels, a higher amount of traffic in the network can be served with more balanced transmissions.

We can further observe from figure 6.4 that the advantage of MIMO-aware routing is even more significant with the increase of the number of MIMO-channels, as the improvement of λ

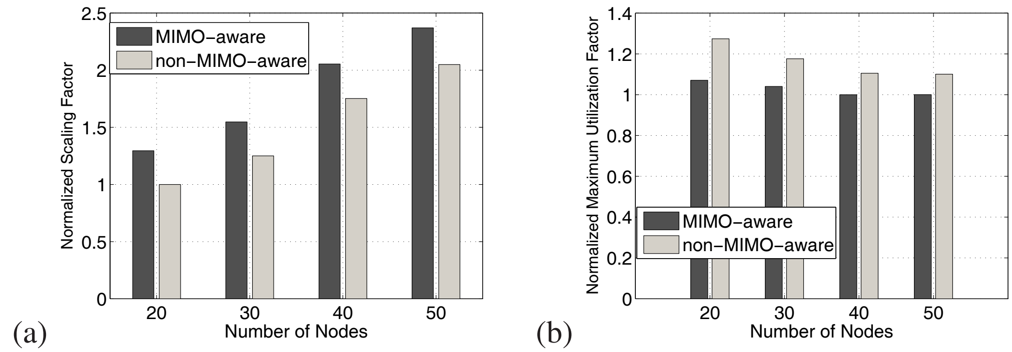


Figure 6.8: Random topology: impact of the number of nodes.

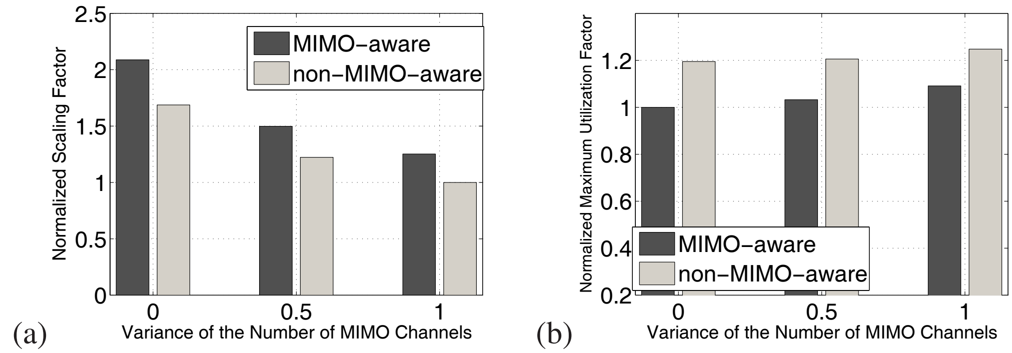


Figure 6.9: Random topology: impact of variance of the number of MIMO channels.

and g^* increases from 10% to 22% and 32% to 45% respectively, when the number of MIMO channels in each link increases from 2 to 4. With more MIMO channels, there are more options for performing more flexible routing.

6.6.2 Performance in Random Topology

Figures 6.5-6.9 show the performance of our routing algorithms in random topologies. A random topology is generated by populating nodes randomly in a 500×500 square area. The transmission range is set as 100, and each topology generated is ensured to be connected. For each flow, the source and destination are randomly selected from the set of nodes in the network. Each data point is obtained by averaging over 10 different random topologies.

In figure 6.5, a 33% increase in λ and a 45% decrease in g^* are achieved with increasing traffic demand. As the number of flows in the network increases in figure 6.6, MIMO-aware routing outperforms its counterpart by up to 45% higher λ and 32% lower g^*. The results in random topologies are consistent with that in the grid topology and demonstrate that being MIMO-aware is an effective way to leverage MIMO benefits and improve routing performance. The better performance also exists for different number of MIMO-channels, as in figure 6.7.

In figure 6.8, the number of nodes in the network is varied from 20 to 50. As more nodes in the fixed area bring more links, we can see an increase in λ and a decrease in g^*, and MIMO-aware routing remains to get better performance in all cases of node density. So far, all the cases we studied assume nodes have the same size of antenna array all over the network.

In figure 6.9, N_v^{ant} is assumed to be normally distributed with a given mean 3 and a variance from 0 to 1, to simulate the more practical case where the antenna array sizes are heterogeneous. As the variance increases, the degree of heterogeneity increases, and the performance degrades due to the decrease in the degree-of-freedom of node pairs. In all the cases we studied, MIMO-aware routing achieves improvement of about 20% for both λ and g^*, which indicates that it is more robust to heterogeneity than non-MIMO-aware strategy and further justifies its effectiveness.

To conclude, the simulation results demonstrate that MIMO-aware routing outperforms conventional routing in MIMO-based wireless mesh networks, and our proposed algorithms are effective.

6.7 Conclusions

As a promising technology to improve transmission capacity and reliability in wireless mesh networks, MIMO has been studied extensively in physical and MAC layers, but has not drawn much attention from the network layer's perspective. In this chapter, we propose the concept of MIMO-aware routing and investigate how it can further leverage the advantages brought by MIMO. We first present constraints that capture the characteristics of MIMO transmissions, and mathematically formulate the MIMO-enabled multi-source multi-destination multi-hop routing problem into a multi-commodity flow problem. We then propose a centralized algorithm to provide an approximate solution to achieve maximum concurrent flow in the network, as well as a distributed algorithm that minimizes the maximum congestion of link/MIMO-channels. The performance of our algorithms is evaluated through simulations with varied traffic demands, number of flows, number of network nodes, as well as different antenna setup and available MIMO channels. The results demonstrate that our MIMO-aware routing algorithm significantly outperforms the routing scheme that does not consider MIMO transmission features and constraints in all the test scenarios. The results in this chapter provide a basis for our future work on practical MIMO-aware routing protocol design.

Chapter 7

Deployment of MIMO Relay

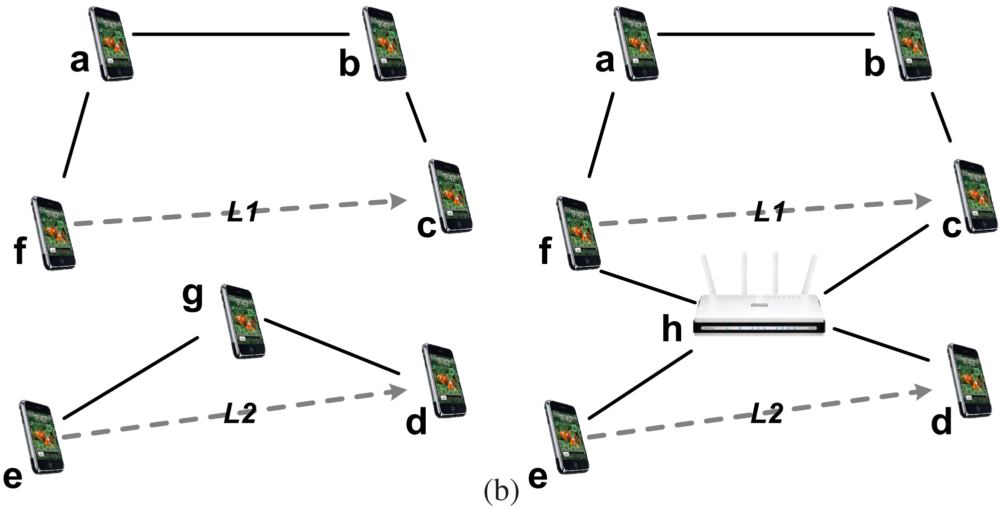
Over the past a few years, uncoordinated multi-hop wireless networks such as ad hoc networks, wireless mesh networks, and sensor networks have gone through rapid development. They are widely used in both military and civilian applications. In wireless networks, factors such as energy depletion, harsh environmental conditions, and malicious attacks may result in node failures. An active link could thus become *broken* and a network tends to lose connectivity. Moreover, different links have different quality depending on the channel conditions, and thus have different link capacities. In a wireless environment, the channel condition may experience a significant change due to reasons such as the variation of weather and the existence of obstacles. As a result, an active link may become too weak for data transmission, and thus become a *bottleneck* in an end-to-end path. The links that fall into the afore-mentioned categories are called *weak links* in general throughout this chapter.

In order to have the network perform properly, it is of significant importance to deal with the weak links to restore connectivity as well as to provide acceptable throughput and ensure transmission reliability in a severe environment. A group of weak links may be close to each other, as their channel degradation is caused by the same reason or they are within a heavy-loaded bottleneck region (i.e. near a data sink in the sensor networks) thus the nodes are more likely to run out of the energy. A practical option is to deploy a small number of more powerful relay nodes to re-establish the network connectivity while meeting traffic requirements. In order to reduce the cost of relay node deployment, we would like to place as few new nodes as possible.

In wireless sensor networks, there are some studies on placing relay nodes to provide the connectivity and/or prolong the network lifetime [75–79]. In addition to maintaining the connectivity, in general wireless networks, it is also important to guarantee the desirable rate over data transmission paths. For example, it is critical to ensure uninterrupted monitoring of a remote site through video cameras. Different from the previous work, we notice that the traffic requirements can impact the relay placement result and significantly increase the deployment challenge. Although traffic is generally considered in routing and scheduling, a deployment algorithm that takes into account the statistical traffic information could help improve the efficiency of MAC schemes and thus the overall network performance. In the literature work, Steiner-tree-based schemes have often been exploited for relay placement, which cannot satisfy the traffic requirements of the weak links. It is therefore important to introduce a new strategy that can enable a larger network capacity and higher transmission reliability especially in a severe network environment.

Multiple-input multiple-output (MIMO) technique has been proven to be able to provide high spectral efficiency and increase channel capacity substantially through multiple spatial channels without need of the additional spectrum. With multiple antennas at the transmitter and/or receiver, a MIMO system takes advantage of multiplexing to simultaneously transmit multiple data streams to increase the wireless data rate and diversity to optimally combine signals from different transmission streams to increase the transmission reliability and range. MIMO technique is considered as one of the most promising techniques for future wireless networks, and has been adopted by 802.11n, WiMAX, and LTE. To meet the high data rate requirements, more and more wireless devices are equipped with multiple antennas. As the cost of a MIMO node is usually higher than a regular node, it may not be economically efficient to have all nodes in a network equipped with multiple antennas. However, it is beneficial to deploy a small set of MIMO nodes to assist weak links, which can significantly improve the overall network performance in a severe environment.

Specifically, MIMO relays can address issues that cannot be handled by simply adding more conventional single-antenna relay nodes. As an example, in figure 7.1 (a), the links L_1 from node f to node c and L_2 from node e to node d are detected to be weak links. There are two conventional ways to exploit relay nodes. One is using multi-hop intermediate nodes as relays, so the transmission for L_1 could be redirected via the route $f \rightarrow a \rightarrow b \rightarrow c$. This strategy raises the traffic load over the relay routes as well as increases the transmission latency, and it cannot be guaranteed that there are available neighboring nodes to serve as multi-hop relays. This method may temporarily facilitate transmissions of a weak link, but not if the inferiority of the link is permanent or lasts for a long period of time. Moreover, the consecutive hops cannot transmit simultaneously and the total throughput is reduced as a



(a) (b)

Figure 7.1: Assisting weak links with relays: (a) single antenna and multihop relays; (b) MIMO relay.

result. As the other way to exploit relay, a regular node with single antenna can be deployed near a weak link, for example, node g could assist the weak link L_2. Even if node g is close to another weak link such as L_1, it cannot simultaneously assist it as g can only communicate with one node at a time. Due to this transmission limitation, deployment of multiple single-antenna relays cannot meet the total traffic requirement of weak links as the relays cannot transmit concurrently in the same neighborhood. In such cases, MIMO relays can better assist weak links and provide a higher performance gain without extra spectrum cost. As in figure 7.1 (b), when placed close to nodes c, d, e and f, a MIMO node h with a multi-antenna array could simultaneously assist links L_1 and L_2 , i.e., it could receive concurrently from f and e and transmit concurrently to c and d. Thanks to spatial diversity and multiplexing, these concurrent transmissions can enjoy higher rates than single-antenna transmissions, and h can even reach a larger area to assist more weak links by using spatial diversity to extend its communication range. It is therefore critical to develop techniques that can efficiently deploy MIMO relays and harvest their multiplexing and diversity gains for a higher network performance.

A MIMO node could provide various transmission rates and ranges thus meet different transmission needs with an appropriate configuration of the antennas. This flexibility can bring in a significant advantage. However, it also makes the deployment of MIMO nodes much more challenging than conventional relay placement. In order to fully take advantage of MIMO features, it is necessary to identify the specific constraints of MIMO transmissions, and design appropriate transmission strategy to flexibly determine the specific transmission mode of a MIMO relay node when facilitating a specific weak link according to the network topology, channel conditions and traffic requirements. As the relay deployment is made for a higher network performance over a relatively longer period, the provisioning is based on statistical channel conditions and traffic requirements over a period of time. If the relay nodes are mobile (i.e., robots) or mobile agents are available to move the relay nodes, the relay positions can be adjusted over time.

In this chapter, we aim to design MIMO relay node placement algorithms to facilitate transmissions over weak links in a multi-hop wireless network so that higher performance can be achieved with the minimum deployment cost and the traffic requirements of the weak links are satisfied. Our work is different from the previous work in that:

1. It is the first work that considers deployment of MIMO nodes in wireless multi-hop networks. Different from conventional deployment problems, MIMO nodes are with different transmission ranges and rates when different transmission strategies are configured.

2. The deployment problem not only guarantees the full coverage of the weak links that require assistance for connectivity, but also opts to minimize the number of MIMO nodes while considering the traffic demand of flows in the deployment to provide performance provisioning.

3. We perform cross-layer optimization to flexibly select MIMO transmission strategies for each of the weak links facilitated depending on the network topology, statistical channel conditions and traffic demands.

The rest of this chapter is organized as follows. We discuss related work in Section 7.1. We introduce background and describe the system model in Section 7.2, and formulate the problem in Section 7.3. The centralized and distributed deployment scheme are proposed in Section 7.4 and Section 7.5, respectively. Simulation results are presented in Section 7.6. The chapter is concluded in Section 7.7.

7.1 Related Work

Relay placement in wireless sensor networks (WSN) has been studied over the past several years, where the focuses have been on improving energy efficiency [77] or minimizing the number of relays to guarantee network connectivity [75, 76, 78, 79], assuming a homogeneous transmission range for both relay nodes and sensor nodes. The limited studies [75, 76] that do not assume uniform-range often consider specific WSN architectures, i.e. tiered and/or with base-station, instead of a general multi-hop wireless network we study here. More recently, the authors in [78, 79] further extended the problem of relay deployment to heterogeneous wireless sensor networks, with the assumption that the candidate relay positions are known. The finding of candidate positions is a challenging problem itself. The objectives of these existing studies are constrained to connectivity, with little consideration of traffic requirements or throughput optimization. In an environment with severe channel conditions such as scattering and fading or with a number of weak links close by, placing simple relay nodes cannot ensure the transmission quality. In this work, we consider the deployment of MIMO relay nodes with higher capacity and various operational modes to address all these issues in an integrated and coherent manner.

Besides relay placement in WSN, the placement of Internet transit access points was studied in [80] to provide Internet connectivity in multi-hop wireless networks. Gateway placement for throughput optimization in multi-hop wireless mesh networks was addressed in [81], and joint mobile backbone node placement and regular node assignment was proposed in [82]. The solutions proposed in the above work cannot be applied to MIMO relay deployment, which is more challenging as a MIMO node can be configured flexibly with different transmission ranges and rates.

The application of MIMO technique in wireless mesh and ad hoc networks has gained increasing attention in recent years. Many efforts have been made in developing efficient MAC [12, 16, 19, 29] and routing [26, 67] schemes to enable MIMO communications in ad hoc networks. However, to the best of our knowledge, there is no study on deployment of MIMO relay nodes in meshed wireless networks. The specific features of MIMO technique promise great potential to improve network performance, but also bring in new challenges in both identifying the constraints and designing proper MIMO relay placement algorithms.

7.2 Model Description

Deployment algorithms can be classified into two types, global and local. Based on the knowledge of complete network topology and all the flow traffic, a global scheme may achieve a better performance, but will incur a much higher cost for collecting network information and potential global network reconfiguration (i.e. change of paths). Alternatively, the *local* deployment identifies the set of broken/bottleneck links, and uses MIMO nodes to cover these links

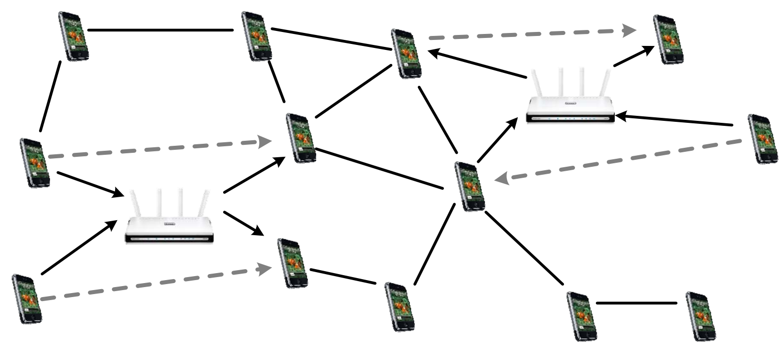


Figure 7.2: Illustration of MIMO relay placement.

so that the broken links can be bridged and the bottleneck capacities of flows can be improved. In this chapter, we focus on the local deployment for better distributed implementation.

In this chapter, we consider a multi-hop wireless network with two types of nodes, regular nodes (RN) with one antenna each, and MIMO relay nodes (MR) each equipped with an array of antennas. The locations of the regular nodes are fixed and pre-determined. We consider the problem of deploying a set of MRs to facilitate the transmissions of RNs, as illustrated in Figure 7.2. In addition to providing much more powerful transmission capability, MIMO nodes also have the flexibility of switching between different strategies exploiting two types of gains of MIMO, *spatial multiplexing* and *spatial diversity*. In a rich scattering environment where the transmission channels for different streams are differentiable and independent, i.e., *orthogonal*, multiple independent data streams can be transmitted between a transmission node pair through spatial multiplexing. Alternatively, different types of diversity techniques can be exploited to improve the transmission reliability or range.

The term *degree of freedom* (DoF) is widely used to describe the dimension of the space over which communication can take place. The DoF of a MIMO link is limited by the antenna array sizes of the end nodes as well as the channel conditions, and a MIMO operational mode takes up a specific number of DoFs. Generally speaking, the value of DoF for each node is close to its antenna array size in a rich scattering environment. A group of RNs can be assigned to the same MR to form a virtual MIMO node, taking advantage of multi-user MIMO [32] and spatial multiplexing to provide multiple access. Therefore, each MR can *simultaneously* assist several weak links, which not only effectively reduces the number of MRs needed but also provides a higher transmission capacity to relieve a bottleneck area. We call this strategy *MAS* for multiple access. Using this mode, an MR simultaneously receives data uplink from multiple RNs through cooperative spatial multiplexing and transmit to several RNs downlink using zero-forcing multi-user beamforming. The available DoF of the MR is thus shared by several

RNs. In addition, multiple antennas can be used to improve the link capacity when an MR receives from or transmits to one RN, taking advantage of the power gain. Alternatively, we can use MIMO for an extended transmission range to achieve a guaranteed reliability, denoted as mode *RANGE*, which exploits a diversity mode, i.e. uplink receiving with spatial diversity or maximal ratio combining and downlink transmission using space-time coding/selection diversity/beamforming. As the receiving signal to noise ratio (SNR) is a function of the diversity order, the levels of DoFs and MR uses for transmission can be adjusted to communicate with RNs at different distances.

The *MAS* mode and the *RANGE* mode can be combined by adjusting the DoF used for each to form different transmission strategies, each with different transmission range and link capacity. For instance, an MR with 4 DoFs can assist 2 RNs simultaneously if each of them takes 2 DoFs for *RANGE*. The strategies of MRs can be saved in a table, so that for each combination of modes and the number of DoFs to use, the corresponding rate and range can be determined easily based on the statistics of the channel and traffic conditions.

7.3 Problem Formulation

In this section, we first introduce the notations, then identify the specific constraints for deployment, and finally mathematically formulate the problem.

7.3.1 Notations

The set of weak links in the network is denoted as $\mathcal{L}$ and $|\mathcal{L}| = L$. For a weak link $l_j \in \mathcal{L}$, its corresponding sender and receiver nodes are denoted as $t(l_j)$ and $h(l_j)$ respectively. We consider two types of nodes in our problem, regular nodes denoted as a set $\mathcal{RN}$ are the end nodes of links in $\mathcal{L}$, and MIMO nodes denoted as a set $\mathcal{MR}$ serve as relays for the weak links. Obviously, $\mathcal{RN} = \{t(l_j), h(l_j) | l_j \in \mathcal{L}\}$. For a MIMO node $m_i \in \mathcal{MR}$, its available degree-of-freedom value is denoted as α_i and its corresponding position is denoted as p_i. A MIMO node that covers l_j is denoted as $m(l_j)$, and needs to receive data from $t(l_j)$ and forward the data to $h(l_j)$. A MIMO node m_i can choose from multiple strategies, denoted as a set $\mathcal{A}_i$, depending on the specific requirement of the weak link and the environmental condition. The k-th strategy in $\mathcal{A}_i$, i.e. $\mathcal{A}_i(k)$, is a pair $(\mathcal{A}_i^U(k), \mathcal{A}_i^D(k))$ for uplink and downlink transmissions respectively, with the corresponding receiving and transmission ranges (which could be an equivalent range based on the target received signal strength) denoted as $R_{i,k}^U$ and $R_{i,k}^D$ and the average capacity represented as $C_{i,k}^U$ and $C_{i,k}^D$ respectively. Correspondingly, it costs $O_{i,k}^U$ and $O_{i,k}^D$ DoFs for uplink and downlink transmissions respectively. In order to support the traffic demands of weak links, we denote the aggregate traffic requirement of a weak link l_j in a unit transmission period as F_j.

7.3.2 Deployment Constraints

The deployment will meet the basic traffic demands of weak links and resource constraints of MIMO relays. The sender and receiver of a weak link can be regarded as a local source and destination with the flow requirement of the aggregate link traffic. In this section, we first study the flow constraints during the packet scheduling in each time slot at the link-layer with MIMO relays, and then translate the rate variables into the deployment constraints for performance-guarantee. Although traffic of flows are not constant, and the channel conditions and thus link rate could vary, their long-term statistics are relatively stable and can serve for the resource provisioning purpose. Our deployment algorithm does not depend on specific MAC, and the per-slot traffic scheduling between weak links and MIMO relay nodes is out of our scope. We use $x_{i,j,k}^U$ to denote the flow on the uplink transmission from $t(l_j)$ to MIMO node m_i using strategy $\mathcal{A}_i^U(k)$, and $x_{i,j,k}^D$ to denote the flow on the downlink transmission from m_i to $h(l_j)$ using strategy $\mathcal{A}_i^D(k)$.

A necessary condition to meet the traffic requirement is the existence of the link flow $\{x_{i,j,k}^U, x_{i,j,k}^D\}$ that satisfies the following *flow conservation constraints*:

$$\sum_k x_{i,j,k}^U = \sum_k x_{i,j,k}^D, \forall l_j \in \mathcal{L}, m_i = m(l_j); \tag{7.1}$$

$$\sum_k x_{i,j,k}^U = F_j, \forall l_j \in \mathcal{L}, m_i = m(l_j); \tag{7.2}$$

$$x_{i,j,k}^U = x_{i,j,k}^D = 0, \forall l_j \in \mathcal{L}, m_i \neq m(l_j); \tag{7.3}$$

where (7.1) assures that the incoming traffic equals the outgoing traffic for the link l_j which is associated with a MIMO node m_i; and (7.2) guarantees that the aggregate traffic requirement of each weak link can be satisfied with the help of MIMO relay nodes. Denote the weak links that are within the range of m_i as $\mathcal{L}_i$, which may include the weak links that are associated with m_i, i.e. $\forall j\, s.t. m(l_j) = m_i$, as well as other weak links whose transmissions interfere with the transmission or reception of the node m_i. As the *flow capacity constraint*, the total transmissions on a MIMO channel should not exceed its capacity:

$$\sum_{j:l_j\in\mathcal{L}_i} x^U_{i,j,k} \le C^U_{i,k}, \forall \mathcal{A}_i(k) \in \mathcal{A}_i, m_i \in \mathcal{MR}; \tag{7.4}$$

$$\sum_{j:l_j\in\mathcal{L}_i} x^D_{i,j,k} \le C^D_{i,k}, \forall \mathcal{A}_i(k) \in \mathcal{A}_i, m_i \in \mathcal{MR}. \tag{7.5}$$

While constraints (7.1)-(7.5) are conventional for flow problems, the use of MIMO technique imposes new constraints. Even though the DoF constraints are generally formulated at the MAC layer, they actually have a significant impact on resource provisioning and the deployment strategy of MIMO nodes. Let $I^U_{i,j,k,\tau}/I^D_{i,j,k,\tau}$ be the indicator variable that has value 1 if and only if $m(l_j) = m_i$ and the link from $t(l_j)$ to m_i/from m_i to $h(l_j)$ is *active* using MIMO strategy $\mathcal{A}^U_i(k)$/$\mathcal{A}^D_i(k)$ in the time slot τ. During the deployment phase, there is no knowledge on the group of nodes scheduled to transmit together, so the actual interference information is unknown. We assume the transmissions are over orthogonal channels and constrain the number of concurrent transmissions with the available number of DoFs, while throughput reduction due to interference beyond the transmission range or due to uncanceled interference as a result of physical decoder limit can be mitigated with a certain level of the over-provisioning of resources. To satisfy the DoF constraint at the transmitter side, the DoF used by a node m_i for all its active downlink transmissions must be no larger than its available DoF, i.e. α_i, in each time slot τ:

$$\sum_{j:l_j\in\mathcal{L}_i} \sum_k I^D_{i,j,k,\tau} O^D_{i,k} \le \alpha_i, \forall m_i \in \mathcal{MR}. \tag{7.6}$$

Similarly, corresponding to the receiver's DoF constraint, the DoF number required to decode the receiving transmissions should not exceed the receiving capability of the node:

$$\sum_{j:l_j\in\mathcal{L}_i} \sum_k I^U_{i,j,k,\tau} O^U_{i,k} \le \alpha_i, \forall m_i \in \mathcal{MR}. \tag{7.7}$$

For a time period with T time slots, define $g^U_{i,j,k} = \frac{x^U_{i,j,k}}{C^U_{i,k}}$ and $g^D_{i,j,k} = \frac{x^D_{i,j,k}}{C^D_{i,k}}$ as the *utilization* of uplink and downlink channels of MIMO node m_i using strategy $\mathcal{A}^U_i(k)$/$\mathcal{A}^D_i(k)$ respectively. Note that we also have $g^U_{i,j,k} = \frac{1}{T}\sum_{1\le\tau\le T} I^U_{i,j,k,\tau}$ and $g^D_{i,j,k} = \frac{1}{T}\sum_{1\le\tau\le T} I^D_{i,j,k,\tau}$ for all l_j. Adding equations (7.6) and (7.7) over all the T time slots and dividing by T results in the constraints:

$$\sum_{j:l_j \in \mathcal{L}_i} \sum_k \frac{x^U_{i,j,k}}{C^U_{i,k}} O^U_{i,k} \leq \alpha_i, \forall m_i \in \mathcal{MR}; \tag{7.8}$$

$$\sum_{j:l_j \in \mathcal{L}_i} \sum_k \frac{x^D_{i,j,k}}{C^D_{i,k}} O^D_{i,k} \leq \alpha_i, \forall m_i \in \mathcal{MR}. \tag{7.9}$$

7.3.3 The Deployment Problem

Based on the system model and the notations, the problem is then formulated as follows.

Objective: Find a set $\mathcal{MR}$ of MIMO relay nodes that achieves the minimum cardinality $\min |\mathcal{MR}|$ and determine the position p_i for each MIMO node $m_i \in \mathcal{MR}$ and the values of all flows $\{x^U_{i,j,k}, x^D_{i,j,k}\}$ associated with weak links, subject to the following ***constraints***:

(a) For each weak link $l_j \in \mathcal{L}$, a MIMO node $m(l_j) = m_i$ is assigned to assist it with the strategy set A_i. Specifically, $t(l_j)$ and $h(l_j)$ are within the range of $R^U_{i,k}$ and $R^D_{i,k}$ with regard to m_i, when uplink strategy $\mathcal{A}^U_i(k)$ and downlink strategy $\mathcal{A}^D_i(k)$ are used, where $(\mathcal{A}^U_i(k), \mathcal{A}^D_i(k)) \in \mathcal{A}_i$. Equations (7.1)-(7.3) are satisfied for each l_j.

(b) For each deployed MIMO node $m_i \in \mathcal{MR}$, equations (7.4), (7.5), (7.8) and (7.9) are satisfied, so the deployment of MIMO nodes can provide performance provisioning.

The above formulation is relatively descriptive. If the candidate positions for MIMO nodes are known and denoted as a set $\mathcal{P}$, the problem can be formulated more clearly as a programming problem. Let y_i be the indicator that equals 1 if a MIMO node is placed at the position $p_i \in \mathcal{P}$, otherwise $y_i = 0$. Let z_{ijk} be the indicator that equals 1 if the weak link l_j is assigned to a MIMO node at p_i using the strategy $\mathcal{A}^U_i(k)$/$\mathcal{A}^D_i(k)$ for uplink/downlink transmission respectively; otherwise, $z_{ijk} = 0$. The problem is then reformulated as

$$\min \sum_{i=1}^{|\mathcal{P}|} y_i, \tag{7.10}$$

$$\text{subject to:} \quad \sum_{i=1}^{|\mathcal{P}|}\sum_{k} z_{ijk} = 1, \forall j = 1, \dots, |\mathcal{L}|; \tag{7.11}$$

$$\sum_{j=1}^{|\mathcal{L}|}\sum_{k} z_{ijk}\frac{F_j}{C_{i,k}^U}O_{i,k}^U \leq \alpha_i, \forall i = 1, \dots, |\mathcal{P}|; \tag{7.12}$$

$$\sum_{j=1}^{|\mathcal{L}|}\sum_{k} z_{ijk}\frac{F_j}{C_{i,k}^D}O_{i,k}^D \leq \alpha_i, \forall i = 1, \dots, |\mathcal{P}|; \tag{7.13}$$

$$\begin{gathered} \sum_{j=1}^{|\mathcal{L}|} z_{ijk}F_j \leq C_{i,k}^U, \sum_{j=1}^{|\mathcal{L}|} z_{ijk}F_j \leq C_{i,k}^D, \\ z_{ijk} \leq y_i, y_i = \{0,1\}, z_{ijk} = \{0,1\}, \\ i = 1, \dots, |\mathcal{P}|, j = 1, \dots, |\mathcal{L}|, k = 1, \dots, |\mathcal{A}_i|; \end{gathered} \tag{7.14}$$

where (7.11) assigns one MIMO node to a weak link obeying constraint (a) in the above problem formulation, (7.12)(7.13) reflect constraint (b) and are together called *feasibility constraints*, and (7.14) ensures the flow constraints as well as the correct relation between parameter y_i and z_{ijk}. Although candidate positions are assumed to be known in many literature work, they are actually quite challenging to be found.

NP-hardness: We now briefly analyze the complexity of the deployment problem with candidate positions. Consider a simplified version of our problem where each MR node works at only one mode and has unlimited capacity. It can be shown that the NP-complete Vertex Cover problem in planar graph with maximum degree 3 is polynomial-time reducible to our simplified problem. Following the first two steps of transformation as in [83], an arbitrary planar graph G_a of maximum degree 3 can be transformed in polynomial-time to the MIMO relay deployment problem, with $u_{i,l}$ in the step 2 as the candidate positions. It is then easy to verify that G_a has a vertex cover set of size N in a planar graph with degree at most 3 if and only if the deployment problem has a solution of size $N+\frac{1}{2}\sum_{e_i\in E(G_a)}(|e_i|-1)$, where $E(G_a)$ is the edge set of graph G_a. Therefore, the MIMO relay deployment problem with candidate positions is NP-hard.

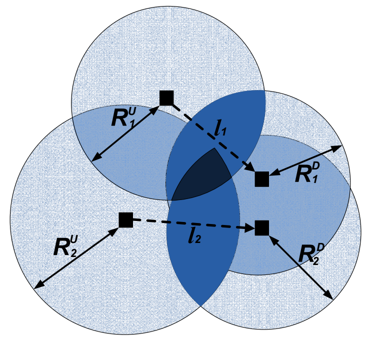


Figure 7.3: Illustration of the optimum position.

7.4 Centralized Deployment

Our problem consists of two coupled subproblems: 1) where to exactly place the MRs, and 2) the assignment of weak links to the MRs with specific transmission strategies. As seen in Section 7.3, the optimum position of an MR depends on the links it is assigned to cover and the transmission ranges (which depend on the strategies) it uses to cover them. On the other hand, the position of an MR impacts the number of links it can cover and the transmission strategy it needs to use. Different from the literature work, e.g. [78, 79], where the candidate positions are assumed to be known, the main difficulty in finding the optimum position is that there are an infinite number of potential locations for the MIMO nodes.

In this section, we jointly consider the two subproblems. We first discuss three possible ways of narrowing down the search space of the optimum MR positions. Then based on the candidate positions, we provide two schemes to determine the association of weak links with MRs and the transmission strategies: a polynomial time approximation scheme (PTAS) and an iterative centralized algorithm with lower complexity.

7.4.1 The Candidate Positions of MIMO Nodes

We first discuss several ways of finding the candidate positions of MIMO nodes, and then compare their impact on the performance later in Section 7.6.

Optimum positions

The position of a MIMO node that can assist a weak link l_{j_1} with strategy k_1 should be inside the intersection area (which is a lune shape) of the circles centered at $t(l_{j_1})$ and $h(l_{j_1})$ with radiuses $R^U_{k_1}$ and $R^D_{k_1}$ respectively. Intuitively, a position at the center of the intersection area of the lune shapes formed by the weak links could be considered as an optimum position of a MIMO node to cover these weak links (Figure 7.3). However, the complexity is high to find the intersection of all possible lune-shapes and determine the optimum position.

1-center positions

Considering the coverage of end nodes instead of links, the candidate positions can be regarded as a subset of the candidate positions of 1-center problem. According to [82], the following fact holds for the 1-center problem.

Fact 1: The unique 1-center location of a set of nodes V, denoted as $1C(V)$, is defined by:

1. A pair of nodes $a, b \in V$. If this is the case, then $1C(V)$ is located at the midpoint of a, b.

2. A triplet of nodes $a, b, c \in V$ that form an acute triangle. If this is the case, then $1C(V)$ is located at the circumcenter of a, b, c.

3. A single node $a \in V$.

It is therefore reasonable to consider the candidate positions of the MIMO nodes based on the above facts.

Simplified positions

Although using the 1-center positions can significantly reduce the size of the candidate position set, searching over it may still be computationally complicated. In some circumstances, it is possible to tradeoff the accuracy with simplicity, so we can use even more simplified positions.

Vicinity Criterion: Let $\overline{(a, b)}$ be the distance between node a and b in the network. For a link l_{j1} with the sender/receiver $t(l_{j1})$/$h(l_{j1})$ and a link l_{j2} with the sender/receiver $t(l_{j2})$/$h(l_{j2})$, if the maximum of $\{\overline{(t(l_{j1}), t(l_{j2}))}$, $\overline{(t(l_{j1}), h(l_{j2}))}$, $\overline{(h(l_{j1}), t(l_{j2}))}$, and $\overline{(h(l_{j1}), h(l_{j2}))}\}$ is no larger than r^*, a given parameter of the transmission range, links l_{j1} and l_{j2} are considered to be in each other's vicinity under r^*.

The vicinity criterion is based on the simple fact that the distance between two links is bounded by the pairwise distance of the four endpoints. If r^* is the transmission range of an MR node, then two links that are in each other's vicinity can be guaranteed to be covered by one MR that is placed at the mid-point of any of them. Note that the distance here can be equivalent distance depending on the received signal level instead of physical distance between nodes.

7.4.2 Approximation solution

As the deployment problem is NP-hard, it is important to develop an algorithm that can provide some performance bound. In this section, we propose a polynomial time approximate solution using shifting strategy that is specifically tailored for our problem, and prove the approximate ratio through the description of the algorithm. Different from the existing work [83, 84], the problem becomes much harder given the performance provision requirements and the flexibility of MIMO modes.

We first simplify the problem formulated in (7.10)-(7.14) as follows. Assume that the candidate positions of MIMO nodes are known and denoted as a set $\mathcal{P}$, and the set of available operational strategies is the same for all MIMO nodes. Denote the set for all available values of the ranges as $\mathcal{R}$. If a MIMO node placed at position $p_i \in \mathcal{P}$, denoted as node m_i, uses the strategy k, it has the transmission range $R_k \in \mathcal{R}$ and capacity $C_k \in \mathcal{C}$. A MIMO node can activate several strategies of operation, as long as constraints (7.12) and (7.13) are satisfied.

We can then have the following theorem for our problem.

Theorem There exists a polynomial-time approximation scheme for the deployment problem with above simplification.

Proof: To approximate the deployment problem, we consider the shifting strategy [83, 84]. To facilitate finding the solution, we first introduce the following graph representation, and then present the proof along with the construction of the solution step by step.

Graph Representation. Set the maximum range $R^M = \max\{R_k\}$ of all the available MIMO modes as the distance unit in the network. Let $\mathscr{R}$ denote the smallest rectangular region that can hold the network graph, with width $\mathscr{R}_w$ and length $\mathscr{R}_l$ normalized to R^M. The candidate position p_i for an MR to cover a weak link l_j using the strategy $A_i(k)$ lies in the lune region $S(j, k)$ that is the intersection of the disks centered at $t(l_j)$ and $h(l_j)$ respectively, both with radius R_k. A weak link may be covered by an MR using different transmission ranges. For the set of all the available ranges $\mathcal{R}$, $|\mathcal{R}|$ layers of planes can be constructed (Figure 7.4), where the lune shapes in each layer is formed with a specific $R_k \in \mathcal{R}$. Therefore, the deployment problem can be considered as selecting the minimum size of position set from $\mathcal{P}$ so that each weak link is covered at least once by positions from the $|\mathcal{R}|$ layers and the feasibility constraints in (7.8) and (7.9) of MIMO nodes placed at the selected positions can

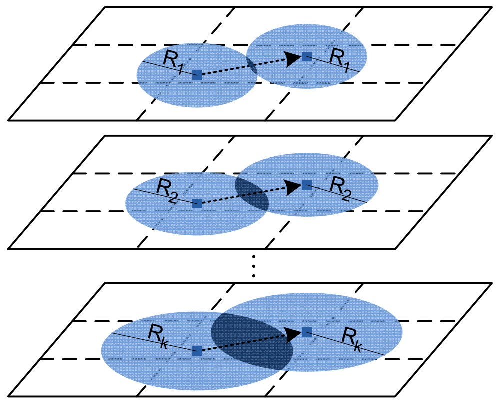


Figure 7.4: The layered graph construction.

be satisfied.

Shifting Strategy. For a positive integer $m > 0$, consider any even integers a, b satisfying $2 \leq a, b \leq m$, with the value of m given at the end of the proof. We partition the region $\mathscr{R}$ into $m \times m$ squares by horizontal lines at $a + k_1 \cdot m$ and vertical lines at $b + k_2 \cdot m$, where k_1 and k_2 are selected as all the possible non-negative integers such that $a + k_1 \cdot m \leq \mathscr{R}_w$ and $b + k_2 \cdot m \leq \mathscr{R}_l$. By exhaustively varying the possible values of a and b, the division of squares is shifted over the plane. Let $S_{a,b}$ denote the set of squares for a fixed pair a, b. A lune is said to belong to a square if and only if its geometric center lies in the square. For a square $s \in S_{a,b}$, let $L(s)$ denote the set of lunes belonging to the square s. The radius of a lune is at most 1 unit and the range is the same for both uplink and downlink. For different layers, the squares actually associate with a set of lunes formed by the same end points but the lunes have different sizes corresponding to different strategies of the MR nodes, as shown in Figure 7.4.

Local Optimum. We assume the node density of the network has an upper bound, so that for each $m \times m$ square in $S_{a,b}$, the number of available candidate positions is bounded by $O(m^2)$. Therefore, the optimum solution within a square can be found through exhaustive search over the square area on all the $|\mathcal{R}|$ layers in polynomial time using complete enumeration of all possible positions for the given constant m and $|\mathcal{R}|$, while guaranteeing that each MIMO node placed at a selected position satisfies the feasibility constraints (7.8) and (7.9). The union of positions selected from all squares in $S_{a,b}$ gives a candidate solution for a given pair of a, b, and the candidate solution for each pair of a, b can be obtained through the shifting strategy. Among all the candidate solutions, the one with the minimum cardinality is considered as our solution, denoted as a set H.

Approximation. Now we analyze the approximation ratio. Let H_o be the optimal solution, and H be the solution obtained by the shifting strategy. For a given pair of a, b, let $H_o(a,*)$, $H_o(*,b)$, $H_o(a,b)$ respectively be the vertices set in H_o that lie in lunes intersecting horizontal lines, vertical lines, and both horizontal and vertical lines. Let $H_o(s)$ be vertices in $H_o \cap L(s)$, $OPT(s)$ be the optimum MIMO nodes set to cover the weak links represented by the lunes in $L(s)$. We have

$$|H| \leq |\bigcup_{s\in S_{a,b}} OPT(s)| \leq \sum_{s\in S_{a,b}} |OPT(s)| \leq \sum_{s\in S_{a,b}} |H_o(s)|. \tag{7.15}$$

Note that based on the definition of the distance unit, positions in lunes that cross an active division line can be used in at most 4 squares. Therefore,

$$\sum_{s\in S_{a,b}} |H_o(s)| \leq |H_o(a,*)| + |H_o(*,b)| + |H_o(a,b)| + |H_o|. \tag{7.16}$$

As the shifting step is set to be two units, all lunes that cross one horizontal(or vertical) line do not intersect with lunes that cross another horizontal(or vertical) line, regardless of their radii. Hence, a position is counted at most once in $H_o(a,*)$ with a changing as well as in $H_o(*,b)$ with b changing. Consequently, we obtain the following inequalities, $\sum_{2\leq a\leq m} |H_o(a,*)| \leq |H_o|$, $\sum_{2\leq b\leq m} |H_o(*,b)| \leq |H_o|$. We can find a pair of (a,b), such that

$$|H_o(a,*)| \leq \frac{2}{m}|H_o|, |H_o(*,b)| \leq \frac{2}{m}|H_o|. \tag{7.17}$$

Therefore,

$$\begin{aligned} |H_o(a,b)| &= |H_o(a,*) \bigcap H_o(*,b)| \\ &\leq \min\{|H_o(a,*)|, |H_o(*,b)|\} \leq \frac{2}{m}|H_o|. \end{aligned} \tag{7.18}$$

Combing inequalities (7.15)-(7.18), we finally have

$$|H| \leq (\frac{2}{m} + \frac{2}{m} + \frac{2}{m} + 1)|H_o| = (1 + \frac{6}{m})|H_o|. \tag{7.19}$$

As a result, given any $\epsilon > 0$, let $m > 0$ be the smallest even integer such that $(6/m) \leq \epsilon$ and find a solution H using the shifting strategy with a specific value m, the solution H thus can achieve $1 + \epsilon$ approximation ratio. □

To search for the local optimum for each pair of a, b, $O(|\mathcal{R}|m^2)$ candidate positions need to be examined and the number of shifts is bounded by $m^2/4$. Considering $(6/m) \leq \epsilon$, the time complexity of the shifting algorithm is bounded by $O(|\mathcal{R}|(1/\epsilon)^4)$.

7.4.3 A Centralized Algorithm

Although an approximate solution is proposed in the previous subsection, it is still desirable to have some lower complexity algorithm. A greedy centralized algorithm is proposed in Algorithm 15. It is inspired by the fact that the greedy algorithm gives the best approximate solution for the general set cover problem.

Two concepts are used here, *covering degree* of a MIMO position i is used to describe the number of weak links that a MIMO node at position i can cover, and *covered degree* of a weak link l_j is used to describe how many MIMO positions can cover l_j. Sort the available transmission ranges $R_k \in \mathcal{R}$ in the ascending order. k_i is the index of the strategy with a corresponding range for a MIMO node at position i.

Among the candidate positions in $\mathcal{P}$, the positions that have been selected are included in the set $\mathcal{P}^*$. The algorithm initially uses the MIMO strategy with the shortest range. The candidate positions in the set $\mathcal{P}_0$ are considered in each run, and the positions that are excluded from being selected in the current run are in the set $\tilde{\mathcal{P}}$. A *communication graph* is constructed as in line 6, where an edge is formed between a candidate position and the end node of a weak link if their distance is less than or equal to the preset range. Note that before a relay is deployed, the channel condition between the end nodes of a weak link and the relay is not known. So we can only use the position information as a guidance to look for the candidate deployment positions. After the relay is deployed, the end nodes will adjust their power to communicate with the relay node assigned based on the actual channel condition. In order to reduce the total number of MR nodes, candidate positions are selected based on their covering degree, as in line 7, so that the position that potentially covers more weak links is favored. After a position is selected, adjustment is performed based on the feasibility constraints. If the constraints are satisfied, as in lines 8-9, the weak links covered are associated with this position, and the *DoF* of the corresponding MR node is updated. The remaining DoFs are used to cover other weak links and preferably considered in the next iteration, as in line 5. If the feasibility constraints are not satisfied, as in lines 10-13, the weak links with a lower covered degree are preferably covered by this position, as the ones with a higher covered degree generally have a higher opportunity of being covered by other positions. The position is then added to the set $\tilde{\mathcal{P}}$, whose positions will be excluded in the next iteration. Lines 4-14 work iteratively until all the weak links are covered and MRs all satisfy the feasibility constraints.

Algorithm 15 Centralized Deployment (at a central controller)

1: **Input:** The set of weak links $\mathcal{L}$ and the candidate MIMO node positions $\mathcal{P}$
2: **Output:** The selected positions to place MIMO nodes $\mathcal{P}^*$ with association of weak links $\{z_{ijk}\}$
3: Initialize: $\mathcal{P}^* = \emptyset$, $\tilde{\mathcal{P}} = \emptyset$, $L = \mathcal{L}$, $k_i = 1, \forall i$
4: **while** $L \neq \emptyset$ **do**
5:    If $\mathcal{P}^* \cap (\mathcal{P} \setminus \tilde{\mathcal{P}}) \neq \emptyset$, $\mathcal{P}_0 = \mathcal{P}^* \cap (\mathcal{P} \setminus \tilde{\mathcal{P}})$; else $\mathcal{P}_0 = \mathcal{P} \setminus \tilde{\mathcal{P}}$.
6:    For $\forall p_i \in \mathcal{P}_0$, use R_{k_i} to construct a communication graph with the end nodes of weak links in L. Count the *covering degree* of an MR placed at p_i, and the *covered degree* for each weak link.
7:    Select $p_n \in \mathcal{P}_0$ with the highest covering degree.
8:    **if** the feasibility constraints of m_n are satisfied **then**
9:       Remove the weak links covered by m_n from L; Update the remaining DoF of m_n. Let $k_n = k_n + 1$ if the remaining DoF is larger than 0 and k_n is not the maximum layer; otherwise, add p_n into $\tilde{\mathcal{P}}$. If $p_n \notin \mathcal{P}^*$, add p_n into $\mathcal{P}^*$.
10:    **else**
11:       Sort the weak links in the ascending order of their covered degrees;
12:       Tentatively select the weak links to cover in this order until the constraints of m_n cannot be satisfied;
13:       Remove the selected weak links covered by m_n from L. If $p_n \notin \mathcal{P}^*$, add p_n into $\mathcal{P}^*$. Add p_n into $\tilde{\mathcal{P}}$.
14:    **end if**
15: **end while**

7.5 Distributed Deployment

As there are no actual nodes available at the candidate positions and the existing nodes themselves may not be able to communicate due to the weak links, we resort to mobile agents to determine the final deployment positions. A mobile agent can move to the candidate positions, cooperates and communicates with other mobile agents when they are within each other's transmission range as well as communicates with RN nodes closed by using all the available MIMO modes. Mobile agents collect the topology information of the network and traffic requirements of weak links independently, and each agent coordinates with its neighbors in reaching a consensus on how many MRs are required and the deployment strategy. The communications between mobile agents are in an ad hoc manner, and there is no central controller to manage them. The details of information acquisition and communications process are beyond the scope of this chapter.

Generally, there are two tasks for the agents. One is to find out where the weak links and the candidate locations of MRs are, and the other is to decide where to actually place the MR nodes and how to assign weak links to the MR nodes using appropriate transmission strategy. The agents exhaustively go through the whole area twice to accomplish the tasks. First, given the number of agents, the area of the network is divided into equal-size stripes. Each agent goes over its assigned stripe to "meet" with each RN within the stripe to collect the information of weak links and aggregate traffic requirements. It can determine the candidate MR positions within the stripe based on the network topology and the collected information. In the second time, all agents move together and stay close as a group, collaboratively checking the candidate positions and determining the placement of MRs distributively, as in Algorithm 16.

The algorithm shares the same nature as the centralized Algorithm 15. Each agent is assigned a sequence of candidate positions and will check one of the positions p_i in each move to determine if it can be selected to deploy an MR. The parameter INPROCESS is used to indicate if the current move is completed for agent n, and $\widetilde{\alpha_i}$ is the remaining DoF that can be used for the potential MR at position p_i, i.e., a potential m_i. The weak links that can be covered by m_i is determined starting from its lowest transmission range, until all its DoF is used up. An agent needs to communicate with end RN nodes of weak links to make sure it can reach them and also determine its covering degree of weak links, and share the information with its two-hop neighboring agents (which potentially cover some common weak links), as in lines 6 and 7. Similar to the centralized algorithm, the positions with a higher covering degree are favored. If the feasibility constraints are satisfied for covering the weak links in its range and the remaining DoF is larger than zero, the remaining DoF of m_i is used to cover more weak links and will be preferably used; otherwise, the weak links with lower covered degrees are preferably covered. When an agent at a selected position finishes determining all possible weak links to cover, it will send an END message to inform the neighboring agents, and the ones which also have this position in their assigned candidate list learn that this position is no longer active. These are shown in lines 12-18. In lines 20-23, neighboring agents cooperatively determine the selection of MR positions and the association of weak links with appropriate strategies. When the process is completed, if the adjacent candidate

Algorithm 16 Distributed Deployment (for each mobile agent n in each move)

Input: The set of uncovered weak links $\mathcal{L}$ in the network and the candidate MIMO node positions p_i assigned for n in the current move

Output: Decide if p_i is selected as a position for an MR node, if yes, also determine the association of weak links $\{z_{ijk}\}$

Initialize: INPROCESS = 1, $\widetilde{\alpha}_i = \alpha_i$, $k = 1$

while INPROCESS **do**

$\mathcal{L}' = \emptyset$, $\mathcal{L}'' = \emptyset$.

Use range R_k to communicate with the end nodes of weak links in $\mathcal{L}$, find the subset $\mathcal{L}'$ it can cover; count the covering degree for an MR placed at p_i, i.e., m_i.

Share the information of covering degree of p_i with the 2-hop neighboring agents if p_i is not selected.

if p_i is the position with the highest covering degree among active positions within 2-hop neighborhood **then**

Mark p_i as selected if it hasn't been marked.

end if

if p_i is selected and the covering degree is not zero **then**

if the feasibility constraints of m_i are satisfied when covering all the weak links in $\mathcal{L}'$ **then**

Send out $\mathcal{L}'$, and $\mathcal{L} = \mathcal{L} - \mathcal{L}'$. Update $\widetilde{\alpha}_i$.

If $\widetilde{\alpha}_i > 0$ and k is not the maximum, let $k = k + 1$; otherwise, send out END to other agents, and INPROCESS = 0.

else

Tentatively add in $\mathcal{L}''$ the weak links of $\mathcal{L}'$ in the ascending order of their covered degrees, until the constraints of m_i cannot be satisfied.

Send out $\mathcal{L}''$, $\mathcal{L} = \mathcal{L} - \mathcal{L}''$. Also send out END to other agents. INPROCESS = 0.

end if

else

Listen to other nodes' messages, update $\mathcal{L}$ and the covering degree of weak links in $\mathcal{L}$.

if the covering degree is 0 **then**

If k is the maximum value, INPROCESS = 0; else $k = k + 1$.

end if

end if

end while

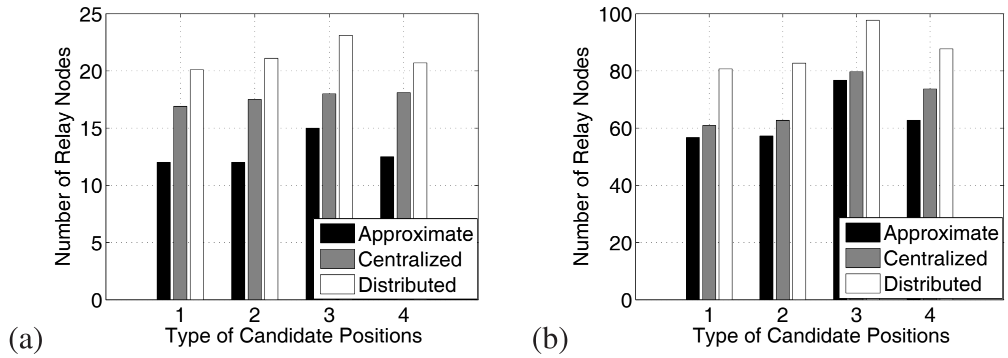


Figure 7.5: Impact of the candidate positions: (a) in sparse network; (b) in dense network.

positions of p_i have all been explored or the DoF of m_i is used up, agent n should move to the next position; otherwise, it should stay in the current position, as it is on the boundary of the current area and could use its remaining degree to serve some weak links of the next area to explore.

7.6 Performance Evaluation

We evaluate the performance of the proposed algorithms through simulations. We implemented the approximate algorithm, the centralized heuristic algorithm and the distributed algorithm proposed in Sections 7.4.2, 7.4.3 and 7.5, as well as a reference scheme where the relay nodes are only equipped with one antenna. The performance metric is the objective of our proposed algorithms, namely the minimum number of relay nodes to fully cover the weak links in the network while satisfying the aggregate traffic requirement of each weak link.

The node locations are generated within a 1500×1500 area according to two-dimensional uniform random distribution. The regular nodes (RN) are assumed to have the same range. Suppose that the ratio of weak links over all the connected links is γ and the weak links are randomly distributed. Each MIMO relay (MR) node is equipped with an array of antennas to facilitate MIMO transmission. For an MR node m_i with α_i available DoFs, we consider up to $2\alpha_i$ transmission strategies available to take advantage of the *MAS* and *RANGE* modes discussed in Section 7.2, each with empirical parameters of capacity, range and the required value of DoF. For simplicity, all weak links are assumed to have the same traffic requirement, whose value is normalized to the capacity of MIMO-channel with the degree-of-freedom 1. The default values of γ and α_i are 0.5 and 4 respectively, and the network has 150 nodes if not

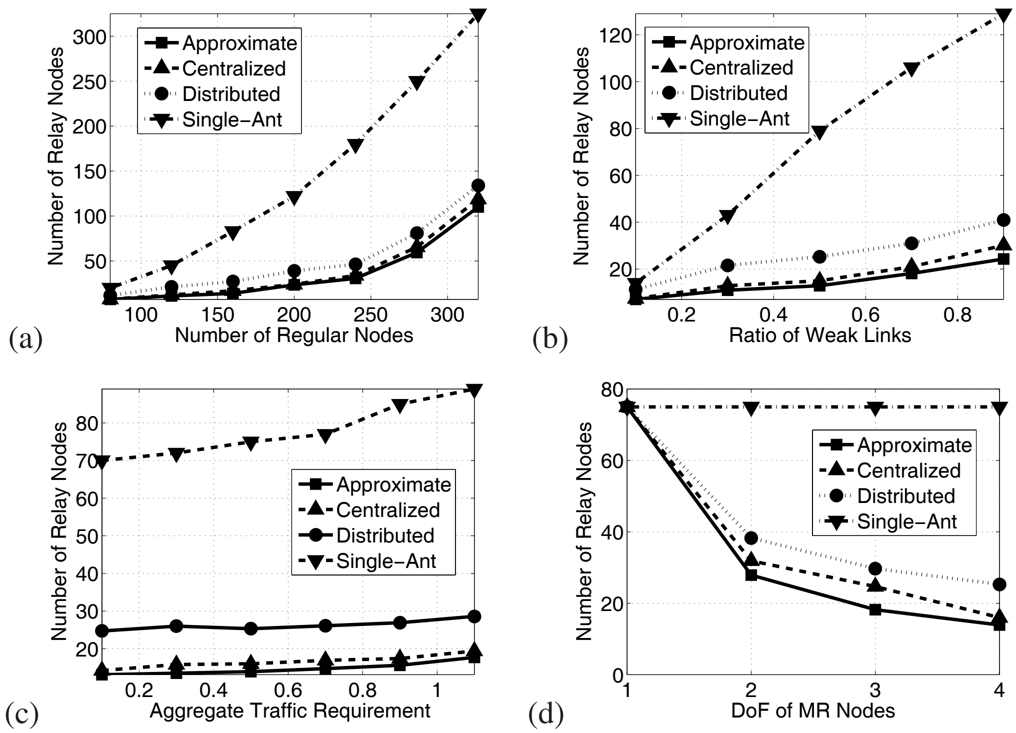
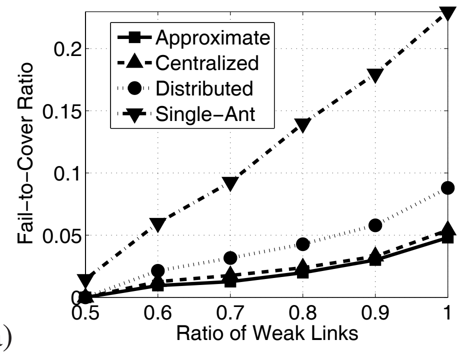


Figure 7.6: Performance of the algorithms: (a) impact of the node density, (b) impact of the ratio of weak links, (c) impact of the traffic requirement, (d) impact of the DoF value.

otherwise specified. The parameter ϵ for the approximate algorithm is set as 3 so the PTAS has approximation ratio of 4. For each simulation setup we take 100 runs and the average result is reported.

In Fig. 7.5, we first study the impact of candidate node positions as discussed in Section 7.4.1. As the detection of optimum positions is coupled with the selection of MR strategies and thus is prohibitive to track, we compare the 1-center positions (type 1) and the simplified positions (type 2), with two types of grid-based positions, sparsely and densely distributed with distances between adjacent nodes to be 300 (type 3) and 50 (type 4) respectively. The node density are set to 60 and 300 to represent a sparse network and dense network respectively. It can be seen that for all the three proposed algorithms, position type 1 and 2 achieve very close number of MR nodes in both network scenarios, which indicates that it is sufficient to use type 2 positions to reduce the deployment complexity while maintaining the accuracy of candidate positions. Both type 1 and 2 perform better than the grid positions, especially type 3, as it does not take the actual positions of weak links into consideration and also does not provide fine-grained candidate positions. Moreover, type 3 positions have the chance of



(a)

Figure 7.7: The Fail-to-Cover Ratio for different traffic requirements.

failing to cover all the weak links, as sparsely distributed MRs may not be able to reach the end nodes of some weak links. The densely distributed grid positions are seen to have the similar performance as that of type 1 and 2 at the cost of much higher computational complexity, as the number of the type 4 positions can be quite large especially in a dense network.

We then evaluate the performance of our algorithms using the type 2 candidate positions. We compare our centralized and distributed heuristic algorithms with the approximate algorithm with approximation ratio 4, as well as a centralized reference scheme where the relay nodes are only equipped with an antenna each. Both the node density and the ratio of weak links can impact the density of weak links in the network, as in Fig. 7.6(a) and (b). By employing MIMO nodes as relays, where the multiplexing and diversity features are exploited, our proposed algorithms require only up to $1/3$ the number of relay nodes compared with the single antenna relays. Note that the reduction of the relay number cannot be the same as the size of the antenna array, as all the antennas from the same node are constrained to the same location. With the increased aggregate traffic requirements of weak links, as in Fig. 7.6(c), 30% more relay nodes are required for single antenna case while only 16% more is needed for MIMO relay nodes, as our algorithms enable the MR nodes to flexibly select the appropriate transmission strategies based on the deployment conditions and better satisfy the traffic requirements of weak links. In Fig. 7.6(d), we show the performance when the DoF value of MR nodes are varied. A larger value of DoF brings the opportunity for more options of available strategies, and also provides the potential of higher capacity and longer range. Thus the number of MRs is significantly reduced. In all the figures, it is obvious that our centralized algorithm achieves very close performance with the approximate algorithm, which verifies its effectiveness in achieving desirable performance with lower complexity. The distributed algorithm requires 61% more MR nodes compared with the approximate algorithm, however, it still significantly outperforms the single-antenna case with 67% fewer relay nodes.

As discussed earlier, simply increasing the number of single-antenna relays cannot substitute MIMO relays due to interference. As the traffic requirements increase, there is a possibility that some of the weak links can not be successfully covered to provide performance provisioning for them. A new metric called Fail-to-Cover Ratio is introduced to denote the ratio of the links that cannot be covered over all the weak links. In Fig 7.7, MIMO relays achieves up to 86% lower Fail-to-Cover Ratio compared with single-antenna relays, thanks to the higher capacity of MIMO with exploration of spatial DoFs. This demonstrates that the deployment of MIMO relay is necessary in cases where single-antenna relays could not provide the sufficient coverage, and our deployment schemes are effective in achieving the coverage especially for relieving a traffic bottleneck.

7.7 Conclusions

In this chapter, we propose MIMO-relay deployment algorithms which exploit MIMO features to flexibly select among various possible transmission strategies based on network conditions to effectively bridge weak links and provide performance provisioning. We first present constraints that capture the characteristics of transmissions between MIMO relays and regular nodes, and mathematically formulate the MIMO relay deployment problem with the objective of minimizing the number of relays while satisfying the transmission requirement of each weak link. We then propose a polynomial-time approximation scheme (PTAS) to provide a performance upper bound in the centralized scenario, as well as a centralized heuristic algorithm with reduced complexity and a distributed algorithm that can facilitate practical in-field implementation. The performance of our algorithms is evaluated through simulations with varied node density, percentage of weak links, aggregate traffic requirement and DoF value of MR nodes. The results demonstrate that MIMO relays can more effectively assist weak links, especially for relieving traffic bottlenecks, and the proposed heuristic algorithm achieves very close performance compared with the upper bound provided by the PTAS.

Chapter 8

Conclusions

The purpose of this dissertation is to model the realistic physical layer characteristics and constraints to assist the design of algorithms and protocols that enhance the performance and efficiency of MIMO ad hoc networks. A suite of algorithms and schemes for MIMO ad hoc networks has been proposed over multiple network layers. To summarize, we have made contributions in the following aspects:

- *Opportunistic and Cooperative Spatial multiplexing* [19, 20]: Formulated a concrete physical model for MIMO ad hoc networks, and presented cross-layer algorithms which take advantage of physical layer channel information to opportunistically schedule cooperative spatial multiplexed transmissions between nodes to maximize the network throughput. Proposed a novel scheduling algorithm and protocol to exploit the multiuser diversity and spatial diversity by taking advantage of the meshed topology, while also supporting user transmission quality requirement.
- *Adaptive Scheduling in Heterogeneous MIMO networks* [21, 22]: Proposed a holistic distributed scheduling algorithm that can adaptively select different transmission strategies based on the antenna array size, channel condition, traffic demand and multiuser diversity to effectively relieve the bottleneck effect caused by nodes with smaller antenna arrays, and avoid transmission failure due to violation of channel constraint.
- *Adaptive Exploitation of Cooperative Relay* [23, 24]: Exploited cooperative relay transmission in a MIMO-based ad hoc network to cope with harsh channel condition. Designed both centralized and distributed scheduling algorithms to integratively support adaptive use of cooperative relay in a MIMO-based ad hoc network, and a MAC protocol to implement the distributed algorithm. The scheduling scheme can efficiently

invoke relay transmission without introducing significant signaling overhead as conventional relay schemes, and seamlessly integrate relay transmission with multiplexed MIMO transmission.

- *Distributed Interference Management* [25]: Investigated the physical model of interference management for MIMO networks and made the first effort to design a distributed mechanism for MIMO transmission with adaptive interference cancellation at both the transmitter and receiver. Specifically, we allow transmitters to allocate their DoF to transmit multiple data streams and cancel interference towards a selected set of receivers, and receivers could use their antenna array for receiving and cancel the residual interference.
- *MIMO-Aware Routing* [26][27]: Identified the specific opportunities and constraints brought by MIMO transmissions, with flexible transmission strategy selection and node cooperation. Formulated the MIMO-enabled multi-source multi-destination multi-hop routing problem into a multi-commodity flow problem. Developed a polynomial time approximation solution that maximizes the scaling factor for the concurrent flows as well as a distributed algorithm to minimize the congestion in the network links.
- *Deployment of MIMO Relay* [28]: Proposed the very first strategy to deploy MIMO nodes as relays to assist weak links in wireless networks, with the aim of reducing the number of relay nodes and providing performance provisioning. Provided a polynomial-time approximation scheme (PTAS) algorithm, as well as a centralized and a distributed algorithms to effectively determine the MIMO relay nodes positions over the network and flexibly select various transmission strategies to further leverage the advantages brought by MIMO.

The research contained in this dissertation combines algorithm design, protocol design, analytical, and simulation techniques. We expect the research results to have a significant impact on the fundamental design principles and infrastructures for the development of future wireless ad hoc networks. Moreover, we expect the outcome of this research to inspire and boost the exploitation of MIMO and other advance technologies in wireless networks [85–87].

Bibliography

[1] IEEE Strandards. *IEEE Standard 802.16e*.

[2] IEEE Strandards. *IEEE Standard 802.11n*.

[3] D. Gesbert, M. Shafi, D. shan Shiu, P. J. Smith, and A. Naguib. From theory to practice: an overview of MIMO space-time coded wireless systems. *IEEE J. Select. Areas Commun.*, (3):281–302, April 2003.

[4] G. J. Foschini. Layered space-time architecture for wireless communication in fading environments when using multi-element antennas. *Bell Labs Tech. J.*, (2):41–59, 1996.

[5] P. W. Wolniansky, G. J. Foschini, G. D. Golden, and R. A. Valenzuela. V-BLAST: An architecture for realizing very high data rates over the rich-scattering wireless channel. In *Proc. of ISSSE-98*, September 1998.

[6] S. M. Alamouti. A simple transmit diversity technique for wireless communications. *IEEE J. Select. Areas Commun.*, pages 1451–1458, October 1998.

[7] V. Tarokh, N. Seshardi, and A. Calderbank. Space-time codes for high data rate wireless communication: Performance criteria and code construction. *IEEE Trans. Inform. Theory*, (2):744–765, March 1998.

[8] V. Tarokh, N. Seshardi, and A. Calderbank. Space-time block codes from orthogonal designs. *IEEE Trans. Inform. Theory*, (5):1456–1467, July 1999.

[9] W. Ajib and D. Haccoun. An overview of scheduling algorithms in MIMO-based fourth-generation wireless systems. *IEEE Network*, (5):43–48, Sept.-Oct. 2005.

[10] V.K.N. Lau, Y. Liu, and T. A. Chen. Optimal multi-user space time scheduling for wireless communications. In *Proc. IEEE VTC 2002-Fall*, pages 1939–1942, 2002.

[11] D. Aktas and H. El Gamal. Multiuser scheduling for MIMO wireless systems. In *Proc. IEEE VTC 2003-Fall*, pages 1743–1747, 2003.

[12] M. Hu and J. Zhang. MIMO ad hoc networks: Medium access control, saturation throughput, and optimal hop distance. *Journal of Communications and Networks, Special Issue on Mobile Ad Hoc Networks*, pages 317–330, 2004.

[13] J-S. Park, A. Nandan, M. Gerla, and H. Lee. SPACE-MAC: Enabling spatial-reuse using MIMO channel-aware MAC. In *Proc. IEEE ICC 2005*, May 2005.

[14] M. Levorato, S. Tomasin, P. Casari, and M. Zorzi. Analysis of spatial multiplexing for cross-layer design of MIMO ad hoc networks. In *Proc. IEEE VTC-2006 Spring*, pages 1146–1150, May 2006.

[15] P. Casari, M. Levorato, and M. Zorzi. DSMA: an access method for MIMO ad hoc networks based on distributed scheduling. In *Proc. ACM IWCMC 2006*, July 2006.

[16] K. Sundaresan, R. Sivakumar, M. Ingram, and T-Y. Chang. A fair medium access control protocol for ad-hoc networks with MIMO links. In *Proc. IEEE INFOCOM 2004*, June 2004.

[17] K. Sundaresan and R. Sivakumar. A unified MAC layer framework for ad-hoc networks with smart antennas. In *Proc. ACM MobiHoc 2004*, pages 244–255, May 2004.

[18] M. Park, R. Heath, and S. Nettles. Improving throughput and fairness of MIMO ad hoc networks using antenna selection diversity. In *Proc. IEEE Globecom 2004*, November 2004.

[19] S. Chu and X. Wang. Opportunistic and cooperative spatial multiplexing in MIMO ad hoc networks. In *Proc. ACM MobiHoc 2008*, pages 63–72, May 2008.

[20] S. Chu and X. Wang. Opportunistic and cooperative spatial multiplexing in MIMO ad hoc networks. *IEEE Trans. on Networking*, (5):1610–1623, Oct. 2010.

[21] S. Chu and X. Wang. Adaptive and distributed scheduling in heterogeneous MIMO-based ad hoc networks. In *Proc. IEEE MASS 2009*, pages 217–226, Oct 2009.

[22] S. Chu, X. Wang, and Y. Yang. Adaptive scheduling in MIMO-based ad hoc networks. under submission.

[23] S. Chu and X. Wang. Adaptive exploitation of cooperative relay for high performance communications in MIMO ad hoc networks. In *Proc. IEEE MASS 2010*.

[24] S. Chu, X. Wang, and Y. Yang. Exploiting cooperative relay for high performance communications in MIMO ad hoc networks. under revision.

[25] S. Chu, X. Wang, and Z. Zhang. Distributed interference management for MIMO-based wireless mesh networks. under submission.

[26] S. Chu and X. Wang. MIMO-aware routing in wireless mesh networks. In *Proc. IEEE INFOCOM 2010*.

[27] S. Chu and X. Wang. Routing in multi-hop wireless networks with MIMO awareness. under submission.

[28] S. Chu, X. Wang, and M. Li. Enforcing high-performance operation of multi-hop wireless networks with MIMO relays. under submission.

[29] M. Zorzi, J. Zeidler, A. Anderson, B. Rao, J. Proakis, A.L. Swindlehurst, M. Jensen, and S. Krishnamurthy. Cross-layer issues in MAC protocol design for MIMO ad hoc networks. *IEEE Wireless Communication Magzine*, (4):62–76, Aug. 2006.

[30] S. Tomasin M. Levorato, P. Casari and M. Zorzi. Physical layer approximations for cross-layer performance analysis in mimo-blast ad hoc networks. *IEEE Trans. on Wireless Communications*, Dec. 2007.

[31] Y. Pan and S. Aissa. Performance analysis of selective space-time coding and selection diversity under perfect and imperfect CSI. In *Proc. PIMRC 2005*, pages 2371–2375, September 2005.

[32] D. Tse and P. Viswanath. *Fundamentals of Wireless Communication*. Cambridge University Press, May 2005.

[33] B. Vucetic and J. Yuan. *Space-Time Coding*. New York: Wiley, 2003.

[34] L. Bao and J.J. Garcia-Luna-Aceves. *Algorithms and Protocols for Wireless and Mobile Networks*. CRC/Hall Publisher, 2004.

[35] A. Ephremides and T.V. Truong. Scheduling broadcasts in multihop radio networks. *IEEE Transactions on Communications*, (4):456–460, April 1990.

[36] R. M. de Moraes, H. R. Sadjadpour, and J. J. Garcia-Luna-Aceves. Many-to-many communication: A new approach for collaboration in MANETs. In *Proc. IEEE INFOCOM 2007*, pages 1829–1837, May 2007.

[37] R. Bhatia and L. Li. Throughput optimization of wireless mesh networks with MIMO links. In *Proc. IEEE INFOCOM 2007*, May 2007.

[38] B. Hamdaoui and K.G. Shin. Characterization and analysis of multi-hop wireless mimo network throughput. In *Proc. ACM MobiHoc '07*, pages 120–129, 2007.

[39] J. Liu, Y.T. Hou, Y. Shi, and H. Sherali. Cross-layer optimization for mimo-based wireless ad hoc networks: Routing, power allocation, and bandwidth allocation. *IEEE J. Select. Areas of Commun.*, (6):913–926, August 2008.

[40] R. de Lacerda, L. S. Cardoso, R. Knopp, D. Gesbert, and M. Debbah. EMOS platform: Real-time capacity estimation of MIMO channels in the UMTS-TDD band. In *Proc. ISWCS 2007*, pages 782–786.

[41] W. Navidi and T. Camp. Stationary distributions for the random waypoint mobility model. *IEEE Trans. on Mobile Computing*.

[42] A. Sendonaris, E. Erkip, and B. Aazhang. User cooperation diversity part I and part II. *IEEE Trans. Commun.*, (11):1927–1948, November 2003.

[43] J. N. Laneman, D. N. C. Tse, and G. W. Wornell. Cooperative diversity in wireless networks: Efficient protocols and outage behavior. *IEEE Trans. Inform. Theory*, (12):3062–3080, December 2004.

[44] Q. Zhang, J. Jia, and J. Zhang. Cooperative relay to improve diversity in cognitive radio networks. *IEEE Communications Magazine*, (2):111–117, February 2009.

[45] S. Cui, A. J. Goldsmith, and A. Bahai. Energy-efficiency of MIMO and cooperative MIMO in sensor networks. *IEEE J. Select. Areas of Commun.*, (6):1089–1098, August 2004.

[46] G. Jakllari, S. V. Krishnamurthy, M. Faloutsos, P. V. Krishnamurthy, and O.A Ercetin. Framework for distributed spatio-temporal communications in mobile ad hoc networks. In *Proc. IEEE INFOCOM 2006*, pages 1–13, April 2006.

[47] Y. Shi, S. Sharma, Y. T. Hou, and S. Kompella. Optimal relay assignment for cooperative communications. In *Proc. ACM Mobihoc 2008*, May 2008.

[48] A.S.Ibrahim, A.K.Sadek, W. Su, and K.J.R.Liu. Cooperative communications with relay-selection: when to cooperate and whom to cooperate with? *IEEE Transactions on Wireless Communications*, 7(7):2814 –2827, july 2008.

[49] Y.Wei, F.R.Yu, and M.Song. Distributed optimal relay selection in wireless cooperative networks with finite-state markov channels. *IEEE Transactions on Vehicular Technology*, 59(5):2149 –2158, jun 2010.

[50] M. Yuksel and E. Erkip. Multiple-antenna cooperative wireless systems: A diversityc-multiplexing tradeoff perspective. *IEEE Trans. Inform. Theory*, Octorber 2007.

[51] K. Sundaresan and R. Sivakumar. Cooperating with smartness: Using heterogeneous smart antennas in ad-hoc networks. In *Proc. INFOCOM 2007*, pages 303–311, May 2007.

[52] M. Levorato, S. Tomasin, and M. Zorzi. Cooperative spatial ultiplexing for ad hoc networks with hybrid ARQ: system design and performance analysis. *IEEE Trans. Commun.*, (9):1545–1555, September 2008.

[53] Mischa Schwartz. *Mobile Wireless Communications*. Cambridge University Press, 2005.

[54] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin. *Network flows: theory, algorithms and applicationss*. Prentice Hall, 1993.

[55] Q.H. Spencer, A.L. Swindlehurst, and M. Haardt. Zero-forcing methods for downlink spatial multiplexing in multiuser MIMO channels. *IEEE Transactions on Signal Processing*, 52(2):461–471, 2004.

[56] T. Yoo and A. Goldsmith. On the optimality of multiantenna broadcast scheduling using zero-forcing beamforming. *IEEE Journal on Selected Areas in Communications*, 24(3):528–541, March 2006.

[57] K. Sundaresan, R. Sivakumar, M. Ingram, and T-Y. Chang. Medium access control in ad hoc networks with MIMO links: Optimization considerations and algorithms. *IEEE Trans. Mobile Comp.*

[58] J. Liu, Y. Shi, and Y.T. Hou. A tractable and accurate cross-layer model for multi-hop MIMO networks. In *Proc. IEEE INFOCOM '10*, March 2010.

[59] Y. Shi, J. Liu, C. Jiang, C. Gao, and Y. T. Hou. An optimal MIMO link model for multi-hop wireless networks. In *Proc. IEEE INFOCOM '11*, April 2011.

[60] L.E. Li, R. Alimi, D. Shen, H. Viswanathan, and Y.R. Yang. A general algorithm for interference alignment and cancellation in wireless networks. In *Proc. IEEE INFOCOM 2010*, pages 1 –9, march 2010.

[61] R. Srinivasan, D. Blough, and P. Santi. Optimal one-shot stream scheduling for mimo links in a single collision domain. In *Proc. IEEE SECON 2009*, pages 700–708, 2009.

[62] E. Gelal, K. Pelechrinis, I. Broustis, T.-S. Kim, S.V. Krishnamurthy, and B. Rao. Topology control for effective interference cancellation in multi-user mimo networks. In *Proc. IEEE INFOCOM '10*, March 2010.

[63] S. Gollakota, S. D. Perli, and D. Katabi. Interference alignment and cancellation. In *Proc. ACM SIGCOMM 2009*, pages 159–170, 2009.

[64] E. Aryafar, N. Anand, T. Salonidis, and E. W. Knightly. Design and experimental evaluation of multi-user beamforming in wireless LANs. In *Proc. IEEE MOBICOM '10*, 2010.

[65] K. Lin, S. Gollakota, and D. Katabi. Random access heterogeneous mimo networks. In *Proc. ACM SIGCOMM 2011*, 2011.

[66] K. Tan, H. Liu, J. Fang, W. Wang, J. Zhang, M. Chen, and G. M. Voelker. SAM: enabling practical spatial multiple access in wireless LAN. In *Proc. ACM MobiCom '09*, pages 49–60, September 2009.

[67] K. Sundaresan and R. Sivakumar. Routing in ad-hoc networks with MIMO links. In *Proc. IEEE ICNP 2005*, pages 85–98, 2005.

[68] A. Gkelias, F. Boccardi, C.H. Liu, and K.K. Leung. MIMO routing with QoS provisioning. In *Proc. ISWPC 2008*.

[69] K. C. Lin, S. Gollakota, , and D. Katabi. Random access heterogeneous mimo networks. In *Proc. ACM SIGCOMM 2011*, 2011.

[70] M. Kodialam and N. Thyaga. Characterizing the capacity region in multi-radio multi-channel wireless mesh networks. In *Proc. ACM Mobicom '05*, pages 73–87, May 2005.

[71] 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 '05*, pages 58–72, May 2005.

[72] X. Lin and S. Rasool. A distributed joint channel-assignment, scheduling and routing algorithm for multi-channel ad-hoc wireless networks. In *Proc. IEEE Infocom '07*, pages 58–72, May 2007.

[73] N. Garg and J. Koenemann. Faster and simpler algorithms for multicommodity flow and other fractional packing problems. In *Proc. FOCS '98*.

[74] B. Awerbuch, R. Khandekar, and S. Rao. Distributed algorithms for multicommodity flow problems via approximate steepest descent framework. In *Proc. SODA '07*.

[75] W. Zhang, G. Xue, and S. Misra. Fault-tolerant relay node placement in wireless sensor networks: problems and algorithmss. In *Proc. IEEE Infocom '07*, pages 1649–1657, May 2007.

[76] X. Cheng, D.-Z. Du, L. Wang, and B. Xu. Relay sensor placement in wireless sensor networks. *Wirel. Netw.*, 14(3):347–355, 2008.

[77] Y.T. Hou, Y. Shi, H.D. Sherali, and S.F. Midkiff. On energy provisioning and relay node placement for wireless sensor networks. *IEEE Transactions on Wireless Communications*, 4(5):2579–2590, 2005.

[78] X. Han, X. Cao, E.L. Lloyd, and C.-C. Shen. Fault-tolerant relay node placement in heterogeneous wireless sensor networks. *IEEE Transactions on Mobile Computing*, 9(5):643–656, may 2010.

[79] S. Misra, S.D. Hong, G. Xue, and J. Tang. Constrained relay node placement in wireless sensor networks: Formulation and approximations. *IEEE/ACM Transactions on Networking*, 18(2):434–447, april 2010.

[80] R. Chandra, L. Qiu, K. Jain, and M. Mahdian. Optimizing the placement of internet taps in wireless neighborhood networks. In *Proc. IEEE ICNP '04*, pages 271–282, 2004.

[81] F. Li, Y. Wang, X.-Y. Li, A. Nusairat, and Y. Wu. Gateway placement for throughput optimization in wireless mesh networks. *Mob. Netw. Appl.*, 13(1-2):198–211, 2008.

[82] A. Srinivas and E. Modiano. Joint node placement and assignment for throughput optimization in mobile backbone networks. In *Proc. IEEE Infocom '08*, pages 1804–1812, April 2008.

[83] D. Dong, Y. Liu, and X. Liao. Self-monitoring for sensor networks. In *Proc. ACM MobiHoc '08*, pages 431–440, 2008.

[84] D. S. Hochbaum and W. Maass. Approximation schemes for covering and packing problems in image processing and VLSI. *J. ACM*, 32(1):130–136, 1985.

[85] C. Gao, S. Chu, and X. Wang. A distributed scheduling algorithm on MIMO empowered cognitive radio ad hoc networks. under submission.

[86] P. Wei, S. Chu, X. Wang, and Yu Zhou. Deployment of a reinforcement backbone network with constraints of connection and resources. In *Proc. ICDCS'10*, Jun. 2010.

[87] S. Chu, P. Wei, X. Zhong, X. Wang, and Y. Zhou. Deployment of a connected reinforced backbone network with a limited number of backbone nodes. under revision.