Coding for Wireless Broadcast and Network Secrecy

Thesis by

Tao Cui

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

California Institute of Technology
Pasadena, California

2010
(Defended September 4th, 2009)

For my parents and my wife Tingting, who offered me endless encouragement, love, and support.

Acknowledgements

This is perhaps the easiest and hardest chapter that I have to write. It will be simple to name all the people who made this thesis possible, but it will be tough to thank them enough.

I would like to thank my advisor, Professor Tracey Ho for her support, encouragement and guidance on my research and career. Her constant curiosity, passion, insight, kindness and genuine concern for her students made working with her a memorable experience. I have been privileged to learn from her in the past few years, and I am very grateful for the excellent research environment and strict training she gave to me. My thesis committee (Michelle Effros, Babak Hassibi, Steven Low, and P. P. Vaidyanathan) has been very supportive and helpful during my graduate study. I would like to thank all of them for their insight and suggestions on my research.

It is my great honor and pleasure working with all the passionate and creative people in Professor Tracey Ho and Professor Michelle Effros' lab. I gave my deep gratitude to Derek Leong, Sukwon Kim, Christopher Sungwook Chang, Mayank Bakshi, Simon Chien, and Svitlana Vyetrenko. Special thanks to Dr. Lijun Chen, Dr. Feifei Gao and Professor Jörg Kliewer, my collaborators, for all the work they have done. It has been a great experience working and studying with them.

Finally I would like to give my deepest gratitude to my wife Tingting Wang for her love, encouragement, support and patience in my life. I thank her for all the things she gave to me. I am forever indebted to my parents for their understanding, endless love, patience and encouragement when it was most required. I feel fortunate to have been brought up by two of the most kind people I have ever met.

Abstract

In the first part of this thesis, we exploit wireless broadcast across different layers in wireless networks. The wireless channel is distinguished by its broadcast nature. Wireless broadcast provides a fertile ground to improve the efficiency of existing wireless networks and design new ones.

Specifically, we first consider relaying strategies for memoryless two-way relay channels at the physical layer. We generalize networking layer network coding operating on a finite field to physical layer network coding, which is a mapping from the relay's received signal to its transmitted signal. We analyze the symbol-error performance of several relay strategies, and optimize the relay function via functional analysis. Our results indicate that the interference caused by wireless broadcast can be exploited to improve the spectrum efficiency.

We then develop a cross-layer framework with wireless broadcast, which integrates rate control, network coding and scheduling in transport, network and link layers. Under the primary interference model, we show that the link scheduling problem is the maximum weighted hypergraph matching problem, which is NP-complete. We propose several distributed approximation algorithms and bound their worst case performance.

Next, we describe a new class of medium access control (MAC) protocol, which uses successive interference cancelation to resolve packet collision due to wireless broadcast. Each user is allowed to transmit at different data rates chosen randomly from an appropriately determined set of rates. We characterize the throughput of the proposed protocol compared to that with a centralized controller. A game-theoretic framework along with the dynamic algorithms is proposed to achieve the desired throughput optimal equilibrium, which provides a valuable perspective to understand existing MAC protocols and a general framework to design new ones to improve the system performance.

In the second part of this thesis, we consider the problem of secure transmission in the presence of a wiretapper. Due to wireless broadcast, wireless signals are particularly easy to jam and intercept. We derive the secrecy capacity region for the case when the location of the wiretapped links is known and propose several achievable strategies for the case when such information is unknown. We give an example to show that the secrecy capacities of the two cases are generally unequal and show that in both cases computing the secrecy capacity is NP-complete.

Contents

Acknowledgements **iv**

Abstract **v**

1 Introduction **1**

1.1 Wireless Broadcast: Double-Edged Sword 2

1.2 Network Coding: A New Communication Paradigm 4

1.3 Outline and Contributions 6

2 Memoryless Relay Strategies for Two-Way Relay Channels **9**

2.1 Introduction 9

2.2 System Model 11

2.3 Relay Strategies for the BPSK Case 13

2.3.1 Non-Abs-Based Strategies 14

2.3.1.1 Amplify-and-Forward 14

2.3.1.2 Detect-and-Forward 15

2.3.1.3 Estimate-and-Forward 18

2.3.1.4 Optimized Relay Function 19

2.3.2 Abs-Based Strategies 20

2.3.2.1 Abs-Based Amplify-and-Forward (AAF) 20

2.3.2.2 Abs-Based Detect-and-Forward (ADF) 20

2.3.2.3 Abs-Based Estimate-and-Forward (AEF) 22

2.3.2.4 Optimized Relay Strategy 22

2.4 Comparison Between Two Classes of Strategies 27

2.5 Higher Order Constellations 28

2.6 Peak Power Constraint 32

2.7 Simulation Results 35

2.8 Conclusion 41

2.9 Appendix 42

3 Distributed Scheduling in Wireless Networks Exploiting Broadcast and Network Coding 44

3.1 Introduction 44

3.2 Related Work 46

3.3 System Model 47

3.4 Cross-Layer Design with Broadcast Advantage and Network Coding . 48

3.5 Link Scheduling 52

3.5.1 Problem Formulation 53

3.5.2 Local Optimal Algorithms 54

3.5.3 Randomized Algorithm 63

3.5.4 Hybrid Algorithm 66

3.6 Simulation Results 67

3.6.1 Wireless Butterfly Network 68

3.6.2 Random Networks 71

3.7 Conclusion 72

4 Multiple Access Random Medium Access Control 74

4.1 Introduction 74

4.2 A Motivating Example 76

4.3 Multiple Access MAC in Aloha Type Networks 79

4.3.1 MAC on AWGN Multiple Access Channels 79

4.3.2 Achievable Results 80

4.3.3 Fixed Number of Transmission Rates 84

4.4 Multiple Access MAC in WLAN 87

4.5 Game-Theoretic Model of Contention Control 90

4.5.1 Random Access Game 91

4.5.2 Utility Function Design 94

4.5.2.1 Reverse Engineering from Existing Protocols 94

4.5.2.2 Reverse Engineering from Desired Operating Points . 95

4.5.2.3 Forward Engineering by Heuristics 97

4.6 Dynamics of Random Access Game 99

4.6.1 Basic Dynamic Algorithms 99

4.6.1.1 Best Response 99

4.6.1.2 Gradient Play 100

4.6.2 Asynchronous Dynamic Algorithms 102

4.6.2.1 Best Response 102

4.6.2.2 Gradient Play 103

4.6.3 Dynamic Algorithms under Estimation Uncertainties 104

4.6.4 Equilibrium Selection 105

4.7 Rate Splitting 107

4.8 Simulation Results 110

4.8.1 Aloha Type Networks 110

4.8.2 Half Duplex WLAN 114

4.8.2.1 Throughput Comparison with Capacity Formula . . 114

4.8.2.2 Throughput Comparison with Convolutional Codes . 116

4.9 Conclusion 119

4.10 Appendix 120

4.10.1 Proof of Theorem 4.3 120

4.10.2 Proof of Theorem 4.16 122

4.10.3 Proof of Theorem 4.18 124

4.10.4 Proof of Theorem 4.19 125

4.10.5 Proof of Theorem 4.20 126

4.10.6 Proof of Theorem 4.21 127

5 Secure Network Communications 128

5.1 Introduction 128

5.2 Network Model and Problem Formulation 129

5.3 Secrecy Capacity Region When the Location of the Wiretapped Links is Known 131

5.4 Achievable Strategies when the Location of the Wiretapped Links is Unknown 136

5.5 Unachievability of Cut Set Bound 141

5.5.1 Restricted Wiretap Set (Scenario 3) 142

5.5.2 Unequal Link Capacities (Scenario 4) 147

5.6 NP-Completeness 148

5.7 Conclusion 151

6 Conclusions and Future Research 152

Bibliography 156

List of Figures

1.1 An example of wireless broadcast. 3

1.2 An example of a wired network requiring coding to achieve multicast capacity. The network consists of directed unit capacity links, and a source node s multicasting the same information to two receivers t_1 and t_2. b_1 and b_2 are two symbols to be multicast chosen from a finite field of size greater than 2. The presence of the bottleneck link from 3 to 4 necessitates coding on that link in order to achieve the multicast rate 2. 5

1.3 An example of a wireless network using coding to improve spectrum efficiency. Node 1 and node 2 want to send a packet to each other with the help of relay node 3. There does not exist a direct link between node 1 and node 2. Without network coding, four transmissions are required, while only three transmissions are needed when the relay node broadcasts the XOR of the packets received from two terminals. 6

2.1 Two-way relay channel. 12

2.2 The optimized non-abs-based relay function at different SNRs and with different values of h_1 and h_2. 35

2.3 Comparison of function $f(u)$ in different abs-based schemes with $\sigma_r^2 = \sigma_s^2$, $h_1 = h_2 = 1$ and $P_r = P_s = 1$. 36

2.4 Performance comparison between different abs-based and non-abs-based strategies when $h_1 = 1$ and $h_2 = 0.8$, $P_r = P_s = 1$. The subfigure shows the crossover between the abs-based and non-abs-based strategies. . . 37

2.5 Performance comparison between different abs-based and non-abs-based strategies when $h_1 = 1$ and $h_2 = 0.5$, $P_r = P_s = 1$. The subfigure shows the crossover between the abs-based and non-abs-based strategies. . . . 38

2.6 Performance comparison of AF, AAF and ADF with the AF scheme for one-way relay channel when $\sigma_r^2 = \sigma_s^2$, $h_1 = h_2$ and $P_r = P_s = 1$. 39

2.7 Performance comparison of ADF with DF under different scenarios. . . 39

2.8 Comparison of relay functions for AAF, ADF, and EF with $\sigma_r^2 = \sigma_s^2$ where both terminals use 4-PAM. 40

2.9 SER comparison of ADF and AAF relay functions for 4-PAM with $M = |\mathcal{V}'| = 4, 5, 6, 7$ and $\sigma_r^2 = \sigma_s^2$. The subfigure shows the crossover between different strategies. 40

3.1 Wireless butterfly network. 48

3.2 The evolution of source s_1's rate versus the number of iterations with fixed stepsize $\gamma = 0.01$ for the network in Fig. 3.1, where our cross-layer design with maximum weighted hypergraph matching, Algorithm 3.1 and Algorithm 3.2, and the algorithm in [18] with maximum weighted graph matching and local greedy matching are compared. 69

3.3 The evolution of source s_1's rate versus the number of iterations with fixed stepsize $\gamma = 0.01$ for the network in Fig. 3.1, where our cross-layer design with maximum weighted hypergraph matching, Algorithm 3.3, Algorithm 3.4, and Algorithm 3.5, and the algorithm in [18] with maximum weighted graph matching. 70

4.1 An illustration example. 76

4.2 R_k versus k in a network with $N = 30$ users and $P = 10$, $\sigma^2 = 1$. . . . 84

4.3 Comparison of the total throughput versus P with different k in a network with $N = 50$ users and $\sigma^2 = 1$. 86

4.4 Comparison of throughput by using optimal p_k and asymptotically optimal p_k with $P = 1$, $\sigma^2 = 1$, $\eta_1 = 2$ and $\eta_2 = 0.25$. 90

4.5 Comparison of different schemes with $\sigma^2 = 1$ and different P in a network with 2 users. 109

4.6 Achievable throughput comparison of different strategies as a function of N when $P = 10$ and $\sigma^2 = 1$ (SNR=10 dB). 111

4.7 Achievable throughput comparison of different strategies as a function of P in a network with $N = 30$ users and $\sigma^2 = 1$. 111

4.8 Achievable throughput comparison of different strategies as a function of N when $P = 10$ and $\sigma^2 = 1$ (SNR=10 dB). A finite number K of transmission rates is chosen in the proposed scheme. 113

4.9 Achievable throughput comparison of different strategies as a function of P in a network with $N = 30$ users and $\sigma^2 = 1$. A finite number K of transmission rates is chosen in the proposed scheme. 113

4.10 Achievable throughput comparison of different strategies as a function of N in WLAN when $P = 10$ and $\sigma^2 = 1$ (SNR=10 dB). 115

4.11 Achievable throughput comparison of different strategies as a function of P in a WLAN with $N = 30$ users and $\sigma^2 = 1$. 116

4.12 Dynamics of the protocol without SIC and the proposed protocol in a network with $N = 30$ nodes. We choose $\xi_1 = 0.0722$ and $\xi_2 = 0.7574$ in (4.34), $\xi = 0.8296$ and $\alpha = 2$ in (4.46), and the stepsize $\epsilon_i(t) = 0.05$ in (4.50). 117

4.13 Throughput comparison of different protocols as a function of N when $P = 10$ and $\sigma^2 = 1$ (SNR=10 dB). 1/2 rate convolutional code with BPSK is assumed. 118

4.14 Fairness comparison of different protocols in a network with $N = 30$ nodes, $P = 10$ and $\sigma^2 = 1$ (SNR=10 dB). 1/2 rate convolutional code with BPSK is assumed. 119

5.1 Illustration of strategy 1, an achievable construction where random keys are injected by the source and possibly canceled at intermediate nodes. In this figure, $k = 2$ and only the 5 links in the first layer can be wiretapped. 138

5.2 Example of usefulness of strategy 2. 140

5.3 An example to show that the secrecy rate without knowledge of wiretapping set is smaller than that with such knowledge. The wiretapper can wiretap any three of the five links in the middle layer. 142

5.4 Example of NP-completeness proof for the case with knowledge of wiretapping set 149

List of Tables

3.1 Comparison of different algorithms in the wireless butterfly network . . 71

3.2 Comparison of different algorithms in random networks with 10 nodes, 1 source and 2 sinks 72

3.3 Comparison of different algorithms in random networks with 10 nodes, 1 source and 4 sinks 72

3.4 Comparison of different algorithms in random networks with 10 nodes, 1 source and 6 sinks 72

3.5 Comparison of different algorithms in random networks with 15 nodes, 1 source and 3 sinks 73

4.1 Parameters in Half Duplex WLAN Simulations 114

Chapter 1

Introduction

The explosive growth in wireless networks over the last few years resembles the rapid growth of the Internet within the last decade. Wireless networks have impacted the way we live and do business, and the world has become increasingly mobile. Wireless networks have continued to develop, and their uses have grown significantly in military communication, commercial communication, and emergency services. For example, cell phones and wireless Personal Digital Assistants (PDAs) have become so commonplace in our lives that it is easy to forget that several years ago, they were a rarity.

The wireless era started on 13 May 1897 when Marchese Guglielmo Marconi sent the first ever wireless communication over open sea. In 1948, Claude Elwood Shannon published the landmark paper "A Mathematical Theory of Communication" and founded information theory. Since then, many researchers have tried to find practical error correcting codes to achieve Shannon's channel capacity. In 1993, the discovery of Turbo codes showed that Shannon's limit can be approached [8]. In parallel, commercial communication systems have evolved from the first generation using frequency division multiple access (FDMA) to the fourth generation using multiple-input and multiple-output (MIMO) and orthogonal frequency division multiplexing (OFDM) with turbo codes [1], which can almost achieve the physical layer limit predicted by Shannon. Because the point-to-point channel is well studied at the physical layer, existing wireless network designs are built on the point-to-point abstraction. Routing and rate control protocols for wired networks are applied on this abstraction directly.

Wireless network design based on this abstraction, we argue, is not going to meet the increasing demand for wireless multimedia services and high-speed Internet access. Significantly new designs are hence necessary to efficiently utilize wireless resources, to support new applications and to meet current and future demands on wireless networks.

This thesis examines new techniques for exploiting wireless broadcast to improve the throughput and efficiency of wireless networks and using network coding to enable secure communications. Network coding as well as optimization theory and game theory frameworks are used to develop highly efficient and secure means of wireless transport.

1.1 Wireless Broadcast: Double-Edged Sword

Like all networks, wireless networks transmit data over a network medium. The medium is a form of electromagnetic radiation. The wireless channel is distinguished by its broadcast nature. When omnidirectional antennas are used, every transmission by a node can be received by all nodes that lie within its communication range. The broadcast feature of wireless networks makes their design and control very challenging and imposes strong constraints on the system designer.

Wireless broadcast may lead to interference, which is different from transmission in wired networks, where different nodes' data transmissions do not interference with each other. For example, when two senders transmit simultaneously to a common receiver, the packets collide. Traditional wireless networks have been designed to prevent senders from interfering. Different strategies have been proposed for this purpose. Reservation based schemes such as time division multiple access (TDMA) and frequency division multiple access reserve the medium to a specific node. Random access methods, such as carrier sense multiple access with collision avoidance (CSMA/CA) protocol [2, 49], require each node wishing to transmit to first listen to the channel for a predetermined amount of time so as to check for any activity on the channel. If the channel is sensed to be "idle" then the node is permitted to transmit.

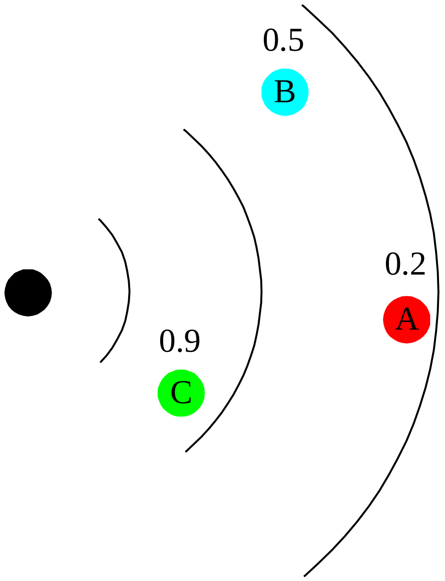


Figure 1.1: An example of wireless broadcast.

If the channel is sensed to be “busy”, the node has to defer its transmission. But this mechanism may incur the hidden terminal problem [9] when two senders that cannot sense each other transmit simultaneously to a common receiver.

Wireless broadcast, on the other hand, can also be exploited to increase throughput and improve reliability of wireless networks. At the physical layer, packet collision results in the sum of the two colliding signals. With advanced information theory and signal processing techniques such as successive interference cancelation [22] and network coding [7], the collided signals can be recovered. But simultaneous signal transmissions increase the spectrum efficiency. Wireless broadcast can also be beneficial at higher layers. For example, in Fig. 1.1, the source sends a packet. Its neighbors A, B, and C can receive the packet with the probability labeled beside each node. The random variables describing these events are independent. We can see that the probability that none of the neighbors receives the packet is only $0.1 \times 0.5 \times 0.8 = 0.04$. If all the neighbors have a route to the destination, all of them can be possible next hops for the packet. Protocols exploiting such diversity can improve the reliability of wireless networks. Improved reliability also implies a reduced number of transmissions and hence power saving.

1.2 Network Coding: A New Communication Paradigm

Network coding is an important technique to exploit wireless broadcast and achieve secure communications. In today's packet networks, each node's functions are limited to the forwarding or replication of received packets. Network coding generalizes network operation beyond traditional routing. Each network node is allowed to perform arbitrary operations on signals from different incoming links. It has been shown that the ability of the network to transfer information can be significantly improved [7] by using network coding. The first example showing the usefulness of network coding was given in [7], which is replicated in Figure 1.2. This example shows that coding within the network may be necessary in order to achieve the maximum possible multicast transmission rate.

Since the introduction of network coding in [7], network coding has attracted significant interest from various research communities. A large body of research has focused on the multicast network coding problem where a source needs to deliver the same packets to a set of receivers. It was shown in [7] that the capacity of the network is equal to the size of the minimum cut that separates the source and any terminal. In a subsequent work, Li *et al.* [54] proved that linear network codes are sufficient to achieve the capacity of the network. An algebraic framework for linear network codes on directed graphs was developed by Koetter and Médard [50]. This framework was used by Ho *et al.* [39, 40] to construct random distributed network coding, which achieves the network capacity with probability exponentially approaching 1 with the code length. Jaggi *et al.* [45] proposed a polynomial-time algorithm for systematically finding feasible network codes. All these papers treat wired networks.

For wireless networks with coding done only across packets within the same session (intrasession network coding), the rate stability region for a wireless network with and

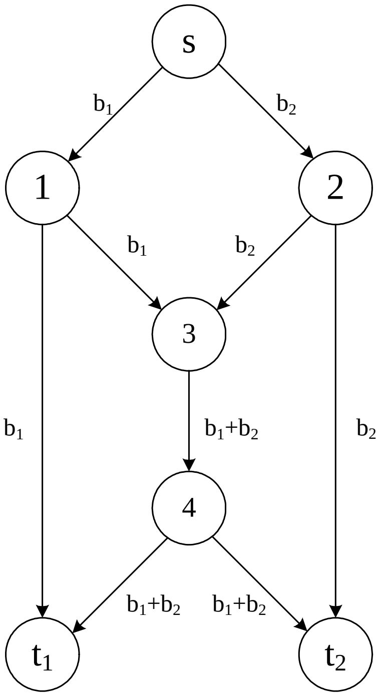


Figure 1.2: An example of a wired network requiring coding to achieve multicast capacity. The network consists of directed unit capacity links, and a source node s multicasting the same information to two receivers t_1 and t_2. b_1 and b_2 are two symbols to be multicast chosen from a finite field of size greater than 2. The presence of the bottleneck link from 3 to 4 necessitates coding on that link in order to achieve the multicast rate 2.

without correlated sources is characterized in [42]. The rate control problem is studied in [18]. Opportunistic XOR coding, which allows coding between packets across different sessions (intersession network coding), is proposed in [48]. Constructive XOR coding across pairs of unicasts is considered in [77] using a linear optimization approach. Dynamic backpressure is applied in [26,41]. A typical example to highlight the utility of network coding in wireless networks is given in Fig. 1.3, where node 1 and node 2 want to send a packet to each other with the help of relay node 3. There does not exist a direct link between node 1 and node 2. By using network coding at the networking layer, the relay node broadcasts the XOR of the packets received from two terminals, which reduces the number of transmissions.

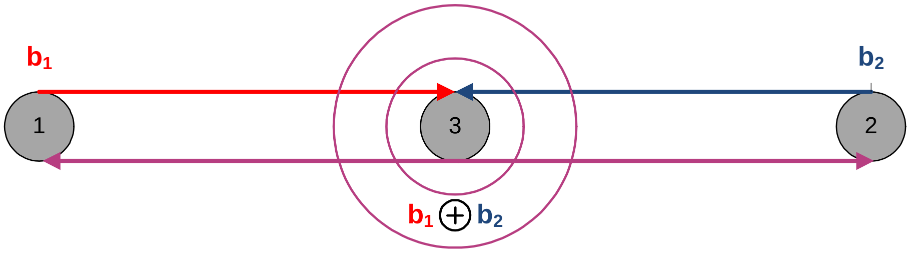


Figure 1.3: An example of a wireless network using coding to improve spectrum efficiency. Node 1 and node 2 want to send a packet to each other with the help of relay node 3. There does not exist a direct link between node 1 and node 2. Without network coding, four transmissions are required, while only three transmissions are needed when the relay node broadcasts the XOR of the packets received from two terminals.

1.3 Outline and Contributions

Each of the problems studied in this thesis demonstrates an aspect of coding benefit in a network scenario of practical relevance. We show that wireless broadcast opens up powerful new ways to consider and approach a number of theoretical and practical wireless networking issues. The outline and contributions are as follows.

In Chapter 2, we consider using network coding at the physical layer to improve spectrum efficiency. We propose relaying strategies for uncoded two-way relay channels motivated by the example in Fig. 1.3, where two terminals transmit simultaneously to each other with the help of a relay. Different from Fig. 1.3 where network coding is applied at the networking layer assuming an error free packet is supplied by the physical layer, we consider general mappings from the relay's received signal to its transmitted signal at the physical layer. For binary antipodal signaling, a class of so called absolute (abs)-based schemes is proposed in which the processing at the relay is solely based on the absolute value of the received signal. We analyze and optimize the symbol-error performance of existing and new abs-based and non-abs-based strategies, including abs-based and non-abs-based versions of amplify and forward (AF), detect and forward (DF), and estimate and forward (EF). Additionally, we optimize the relay function via functional analysis such that the average probability of error is minimized in high signal-to-noise ratio (SNR) regime. The optimized function behaves like abs-AF at low SNR and like abs-DF at high SNR, respectively; EF behaves similarly to the optimized function over the whole SNR range. We find the conditions under which each class of strategies is preferred. All these results are also generalized to higher order constellations, where finding the relay mapping is converted to a graph coloring problem.

In Chapter 3, we consider cross-layer optimization in wireless networks with wireless broadcast, focusing on the problem of distributed scheduling of broadcast links. The goal is to integrate various protocol layers into a unified framework to take advantage of wireless broadcast. This framework integrates rate control, network coding and scheduling at the transport, network and link layers. The link scheduling problem, however, requires a centralized controller. Under the primary interference model, the link scheduling problem is equivalent to a maximum weighted hypergraph matching problem that is NP-complete. To solve the scheduling problem distributedly, locally greedy and randomized approximation algorithms are proposed and shown to have bounded worst-case performance. With random network coding, we obtain a fully distributed cross-layer design. Numerical results show promising throughput gain using the proposed algorithms, and surprisingly, in some cases even with less complexity than cross-layer design without the broadcast advantage.

In Chapter 4, we develop a new class of medium access control (MAC) protocol, which allows each user to transmit at different data rates chosen randomly from an appropriately determined set of rates. Different from traditional protocols which use a collision model and consider packet collision due to wireless broadcast as harmful, multiple packets can be received simultaneously in the proposed method by using successive interference cancelation. In slotted Aloha type Gaussian networks, we show that the achievable total throughput of the proposed protocol is at least a constant fraction of the centralized multiple access channel sum rate when the number of transmission rates at each node is equal to the number of users in the network. We also study the case when only a limited number of transmission rates is available at each node. To achieve the desired throughput-optimal equilibrium, a game-theoretic framework is proposed. We study the design of random access games, characterize their equilibria, study their dynamics, and propose distributed algorithms to achieve the equilibria. This provides a valuable perspective to understand existing MAC protocols and a general framework to design new ones to improve the system performance. Extension to rate splitting is also discussed. Simulation results show that the proposed protocol can achieve a significant throughput gain over conventional Aloha. In a single cell WLAN, the proposed protocol not only achieves a higher throughput but also provides a better short term fairness.

In Chapter 5, we consider the problem of secure transmission in the presence of a wiretapper. Secure multicast network coding over erasure networks with unequal link capacities is studied in the presence of a wiretapper that can wiretap any subset of k links. Existing results show that for the case of equal (unit) link capacities, the secrecy capacity is the same whether or not the location of the wiretapped links is known, and can be achieved by injecting k random keys at the source which are decoded at the sink along with the message. In contrast, we show that for unequal link capacities, the secrecy capacity is not the same in general when the location of the wiretapped links is known as when it is unknown. We give achievable strategies where random keys are canceled at intermediate non-sink nodes, or injected at intermediate non-source nodes. Furthermore, we show that computing the secrecy rate is NP-complete both when the location of the wiretapped links is known and when it is unknown.

We conclude the thesis in Chapter 6 with a discussion of the preceding results and some directions for further work.

Chapter 2

Memoryless Relay Strategies for Two-Way Relay Channels

In this chapter, we consider network coding design at the physical layer for two-way relay channels. Several relaying strategies are proposed and optimized to improve spectrum efficiency.

2.1 Introduction

Two-way communication is a common scenario where two parties simultaneously transmit information to each other. The two-way channel was first considered by Shannon [72], who derived inner and outer bounds on the capacity region. Recently, the two-way relay channel (TWRC) has drawn renewed interest from both academic and industrial communities [24, 37, 52, 65, 67, 68, 83] due to its potential application to cellular networks and peer-to-peer networks. AF and DF protocols for one-way relay channels are extended to the half-duplex Gaussian TWRC in [68] and the general full-duplex discrete TWRC in [67]. In [37], network coding is used to increase the sum-rate of two users. With linear network coding, each node in a network is allowed to perform algebraic operations on received packets instead of only forwarding or replicating received packets. Most of these works [37, 67, 68] focus on capacity bounds for strategies similar to those for one-way relay channels [51]. Furthermore, physical layer network coding (PNC) is considered in [83] for two-way AWGN relay channels. Also, two partial detect and forward (PDF) schemes are proposed in [24] for distributed space time coding to achieve diversity in two-way relay fading channels with multiple relays. A variety of works [24, 83] propose new relaying strategies without addressing their optimality.

In this chapter, we consider an uncoded scenario with memoryless relays which is beneficial in those situations when the relay is under a strict complexity or latency constraint. The former case applies, for example, if the relay is part of a sensor network with battery powered nodes, where the complexity for relaying the partner nodes' data must be kept small. Also, minimizing the end-to-end delay in networked communication is important in real-time applications with feedback, where typically a bidirectional unicast session is established.

In the following work, we analyze and optimize the symbol error probability at each receiver without considering the effect of any end-to-end channel coding that may be applied. We first derive the symbol error probabilities for existing amplify and forward (AF) and detect and forward (DF) schemes for TWRCs using binary antipodal signaling. Noting the performance limitations of these existing schemes, we develop a number of new schemes. We classify both existing and new schemes into two categories: absolute (abs)-based schemes, where the relay transmits an instantaneous function of the *absolute value* of the received signal, and non-abs-based schemes where the sign of the received signal is preserved by the instantaneous relay function. The advantage of abs-based schemes is that for binary antipodal signaling at the terminals the relay performs a constellation compression such that the transmitted signal from the relay is again an antipodal signal with only two constellation points. In fact, the abs-based scheme bears resemblance to network coding where the relay performs an XOR on the decoded data from the terminals [48]. However, in an abs-based scheme the relay receives the real-valued sum of the data from the two terminals plus noise on the physical layer, whereas in network coding the addition is performed over a finite field on the network layer. In contrast to abs-based schemes, in the case of binary antipodal signaling non-abs-based schemes require the relay to transmit four constellation points, which may lead to a larger transmit power and higher decoding complexity. However, as we will see, the relative performance of abs- and non-abs-based schemes depends on the characteristics of the channels between terminals and relay.

Specifically, the abs-based schemes include an abs-based AF (AAF) scheme, an abs-based DF (ADF) scheme and a novel estimate and forward (EF) strategy by extending the EF scheme in [33] for the one-way relay channel to TWRCs, all of which can substantially outperform existing schemes. Besides characterizing the performance of different schemes, we also optimize the relay strategy within the class of abs-based strategies via functional analysis, where the solution minimizes the average probability of error at the terminals[1] over all possible relay functions at high SNR, and generally outperforms all other strategies we consider. This approach can be seen as a generalization of the result from [4] for the one-way case. The optimized relay function is shown to be a Lambert W function parameterized on the noise power and the transmission energy. Interestingly, the optimized function looks like the AAF scheme at low SNR and like the ADF scheme at high SNR. The EF strategy leads to a relay function which is similar in shape to the optimized function in all SNRs. We also prove that DF performs better than ADF if the two-way channel is very asymmetric or the relay has greater power than the two terminals, while ADF performs better than DF in more symmetrical channels or when the relay has roughly the same power as the terminals. These results will also be generalized to higher order constellations at the terminals such as quadrature amplitude modulation (QAM).

Notation: In the following, $p_X(x)$ denotes the probability density function (pdf) of a random variable X, and $\mathcal{G}(x, \sigma^2) \triangleq \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{x^2}{2\sigma^2}\right)$ denotes the pdf of a normal random variable X with mean 0 and variance σ^2. $Q(\cdot)$ represents the Q-function.

2.2 System Model

The system model is illustrated in Fig. 2.1, where the X_i are the transmitted symbols from some given constellation at terminal i, $i = 1, 2$, Y_i are the received symbols at the

[1] An alternative objective would be to minimize the maximum of the two terminals' error probabilities, which gives the same result in high SNR, but is in general more complicated to work with mathematically.

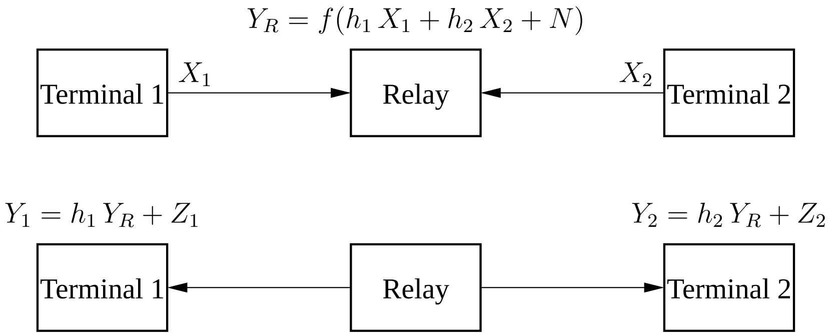


Figure 2.1: Two-way relay channel.

terminals, and Y_R is the transmitted symbol at the relay. Communication takes place in two phases. In the multiple-access (MAC) phase, both terminals simultaneously send a block of data symbols to the relay, which generates $Y_R = f(h_1 X_1 + h_2 X_2 + N)$ with the relay function $f(\cdot)$. Here, h_1 and h_2 represent deterministic attenuation factors for the terminal-to-relay and relay-to-terminal channels, which could for example represent a single realization of a fading process. Throughout this chapter, we assume that $h_1 \geq h_2 \geq 0$ without loss of generality. The quantity N represents the additive white Gaussian noise (AWGN) at the relay with mean zero and variance σ_r^2. In the broadcast phase, the relay transmits Y_R to both terminals 1 and 2. Let Z_i be the AWGN at terminal i with mean zero and variance $\sigma_{s_i}^2$. The discrete-time model for the TWRC can therefore be written as

$$Y_i = h_i\, f(h_1 X_1 + h_2 X_2 + N) + Z_i, \quad i = 1, 2. \tag{2.1}$$

For the sake of brevity, we also define the received signal at the relay as $U = h_1 X_1 + h_2 X_2 + N$. Since each terminal knows what it has sent to the relay in the MAC phase, it can recover the information from the other terminal based on the received Y_i and its own *a priori* knowledge of X_i. In addition, we impose an average power constraint on X_i: $E\{|X_i|^2\} \leq P_s, \quad i = 1, 2$, as well as on the output of the relay: $E\{|f(h_1X_1 + h_2X_2 + N)|^2\} \leq P_r$.

We assume for notational simplicity that the noise variance at the two terminals is the same, i.e., $\sigma_{s_1}^2 = \sigma_{s_2}^2 = \sigma_s^2$; extensions to the more general case are straightforward.

Also, it is assumed that the terminals and the relay know h_1 and h_2, which may be obtained by using channel estimation at the relay or the feedback channel from the two terminals, see e.g., [23]. Further, we assume that the two terminals are perfectly synchronized and compensate for channel phase prior to transmission. Under these assumptions, the channel coefficients h_1 and h_2 are used as real-valued attenuation factors. Alternatively, the synchronization approach from [47] could be applied at the terminals; In this approach, pilot symbols are used to estimate the phase differences between the two terminal signals in the signal received from the relay.

We focus on symbol error probability as a performance metric: each terminal is assumed to perform a hypothesis test to decide which symbol was transmitted by the other terminal; we do not consider the effect of any end-to-end channel coding that may be applied. Note that (2.1) applies to both a half duplex system with two time slots, where the transmission from one terminal to the other takes place in a multiple-access and a broadcast time slot, or a full duplex system.

2.3 Relay Strategies for the BPSK Case

We begin by considering BPSK; an extension to higher order constellations is given in Section 2.5. Each terminal transmits $X_i = \pm\sqrt{P_s}$. We consider two classes of relay strategies: absolute value strategies, where the relay transmits a non-decreasing function of $|U|$, and non-absolute value strategies, where the relay transmits an odd non-decreasing function of U.

We first show that the error probability is minimized if the terminals employ threshold detection as follows.

- For non-abs-based strategies: If $x_i = \sqrt{P_s}$ has been sent in the MAC phase then terminal i decodes to $\sqrt{P_s}$ if $y_i \geq v_i$ and $-\sqrt{P_s}$ otherwise, where v_i is its detection threshold and y_i is the value of its received symbol Y_i. Likewise, if $x_i = -\sqrt{P_s}$ has been sent, then terminal i decodes to $\sqrt{P_s}$ if $y_i \geq -v_i$ and on $-\sqrt{P_s}$ otherwise.

- For abs-based strategies: each terminal decodes to either ($X_1 = \sqrt{P_s}$, $X_2 = \sqrt{P_s}$) if $X_i = \sqrt{P_s}$ or ($X_1 = -\sqrt{P_s}$, $X_2 = -\sqrt{P_s}$) if $X_i = -\sqrt{P_s}$, if $Y_i > v_i$. Otherwise, if the received signal is smaller than the threshold v_i receiver i decodes to ($X_1 = \sqrt{P_s}$, $X_2 = -\sqrt{P_s}$) or ($X_1 = -\sqrt{P_s}$, $X_2 = \sqrt{P_s}$), depending on the value of X_i.

Theorem 2.1 *When each terminal transmits $\sqrt{P_s}$ and $-\sqrt{P_s}$ with equal probability, for any given non-abs-based relay function $f(U)$ or abs-based relay function $f(|U|)$ where f is a non-decreasing function of U or $|U|$, respectively, threshold detection at the terminals minimizes the probability of error.*

The proof is given in the Appendix in Section 2.9.

2.3.1 Non-Abs-Based Strategies

The average probability of error at terminal 1 is

$$\begin{aligned}
P_e^{(1)} =& \frac{1}{4}\Big(\Pr(y_1 < v_1 | x_1 = x_2 = \sqrt{P_r}) + \Pr(y_1 > v_1 | x_1 = \sqrt{P_r}, x_2 = -\sqrt{P_r}) \\
&+ \Pr(y_1 < -v_1 | x_1 = -\sqrt{P_r}, x_2 = \sqrt{P_r}) + \Pr(y_1 > -v_1 | x_1 = x_2 = -\sqrt{P_r})\Big) \\
=& \frac{1}{2} + \frac{1}{2}\int_{-\infty}^{+\infty} \left(\mathcal{G}\left(u - (h_1+h_2)\sqrt{P_s}, \sigma_r^2\right) - \mathcal{G}\left(u - (h_1-h_2)\sqrt{P_s}, \sigma_r^2\right)\right) \\
&\times \left[\int_{-\infty}^{v_1} \mathcal{G}\left(y - h_1 f(u), \sigma_s^2\right) dy\right] du.
\end{aligned} \tag{2.2}$$

By symmetry, the average probability of error at terminal 2 is given by interchanging subscripts $_1$ and $_2$.

2.3.1.1 Amplify-and-Forward

We analyze the performance of amplify and forward [68], where a linear function $f(\cdot)$ is used. To satisfy the average power constraint at the relay, $f(\cdot)$ is equal to $f(u) = \sqrt{\frac{P_r}{(h_1^2+h_2^2)P_s+\sigma_r^2}}u$. The resulting output at terminal i is

$$Y_i = h_i\sqrt{\frac{P_r}{(h_1^2+h_2^2)P_s+\sigma_r^2}}(X_1+X_2) + \left(h_i\sqrt{\frac{P_r}{(h_1^2+h_2^2)P_s+\sigma_r^2}}N + Z_i\right), \; i = 1, 2. \tag{2.3}$$

Therefore, when x_1 and x_2 are transmitted, the conditional pdf of the output Y_i is

$$p_{Y_i|X_1,X_2}(y_i|x_1,x_2) = \mathcal{G}\left(y_i - h_i\sqrt{\frac{P_r}{(h_1^2+h_2^2)P_s+\sigma_r^2}}(x_1+x_2), \frac{h_i^2 P_r \sigma_r^2}{(h_1^2+h_2^2)P_s+\sigma_r^2} + \sigma_s^2\right), \tag{2.4}$$

where $\mathcal{G}(x,\sigma^2)$ is defined at the end of Section 2.1. Given x_i, we observe from (2.4) that terminal i's decoding threshold is $v_i = h_i\sqrt{\frac{P_r}{(h_1^2+h_2^2)P_s+\sigma_r^2}}x_i$. Therefore, the average probability of error at terminal i, $i = 1, 2$ is

$$P_e^{(i)} = Q\left(\sqrt{\frac{h_i^2 P_r P_s}{h_i^2 P_r \sigma_r^2 + (h_1^2+h_2^2)P_s\sigma_s^2 + \sigma_r^2\sigma_s^2}}\right). \tag{2.5}$$

2.3.1.2 Detect-and-Forward

In DF the relay performs hard decisions and maps each decision region to a fixed value that it transmits, i.e.,

$$f(u) = \begin{cases} a, & \text{if } u \geq w, \\ b, & \text{if } w > u \geq 0, \\ -f(-u), & \text{otherwise,} \end{cases} \tag{2.6}$$

The error probability at the terminals is optimized over the relay threshold w, relay transmit values a and b, and the terminal detection thresholds v_1 and v_2, subject to the average power constraint at the relay. Substituting (2.6) into (2.2), the average probability of error at terminal 1 can be written as

$$\begin{aligned} P_e^{(1)} =& \frac{1}{2} + \frac{1}{2}\int_0^w A(u)du \underbrace{\left[\int_{-\infty}^{v_1} \mathcal{G}\left(y - h_1 b, \sigma_s^2\right) dy\right]}_{C(v_1,b)} + \frac{1}{2}\int_w^{+\infty} A(u)du \underbrace{\left[\int_{-\infty}^{v_1} \mathcal{G}\left(y - h_1 a, \sigma_s^2\right) dy\right]}_{D(v_1,a)} \\ &+ \frac{1}{2}\int_0^w B(u)du \underbrace{\left[\int_{-\infty}^{v_1} \mathcal{G}\left(y + h_1 b, \sigma_s^2\right) dy\right]}_{E(v_1,b)} + \frac{1}{2}\int_w^{+\infty} B(u)du \underbrace{\left[\int_{-\infty}^{v_1} \mathcal{G}\left(y + h_1 a, \sigma_s^2\right) dy\right]}_{F(v_1,a)}. \end{aligned} \tag{2.7}$$

where

$$\begin{aligned} A(u) &\triangleq \mathcal{G}\left(u-(h_1+h_2)\sqrt{P_s},\sigma_r^2\right)-\mathcal{G}\left(u-(h_1-h_2)\sqrt{P_s},\sigma_r^2\right), \\ B(u) &\triangleq \mathcal{G}\left(u+(h_1+h_2)\sqrt{P_s},\sigma_r^2\right)-\mathcal{G}\left(u+(h_1-h_2)\sqrt{P_s},\sigma_r^2\right). \end{aligned} \tag{2.8}$$

Taking the partial derivative of $P_e^{(1)} + P_e^{(2)}$ with respect to w and setting this to zero, we obtain

$$\frac{\partial(P_e^{(1)}+P_e^{(2)})}{\partial w} = A(w)\left(C(v_1,b)-D(v_1,a)\right)+B(w)\left(E(v_1,b)-F(v_1,a)\right)+\frac{\partial P_e^{(2)}}{\partial w}=0. \tag{2.9}$$

As the optimal solution of w in (2.9) depends on a, b, v_1, v_2 in a complicated way, it is hard to solve (2.9) directly. One way to approximate the optimal solution is to use an iterative method. At the beginning of the k-th iteration, assuming that $w^{(k)}$ is given ($w^{(0)} = h_1\sqrt{P_s}$), we can optimize $a^{(k)}, b^{(k)}, v_1^{(k)}, v_2^{(k)}$ as follows. When $w^{(k)}, a^{(k)}, b^{(k)}$ are given, $v_1^{(k)}, v_2^{(k)}$ can be written as a function of $a^{(k)}, b^{(k)}$ by minimizing the average error probability. Finally, we perform a two dimensional search over $a^{(k)}, b^{(k)}$. Then, $w^{(k+1)}$ can be obtained from (2.9) by using $a^{(k)}, b^{(k)}, v_1^{(k)}, v_2^{(k)}$. The process repeats until convergence or the maximum number of iterations is achieved. In our experiments, fewer than five iterations were required before convergence. Even though this process does not guarantee convergence to the global minimum, it seems to work well in our experiments.

Alternatively, at high SNR, when $w = h_1\sqrt{P_s}$, we have $A(w) = 0$ and the other Q function and Gaussian terms in (2.9) tend to 0. A suboptimal solution to (2.9) can be approximated with this w. By substituting $w = h_1\sqrt{P_s}$ into (2.7), taking the partial derivative of (2.7) with respect to v_1, and setting the resulting equation to zero we obtain the thresholds

$$v_1 = \frac{h_1(a+b)}{2}, \quad v_2 = \frac{h_2(a-b)}{2}. \tag{2.10}$$

We can then derive the optimal a and b subject to the power constraint at the relay by substituting (2.10) into (2.7). From the resulting expression, by discarding small terms at high SNR, we can then show that $P_e^{(1)} + P_e^{(2)}$ can be approximated[2] as

$$P_e^{(1)}+P_e^{(2)} \approx Q\left(\frac{h_1(a-b)}{2\sigma_s}\right)+Q\left(\frac{h_2(a+b)}{2\sigma_s}\right)+Q\left(\frac{h_2\sqrt{P_s}}{\sigma_r}\right)\left(1+\frac{1}{2}Q\left(\frac{h_2(3b-a)}{2\sigma_s}\right)\right). \tag{2.11}$$

To find the optimal a, b, we need to minimize (2.11) subject to $a^2+b^2=2P_r$. Whether the first two terms or the third term dominates depends on the relative values of $P_r, P_s, \sigma_r, \sigma_s, h_1$ and h_2. If we optimize the first two terms of (2.11), we find that

$$\frac{b}{a} = \frac{h_1-h_2}{h_1+h_2}, \quad a^2+b^2 = 2P_r. \tag{2.12}$$

Substituting (2.12) back into (2.11), we obtain

$$P_e^{(1)}+P_e^{(2)} \approx 2Q\left(\sqrt{\frac{P_r}{h_1^2+h_2^2}}\frac{h_1h_2}{\sigma_s}\right)+Q\left(\frac{h_2\sqrt{P_s}}{\sigma_r}\right)\left(1+\frac{1}{2}Q\left(\sqrt{\frac{P_r}{h_1^2+h_2^2}}\frac{h_2(h_1-2h_2)}{\sigma_s}\right)\right). \tag{2.13}$$

Note that (2.12) agrees with the straightforward DF, where the relay first finds a point from the set $\{-h_1-h_2, -h_1+h_2, h_1-h_2, h_1+h_2\}$ with the minimum Euclidean distance from the received signal and then transmits a scaled version of this point.

If we optimize the third term of (2.11), we find that

$$a = \sqrt{\frac{9P_r}{5}}, \quad b = \sqrt{\frac{P_r}{5}}. \tag{2.14}$$

Substituting (2.14) back into (2.11), we obtain

$$P_e^{(1)} + P_e^{(2)} \approx Q\left(\sqrt{\frac{P_r}{5}}\frac{h_1}{\sigma_s}\right) + Q\left(\sqrt{\frac{4P_r}{5}}\frac{h_2}{\sigma_s}\right) + \frac{5}{4}Q\left(\frac{h_2\sqrt{P_s}}{\sigma_r}\right). \tag{2.15}$$

Note that (2.14) corresponds to the uniform constellation where the distances between any two adjacent constellation points are identical. Comparing (2.13) with (2.15), we find that when $\sqrt{\frac{5P_s\sigma_s^2}{P_r\sigma_r^2}} < \frac{h_1}{h_2} < 2$ we should choose (2.14), which means that the first two terms in (2.11) dominate; otherwise, (2.12) is preferred which means the third term in (2.11) dominates.

[2] Actually $\max(P_e^1, P_e^2)$ dominates, which means that at high SNR optimizing $P_e^{(1)} + P_e^{(2)}$ yields the same function as optimizing $\max(P_e^1, P_e^2)$.

When $h_1 = h_2$, (2.12) leads to

$$a = \sqrt{2P_r}, \quad b = 0, \tag{2.16}$$

where the relay decodes only three points as $h_1 - h_2 = 0$.

Numerical simulations in Section 2.7 reveal that when h_1/h_2 is close to one, (2.16) performs better than both (2.12) and (2.14) where a performance close to the optimal solution is obtained. As h_1/h_2 increases, (2.12) and (2.14) outperform (2.16) at high SNR. But (2.16) still performs better than (2.12) and (2.14) at low SNR, where removing a constellation point results in power savings and performance improvements.

2.3.1.3 Estimate-and-Forward

In this strategy the relay transmits a scaled version of the MMSE estimate of $h_1X_1 + h_2X_2$ given its observation u, i.e., we consider a function

$$\begin{aligned} g(u) &= E\{h_1x_1 + h_2x_2|u\} \\ &= \frac{\sinh\left(\frac{(h_1+h_2)\sqrt{P_s}u}{\sigma_r^2}\right) e^{-\frac{(h_1+h_2)^2P_s}{2\sigma_r^2}}(h_1+h_2) + \sinh\left(\frac{(h_1-h_2)\sqrt{P_s}u}{\sigma_r^2}\right) e^{-\frac{(h_1-h_2)^2P_s}{2\sigma_r^2}}(h_1-h_2)}{\cosh\left(\frac{(h_1+h_2)\sqrt{P_s}u}{\sigma_r^2}\right) e^{-\frac{(h_1+h_2)^2P_s}{2\sigma_r^2}} + \cosh\left(\frac{(h_1-h_2)\sqrt{P_s}u}{\sigma_r^2}\right) e^{-\frac{(h_1-h_2)^2P_s}{2\sigma_r^2}}}\sqrt{P_s} \end{aligned} \tag{2.17}$$

and set the relay function $f(u)$ to be a scaled version of $g(u)$ to satisfy the power constraint. We find that $g(u)$ in (2.17) is close to the straightforward DF (2.12) at high SNR.

2.3.1.4 Optimized Relay Function

The optimal relay function minimizes the sum of average probabilities of both terminals subject to the average power constraint, i.e.,

$$\begin{aligned}
G(f) &= P_e^{(1)} + P_e^{(2)} \\
=&1 + \frac{1}{2}\int_{-\infty}^{+\infty} \left(\mathcal{G}\left(u-(h_1+h_2)\sqrt{P_s},\sigma_r^2\right) - \mathcal{G}\left(u-(h_1-h_2)\sqrt{P_s},\sigma_r^2\right)\right) \\
&\qquad\qquad \times \left[\int_{-\infty}^{v_1} \mathcal{G}\left(y-h_1 f(u),\sigma_s^2\right) dy\right] du \qquad (2.18) \\
&+ \frac{1}{2}\int_{-\infty}^{+\infty} \left(\mathcal{G}\left(u-(h_2+h_1)\sqrt{P_s},\sigma_r^2\right) - \mathcal{G}\left(u-(h_2-h_1)\sqrt{P_s},\sigma_r^2\right)\right) \\
&\qquad\qquad \times \left[\int_{-\infty}^{v_2} \mathcal{G}\left(y-h_2 f(u),\sigma_s^2\right) dy\right] du.
\end{aligned}$$

The optimal relay function is the solution of the following problem.

$$\begin{aligned}
&\min_{f,v_1,v_2} G(f) \\
&\text{subject to} \int_0^{+\infty} \mathcal{G}\left(u-(h_1+h_2)\sqrt{P_s},\sigma_r^2\right) f^2(u)du \\
&\quad + \int_0^{+\infty} \mathcal{G}\left(u-(h_1-h_2)\sqrt{P_s},\sigma_r^2\right) f^2(u)du \qquad (2.19) \\
&\quad + \int_0^{+\infty} \mathcal{G}\left(u+(h_1+h_2)\sqrt{P_s},\sigma_r^2\right) f^2(u)du \\
&\quad + \int_0^{+\infty} \mathcal{G}\left(u+(h_1-h_2)\sqrt{P_s},\sigma_r^2\right) f^2(u)du = 2P_r.
\end{aligned}$$

To solve the functional optimization problem (2.19), we first fix v_1 and v_2 and derive the relay function as a function of v_1 and v_2 via the Lagrange dual. Then the relay function is substituted into the objective function and the resulting equation is minimized over v_1 and v_2 by performing a line search around v_1 and v_2 in the optimal DF strategy. Since we do not have a convex optimization problem, the obtained solution may be a local optimum. The closed-form solution of (2.19) is hard to obtain. Nevertheless, we plot the optimized non-abs-based relay function at different SNRs and with different h_1 and h_2 in Fig. 2.2.

2.3.2 Abs-Based Strategies

In this subsection, we consider abs-based strategies, where in particular, we will provide detailed derivations for the special case $h_1 = h_2 = 1$. The derivations for the general case $h_1 > h_2$ are analogous and will only be briefly discussed due to space limitations. As a starting point for the following discussions, we note that generally for abs-based schemes the average error probability at each terminal i can be written as

$$P_e = \frac{1}{2}\Pr(y > v_i | x_1 \neq x_2) + \frac{1}{2}\Pr(y < v_i | x_1 = x_2). \tag{2.20}$$

For $h_1 = h_2$, we have $v_1 = v_2 = v$.

2.3.2.1 Abs-Based Amplify-and-Forward (AAF)

In this scheme, the relay first takes the absolute value of the received signal and then subtracts a positive constant C from the resulting signal, i.e.,

$$f(u) = \beta\left(|u| - C\right), \tag{2.21}$$

where β is a coefficient to maintain the average power constraint at the relay. From (2.20), the average error probability at terminal 1 for $h_1 = h_2 = 1$ can be written as

$$\begin{aligned} P_e =& \frac{1}{2}\left(1 + \int_0^{+\infty}\left(\mathcal{G}\left(u - 2\sqrt{P_s}, \sigma_r^2\right) + \mathcal{G}\left(u + 2\sqrt{P_s}, \sigma_r^2\right) - 2\mathcal{G}\left(u, \sigma_r^2\right)\right)\right. \\ & \left. \times \left[\int_{-\infty}^{v} \mathcal{G}\left(y - \beta\left(u - C\right), \sigma_s^2\right) dy\right] du\right). \end{aligned} \tag{2.22}$$

The optimal solution is given by minimizing (2.22) with respect to both v and C, which is done numerically since an analytical solution is hard to obtain. The optimal solution depends on the SNR values, but we have observed experimentally that the optimal threshold is very close to zero. So, a simple solution, in particular if the SNR is not accurately known, is to set $v = 0$ and $C = h_1\sqrt{P_s}$ or $C = h_1\sqrt{P_s} + \sigma_r/\sqrt{2}$.

2.3.2.2 Abs-Based Detect-and-Forward (ADF)

In ADF, the relay performs hard decisions, based on the absolute value of the received signal, to decide whether $2\sqrt{P_s}$, 0, or $-2\sqrt{P_s}$ is received. The relay does not actually detect x_1 and x_2, but only the mixture $h_1x_1 + h_2x_2$. To satisfy the relay's average power constraint, $\sqrt{P_r}$ and $-\sqrt{P_r}$ are transmitted, i.e.,

$$f(u) = \begin{cases} \sqrt{P_r}, & \text{if } |u| \geq w, \\ -\sqrt{P_r}, & \text{otherwise,} \end{cases} \tag{2.23}$$

where w is a threshold which will be determined below. Note that a related detect-and-forward scheme for the TWRC is already proposed in [83] as physical layer network coding. In the following, we extend this work by providing a detailed analysis of the end-to-end error probability.

For the case $h_1 = h_2 = 1$ the average error probability at each terminal (2.20) can be written as

$$\begin{aligned} P_e = \frac{1}{2} + \frac{1}{2}\int_0^w &\left(\mathcal{G}\left(u - 2\sqrt{P_s}, \sigma_r^2\right) + \mathcal{G}\left(u + 2\sqrt{P_s}, \sigma_r^2\right) - 2\mathcal{G}\left(u, \sigma_r^2\right)\right) du \\ &\times \int_{-\infty}^{v} \left(\mathcal{G}\left(y + \sqrt{P_r}, \sigma_s^2\right) - \mathcal{G}\left(y - \sqrt{P_r}, \sigma_s^2\right)\right) dy. \end{aligned} \tag{2.24}$$

Eq. (2.24) has the nice property that the optimization with respect to w and v is separated. By minimizing (2.24) over w and v we find that the optimal value of w is

$$w = \sqrt{P_s}\left(1 + \frac{\sigma_r^2}{2P_s}\log\left(1 + \sqrt{1 - e^{-4P_s/\sigma_r^2}}\right)\right), \tag{2.25}$$

and the optimal value of v is $v = 0$, which gives

$$P_e = \frac{1}{2} + \frac{1}{2}\left(Q\left(\frac{2\sqrt{P_s} - w}{\sigma_r}\right) + 2Q\left(\frac{w}{\sigma_r}\right) - Q\left(\frac{2\sqrt{P_s} + w}{\sigma_r}\right) - 1\right)\left(1 - 2Q\left(\frac{\sqrt{P_r}}{\sigma_s}\right)\right). \tag{2.26}$$

When $\sigma_r^2 \to 0$ the optimal w converges to $\sqrt{P_s}$. Note that due to the separation of w and v in (2.24), the optimal w also minimizes the error probability of detection at the relay. When $h_1 > h_2$, we obtain $w = h_1\sqrt{P_s}$ at high SNR.

2.3.2.3 Abs-Based Estimate-and-Forward (AEF)

In this strategy the relay transmits its minimum mean squared error (MMSE) estimate of $|h_1x_1+h_2x_2|$. We first address the case $h_1 = h_2 = 1$ and derive the MMSE estimator

$$g(u) = E\left\{|x_1+x_2|\big|u\right\} = \frac{2\sqrt{P_s}\cosh\left(\frac{2\sqrt{P_s}u}{\sigma_r^2}\right)}{e^{2P_s/\sigma_r^2}+\cosh\left(\frac{2\sqrt{P_s}u}{\sigma_r^2}\right)}. \tag{2.27}$$

The relay function $f(u)$ is then a scaled version of $g(u) - C$, i.e.,

$$f(u) = \begin{cases} \beta\left(g(u) - C\right), & \text{if } u \geq 0, \\ f(-u), & \text{otherwise,} \end{cases} \tag{2.28}$$

where C is a constant as in AAF and $\beta \geq 0$ is a scaling factor to satisfy the average power constraint $E\{f^2(u)\} = P_r$. Optimization of the terminal decoding thresholds is similar to that for AAF. Analogous to the above derivation, for $h_1 > h_2$ we obtain $g(u)$ as

$$\begin{aligned} g(u) = &\frac{|h_1+h_2|\sqrt{P_s}e^{-\frac{(h_1+h_2)^2P_s}{2\sigma_r^2}}\cosh\left(\frac{(h_1+h_2)\sqrt{P_s}u}{\sigma_r^2}\right)}{e^{-\frac{(h_1+h_2)^2P_s}{2\sigma_r^2}}\cosh\left(\frac{(h_1+h_2)\sqrt{P_s}u}{\sigma_r^2}\right)+e^{-\frac{(h_1-h_2)^2P_s}{2\sigma_r^2}}\cosh\left(\frac{(h_1-h_2)\sqrt{P_s}u}{\sigma_r^2}\right)} \\ &+\frac{|h_1-h_2|\sqrt{P_s}e^{-\frac{(h_1-h_2)^2P_s}{2\sigma_r^2}}\cosh\left(\frac{(h_1-h_2)\sqrt{P_s}u}{\sigma_r^2}\right)}{e^{-\frac{(h_1+h_2)^2P_s}{2\sigma_r^2}}\cosh\left(\frac{(h_1+h_2)\sqrt{P_s}u}{\sigma_r^2}\right)+e^{-\frac{(h_1-h_2)^2P_s}{2\sigma_r^2}}\cosh\left(\frac{(h_1-h_2)\sqrt{P_s}u}{\sigma_r^2}\right)}. \end{aligned} \tag{2.29}$$

2.3.2.4 Optimized Relay Strategy

In this section, we optimize the average probability of error over even functions $f(\cdot)$ at the relay. Our approach generalizes the result from [4] for the one-way case. For

$h_1 = h_2$ the average probability of error can be obtained from (2.20) as

$$\begin{aligned} P_e(f) =& \frac{1}{2} + \frac{1}{2}\int_0^{+\infty} \underbrace{\left(\mathcal{G}\left(u+2\sqrt{P_s},\sigma_r^2\right) + \mathcal{G}\left(u-2\sqrt{P_s},\sigma_r^2\right) - 2\mathcal{G}\left(u,\sigma_r^2\right)\right)}_{\triangleq B(u)} \\ & \times \underbrace{\left[\int_{-\infty}^{v} \mathcal{G}\left(y-f(u),\sigma_s^2\right)dy\right]}_{\triangleq A(f)} du, \end{aligned} \tag{2.30}$$

which holds since $B(u)$ is an even function in u. Let

$$D(u) \triangleq \mathcal{G}\left(u+2\sqrt{P_s},\sigma_r^2\right) + \mathcal{G}\left(u-2\sqrt{P_s},\sigma_r^2\right) + 2\mathcal{G}\left(u,\sigma_r^2\right). \tag{2.31}$$

Our optimization problem is

$$\min_{f,v}\ H(f) = \int_0^{+\infty} B(u)A(f)du \quad \text{subject to} \quad \frac{1}{2}\int_0^{+\infty} D(u)f^2(u)du \le P_r, \tag{2.32}$$

which can be solved by considering the Lagrangian

$$\phi(\lambda, f) = H(f) + \frac{\lambda}{2}\left(\int_0^{+\infty} D(u)f^2(u)du - 2P_r\right), \tag{2.33}$$

where $\lambda \ge 0$ is the Lagrange multiplier of the average power constraint. Differentiating $\phi(\lambda, f)$ with respect to $f(u)$ for each u and setting the result to zero, we obtain after rearranging

$$\frac{\mathcal{G}\left(f(u)-v,\sigma_s^2\right)}{f(u)} = \lambda\frac{D(u)}{B(u)}. \tag{2.34}$$

Since $\lambda > 0, D(u) > 0$, and if $|u| \ge w$ we have $B(u) \ge 0$ (and $B(u) < 0$ otherwise), we obtain

$$\begin{cases} f(u) \ge 0, & \text{if } |u| \ge w, \\ f(u) < 0, & \text{otherwise,} \end{cases} \tag{2.35}$$

where w is the relay hard decision threshold defined in (2.25).

Lemma 2.2 *For $f(u)$ satisfying*

$$\begin{cases} f(u) \geq v, & \textit{if } |u| \geq w, \\ f(u) < v, & \textit{otherwise}, \end{cases} \tag{2.36}$$

$P_e(f)$ in (2.30) is a strictly convex function in f (when considering functions that differ on a set of non-zero measure).

Proof. Let f and g be two functions satisfying (2.36), and let $\lambda \in [0,1]$ and $\gamma = 1-\lambda$. Clearly, $\lambda f + \gamma g$ also satisfies (2.36). Then,

$$\frac{\partial^2 A(f)}{\partial f^2} = \frac{1}{2\sigma_s^2}\left(f(u) - v\right)\mathcal{G}\left(v - f(u), \sigma_s^2\right), \tag{2.37}$$

is nonnegative if $f(u) \geq v$ and negative otherwise. Since $B(u)\frac{\partial^2 A(f)}{\partial f^2}$ is nonnegative for $|u| \geq w$ and positive otherwise, we have

$$P_e(\lambda f + \gamma g) = \frac{1}{2} + \frac{1}{2}\int_0^{+\infty} B(u)A(\lambda f + \gamma g)du \leq \lambda P_e(f) + \gamma P_e(g).$$

■

If $v = 0$, then (2.34) can be further simplified to be

$$\frac{e^{-\left(f(u)/\sqrt{2\sigma_s^2}\right)^2}}{f(u)/\sqrt{2\sigma_s^2}} = \lambda 2\sqrt{\pi}\sigma_s^2 \frac{\cosh\left(\frac{2\sqrt{P_s}u}{\sigma_r^2}\right) + e^{2P_s/\sigma_r^2}}{\cosh\left(\frac{2\sqrt{P_s}u}{\sigma_r^2}\right) - e^{2P_s/\sigma_r^2}}, \tag{2.38}$$

which can be solved to obtain the following expression for $f(u)$:

$$f(u) = \begin{cases} \sqrt{\sigma_s^2 W\left(\frac{1}{2\pi\lambda^2\sigma_s^4}\left[\frac{\cosh\left(\frac{2\sqrt{P_s}u}{\sigma_r^2}\right) - e^{2P_s/\sigma_r^2}}{\cosh\left(\frac{2\sqrt{P_s}u}{\sigma_r^2}\right) + e^{2P_s/\sigma_r^2}}\right]^2\right)}, & \text{if } u \geq w, \\ -\sqrt{\sigma_s^2 W\left(\frac{1}{2\pi\lambda^2\sigma_s^4}\left[\frac{\cosh\left(\frac{2\sqrt{P_s}u}{\sigma_r^2}\right) - e^{2P_s/\sigma_r^2}}{\cosh\left(\frac{2\sqrt{P_s}u}{\sigma_r^2}\right) + e^{2P_s/\sigma_r^2}}\right]^2\right)}, & \text{if } w > u \geq 0, \\ f(-u), & \text{if } u < 0. \end{cases} \tag{2.39}$$

Here, $W(\cdot)$ denotes the Lambert W function, defined by $W(x)e^{W(x)} = x$, and λ is such that the power constraint is satisfied with equality.

Note that $f(u)$ in (2.39) is derived from the Lagrange dual without any assumption on the convexity of the problem, which may not be a true optimal solution. However, (2.39) indeed satisfies (2.35), which means that it is optimal within the class of functions satisfying (2.35). By Lemma 2.2 and the convexity of $f^2(u)$ in $f(u)$, the set of functions satisfying (2.35) and the power constraint of (2.32) is a convex function set. The optimization under the constraint (2.35) is thus convex, and there is no duality gap. Therefore, (2.39) is the optimal solution when $v = 0$, which can be achieved in the high SNR regime as shown below.

At high SNR, since

$$\lim_{\sigma_r^2 \to 0} \frac{\mathcal{G}\left(u + 2\sqrt{P_s}, \sigma_r^2\right) + \mathcal{G}\left(u - 2\sqrt{P_s}, \sigma_r^2\right) - 2\mathcal{G}\left(u, \sigma_r^2\right)}{\mathcal{G}\left(u + 2\sqrt{P_s}, \sigma_r^2\right) + \mathcal{G}\left(u - 2\sqrt{P_s}, \sigma_r^2\right) + 2\mathcal{G}\left(u, \sigma_r^2\right)} = \begin{cases} 1, & \text{if } |u| \geq w, \\ -1, & \text{if } |u| < w, \end{cases} \tag{2.40}$$

from (2.34) we obtain

$$f(u) = \begin{cases} C_1, & \text{if } |u| \geq w, \\ -C_2, & \text{if } |u| < w, \end{cases} \tag{2.41}$$

where $C_1, C_2 > 0$ are constants. Substituting (2.41) back into (2.34), we find that

$$\frac{\mathcal{G}\left(C_1 - v, \sigma_s^2\right)}{C_1} = \lambda = \frac{\mathcal{G}\left(C_2 + v, \sigma_s^2\right)}{C_2}, \tag{2.42}$$

which gives

$$v = \frac{\log C_1 - \log C_2}{C_1 + C_2}\sigma_s^2 + \frac{C_1 - C_2}{2} \xrightarrow[\sigma_s^2 \to 0]{} \frac{C_1 - C_2}{2}. \tag{2.43}$$

Substituting (2.43) into (2.42), we obtain $C_1 = C_2 = C$, which corresponds to ADF. Hence, λ can be approximated as

$$\lambda = \frac{\mathcal{G}\left(C, \sigma_s^2\right)}{C}. \tag{2.44}$$

Substituting (2.41)-(2.44) into (2.33) and using (2.26), the dual problem then becomes

$$\min_{C,v} \quad Q\left(\frac{C}{\sigma_s}\right) + \frac{\mathcal{G}\left(C, \sigma_s^2\right)}{C}\left(C^2 - P_r\right). \tag{2.45}$$

Note that at high SNR $Q\left(\frac{C}{\sigma_s}\right)$ can be approximated as $\frac{\sigma_s}{\sqrt{2\pi}C}e^{-\frac{C^2}{2\sigma_s^2}}$, which decreases faster than $\mathcal{G}\left(C,\sigma_s^2\right) = \frac{1}{\sqrt{2\pi}\sigma_s}e^{-\frac{C^2}{2\sigma_s^2}}$. Therefore, the minimum of (2.45) is attained at $v = 0$, $C_1 = C_2 = C = \sqrt{P_r}$ when $\sigma_s^2 \to 0$ and $\sigma_r^2 \to 0$. By substituting $v = 0$ and $C_1 = C_2 = C = \sqrt{P_r}$ into (2.41) and (2.44), we obtain f^* and λ^*, which gives $\min_f \phi(\lambda^*, f) = G(f^*)$ at high SNR. Therefore, there is no duality gap at high SNR and the optimal solution converges to (2.41), which is equivalent to the ADF strategy. In general, the optimal v varies with SNR.

For the case $h_1 > h_2$, minimizing the sum of error probabilities of both terminals can be approximated by minimizing the error probability of terminal 2 at high SNR, which gives

$$f(u) = \begin{cases} \sqrt{\sigma_s^2 W\left(\frac{1}{2\pi\lambda^2 h_1^2\sigma_s^4}\left[\frac{e^{-\frac{(h_1+h_2)^2P_s}{2\sigma_r^2}}\cosh\left(\frac{(h_1+h_2)\sqrt{P_s}u}{\sigma_r^2}\right)-e^{-\frac{(h_1-h_2)^2P_s}{2\sigma_r^2}}\cosh\left(\frac{(h_1-h_2)\sqrt{P_s}u}{\sigma_r^2}\right)}{e^{-\frac{(h_1+h_2)^2P_s}{2\sigma_r^2}}\cosh\left(\frac{(h_1+h_2)\sqrt{P_s}u}{\sigma_r^2}\right)+e^{-\frac{(h_1-h_2)^2P_s}{2\sigma_r^2}}\cosh\left(\frac{(h_1-h_2)\sqrt{P_s}u}{\sigma_r^2}\right)}\right]^2\right)}, & \text{if } u \geq w, \\ -\sqrt{\sigma_s^2 W\left(\frac{1}{2\pi\lambda^2 h_1^2\sigma_s^4}\left[\frac{e^{-\frac{(h_1+h_2)^2P_s}{2\sigma_r^2}}\cosh\left(\frac{(h_1+h_2)\sqrt{P_s}u}{\sigma_r^2}\right)-e^{-\frac{(h_1-h_2)^2P_s}{2\sigma_r^2}}\cosh\left(\frac{(h_1-h_2)\sqrt{P_s}u}{\sigma_r^2}\right)}{e^{-\frac{(h_1+h_2)^2P_s}{2\sigma_r^2}}\cosh\left(\frac{(h_1+h_2)\sqrt{P_s}u}{\sigma_r^2}\right)+e^{-\frac{(h_1-h_2)^2P_s}{2\sigma_r^2}}\cosh\left(\frac{(h_1-h_2)\sqrt{P_s}u}{\sigma_r^2}\right)}\right]^2\right)}, & \text{if } w > u \geq 0, \\ f(-u), & \text{if } u < 0. \end{cases} \tag{2.46}$$

Remarks:

- As seen above, $f(u)$ in (2.39) is optimal when the two terminals' detection thresholds are set to zero. Our experiments show that this relay function outperforms the other strategies in both high and low SNR regimes. One way to optimize jointly over $f(u)$ and v is to solve (2.34) for $f(u)$, which depends on both v and λ. For a given v, we can find λ by satisfying the average power constraint. Finally, v can be found by substituting the resulting function into $H(f)$ and optimizing over v. The optimized function using this approach performs better than (2.39) but is more difficult to implement.

- Above, we have derived the error probabilities for various strategies with fixed h_1 and h_2. To obtain the performance in fading channels, we integrate the obtained error probabilities over the joint pdf of h_1 and h_2. Except for the optimized relay function for non-abs strategies, we give closed-form expression for the other cases at least in high SNR.

2.4 Comparison Between Two Classes of Strategies

The average error probability of non-abs DF can be approximated by applying Chernoff bound-type arguments to (2.13) and (2.15), which gives

$$P_e^{(1)} + P_e^{(2)} \approx \begin{cases} \frac{5}{8} e^{-\frac{h_2^2 P_s}{2\sigma_r^2}}, & \text{if } 2 > \frac{h_1}{h_2} > \sqrt{\frac{5P_s\sigma_s^2}{P_r\sigma_r^2}}, \\ e^{-\frac{h_1^2 h_2^2 P_r}{2(h_1^2+h_2^2)\sigma_s^2}} + \frac{1}{2} e^{-\frac{h_2^2 P_s}{2\sigma_r^2}}, & \text{otherwise.} \end{cases} \tag{2.47}$$

Likewise, we can approximate the average error probability of ADF for $h_1 > h_2$ by using Chernoff bounds on (2.24) (and the corresponding expression for terminal 2) according to

$$P_e^{(1)} + P_e^{(2)} \approx \frac{1}{2}\left(e^{-\frac{h_1^2 P_r}{2\sigma_s^2}} + e^{-\frac{h_2^2 P_r}{2\sigma_s^2}} \right) + e^{-\frac{h_2^2 P_s}{2\sigma_r^2}}. \tag{2.48}$$

In the following, we consider several cases at high SNR. Let $\text{SNR}_r \sim \frac{P_s}{\sigma_r^2}$ and $\text{SNR}_s \sim \frac{P_r}{\sigma_s^2}$.

- If $\text{SNR}_s < \text{SNR}_r$, (2.48) is dominated by $\frac{1}{2} e^{-\frac{h_2^2 P_r}{2\sigma_s^2}}$, while (2.47) is dominated by $e^{-\frac{h_1^2 h_2^2 P_r}{2(h_1^2+h_2^2)\sigma_r^2}}$. Therefore, the average error probability for ADF is at most 1/2 of the one for DF.

- If $\text{SNR}_s > \text{SNR}_r$ and $1+\frac{h_2^2}{h_1^2} > \frac{P_r\sigma_r^2}{P_s\sigma_s^2}$ and $h_1 > 2h_2$, (2.48) is dominated by $e^{-\frac{h_2^2 P_s}{2\sigma_r^2}}$, and (2.47) is dominated by $\frac{1}{2}e^{-\frac{h_2^2 P_s}{2\sigma_r^2}}$. In this case, the average error probability for DF is 1/2 of the one for ADF.

- If $\mathrm{SNR}_s > \mathrm{SNR}_r$ and $1+\frac{h_2^2}{h_1^2} < \frac{P_r\sigma_r^2}{P_s\sigma_s^2}$ and $\frac{h_1}{h_2} < \sqrt{\frac{5P_s\sigma_s^2}{P_r\sigma_r^2}}$, (2.48) is dominated by $e^{-\frac{h_2^2 P_s}{2\sigma_r^2}}$, and (2.47) by $\frac{3}{4}e^{-\frac{h_2^2 P_s}{2\sigma_r^2}}$. Hence, the average error probability for DF is 3/4 of the one for ADF.

- If $\mathrm{SNR}_s > \mathrm{SNR}_r$ and $1+\frac{h_2^2}{h_1^2} < \frac{P_r\sigma_r^2}{P_s\sigma_s^2}$ and $2 > \frac{h_1}{h_2} > \sqrt{\frac{5P_s\sigma_s^2}{P_r\sigma_r^2}}$, (2.48) is dominated by $e^{-\frac{h_2^2 P_s}{2\sigma_r^2}}$, and (2.47) is dominated by $\frac{5}{8}e^{-\frac{h_2^2 P_s}{2\sigma_r^2}}$. This leads to an average error probability for DF which is 5/8 of the one for ADF.

These results suggest that when the channel is very asymmetric or the relay has greater power than the terminals we should use DF. When the relay has almost the same power as the terminals, we prefer ADF where the power savings by using the abs-based operation has a big impact on the overall performance. From Section 2.3.1.2, we know that if h_1/h_2 is close to one, DF with (2.16) performs better than DF with (2.12) or (2.14). Therefore, when the channel is symmetric and the relay has greater power than the terminals we should use DF with (2.16).

2.5 Higher Order Constellations

In industry standards such as the IEEE 802.11 series, usually higher order QAM constellations are employed to achieve high spectral efficiency. In the following, we assume $h_1 = h_2 = 1$ for simplicity. A good mapping function $h(u)$ at the relay should have the property that each terminal can detect the other terminal's signal without loss in the noise free case. We therefore require that

$$\begin{aligned} h(u_1+u_2) &\neq h(u_1'+u_2), \quad \forall u_1 \neq u_1' \\ \text{and } h(u_1+u_2) &\neq h(u_1+u_2'), \quad \forall u_2 \neq u_2', \quad u_i, u_i' \in \mathcal{Q}, \end{aligned} \tag{2.49}$$

$i = 1, 2$, where $\mathcal{Q}$ is the constellation set used by the two terminals. The classification of BPSK strategies into absolute and non-absolute value strategies can be generalized to a classification based on underlying relay mappings $h(u)$ satisfying the above condition. We represent condition (2.49) as an undirected graph $\mathcal{G}$, where each node corresponds to a different value of $u_1 + u_2$ and there is an edge between the node corresponding to $u_1 + u_2$ and the node corresponding to $u_1' + u_2$, $u_1' \neq u_1$. Given this graph, a relay function $h(u)$ satisfies condition (2.49) if and only if it corresponds to a vertex coloring of $\mathcal{G}$ such that no two adjacent nodes have the same color. To find the optimal relay function, we need to consider all possible colorings of graph $\mathcal{G}$. For each coloring, the strategies discussed for BPSK in Section 2.3.1 and Section 2.3.2.4 can be generalized using the underlying mapping $h(u)$ as described below. The one achieving the minimum error rate is optimal. The minimum possible constellation size of the relay function is equal to the chromatic number of $\mathcal{G}$.

Another way of finding a feasible relay mapping $h(u)$ is, as above, to consider the sum $u_1 + u_2 = c_i$, $i = 1, \ldots, 2|\mathcal{V}| - 1$, for all $u_1, u_2 \in \mathcal{V}$, where $\mathcal{V}$ denotes the constellation set at the two terminals[3]. The quantity c_i takes elements from the set $\mathcal{W}$, where $|\mathcal{W}| = 2|\mathcal{V}| - 1$. The underlying (noise free) relay mapping $h(u)$ which maps the set $\mathcal{W}$ to a set $\mathcal{V}'$ of size $M \geq |\mathcal{V}|$ containing the constellation set to be received at the terminals, can now be found for every i by assigning the $k = (i \mod M)$-th element of $\mathcal{V}'$ to the values c_i. In principle, the M elements can be picked from $\mathcal{V}'$ in arbitrary order.

Note that rectangular QAM constellations can be easily transmitted as two PAM signals on quadrature carriers. In the following, we only consider PAM constellations, and we take 4-PAM as an example. The approach can be generalized to higher PAM constellations. For simplicity, we assume that the signal transmitted by the terminals is chosen from the constellation set $\mathcal{V} = \{-3, -1, 1, 3\}$. In the absence of noise, the received signal at the relay is from the set $\mathcal{W} = \{-6, -4, -2, 0, 2, 4, 6\}$. We first consider the class of mapping functions that map $\mathcal{W}$ to $\mathcal{V}' = \mathcal{V}$. For example, we can choose

[3] For the sake of simplicity we assume that the two terminals employ the same constellation set.

X_2 X_1	-3	-1	+1	+3
-3	+1	-3	-1	+3
-1	+3	+1	-3	-1
+1	-1	+3	+1	-3
+3	-3	-1	+3	+1

(2.50)

or

X_2 X_1	-3	-1	+1	+3
+3	+3	-3	-1	+1
-1	+1	+3	-3	-1
+1	-1	+1	+3	-3
+3	-3	-1	+1	+3

(2.51)

It is easy to verify that both (2.50) and (2.51) satisfy the condition in (2.49). Note that (2.51) is the physical network coding operation given in [83] using DF.

AAF can be readily generalized by setting the relay function to be a piecewise linear function based on $h(u)$ such as

$$f(u) = \begin{cases} \beta(u+3), & \text{if } u < -3, \\ \beta(u+5), & \text{if } -2 > u \geq -3, \\ \beta(1-u), & \text{if } 1 > u \geq -2, \\ \beta(-1-u), & \text{if } 2 > u \geq 1, \\ \beta(u-5), & \text{if } 5 > u \geq 2, \\ \beta(u-3), & \text{if } u \geq 5, \end{cases} \tag{2.52}$$

where β is a coefficient to maintain the average power constraint at the relay. The detection at each terminal is similar to the traditional 4-PAM demodulation by comparing with some thresholds. ADF can be adapted similarly. The relay defines hard decision regions for u, and sends a scaled/shifted version of $h(u)$. At high SNR, the ADF relay function based on (2.50) can be obtained as

$$f(u) = \begin{cases} -3\beta, & \text{if } u < -5, \\ -\beta, & \text{if } -3 > u \geq -5, \\ 3\beta, & \text{if } -1 > u \geq -3, \\ \beta, & \text{if } 1 > u \geq -1, \\ -3\beta, & \text{if } 3 > u \geq 1, \\ -\beta, & \text{if } 5 > u \geq 3, \\ 3\beta, & \text{if } u \geq 5. \end{cases} \tag{2.53}$$

For EF, we first consider the function $g(u)$ such that

$$g(u) = \underset{g'(u)}{\arg\min}\, E\left\{ \left|h(x_1 + x_2) - g'(u)\right|^2 \middle| u \right\}. \tag{2.54}$$

$f(u)$ is then a scaled version of $g(u)$, i.e., $f(u) = \beta\, g(u)$, where $\beta \geq 0$ is a scalar to satisfy the average power constraint. At the two terminals, there also exists an optimal decision threshold v. We can optimize v using the same approach as in AAF or just choose the conventional 4-PAM detection threshold. In all strategies, we can also apply a maximum likelihood detector at each terminal, giving

$$\hat{x}_2 = \underset{\tilde{x}_2 \in \mathcal{Q}}{\arg\min} \left|y_1 - f\left(x_1 + \tilde{x}_2\right)\right|^2. \tag{2.55}$$

The relay mapping function can also perform a redundant mapping such that $\mathcal{W} = \{-6, -4, -2, 0,\ 2, 4, 6\}$ is mapped to a set $\mathcal{V}'$ with 5, 6, or 7 elements. For example, when $\mathcal{V}' = \{-4, -2, 0, 2, 4\}$, we can choose $h(u)$ as

X_2 X_1	-3	-1	+1	+3
-3	+2	+4	-2	-4
-1	0	+2	+4	-2
+1	-2	0	+2	+4
+3	-4	-2	0	+2

(2.56)

or, when $\mathcal{V}' = \{-5, -3, -1, 1, 3, 5\}$, we can choose $h(u)$ as

X_2 X_1	-3	-1	+1	+3
-3	+1	+3	+5	-5
-1	-1	+1	+3	+5
+1	-3	-1	+1	+3
+3	-5	-3	-1	+1

(2.57)

When $\mathcal{V}' = \mathcal{W}$, we can simply choose $h(u) = u$. It is easy to verify that (2.56) and (2.57) satisfy the condition in (2.49).

2.6 Peak Power Constraint

We have considered an average power constraint at the relay. A peak power constraint at the relay is also common in practice. With a peak power constraint, the signal transmitted by the relay cannot exceed a threshold $\breve{P}_r$. We limit our discussion to BPSK in the following. The case with higher order constellations can be obtained similarly.

We first consider the optimal abs relay function under a peak power constraint. By following the argument in Section 2.3.2.4, we can obtain the optimization problem to find the optimal relay function as

$$\begin{aligned}
\min_{f,v} \quad G(f) = \int_0^{+\infty} & \Big(\mathcal{G}\left(u - (h_1 + h_2)\sqrt{P_s}, \sigma_r^2\right) + \mathcal{G}\left(u + (h_1 + h_2)\sqrt{P_s}, \sigma_r^2\right) \\
& -\mathcal{G}\left(u - (h_1 - h_2)\sqrt{P_s}, \sigma_r^2\right) - \mathcal{G}\left(u + (h_1 - h_2)\sqrt{P_s}, \sigma_r^2\right)\Big) \\
& \times \left[\int_{-\infty}^{v} \mathcal{G}\left(y - h_1 f(u), \sigma_s^2\right) dy\right] du, \\
\text{subject to} \quad & f^2(u) \leq \breve{P}_r.
\end{aligned} \tag{2.58}$$

Let $B(u) = \mathcal{G}\left(u - (h_1 + h_2)\sqrt{P_s}, \sigma_r^2\right) + \mathcal{G}\left(u + (h_1 + h_2)\sqrt{P_s}, \sigma_r^2\right) - \mathcal{G}\left(u - (h_1 - h_2)\sqrt{P_s}, \sigma_r^2\right) - \mathcal{G}\left(u + (h_1 - h_2)\sqrt{P_s}, \sigma_r^2\right)$ and w be its root in $[0, +\infty)$. It is clear that $B(u) > 0$ when

$u > w$ and $B(u) < 0$ when $u < w$. Therefore, to minimize $G(f)$, we should choose

$$f(u) = \begin{cases} \sqrt{\breve{P}_r}, & \text{if } u \geq w, \\ -\sqrt{\breve{P}_r}, & \text{if } w > u \geq 0, \\ f(-u), & \text{if } u < 0. \end{cases} \tag{2.59}$$

Taking the partial derivative of $G(f)$ with respect to v, we obtain

$$\frac{\partial G(f)}{\partial v} = \int_0^{+\infty} B(u)\mathcal{G}\left(v - f(u), \sigma_s^2\right) dy du, \tag{2.60}$$

which is equal to zero if $v = 0$.

For the non-abs strategy, the sum of average error probabilities at both terminals is

$$\begin{aligned} & P_e^{(1)} + P_e^{(2)} \\ =& 1 + \frac{1}{2}\int_{-\infty}^{+\infty} \left(\mathcal{G}\left(u - (h_1 + h_2)\sqrt{P_s}, \sigma_r^2\right) - \mathcal{G}\left(u - (h_1 - h_2)\sqrt{P_s}, \sigma_r^2\right)\right) \\ & \qquad \times \left[\int_{-\infty}^{v_1} \mathcal{G}\left(y - h_1 f(u), \sigma_s^2\right) dy\right] du \\ & + \frac{1}{2}\int_{-\infty}^{+\infty} \left(\mathcal{G}\left(u - (h_2 + h_1)\sqrt{P_s}, \sigma_r^2\right) - \mathcal{G}\left(u - (h_2 - h_1)\sqrt{P_s}, \sigma_r^2\right)\right) \\ & \qquad \times \left[\int_{-\infty}^{v_2} \mathcal{G}\left(y - h_2 f(u), \sigma_s^2\right) dy\right] du, \end{aligned} \tag{2.61}$$

subject to $f^2(u) \leq \breve{P}_r$. As the constraint on $f(u)$ is imposed on each u, minimizing (2.61) is equivalent to

$$\begin{aligned} \min_{f(u)} \quad & \mathcal{G}\left(u - (h_1 + h_2)\sqrt{P_s}, \sigma_r^2\right) \left[\int_{-\infty}^{v_1} \mathcal{G}\left(y - h_1 f(u), \sigma_s^2\right) dy + \int_{-\infty}^{v_2} \mathcal{G}\left(y - h_2 f(u), \sigma_s^2\right) dy\right] \\ & - \mathcal{G}\left(u - (h_1 - h_2)\sqrt{P_s}, \sigma_r^2\right) \left[\int_{-\infty}^{v_1} \mathcal{G}\left(y - h_1 f(u), \sigma_s^2\right) dy + \int_{-\infty}^{v_2} \mathcal{G}\left(y + h_2 f(u), \sigma_s^2\right) dy\right] \\ & - \mathcal{G}\left(u + (h_1 - h_2)\sqrt{P_s}, \sigma_r^2\right) \left[\int_{-\infty}^{v_1} \mathcal{G}\left(y + h_1 f(u), \sigma_s^2\right) dy + \int_{-\infty}^{v_2} \mathcal{G}\left(y - h_2 f(u), \sigma_s^2\right) dy\right] \\ & + \mathcal{G}\left(u + (h_1 + h_2)\sqrt{P_s}, \sigma_r^2\right) \left[\int_{-\infty}^{v_1} \mathcal{G}\left(y + h_1 f(u), \sigma_s^2\right) dy + \int_{-\infty}^{v_2} \mathcal{G}\left(y + h_2 f(u), \sigma_s^2\right) dy\right], \\ \text{subject to} \quad & f^2(u) \leq \breve{P}_r. \end{aligned} \tag{2.62}$$

It is complicated to solve (2.62) exactly. We consider a high SNR approximate solution instead. When $u > h_1\sqrt{P_s}$, the objective function in (2.62) is dominated by the first term. Therefore, we should choose $f(u) = \sqrt{\breve{P}_r}$ if $u > h_1\sqrt{P_s}$. When $0 \leq u < h_1\sqrt{P_s}$, the objective function in (2.62) is dominated by the second term. Minimizing the second term we obtain

$$f(u) = \frac{v_1 - v_2}{h_1 + h_2}. \tag{2.63}$$

In high SNR, the sum of average probabilities of both terminals (2.61) is dominated by

$$Q\left(\frac{h_1\sqrt{\breve{P}_r} - v_1}{\sigma_s}\right) + Q\left(\frac{h_2\sqrt{\breve{P}_r} - v_2}{\sigma_s}\right) + 2Q\left(\frac{h_2v_1 + h_1v_2}{(h_1+h_2)\sigma_s}\right) + F(P_s, \sigma_r), \tag{2.64}$$

where $F(P_s, \sigma_r)$ is only a function of P_s and σ_r. Minimizing (2.64) over v_1 and v_2, we obtain

$$v_1 = \frac{h_1^2}{h_1 + h_2}\sqrt{\breve{P}_r}, \quad v_2 = \frac{h_2^2}{h_1 + h_2}\sqrt{\breve{P}_r}. \tag{2.65}$$

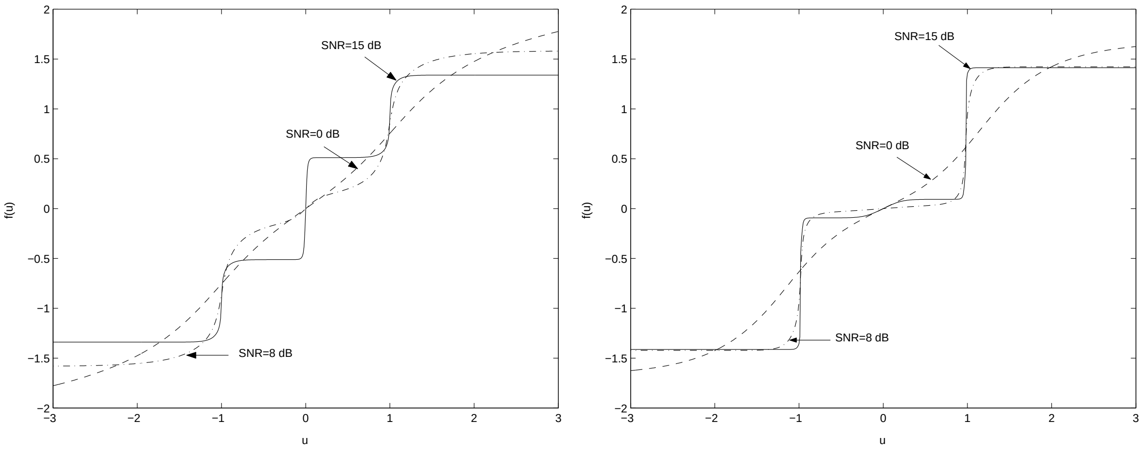
Therefore, the optimal non-abs relay function is

$$f(u) = \begin{cases} \sqrt{\breve{P}_r}, & \text{if } u \geq h_1\sqrt{P_s}, \\ \frac{h_1 - h_2}{h_1 + h_2}\sqrt{\breve{P}_r}, & \text{if } h_1\sqrt{P_s} > u \geq 0, \\ -f(-u), & \text{if } u < 0. \end{cases} \tag{2.66}$$

Remarks:

- In [83], the peak power constraint is implicitly assumed. They do not show the optimality of their relay function. We find that under a peak power constraint, the ADF strategy is the optimal abs strategy, while the optimal non-abs strategy is similar to the DF strategy (2.12) except for the power constraint.

- As with the average power constraint, the non-abs optimal relay may perform better or worse than the abs optimal relay depending on the ratio $\breve{P}_r/P_s$ and the ratio h_1/h_2. The discussion is similar to that in Section 2.4. We do not repeat it here.



(a) $h_1 = 1$ and $h_2 = 0.5$

(b) $h_1 = 1$ and $h_2 = 0.8$

Figure 2.2: The optimized non-abs-based relay function at different SNRs and with different values of h_1 and h_2.

2.7 Simulation Results

In this section, we compare the performance of different strategies with $\sigma_r^2 = \sigma_s^2$ and $P_r = P_s = 1$ in all cases.

Fig. 2.2 shows the optimized non-abs-based relay function for different SNRs and different values of h_1 and h_2. The relay operation behaves like the AF strategy at low SNR and like the DF strategy (2.12) at high SNR. Different abs-based relay functions $f(u)$ are compared in Fig. 2.3, where for AAF we choose $C = \sqrt{P_s} + \sigma_r/\sqrt{2}$. Unlike ADF with a hard limiter, the optimized relay adapts its transmit power according to the signal strength it receives; This is the benefit of the average power constraint. From Fig. 2.3, we can also see that when the SNR is small, the optimized relay function gives a “V”-shaped behavior similar to that of the AAF strategy. As the SNR increases, the behavior of the optimized relay function is more similar to the one for the ADF strategy. This suggests that ADF performs well at high SNR while AAF is effective at low SNR. Interestingly, the relay function of EF has almost the same shape as the optimized relay function at all SNRs.

Fig. 2.4 compares the bit error rate (BER) performance of different abs-based and non-abs-based strategies for BPSK when $h_1 = 1$ and $h_2 = 0.8$. We observe that the

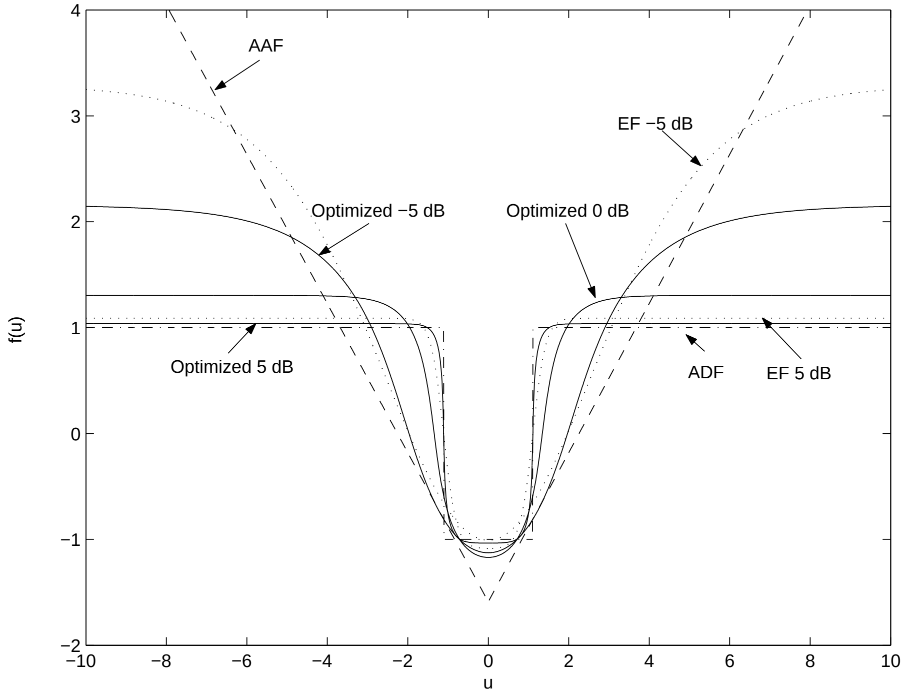


Figure 2.3: Comparison of function $f(u)$ in different abs-based schemes with $\sigma_r^2 = \sigma_s^2$, $h_1 = h_2 = 1$ and $P_r = P_s = 1$.

optimized non-abs-based (abs-based) relay performs like the AF (AAF) strategy at low SNR and like the DF (ADF) strategy at high SNR. Also, EF performs close to the optimized strategy for all SNR values. It can also be seen that in this scenario non-abs-based strategies perform better than abs-based strategies at low SNR and worse than abs-based strategies at high SNR. The reason for this is that non-abs-based strategies do not exploit the *a priori* information about the signal available at each terminal providing; This *a priori* information produces extra redundancy, which is useful particularly at low SNR. A similar behavior is observed in Fig. 2.5, where the case $h_1 = 1$ and $h_2 = 0.5$ is considered. Compared to the results for $h_1 = 1$ and $h_2 = 0.8$ in Fig. 2.4 the threshold SNR below which non-abs-based strategies perform better than abs-based strategies is increased. Thus, non-abs-based strategies are beneficial for asymmetric channels.

In Fig. 2.6 we compare the BER for the AF, AAF, and ADF strategies on the two-way relay channel in the high SNR regime, where we assume that $\sigma_r^2 = \sigma_s^2$, $h_1 = h_2$ and $P_r = P_s = 1$. For the AAF strategy we set $C = 1$. Also, we do not include the

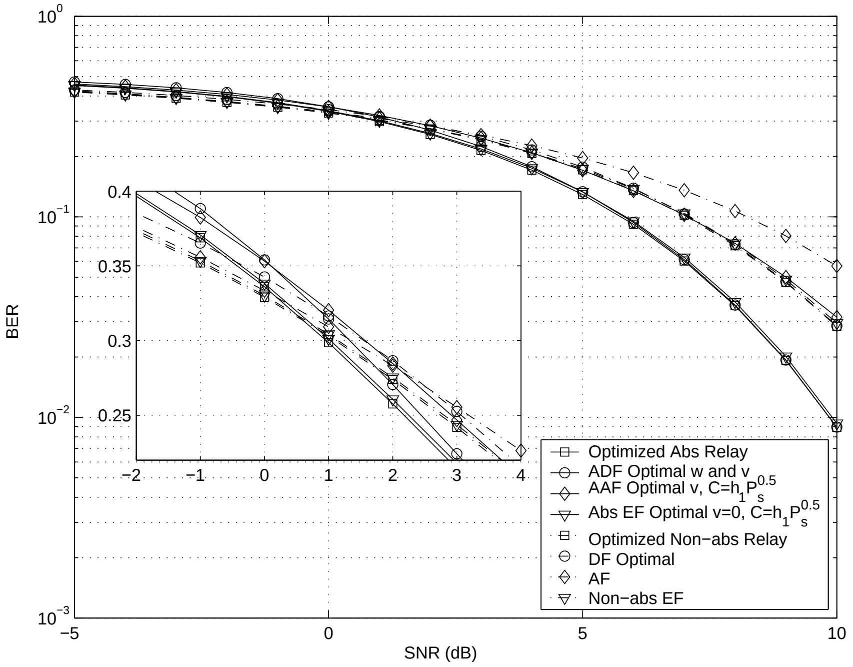


Figure 2.4: Performance comparison between different abs-based and non-abs-based strategies when $h_1 = 1$ and $h_2 = 0.8$, $P_r = P_s = 1$. The subfigure shows the crossover between the abs-based and non-abs-based strategies.

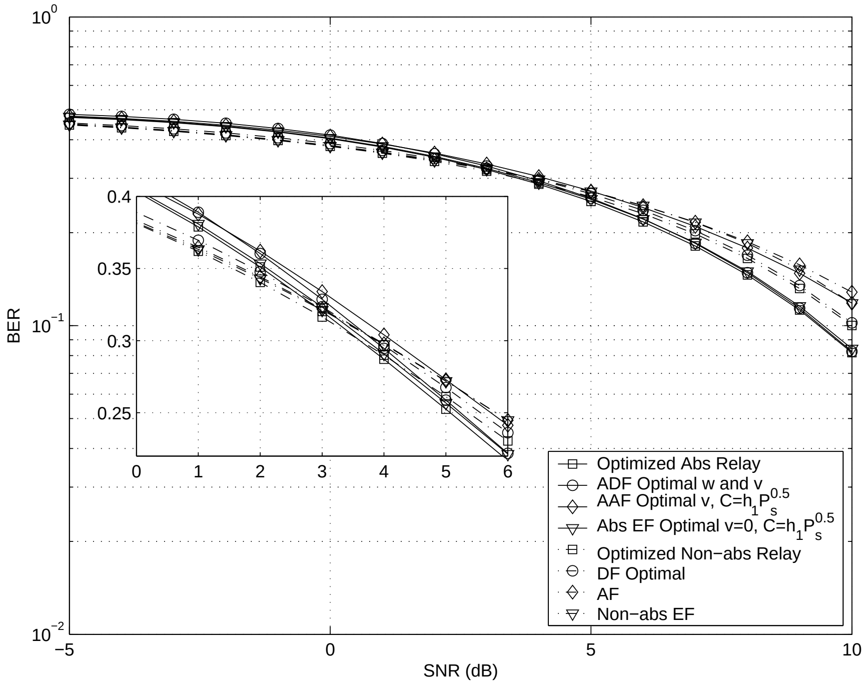


Figure 2.5: Performance comparison between different abs-based and non-abs-based strategies when $h_1 = 1$ and $h_2 = 0.5$, $P_r = P_s = 1$. The subfigure shows the crossover between the abs-based and non-abs-based strategies.

optimized relay and EF strategies as their performances are very close to ADF at high SNR. We observe from Fig. 2.6 that AAF has a 2 dB gain over AF at a BER of 10^{-8}. Finally, we can see from Fig. 2.6 that ADF performs best, where a 2.7 dB gain over AAF at a BER of 10^{-8} can be observed.

In Fig. 2.7 the average error probability of ADF and DF is compared for three different cases. The plotted results agree with our analysis in Section 2.4. Fig. 2.8 compares the behavior of $f(u)$ for AAF, ADF, and EF strategies for $\sigma_r^2 = \sigma_s^2$, where both terminals use 4-PAM, i.e., $M = |\mathcal{V}'| = 4, 5, 6, 7$, and the SNR is chosen to be $5/\sigma_r^2$. The behavior of the relay function in Fig. 2.8 resembles the one in Fig. 2.3 for different strategies. In particular, EF and AAF perform similarly when the SNR is low. As the SNR increases, the EF relay function gives performance resemblig the behavior of the ADF relay function.

In Fig. 2.9 the symbol error rate (SER) of different relay functions using ADF and AAF is compared, where the same parameters as in Fig. 2.8 are used. The

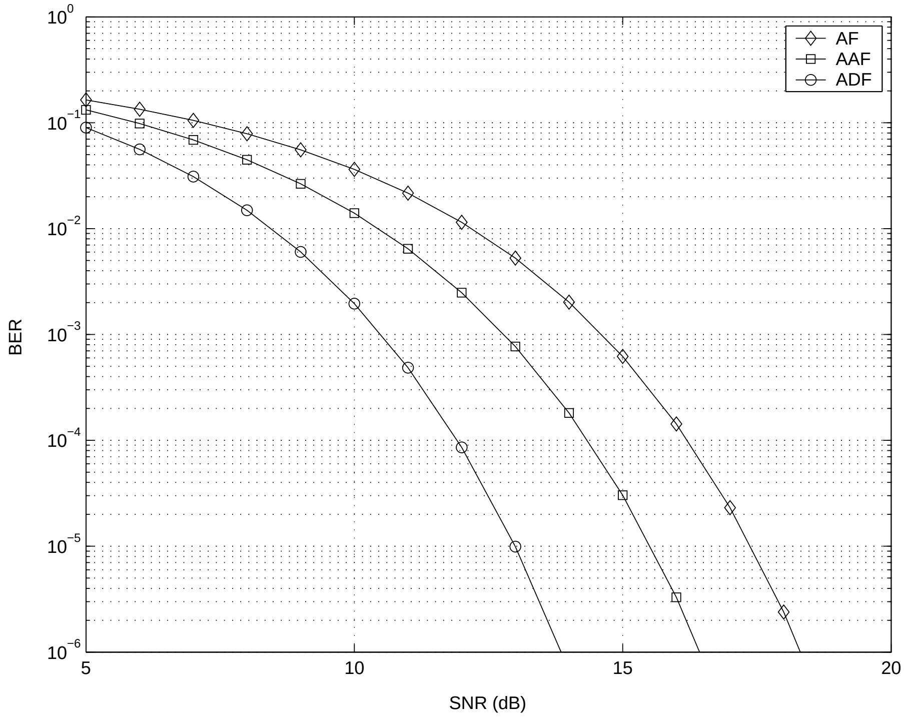


Figure 2.6: Performance comparison of AF, AAF and ADF with the AF scheme for one-way relay channel when $\sigma_r^2 = \sigma_s^2$, $h_1 = h_2$ and $P_r = P_s = 1$.

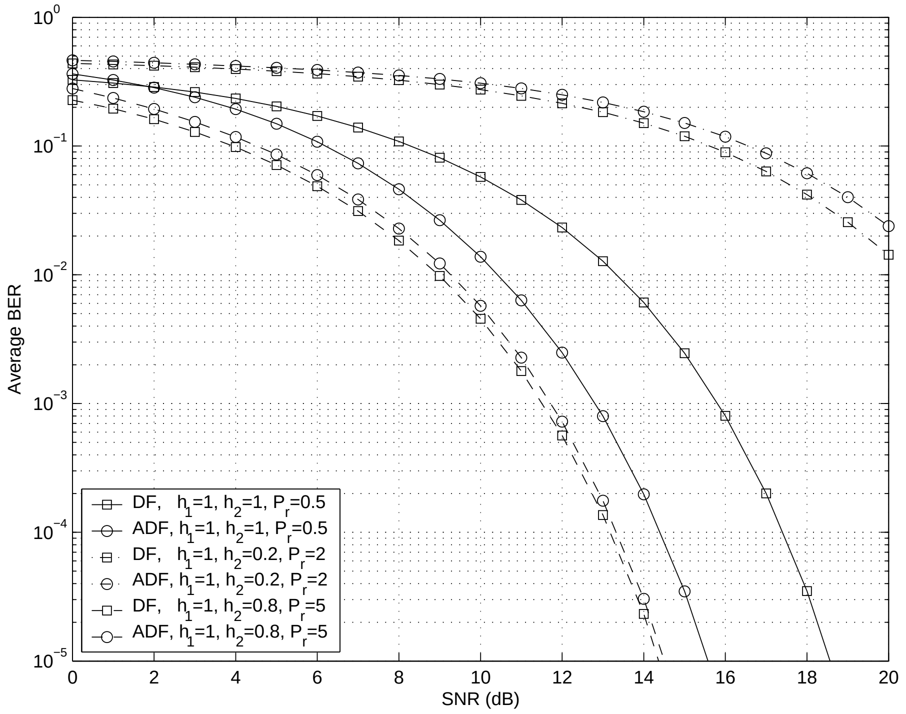


Figure 2.7: Performance comparison of ADF with DF under different scenarios.

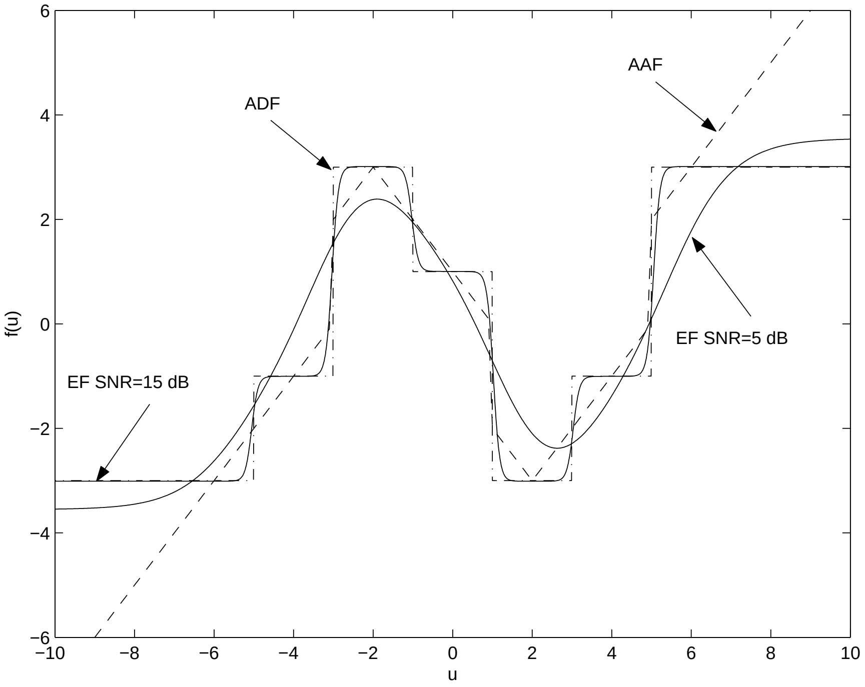


Figure 2.8: Comparison of relay functions for AAF, ADF, and EF with $\sigma_r^2 = \sigma_s^2$ where both terminals use 4-PAM.

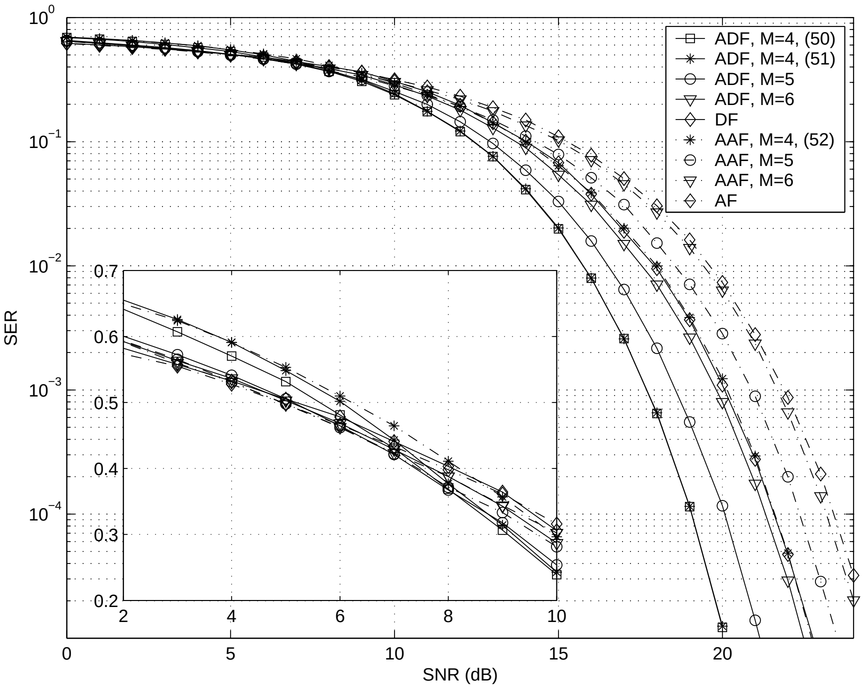


Figure 2.9: SER comparison of ADF and AAF relay functions for 4-PAM with $M = |\mathcal{V}'| = 4, 5, 6, 7$ and $\sigma_r^2 = \sigma_s^2$. The subfigure shows the crossover between different strategies.

performance degrades as M increases. In Fig. 2.9, a comparison between the mappings in (2.50) and (2.51) shows almost identical performance. There are two factors that affect the performance of relay functions with different M. First, a small M indicates a higher compression at the relay, which results in power savings. Second, when M is small, a detection error at the relay may affect the overall performance. At high SNR, it is clear that the power savings dominate the performance of ADF. At low SNR, we find that the performance degrades as M decreases, which means that $M = 7$ achieves the best performance. For example, at SNR= 0 dB, the SERs for $M = 4$, 5, 6, 7 are 0.6904, 0.6472, 0.6428, and 0.6146, respectively. This observation generalizes the one for the BPSK case, where the reason for this behavior is again that the redundancy in the constellation set increases for larger M.

2.8 Conclusion

We have analyzed and optimized relaying strategies for memoryless TWRCs. In particular, we propose abs-based strategies where the relay processes the absolute value of the received signal. These techniques generally outperform non-abs-based strategies in the moderate to high SNR regime since they take into account the side information available at the terminals; This allows for additional power savings. Specifically, we have considered abs- and non-abs-based AF, DF and EF schemes, and also the optimization of the nonlinear processing function at the relay. We found that the non-abs-based DF performs better than the abs-based DF when the two-way channel is very asymmetric or the relay has greater power than the two terminals. ADF performs better than DF when the relay has roughly the same power as the terminals. Although this work does not consider channel coding, the obtained expressions for the error probability allow for a rough determination of the required rate for an end-to-end channel code. Extensions of these results to higher order constellations such as QAM and PAM have also been presented, where similar observations can be made.

2.9 Appendix

In this appendix, we prove Theorem 2.1. We first give the following lemma.

Lemma 2.3 *Let Z be a normal random variable with mean 0, and let $p_U(\mu)$, $p_V(\mu)$ denote two arbitrary probability density functions associated with the random variables U and V that are independent of Z, respectively. If there exists a threshold t for which $p_U(\mu) - p_V(\mu)$ is nonnegative for $\mu \geq t$ and negative otherwise, then there exits a threshold t' for which $p_{U+Z}(\nu) - p_{V+Z}(\nu)$ is nonnegative for $\nu \geq t'$ and negative otherwise.*

Proof. Denote by σ^2 the variance of Z. The result follows since

$$\begin{aligned}
& p_{U+Z}(\nu) - p_{V+Z}(\nu) \\
= & \int_{-\infty}^{\infty} p_Z(\nu-\mu)p_U(\mu)d\mu - \int_{-\infty}^{\infty} p_Z(\nu-\mu)p_V(\mu)d\mu \\
= & \int_{-\infty}^{\infty} \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{(\nu-\mu)^2}{2\sigma^2}\right\} (p_U(\mu) - p_V(\mu))\, d\mu \\
= & \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{\nu^2 - 2t\nu}{2\sigma^2}\right\} \left(\int_{-\infty}^{t} \exp\left\{\frac{2\nu(\mu - t) - \mu^2}{2\sigma^2}\right\} (p_U(\mu) - p_V(\mu))\, d\mu \right. \\
& \left. + \int_{t}^{\infty} \exp\left\{\frac{2\nu(\mu - t) - \mu^2}{2\sigma^2}\right\} (p_U(\mu) - p_V(\mu))\, d\mu \right)
\end{aligned}$$

where $\frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{\nu^2-2t\nu}{2\sigma^2}\right\} > 0$, and both integral terms are nondecreasing functions of ν. ■

Proof of Theorem 2.1. For brevity let $a \triangleq h_1\sqrt{P_s} + h_2\sqrt{P_s}$ and $b \triangleq h_1\sqrt{P_s} - h_2\sqrt{P_s}$.

Case 1: Non-abs strategies. When $x_1 = \sqrt{P_s}$, terminal 1's error-minimizing detection rule is to decide $x_2 = \sqrt{P_s}$ if $p_{f(a+N)+Z_1}(y_1) - p_{f(b+N)+Z_1}(y_1) \geq 0$ and $x_2 = -\sqrt{P_s}$ otherwise. Since $f(U)$ is an increasing function of U, we can apply Lemma 2.3 with $U = f(a+N)$, $V = f(b+N)$ and $Z = Z_1$ to give the result.

Case 2: Abs strategies. When $x_1 = \sqrt{P_s}$, terminal 1's error-minimizing detection rule is to decide $x_2 = \sqrt{P_s}$ if $p_{f(|a+N|)+Z_1}(y_1) - p_{f(|b+N|)+Z_1}(y_1) \geq 0$ and $x_2 = -\sqrt{P_s}$ otherwise. Note that

$$p_{|a+N|}(\mu) - p_{|b+N|}(\mu) = p_{a+N}(\mu) + p_{a+N}(-\mu) - p_{b+N}(\mu) - p_{b+N}(-\mu) = C(\mu)\,(D(\mu) - 1)$$

where $C(\mu) = \left(\exp\{-(\mu-b)^2/2\sigma^2\} + \exp\{-(-\mu-b)^2/2\sigma^2\}\right)/\sigma\sqrt{2\pi} > 0$ and

$$\begin{aligned} D(\mu) &= \frac{\exp\{-(\mu-a)^2/2\sigma^2\} + \exp\{-(-\mu-a)^2/2\sigma^2\}}{\exp\{-(\mu-b)^2/2\sigma^2\} + \exp\{-(-\mu-b)^2/2\sigma^2\}} \\ &= \exp\left\{\frac{-a^2+b^2}{2\sigma^2}\right\}\frac{\exp\{\mu a/\sigma^2\} + \exp\{-\mu a/\sigma^2\}}{\exp\{\mu b/\sigma^2\} + \exp\{-\mu b/\sigma^2\}} \end{aligned}$$

is an increasing function for $\mu \geq 0$. Thus, there exists a threshold t such that $p_{|a+N|}(\mu) - p_{|b+N|}(\mu)$ is nonnegative for $\mu \geq t$ and negative otherwise. Since $f(|U|)$ is a non-decreasing function of $|U|$, we can apply Lemma 2.3 with $U = f(|a+N|)$, $V = f(|b+N|)$ and $Z = Z_1$ to give the result.

In both cases, by symmetry, threshold detection is also optimal when $x_1 = -\sqrt{P_s}$. The result for terminal 2 follows by symmetry. ■

Chapter 3

Distributed Scheduling in Wireless Networks Exploiting Broadcast and Network Coding

In this chapter, we consider cross-layer optimization in wireless networks with wireless broadcast, focusing on the problem of distributed scheduling of broadcast links. The goal is to integrate various protocol layers into a unified framework to take the advantage of wireless broadcast.

3.1 Introduction

Optimization-based cross-layer design for wireless networks has attracted much interest recently, see, e.g., [19, 21, 55, 63] and the references therein. Joint optimization of multiple protocol layers can substantially increase the end-to-end throughput, or reduce power consumption. Most existing works on cross-layer design do not incorporate the exploitation of the wireless broadcast advantage where transmissions from an omnidirectional antenna can be received by any nodes that lie within its communication range. This broadcast advantage can result in throughput improvement and power saving especially with multicasting [78]. In this chapter we consider distributed algorithms for wireless link scheduling that take the broadcast advantage into account. We apply this to a distributed joint optimization of multicast network coding, rate control, and channel access.

We model the wireless network as a directed hypergraph, with wireless broadcast being abstracted as a hyperarc. Scheduling with the broadcast advantage is a hard problem in general. It forms a component of the algorithm proposed in [42], where it is assumed to be solved by a central controller. In this chapter we focus on a simple, so-called primary interference model [35]. Under this interference model, any valid link schedule corresponds to a *hypergraph matching* and the optimal schedule corresponds to a *maximum weighted hypergraph matching.*

The maximum weighted hypergraph matching problem is, however, NP-complete [56]. We thus propose two classes of distributed approximation algorithms to treat the link scheduling problem under the primary interference model. The first algorithm is the locally greedy algorithm, which chooses the locally heaviest hyperedge. We show that this algorithm returns a hypergraph matching with weight at least a constant factor of the maximum weighted hypergraph matching, giving a stability region within a constant factor of the region achievable with any hypergraph matching algorithm. The second algorithm is a randomized algorithm, which always returns a maximal hypergraph matching. This gives a stability region for single hop communication that is at least $1/K$ of the region achievable with any hypergraph matching algorithm, where K is the maximum number of nodes in any hyperedge. The randomized algorithm can be readily turned into a constant-time algorithm.

We also provide a generalization of existing results in cross-layer optimization for multicast network coding in wireless networks. Our cross-layer design uses the framework of utility maximization, see, e.g., [21], which maximizes the aggregate user utilities subject to flow conservation and scheduling constraints on the hypergraph. We then apply duality theory to decompose the problem vertically into rate control, network coding and session scheduling, and link scheduling subproblems, which interact through dual variables. Based on this decomposition, a distributed subgradient algorithm is proposed, whose session and link scheduling components are similar to the back-pressure algorithm in [42], which does not incorporate rate control.

The rest of this chapter is organized as follows. In Section 3.2, we briefly review some related work. Preliminaries are presented in Section 3.3. In Section 3.4, we present our cross-layer design algorithm. Link scheduling algorithms for the primary interference model are given in Section 3.5. Simulation results are presented in Section 3.6. We conclude this chapter in Section 3.7.

3.2 Related Work

Extensive research has been devoted to the cross-layer design for wireless networks but usually without considering network coding, see, e.g., [19,21,55,63]. Similar cross-layer design algorithm is proposed in [17, 55, 63, 64], and in particular, the impact of imperfect scheduling is also studied in [55]. In [64], the network capacity region is characterized, and a joint routing and power allocation policy is proposed to stabilize the system whenever the input rates are within this capacity region.

Network coding extends the functionality of network nodes from storing/forwarding packets to performing algebraic operations on received data. Starting with the work of [7], various potential benefits of network coding have been shown, including robustness to link/node failures and packet losses [57]. It is especially preferred in wireless networks, where the bandwidth is scarce. Distributed random linear coding schemes, see, e.g., [40], have made practical implementation of network coding possible.

For optimization with network coding, Lun *et. al.* [58] propose a dual subgradient method for the problem of minimum cost multicasting with network coding. For rate control, the approach in [19] is extended to network coding in [18]. In [42], the rate stability region for a wireless network with and without correlated sources is characterized. In [70], medium access control and network coding is considered and broadcast advantage is also exploited. A set of conflict-free transmission schedules is predetermined, and the scheduling uses a suboptimal time division mechanism.

The primary interference model was introduced in [35]. In [74], randomized algorithms are proposed, which achieve the capacity region with reduced complexity by comparing a random matching with the current matching. In [61], a distributed implementation of the algorithm in [74] is proposed. Scheduling algorithms based on maximal matching are also considered in works such as [55]. These matching-based algorithms do not consider the broadcast advantage.

3.3 System Model

A wireless network is modeled as a directed hypergraph $\mathcal{H} = (\mathcal{N}, \mathcal{A})$, where $\mathcal{N}$ is the set of nodes and $\mathcal{A}$ is the set of hyperarcs. A hyperarc is a pair (i, J), with $i \in \mathcal{N}$ the start node and $J \subseteq \mathcal{N}$ the set of end nodes, representing a broadcast link from node i to nodes in J. We assume that (i, J) is lossless, i.e., it does not experience packet erasures. When $|J| = 1$ for all $(i, J) \in \mathcal{A}$, the hypergraph reduces to the conventional graph model. A set $\mathcal{M}$ of multicast sessions is transmitted through the network. Each session $m \in \mathcal{M}$ is associated with a set $\mathcal{S}_m \subset \mathcal{N}$ of sources and a set $\mathcal{T}_m \subset \mathcal{N}$ of sinks. In session m, each source $s \in \mathcal{S}_m$ multicasts x^{ms} bits per second to all the sinks in $\mathcal{T}_m$. By the flow conservation condition,

$$\sum_{\{J|(i,J)\in\mathcal{A}\}} \sum_{j\in J} g^{mst}_{iJj} - \sum_{j\in\mathcal{N}} \sum_{\{i|(j,I)\in\mathcal{A},\, i\in I\}} g^{mst}_{jIi} = \sigma^{ms}_i, \forall i \in \mathcal{N},\ s \in \mathcal{S}_m,\ t \in \mathcal{T}_m,\ m \in \mathcal{M}, \quad (3.1)$$

where $\sigma^{ms}_i = x^{ms}$ if $i = s$, $\sigma^{ms}_i = -x^{ms}$ if $i = t$, $\sigma^{ms}_i = 0$ otherwise, and g^{mst}_{iJj} is the information rate from source s to sink t in session m over (i, J) and is intended to node $j \in J$.

Let $\underline{S}(\tau) = \{S_{i,j}(\tau)\}$ denote the matrix process of channel states, where $S_{i,j}(\tau)$ represents the channel state from node i to node j at time τ. In every time slot, node i determines transmission rates on each hyperarc $(i, J) \in \mathcal{A}$ by allocating a power matrix $\underline{P} = \{P_{iJ}\}$ subject to a total power constraint

$$\sum_{(i,J)\in\mathcal{A}} P_{iJ} \le P^{\text{tot}}_i, \forall i \in \mathcal{N}, \quad (3.2)$$

where P^{tot}_i is the maximal total power allowable at node i. Hyperarc rates are determined by a rate-power curve $\underline{r}(\underline{P}, \underline{S}) = \{r_{iJ}(\underline{P}, \underline{S})\}$, where $r_{iJ}(\underline{P}, \underline{S})$ determines the rate at which packets, injected into hyperarc (i, J), are received by all the nodes in J. By time-sharing, the capacity region is the convex hull $\text{Co}(\underline{r}(\underline{P}, \underline{S}))$ of all achievable rate vectors $\underline{r}(\underline{P}, \underline{S})$.

We assume that network coding is done only across packets of the same multicast session. With this setting, we define f^m_{iJ} as the physical flow of session m on hyperarc

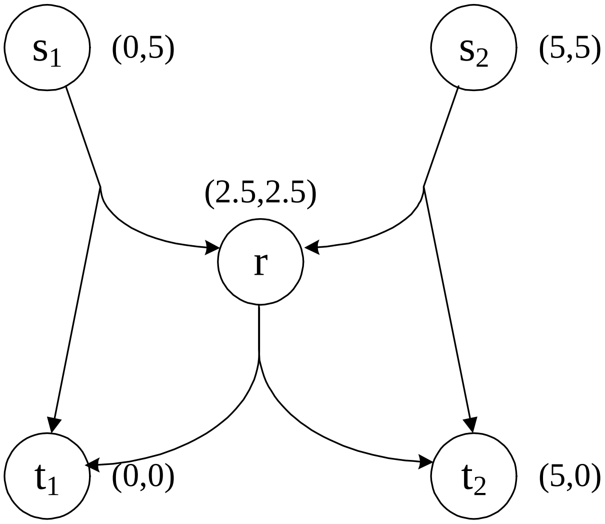


Figure 3.1: Wireless butterfly network.

(i, J) as opposed to the virtual flow g_{iJj}^{mst} in (3.1). By the flow sharing property of network coding [7] and the rate constraint, we have the following two constraints

$$\sum_{s\in\mathcal{S}_m}\sum_{j\in J} g_{iJj}^{mst} \leq f_{iJ}^{m}, \quad \forall(i,J)\in\mathcal{A},\, m\in\mathcal{M},\, t\in\mathcal{T}_m, \tag{3.3}$$

$$\sum_{m\in\mathcal{M}} f_{iJ}^{m} \leq r_{iJ}, \qquad \forall(i,J)\in\mathcal{A}, \tag{3.4}$$

where $\{r_{iJ}\} \in \mathrm{Co}(\underline{r}(\underline{P}, \underline{S}))$, with r_{iJ} the capacity of the hyperarc (i, J).

To illustrate this, consider the network in Figure 3.1. There is a single multicast session m with two source nodes and two sink nodes. The hyperarc $(s_1, \{r, t_1\})$ carries actual flow $f_{s_1\{r,t_1\}}^{m}$ and virtual flows $g_{s_1\{r,t_1\}t_1}^{ms_1t_1}$ to sink t_1, $g_{s_1\{r,t_1\}r}^{ms_1t_1}$ to sink t_1 via r, and $g_{s_1\{r,t_1\}r}^{ms_1t_2}$ to sink t_2 via r. The flow sharing condition (3.3) for this hyperarc is

$$g_{s_1\{r,t_1\}t_1}^{ms_1t_1} + g_{s_1\{r,t_1\}r}^{ms_1t_1} \;\leq\; f_{s_1\{r,t_1\}}^{m}, \quad g_{s_1\{r,t_1\}r}^{ms_1t_2} \leq f_{s_1\{r,t_1\}}^{m} \tag{3.5}$$

3.4 Cross-Layer Design with Broadcast Advantage and Network Coding

In this section, we derive a cross-layer design by using the utility maximization framework. The resulting algorithm is an extension of [18,42]. Each source s of session m is associated with a utility function $U_{ms}(x^{ms})$, which is assumed to be strictly concave, non-decreasing and twice continuously differentiable. We formulate network resource allocation as the following optimization problem

$$
\begin{aligned}
\max_{x^{ms}, g_{iJj}^{mst}, f_{iJ}^{m}, r_{iJ}, P_{iJ}} \quad & \sum_{m\in\mathcal{M}, s\in\mathcal{S}_m} U_{ms}(x^{ms}) \\
\text{s.t.} \quad & \sum_{\{J|(i,J)\in\mathcal{A}\}} \sum_{j\in J} g_{iJj}^{mst} - \sum_{j\in\mathcal{N}} \sum_{\{i|(j,I)\in\mathcal{A},\, i\in I\}} g_{jIi}^{mst} = \sigma_i^{ms}, \\
& \qquad\qquad\qquad\qquad \forall i \neq t, s\in\mathcal{S}_m, t\in\mathcal{T}_m, m\in\mathcal{M}, \\
& \sum_{s\in\mathcal{S}_m,\, j\in J} g_{iJj}^{mst} \le f_{iJ}^m, \ \forall (i,J), t\in\mathcal{T}_m, m\in\mathcal{M}, \\
& \sum_{m\in\mathcal{M}} f_{iJ}^m \le r_{iJ}, \ \forall (i,J), \\
& \{r_{iJ}\} \in \mathrm{Co}(\underline{r}(\underline{P}, \underline{S})), \ \sum_{\{J|(i,J)\in\mathcal{A}\}} P_{iJ} \le P_i^{\mathrm{tot}}, \ \forall i,
\end{aligned}
\tag{3.6}
$$

where the constraints come from equations (3.1)-(3.4). Here we do not include flow conservation equations at destinations, which is automatically guaranteed by the flow conservation at the source and intermediate nodes. Problem (3.6) is strictly convex and has a unique solution with respect to source rates x^{ms}. By relaxing only the first set of constraints, the partial dual function to (3.6) can be decomposed into the following two subproblems

$$
\phi_1(q) = \max_{x^{ms}} \sum_{m,s} U_{ms}(x^{ms}) - \sum_{m,s} \left(\sum_t q_s^{mst} \right) x^{ms}, \tag{3.7}
$$

$$
\begin{aligned}
\phi_2(q) = \max_{g_{iJj}^{mst}, f_{iJ}^m, r_{iJ}, P_{iJ}} & \sum_{i,m,s,t} q_i^{mst} \left(\sum_{\{J|(i,J)\in\mathcal{A}\}} \sum_{j\in J} g_{iJj}^{mst} - \sum_{j\in\mathcal{N}} \sum_{\{i|(j,I)\in\mathcal{A},\, i\in I\}} g_{jIi}^{mst} \right), \\
\text{subject to} & \sum_{s\in\mathcal{S}_m,\, j\in J} g_{iJj}^{mst} \le f_{iJ}^m, \ \sum_{m\in\mathcal{M}} f_{iJ}^m \le r_{iJ}, \ \{r_{iJ}\}\in \mathrm{Co}(\underline{r}(\underline{P}, \underline{S})), \\
& \sum_{\{J|(i,J)\in\mathcal{A}\}} P_{iJ} \le P_i^{\mathrm{tot}},
\end{aligned}
\tag{3.8}
$$

where q_i^{mst} is the Lagrange multiplier at node i for source s and sink t of session m and will be interpreted as congestion price. The first subproblem is rate control. The second is joint network coding and scheduling. Thus, by dual decomposition, the flow optimization problem decomposes into separate "local" optimization problems of transport, and network/link layers, respectively. The two subproblems interact through the dual variable q.

Rate Control: At time τ, given dual variable $q(\tau)$, each source adjusts its sending rate according to the aggregate dual variable $\sum_t q_s^{mst}$ that is generated locally at the source

$$x^{ms}(\tau+1) = U_{ms}'^{-1}\left(\sum\nolimits_t q_s^{mst}(\tau)\right). \tag{3.9}$$

Session Scheduling and Network Coding: Note that (3.8) is equivalent to the following problem

$$\begin{aligned}
&\max_{g_{iJj}^{mst}, f_{iJ}^m, r_{iJ}, P_{iJ}} \sum_{(i,J),m,t} \sum_{s,j\in J} g_{iJj}^{mst}\left(q_i^{mst} - q_j^{mst}\right),\\
&\quad \text{subject to} \sum_{s,\,j\in J} g_{iJj}^{mst} \le f_{iJ}^m, \sum_{m\in\mathcal{M}} f_{iJ}^m \le r_{iJ}, \{r_{iJ}\} \in \mathrm{Co}(\underline{r}(\underline{P},\underline{S})), \sum_{\{J|(i,J)\in\mathcal{A}\}} P_{iJ} \le P_i^{\mathrm{tot}},\\
&= \max_{f_{iJ}^m, r_{iJ}, P_{iJ}} \sum_{(i,J),m} f_{iJ}^m \sum_t \max_{s,j\in J}\left[q_i^{mst} - q_j^{mst}\right]^+,\\
&\quad \text{subject to} \sum_{m\in\mathcal{M}} f_{iJ}^m \le r_{iJ}, \{r_{iJ}\} \in \mathrm{Co}(\underline{r}(\underline{P},\underline{S})), \sum_{\{J|(i,J)\in\mathcal{A}\}} P_{iJ} \le P_i^{\mathrm{tot}},
\end{aligned} \tag{3.10}$$

where $[\cdot]^+$ denotes the projection onto $\mathbb{R}^+$. The last equality in (3.10) comes from the fact that $\max_{g_{iJj}^{mst}} \sum_{s,j\in J} g_{iJj}^{mst}\left(q_i^{mst} - q_j^{mst}\right)$, subject to $\sum_{s,\,j\in J} g_{iJj}^{mst} \le f_{iJ}^m$ is a linear program, so we can always choose an extreme point solution, i.e.,

$$g_{iJj}^{mst} = \begin{cases} f_{iJ}^m, & \text{if } s = \hat{s}^{mt},\ j = \hat{j}^{mt}, \text{ and } q_i^{mst} - q_j^{mst} \ge 0,\\ 0, & \text{otherwise,}\end{cases} \tag{3.11}$$

where $\{\hat{s}^{mt}, \hat{j}^{mt}\} = \arg\max_{s,j\in J}\left(q_i^{mst} - q_j^{mst}\right)$.

Let $\hat{m}_{iJ} = \arg\max\limits_m \sum\limits_t \max\limits_{s,j\in J}\left[q_i^{mst} - q_j^{mst}\right]^+$ be the session with the maximum aggregate differential link prices over hyperarc (i,J). For each hyperarc (i,J), a random linear combination of packets from sources $\hat{s}^{\hat{m}_{iJ}t}$, $\forall t \in \mathcal{T}_{\hat{m}_{iJ}}$, in session $\hat{m}_{iJ}$ is broadcast to all nodes in J at the rate of r_{iJ}, where the packets received by node $j^{\hat{m}_{iJ}t}$ are intended for sink t in session $\hat{m}_{iJ}$. This is equivalent to solving (3.8) by

$$g_{iJj}^{mst}(q) = \begin{cases} r_{iJ}, \quad \text{if } m = \hat{m}_{iJ},\ s = \hat{s}^{mt},\ j = \hat{j}^{mt}, \text{ and } \max\limits_{s,j\in J} \left[q_i^{mst} - q_j^{mst}\right]^+ > 0, \\ 0, \qquad\qquad\qquad \text{otherwise.} \end{cases} \tag{3.12}$$

Link Scheduling and Power Control: Define $w_{iJ} = \max\limits_{m} \sum\limits_{t} \max\limits_{s,j\in J} \left[q_i^{mst} - q_j^{mst}\right]^+$. The joint link scheduling and power control problem becomes

$$\max_{r,P} \sum_{(i,J)\in\mathcal{A}} w_{iJ} r_{iJ}, \ \text{ s.t. } \{r_{iJ}\} \in \mathrm{Co}(\underline{r}(\underline{P},\underline{S})), \quad \sum_{\{J|(i,J)\in\mathcal{A}\}} P_{iJ} \le P_i^{\text{tot}}. \tag{3.13}$$

The problem (3.13) is in general a difficult global optimization problem. In Section 3.5, we will discuss a special interference model such that (3.13) can be solved in a distributed fashion in polynomial time.

Note that our scheduling problem (3.12)-(3.13) generalizes the back-pressure policy in [64, 75] by taking into account the differential backlog between node i and all nodes $j \in J$ instead of only a single node. Clearly, when $J = \{j\}$, our policy reduces to those in [64, 75]. The scheduling problem (3.12)-(3.13) is similar to that in [42], where the former is derived using the optimization framework while the latter is obtained by good intuition.

Dual Variable Update: At time $\tau + 1$, each node i updates its dual variable q according to the subgradient algorithm

$$q_i^{mst}(\tau+1) = \begin{cases} q_i^{mst}(\tau) + \gamma_\tau \left(x^{ms}(\tau) - \sum\limits_{\{J|(i,J)\in\mathcal{A}\}} \sum\limits_{j\in J} g_{iJj}^{mst}(q(\tau)) + \sum\limits_{j\in\mathcal{N}} \sum\limits_{\{i|(j,I)\in\mathcal{A},\, i\in I\}} g_{jIi}^{mst}(q(\tau)) \right), \\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{if } i = s, \\ q_i^{mst}(\tau) + \gamma_\tau \left(\sum\limits_{j\in\mathcal{N}} \sum\limits_{\{i|(j,I)\in\mathcal{A},\, i\in I\}} g_{jIi}^{mst}(q(\tau)) - \sum\limits_{\{J|(i,J)\in\mathcal{A}\}} \sum\limits_{j\in J} g_{iJj}^{mst}(q(\tau)) \right), \\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{otherwise,} \end{cases} \tag{3.14}$$

where γ_τ is a positive stepsize. After node i updates q_i^{mst}, it passes the value to all its neighbors. Note that the algorithm (3.9)-(3.14) only requires nodes to communicate with neighbors.

Now, we discuss the convergence and optimality of this cross layer design. Let the primal function (i.e., the total achieved network utility) be $P(x)$ and let x^* be an optimal value. Let q^* be a dual optimal solution. Define $\bar{x}(\tau) := \frac{1}{\tau}\sum_{k=1}^{\tau} x(k)$, the average data rate up to time τ, and $\bar{q}(\tau) := \frac{1}{\tau}\sum_{k=1}^{\tau} q(k)$, the average dual variable (congestion price) up to time τ. Let $g(q)$ be a subgradient of dual function $\phi(q)$. By using results on the convergence of the subgradient method, see, e.g., [17,63], we can show the following result when the joint link scheduling and power control problem (3.13) is solved exactly.

Theorem 3.1 *If the norm of the subgradients is uniformly bounded, i.e., there exists* $\mathfrak{g}$ *such that* $\|g(q)\|_2 \leq \mathfrak{g}$ *for all* q*, and a constant stepsize* γ *is adopted in (3.14), then the following inequalities hold*

$$\limsup_{\tau\to\infty} \phi(\bar{q}(\tau)) \leq \phi(q^*) + \frac{\gamma\mathfrak{g}^2}{2}, \tag{3.15}$$

$$\liminf_{\tau\to\infty} P(\bar{x}(\tau)) \geq P(x^*) - \frac{\gamma\mathfrak{g}^2}{2}. \tag{3.16}$$

Theorem 3.1 implies that the average source rate and congestion price approach the corresponding optima when the stepsize γ is small enough. The proof is omitted because it is similar to that in [17]. We may also establish the convergence of our cross-layer design in a slightly different sense, by using the standard convergence results for the subgradient method [73].

3.5 Link Scheduling

In this section, we study the joint link scheduling and power control problem (3.13) for networks with primary interference. A system is stable if the queue lengths at all nodes remain finite all the time. Note that the queue length at node i can be written as q_i^{mst}/γ for a constant stepsize γ [55]. A rate vector $\vec{x} = \{x^{ms}\}$ is feasible if there exists a scheduling policy that stabilizes the system with $\vec{x}$. We are interested in those scheduling policies that can stabilize the system for any rate vector within

$\eta\mathbf{\Lambda}$, where $\mathbf{\Lambda}$ denotes the network capacity region characterized by the constraints in (3.6). $\eta \in (0,1]$ is a constant that characterizes the performance of the scheduling policy. For example, in Theorem 3.8, which will be presented in Section 3.5.2, $\eta = \max\{\frac{1}{K}, \frac{1}{\kappa}\}$ for Algorithm 3.1 and Algorithm 3.2. By following the same argument as in [55], we can show that the performance of the joint design with each of our proposed scheduling algorithms is not worse than the design specified by the following optimization problem

$$\max \sum_{m\in\mathcal{M}, s\in\mathcal{S}_m} U_{ms}(x^{ms}), \text{ subject to } \vec{x} \in \eta\mathbf{\Lambda}, \tag{3.17}$$

with appropriate η that is determined by the worst-case performance bound of the corresponding scheduling algorithm.

3.5.1 Problem Formulation

The primary interference model [35] models a situation where each node is equipped with a single transceiver and neighboring nodes can transmit simultaneously using orthogonal CDMA or FDMA channels. Under this model, only those links that do not share nodes can transmit at the same time. Without using the broadcast advantage, any feasible schedule corresponds to a matching [75]. With the broadcast advantage, any feasible schedule corresponds to a hypergraph matching of the hypergraph $\mathcal{H}$, and (3.13) reduces to the maximum weighted hypergraph matching[1] problem.

Let $\mathbf{\Pi}$ denote the set of all hypergraph matchings of the hypergraph $\mathcal{H}$. Assume that if hyperarc (i,J) is active, it transmits at a given rate $r_{iJ}(P_i^{\text{tot}}, \underline{S})$. We can represent a hypergraph matching π as an $|\mathcal{A}|$-dimensional rate vector ξ^π

$$\xi_{iJ}^{\pi} = \begin{cases} r_{iJ}(P_i^{\text{tot}}, \underline{S}), & \text{if } (i,J) \in \pi, \\ 0, & \text{otherwise.} \end{cases} \tag{3.18}$$

The achievable rate region $\text{Co}(\underline{r}(\underline{P}, \underline{S}))$ is then written as

$$\text{Co}(\underline{r}(\underline{P}, \underline{S})) \triangleq \left\{ \mathbf{r} : \mathbf{r} = \sum_{\pi\in\mathbf{\Pi}} \alpha_\pi \xi^\pi,\ \alpha_\pi \geq 0, \sum_{\pi\in\mathbf{\Pi}} \alpha_\pi = 1 \right\}. \tag{3.19}$$

[1] A hypergraph matching is defined as a set of hyperarcs with no pair incident to the same node.

Note that $\mathrm{Co}(\underline{r}(\underline{P}, \underline{S}))$ is a polytope. So, we can always pick up an extreme point maximizer for the scheduling problem (3.13), which corresponds to a maximum weighted hypergraph matching in $\mathcal{H}$.

We first transform the directed hypergraph to an equivalent undirected hypergraph $\tilde{\mathcal{H}} = (\mathcal{V}, \mathcal{E}_h)$, where $\mathcal{H}$ and $\tilde{\mathcal{H}}$ have the same node set. Note that hyperarcs (i, J) and (j, I) mutually interfere and have the same interference/contention relations with other hyperarcs if $\{i\} \cup J = \{j\} \cup I$. Define an undirected hyperedge $e \subseteq \mathcal{V}$ in $\mathcal{E}_h$, which corresponds to all hyperarcs (i, J) such that $e = \{i\} \cup J$. The weight of hyperedge $e \in \mathcal{E}_h$ is

$$\tilde{w}_e = \max_{\{(i,J)\in\mathcal{A},\, \{i\}\cup J = e\}} w_{iJ} r_{iJ}(\underline{P}, \underline{S}). \tag{3.20}$$

The problem (3.13) is then equivalent to the maximum weighted hypergraph matching (or maximum weighted set packing) problem on the weighted hypergraph $\tilde{\mathcal{H}}$.

Unlike the maximum weighted matching problem on graphs which can be computed in polynomial time, the maximum weighted hypergraph matching problem is NP-complete [56]. Also, we would like distributed algorithms. Both factors suggest that we should focus on approximation algorithms.

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be an undirected graph with the same node set as $\mathcal{H}$. We assume that there exists an edge between node i and node j if and only if $\min\{\mathrm{SNR}_{ij}, \mathrm{SNR}_{ji}\} \geq \lambda$, where λ is a predefined threshold. This means that if i can hear j, then j can hear i. Let $N(v)$ denote the neighbor node set of node v in $\mathcal{G}$. We call $\mathcal{G}$ the *connectivity graph* in the following.

3.5.2 Local Optimal Algorithms

A linear time approximation algorithm with bounded worst-case performance for maximum weighted graph matching is proposed in [66], which adds a locally optimal edge into the matching at each step. Motivated by [66], our algorithm adds a locally heaviest hyperedge into the hypergraph matching at each step.

Definition 3.2 ***(locally heaviest hyperedge)****: A hyperedge e is a locally heaviest hyperedge if its weight is at least as large as the weight of all adjacent hyperedges, i.e.,*

$\tilde{w}_e \geq \tilde{w}_f$ for all $f \in \mathcal{E}_h$ such that $f \cap e \neq \emptyset$.

The distributed local optimal hypergraph matching algorithm (DLOHMA) is given in Algorithm 3.1. In Algorithm 3.1, the set Γ_i keeps track of the set of neighbors of node i that are still not matched, which is initialized to be all its neighbors in $\mathcal{G}$. Node i also maintains, for each neighbor $j \in \Gamma_i$, the set $\Gamma_j^{(i)}$ which keeps track of the set of unmatched neighbors of j, and knows the queue lengths of its two hop neighbors. This allows node i to compute the weight $\tilde{w}_e$ of any edge e involving itself, as defined in (3.20). The vector C_i counts the number of matching e_i^* messages that have been received, which is initialized to the null vector (line 3). Each node i broadcasts a matching e_i^* message, where $e_i^* = \{i\} \cup J^*$ is the maximum weight hyperedge in $\tilde{\mathcal{H}}$ containing i (lines 5-8). If node i receives $|J^*|$ matching e_i^* messages, hyperedge e_i^* is added to the hypergraph matching as e_i^* is a locally heaviest hyperedge. It broadcasts a drop e_i^* message to indicate that i is matched and unavailable, and at the same time to tell all nodes in e_i^* that they are matched (lines 26-28). If node i receives a drop e message and node i is not in e, it first checks whether some nodes of e are in Γ_i. If so, i is the direct neighbor of some nodes in e and i broadcasts the drop e message to let i's neighbors (two-hop neighbors of the nodes in e) know that all the nodes in e are matched. If not, i does not need to forward the drop message. Node i then removes the nodes in e from Γ_i and all $\Gamma_j^{(i)}$, $j \in \Gamma_i$. Furthermore, if some nodes in J^* are in e, the hyperedge e_i^* is dropped. Node i then finds another candidate set J^*, and it broadcasts a new matching e_i^* message (lines 16-23). If i receives a drop e message and node i is in e, i will broadcast a drop e message if it did not do so before, i.e., Γ_i is nonempty (line 23).

Note that some nodes in the locally heaviest hyperedge may not be able to hear each other. These nodes cannot receive $|J^*|$ matching e messages and conclude that e is the locally heaviest hyperedge. But at least one node can hear all the other nodes in the hyperedge. This is the reason why we broadcast a drop e message in line 27.

In Algorithm 3.1, we assume that all hyperedges have different weights. If they do not, we can always break ties by adding a small constant ϵ_e to w_e such that $\epsilon_e \neq \epsilon_{e'}$

for all $e \neq e'$ in $\mathcal{E}$. For example, we can change w_{iJ} or r_{iJ} by a small constant. We also assume that the cardinality of all hyperedges in $\mathcal{E}_h$ is bounded from above by a constant K. Let $\kappa = \max_{m \in \mathcal{M}} |\mathcal{T}_m| + 1$.

```
   DLOHMA: $(\mathcal{G})$
1  for each node $i \in \mathcal{V}$ do
2      Broadcast the set $\{\text{SNR}_{ij} | j \in N(i)\}$ to all its neighbor nodes ;
3      Set $C_i = \emptyset$, $\Gamma_i = N(i)$, and $\Gamma_j^{(i)} = N(j)$, $\forall j \in N(i)$;
4  end
5  for each node $i \in \mathcal{V}$ do
6      Find a node set $J^*$ by $J^* = \{j^*\} \cup L^* - i$ where $j^*, L^*$ are obtained via
           $(j^*, L^*) = \underset{j \in \Gamma_i \cup \{i\}, \left\{L | L \subseteq \Gamma_j^{(i)}, i \in L\right\}}{\arg\max} w_{jL} r_{jL}(\underline{P}, \underline{S}),$   (3.21)
       and $w_{jL}$, $r_{jL}$ are defined in (3.13) ;
7      if $J^* \neq \emptyset$ then Broadcast a matching $e_i^* = \{i\} \cup J^*$ message;
8  end
9  while $\exists i$, $\Gamma_i \neq \emptyset$ do
10     if node $i$ receives a message $m$ which is has not received then
11         switch $m$ do
12             case matching $e$
13                 $C_i(e) = C_i(e) + 1$;
14             end
15             case drop $e$
16                 if $i \notin e$ then
17                     if $e \cap \Gamma_i \neq \emptyset$ then Broadcast a drop $e$ message;
18                     Remove the nodes in $e$ from $\Gamma_i$ and all $\Gamma_j^{(i)}$, $j \in \Gamma_i$;
19                     if $e \cap J^* \neq \emptyset$ then
20                         Find a node set $J^*$ by (3.21);
21                         if $J^* \neq \emptyset$ then Broadcast a matching $e_i^* = \{i\} \cup J^*$
                           message;
22                     end
23                 else if $\Gamma_i \neq \emptyset$ then Broadcast a drop $e$ message, and set $\Gamma_i = \emptyset$;
24             end
25         end
26         if $J^* \neq \emptyset$ and $C_i(e_i^*) = |J^*|$ then
27             Broadcast a drop $e_i^*$ message, and set $\Gamma_i = \emptyset$;
28         end
29     end
30 end
```

Algorithm 3.1: Distributed local optimal algorithm.

Proposition 3.3 *The hyperedge e_i^* in line 26 is a locally heaviest hyperedge.*

Proof. From (3.21), $\tilde{w}_e \geq \tilde{w}_f$ for any f that contains i. If node i receives $|J^*|$ matching e messages, we can conclude that $\tilde{w}_e \geq \tilde{w}_f$ for any f that contains i. Therefore, we have $\tilde{w}_e \geq \tilde{w}_f$ for any f such that $f \cap e \neq \emptyset$, and e is a locally heaviest hyperedge. ■

Proposition 3.4 *In Algorithm 3.1, each node i broadcasts at most $\sum_{j \in N(i)} |N(j)| + |N(i)|$ messages.*

Proof. When the algorithm begins, node i first broadcasts a matching message (line 7). After that it broadcasts a matching message only when it receives a drop e message from one of its neighbors and it is not in e. It initiates a drop message only when it gets matched. After that $\Gamma_i = \emptyset$ and no more messages will be sent. It forwards a drop e message only when $i \notin e$ and $e \cap \Gamma_i \neq \emptyset$. From Algorithm 3.1, node i can receive at most $N(i)$ drop messages initiated by its one-hop neighbors and $\sum_{j \in N(i)} (|N(j)| - 1)$ drop messages initiated by its two-hop neighbors. Therefore, the worst case is broadcasting $\sum_{j \in N(i)} |N(j)| - 1$ matching e messages and forwarding $|N(i)|$ drop e messages. This require broadcasting $\sum_{j \in N(i)} |N(j)| + |N(i)|$ messages. ■

Unlike [66], where node i only sends a message to node j, we make use of the broadcast property of wireless communication, which reduces the number of messages.

Theorem 3.5 *The Algorithm 3.1 runs in $O\left(K^3 |\mathcal{E}| \sum_{k=1}^{\min\{\kappa, K\}-1} \binom{K-1}{k}\right)$ time, and the number of time-slots required to finish Algorithm 3.1 is $O(|\mathcal{V}|)$.*

Proof. By Proposition 3.4, each node broadcasts at most $\sum_{j \in N(i)} |N(j)| + |N(i)|$ messages. Thus, there are at most $\sum_{i \in \mathcal{V}} \sum_{j \in N(i)} |N(j)| + |N(i)| \leq (2K+1)|\mathcal{E}|$ broadcasted messages. Each broadcasted message is received by at most $K-1$ neighbor nodes. Therefore, all the nodes receive at most $(2K+1)(K-1)|\mathcal{E}|$ messages. The **while** loop of Algorithm 3.1 (lines 9-30) has at most $(2K+1)(K-1)|\mathcal{E}|$ iterations. In each iteration, we need to perform (3.21) at most once. We can solve (3.21) by performing the inner max first with fixed j and then the outer max by varying j. By the definition of w_{iJ} and assuming that $r_{iJ}(\underline{P}, \underline{S})$ in (3.13) is attained when i sends common information to nodes in J, we can write the inner max of (3.21) as

$$\max_{\{L | L \subseteq \Gamma_j,\, i \in L\}} \min_{l \in L} r_{jl}(\underline{P}, \underline{S}) \cdot \left(\sum_t \max_{s, l \in L} \left[q_j^{mst} - q_l^{mst} \right]^+ \right), \tag{3.22}$$

where $r_{jl}(\underline{P}, \underline{S})$ is the point to point channel capacity of link (j, l). Clearly, given any set L with $|L| > \kappa - 1$, we can always find a subset L' of L such that the weight of L'

is at least that of L because $\sum_t \max_{s,l\in L}\left[q_j^{mst}-q_l^{mst}\right]^+$ in (3.22) contains at most $\kappa-1$ summands, and

$$\min_{l\in L} r_{jl}(\underline{P},\underline{S}) \le \min_{l\in L'} r_{jl}(\underline{P},\underline{S}),\ \forall L'\subseteq L. \tag{3.23}$$

Therefore, we only need to consider those L with $|L|\le\kappa-1$. The number of such L's is at most $\sum_{k=1}^{\min\{\kappa,K\}-1}\binom{K-1}{k}$. Also the number of j in $\Gamma_i\cup\{i\}$ is at most K. Thus, the complexity[2] of solving (3.21) is $O\left(K\sum_{k=1}^{\min\{\kappa,K\}-1}\binom{K-1}{k}\right)$ and the complexity of Algorithm 3.1 is $O\left(K^3|\mathcal{E}|\sum_{k=1}^{\min\{\kappa,K\}-1}\binom{K-1}{k}\right)$.

On the other hand, Algorithm 3.1 is a parallel algorithm. We assume that every message broadcast takes one time-slot. It is easy to see that at least one locally heaviest hyperedge always exists. Let t denote the time-slot that a locally heaviest hyperedge e is found through line 26. Note that at least one node in a hyperedge can hear all the other nodes. From this node, it takes at most one time-slot to let all the nodes in e know that they are matched. It takes at most two time-slots to have this drop e message propagate to all two-hop neighbors of the nodes in e. It takes one time-slot for all one-hop and two-hop neighbors of nodes in e to send a new matching message. Therefore, at the $t+4$ time-slot, we can find the next locally heaviest hyperedge. By removing the nodes in the locally heaviest hyperedge, the number of nodes in $\tilde{\mathcal{H}}$ is reduced at least by two. Therefore, by induction, the algorithm takes at most $O(|\mathcal{V}|)$ time-steps. ■

If we do not consider the complexity of computing (3.21) as in [66], Algorithm 3.1 runs in linear time in the number of edges in the connectivity graph, i.e., $|\mathcal{E}|$ (not $|\mathcal{E}_h|$), which has the same complexity as the algorithms in [66] for finding a locally heaviest matching. This is because we use the broadcast advantage of the wireless communication.

Theorem 3.6 *Algorithm 3.1 computes a hypergraph matching HM_{LO} with at least $\max\{\frac{1}{K},\frac{1}{\kappa}\}$ of the weight of a maximum weighted hypergraph matching HM_{MW}.*

[2] We can sort $\left[q_j^{mst}-q_l^{mst}\right]^+$ and $r_{j,l}$ beforehand so that computing $\max_{l\in L}\left[q_j^{mst}-q_l^{mst}\right]^+$ and $\min_{l\in L} r_{j,l}$ takes $O(1)$ time.

Proof. We show this by induction. Let HM_{LO}^i be the hypergraph matching set after the i-th hyperedge is added, and V_{LO}^i be the set of matched vertices in HM_{LO}^i. The total weight of all the hyperedges in M is denoted as $W(M)$. We need to show that for all i, the following is true

$$W(HM_{\text{LO}}^i) \geq \max\left\{\frac{1}{K}, \frac{1}{\kappa}\right\} W\left(\{e|e \in HM_{\text{MW}}, e \cap V_{\text{LO}}^i \neq \emptyset\}\right). \tag{3.24}$$

Clearly, (3.24) is true for $i = 0$ as $HM_{\text{LO}}^i = \emptyset$. We assume (3.24) is true for $i = k - 1$. Let e^k be the k-th hyperedge added into HM_{LO}. By Proposition 3.3 and the definition of locally heaviest hyperedge,

$$W(e^k) \geq W(e), \forall e \in \mathcal{E}_h, \text{ and } e \cap V_{\text{LO}}^{k-1} = \emptyset. \tag{3.25}$$

All the hyperedges adjacent to the nodes in V_{LO}^{k-1} have been excluded according to Algorithm 3.1. Therefore, we have

$$W(e^k) \geq W(e), \forall e \in HM_{\text{MW}}, e \cap V_{\text{LO}}^{k-1} = \emptyset, \text{ and } e \cap e^k \neq \emptyset. \tag{3.26}$$

Similar to the argument in Theorem 3.6, the size of e^k is at most $\min\{K, \kappa\}$. Thus, e^k intersects with at most $\min\{K, \kappa\}$ hyperedges in HM_{MW}, which indicates

$$W(e^k) \geq \max\left\{\frac{1}{K}, \frac{1}{\kappa}\right\} W\left(\left\{e|e \in HM_{\text{MW}}, e \cap V_{\text{LO}}^{k-1} = \emptyset, \text{ and } e \cap e^k \neq \emptyset\right\}\right). \tag{3.27}$$

Adding both sides of (3.27) and (3.25) with $i = k - 1$, we find (3.25) is still true for $i = k$. Therefore, (3.25) is true for any i, and the theorem is proved. ■

In Algorithm 3.1, some matched nodes may not contribute much to a locally heaviest hyperedge. When these nodes are matched in other hyperedges, they may contribute more, which results in a hypergraph matching with higher weight. Instead of choosing the hyperedge according to its weight, we use the average hyperedge weight, i.e., $\bar{w}_e = \tilde{w}_e/|e|$. We modify Algorithm 3.1 to **Algorithm 3.2** by simply replacing $\tilde{w}_e$ with $\bar{w}_e$. Both the complexity and the approximation ratio of Algorithm 2 are identical to those of Algorithm 3.1.

Theorem 3.7 *Algorithm 3.2 computes a hypergraph matching HM_{LO2} with at least $\max\{\frac{1}{K}, \frac{1}{\kappa}\}$ of the weight of a maximum weighted hypergraph matching HM_{MW}.*

Proof. We show this by induction as Theorem 3.6. Let HM_{LO2}^i be the hypergraph matching set after the i-th hyperedge is added, and V_{LO2}^i be the set of matched vertices in HM^i_{LO2}. The total weight of all the hyperedges in M is denoted as $W(M)$. We need to show that for all i, the following is true

$$W(HM^i_{\mathrm{LO2}}) \geq \max\left\{\frac{1}{K}, \frac{1}{\kappa}\right\} W\left(\{e|e \in HM_{\mathrm{MW}},\, e \cap V^i_{\mathrm{LO2}} \neq \emptyset\}\right). \tag{3.28}$$

Clearly, (3.28) is true for $i = 0$ as $HM^i_{\mathrm{LO2}} = \emptyset$. We assume (3.28) is true for $i = k-1$. Let e^k be the k-th hyperedge added into HM_{LO2}. By Proposition 3.3 and the definition of locally heaviest hyperedge, we have

$$\bar{w}_{e^k} = \frac{W(e^k)}{|e^k|} \geq \frac{W(e)}{|e|} = \bar{w}_e,\ \forall e \in \mathcal{E}_h,\ \text{ and } e \cap V^{k-1}_{\mathrm{LO2}} = \emptyset. \tag{3.29}$$

Therefore, we have

$$\frac{W(e^k)}{|e^k|} \geq \frac{W(e)}{|e|},\ \forall e \in HM_{\mathrm{MW}},\ e \cap V^{k-1}_{\mathrm{LO}} = \emptyset,\ \text{ and } e \cap e^k \neq \emptyset. \tag{3.30}$$

We then have

$$\begin{aligned}\sum_{\{e\in HM_{\mathrm{MW}},\, e\cap V^{k-1}_{\mathrm{LO}}=\emptyset,\text{ and } e\cap e^k\neq\emptyset\}} W(e) &= \sum_{\{e\in HM_{\mathrm{MW}},\, e\cap V^{k-1}_{\mathrm{LO}}=\emptyset,\text{ and } e\cap e^k\neq\emptyset\}} |e|\frac{W(e)}{|e|} \\ &\leq \sum_{\{e\in HM_{\mathrm{MW}},\, e\cap V^{k-1}_{\mathrm{LO}}=\emptyset,\text{ and } e\cap e^k\neq\emptyset\}} |e|\frac{W(e^k)}{|e^k|} \\ &\leq \min\{K,\kappa\}W(e^k).\end{aligned} \tag{3.31}$$

Thus, (3.31) gives

$$W(e^k) \geq \max\left\{\frac{1}{K}, \frac{1}{\kappa}\right\} W\left(\left\{e|e \in HM_{\mathrm{MW}},\, e \cap V^{k-1}_{\mathrm{LO2}} = \emptyset,\ \text{ and } e \cap e^k \neq \emptyset\right\}\right). \tag{3.32}$$

Adding both sides of (3.32) and (3.28) with $i = k-1$, we find (3.28) is still true for $i = k$. Therefore, (3.28) is true for any i, and the theorem is proved. ■

Theorem 3.8 *Both Algorithm 3.1 and Algorithm 3.2 stabilize the system for any rate vector $\vec{x}$ such that $\vec{x} + \epsilon \in \max\{\frac{1}{K}, \frac{1}{\kappa}\}\mathbf{\Lambda}$ for an arbitrarily small $\epsilon \succ 0$.*

Proof. We only show the stability of Algorithm 3.1. Algorithm 3.2 can be shown similarly. Let $\vec{x}$ be any rate vector such that $\check{\vec{x}} + \min\{K,\kappa\}\epsilon \in \mathbf{\Lambda}$. Therefore, there exist flow variables $\check{g}^{mst}_{iJj}$, $\check{f}^m_{iJ}$ and rate variable $\check{r}_{i,J}$ such that the constraints in (3.6) are all satisfied with $x^{ms} = \check{x}^{ms} + \epsilon$ for an arbitrarily small $\epsilon \succ 0$. Let $\tilde{\vec{x}} = \max\{\frac{1}{K}, \frac{1}{\kappa}\}\check{\vec{x}}$, $\tilde{g}^{mst}_{iJj} = \max\{\frac{1}{K}, \frac{1}{\kappa}\}\check{g}^{mst}_{iJj}$, $\tilde{f}^m_{iJ} = \max\{\frac{1}{K}, \frac{1}{\kappa}\}\check{f}^m_{iJ}$ and $\tilde{r}_{i,J} = \max\{\frac{1}{K}, \frac{1}{\kappa}\}\check{r}_{i,J}$. Let $Q^{mst}_i(\tau)$ be the amount of session m data queued at node i for source s and sink t at time

τ, and μ_{iJj}^{mst} be the rate offered to sink t of session m from source s for destination j over link (i, J). Define the Lyapunov function $L(\underline{Q}) = \sum_{i,m,s,t} \left(Q_i^{mst}\right)^2$. Suppose the input rates are $\tilde{x}^{ms}$. By following the same line of proof as in [39, 42], we obtain

$$\begin{aligned}
&E\left\{L(\underline{Q}(\tau+T)) - L(\underline{Q}(\tau))|\underline{Q}(\tau)\right\} \leq 2T^2B|\mathcal{N}| - \\
&2T\sum_{i,m,s,t} Q_i^{mst}(\tau)\left[E\left\{\sum_{\{J|(i,J)\in\mathcal{A}\}}\sum_{j\in J}\mu_{iJj}^{mst} - \sum_{j\in\mathcal{N}}\sum_{\{i|(j,I)\in\mathcal{A},\, i\in I\}}\mu_{jIi}^{mst}\middle|\underline{Q}(\tau)\right\} - \tilde{\sigma}_i^{ms}\right] \\
&\overset{(a)}{=} 2T^2B|\mathcal{N}| - 2T\sum_{(i,J),m,t}\sum_{s,j\in J} E\left\{\mu_{iJj}^{mst}|\underline{Q}(\tau)\right\}\left(Q_i^{mst} - Q_j^{mst}\right) \\
&+ 2T\sum_{(i,J),m,t}\sum_{s,j\in J}\tilde{g}_{iJj}^{mst}\left(Q_i^{mst} - Q_j^{mst}\right) - 2T\sum_m |\mathcal{S}_m|\epsilon,
\end{aligned} \tag{3.33}$$

where B is a constant defined in [42]. Note that Q_i^{mst} is a scaled version of q_i^{mst}. Thus, the second term in (a) is equivalent to the objective function in the session scheduling problem (3.8). Let W_{MW} and W_{LO} be the values of the second term in (a) with maximum weighted hypergraph matching and Algorithm 3.1, respectively. From Theorem 3.6, we have $W_{LO} \geq \max\{\frac{1}{K}, \frac{1}{\kappa}\}W_{MW}$. On the other hand, as maximum weighted hypergraph matching solves (3.8) optimally, we have

$$W_{MW} \geq \sum_{(i,J),m,t}\sum_{s,j\in J}\check{g}_{iJj}^{mst}\left(Q_i^{mst} - Q_j^{mst}\right), \tag{3.34}$$

where $\check{g}_{iJj}^{mst}$ is also a feasible solution to (3.8). Multiply both sides of (3.34) by $\max\{\frac{1}{K}, \frac{1}{\kappa}\}$, we obtain

$$W_{LO} \geq \sum_{(i,J),m,t}\sum_{s,j\in J}\tilde{g}_{iJj}^{mst}\left(Q_i^{mst} - Q_j^{mst}\right). \tag{3.35}$$

Applying the Lyapunov drift lemma of [64] shows that Algorithm 3.1 stabilizes the system with rate vector $\vec{\tilde{x}}$ with $\vec{\tilde{x}} + \epsilon \in \max\{\frac{1}{K}, \frac{1}{\kappa}\}\mathbf{\Lambda}$. ■

The complexity of algorithms 3.1 and 3.2 results in part from the need to propagate te drop e message to all two-hop neighbors of the nodes in e. If we assume that any node can receive the drop e message from its two-hop neighbors, the complexity in Theorem 3.5 can be decreased by a factor of K.

3.5.3 Randomized Algorithm

In this subsection, we consider randomized algorithms to find a maximal hypergraph matching.

Definition 3.9 ***(maximal hypergraph matching):*** *A hypergraph matching HM is maximal if for each hyperedge $e \in \tilde{\mathcal{H}}$, at least one of the following conditions is satisfied:*

- *$e \cap HM \neq \emptyset$, i.e., e has non-empty intersection with at least one hyperedge in HM.*
- *$\tilde{w}_e = 0$, i.e., the number of packets waiting to be transmitted over the hyperedge is zero.*

A distributed randomized hypergraph matching algorithm (DRHMA) is given in Algorithm 3.3. The input of Algorithm 3.3 is a graph $\mathcal{G}'$, which is obtained after deleting all the directed edges (i, j) with $\max\limits_{m,s,t} \left[q_i^{mst} - q_j^{mst}\right]^+ = 0$ from $\mathcal{G}$. This guarantees that all the hyperedges have positive weights. In Algorithm 3.3, the set Γ_i keeps track of the set of neighbors of node i that are still not matched, which is initialized to be all its neighbors in $\mathcal{G}$ (line 1). In each time slot, each unmatched node i attempts to transmit with probability $\frac{1}{|\Gamma_i|}$ (line 5). This choice of probability value is similar to that in [62] for the maximal independent sets problem. If i attempts to transmit, for each neighbor j, it sends a matching request to j with probability $1/2$ (line 6). If i sends request to at least one neighbor, i.e., $S_i \neq \emptyset$, it decides to transmit (line 9). E_i denotes the hyperedge to be added into the matching initialized by i (line 10). If node i does not transmit and it receives several matching requests from its neighbors, it chooses one of them uniformly at random, say j, sets $\Gamma_i = \emptyset$ (i is matched), and broadcasts an "i matched j" message (lines 13-18). Upon receiving an "i matched j" message, node k checks whether $k = j$. If $k = j$, this indicates that i got the matching request from k and it would like to join in the hyperedge initialized by k. Thus, k sets $E_k = E_k \cup \{i\}$ (line 20). If $k \neq j$, this indicates that i got matched to j and k should delete i from Γ_k (line 21). For each node i that decides to transmit, if finally $E_i \neq \emptyset$, E_i is added into the hypergraph matching, and we set $\Gamma_i = \emptyset$ (i is matched). Algorithm 3.3 returns a maximal hypergraph matching.

```
DRHMA: ($\mathcal{G}'$)
for each node $i \in \mathcal{V}$ do  Set $\Gamma_i = N(i)$;
while $\exists i, \Gamma_i \neq \emptyset$ do
    for each node $i \in \mathcal{V}$ and $\Gamma_i \neq \emptyset$ do
        Let $p$ be a random number generated according to the uniform
        distribution on $[0, 1]$.
        if $p < \frac{1}{|\Gamma_i|}$ then
            For each node $j \in \Gamma_i$, with probability $\frac{1}{2}$ add $j$ into set $S_i$;
        end
        if $S_i \neq \emptyset$ then
            Node $i$ decides to transmit and it broadcasts matching messages to
            all nodes in $S_i$;
            Set $E_i = \emptyset$;
        end
    end
    for each node $i \in \mathcal{V}$ and node $i$ does not transmit do
        if node $i$ receives matching messages from several neighbors then
            Node $i$ chooses one of them uniformly at random, say $j$, and sets
            $\Gamma_i = \emptyset$;
            Node $i$ broadcasts a $i$ matched $j$ message;
        end
    end
    while $\exists k$, $k$ receives a $i$ matched $j$ message do
        if  $k = j$ then $E_k = E_k \cup \{i\}$;
        else  $\Gamma_k = \Gamma_k - \{i\}$;
    end
    for each node $i \in \mathcal{V}$, and if $i$ decides to transmit do
        if $E_i \neq \emptyset$ then $E_i$ is added into the hypergraph matching, and set $\Gamma_i = \emptyset$;
    end
end
```

Algorithm 3.3: Distributed randomized hypergraph matching algorithm.

Theorem 3.10 *The expected run time of Algorithm 3.3 is* $O(\log |\mathcal{E}|)$.

Proof. We first give some definitions. A node $v \in \mathcal{V}$ is *bad* if more than 2/3 of the neighbors of v are of higher degree than v. A node is *good* if it is not bad. An edge $e \in \mathcal{E}$ is *bad* if both of its endpoints are bad; otherwise the edge is *good.* To show the expected running time of Algorithm 3.3, we need to show the expected number executions of the **while** loop in Algorithm 3.3. Let $\mathcal{G}_i = (\mathcal{V}_i, \mathcal{E}_i)$ denote the graph after i executions of the while loop, where we only consider those nodes with $\Gamma_v \neq \emptyset$ in $\mathcal{V}_i$.

Let v be a good node of degree $d = |\Gamma_v| > 0$ in $\mathcal{G}_i$ and the neighbors of v be $u_1, \ldots, u_d$. The vertex v has $k \geq \lceil \frac{1}{3}d \rceil$ neighbors such that $d_j = |\Gamma_{u_j}| \leq d$, $j = 1, \ldots, k$. According to Algorithm 3.3, the probability that node u_j sends a matching request to v is $1/(2d_j) \geq 1/(2d)$. The probability that $u_1, \ldots, u_k$ do not broadcast a matching message to v is

$$\prod_{j=1}^{k} \left(1 - \frac{1}{2d_j}\right) \leq \left(1 - \frac{1}{2d}\right)^{d/3} < e^{-1/6}. \tag{3.36}$$

Therefore, the probability that v receives at least one matching message from its neighbors is greater than $1 - e^{-1/6} > 0$. Note that ignoring the matching messages from the neighbors with degree greater than d only decreases this probability. Node v responds to received matching messages only when it decides not to transmit, whose probability is $1 - \frac{1}{d} + \frac{1}{d}\frac{1}{2^d}$. It is not hard to show that $1 - \frac{1}{d} + \frac{1}{d}\frac{1}{2^d}$ is an increasing function in d when $d \geq 1$. Therefore, the probability that node v decides not to transmit is at least $\frac{1}{2}$. Finally, node v is included in the hypergraph matching with probability at least $\frac{1}{2}(1 - e^{-1/6})$. The edges incident to v are either included in the hypergraph matching or deleted from $\mathcal{E}_i$. Note that every good edge is incident with at least one good node. According to Lemma 12.6 in [62], at least half the edges in $\mathcal{E}_i$ are good. Thus, the expected number of edges removed from $\mathcal{E}_i$ is at least $\frac{1}{4}(1 - e^{-1/6})|\mathcal{E}_i|$ or

$$E(|\mathcal{E}_i||\mathcal{E}_{i-1}) \leq |\mathcal{E}_{i-1}|(1 - \alpha) \Rightarrow E(|\mathcal{E}_i|) \leq |\mathcal{E}|(1 - \alpha)^i, \tag{3.37}$$

where $\alpha = \frac{1}{4}(1 - e^{-1/6})$. Therefore, the expected number executions of the **while** loop in Algorithm 3.3 is $O(\log|\mathcal{E}|)$. Each **while** loop requires 2 time-slots and the expected running time of Algorithm 3.3 is also $O(\log|\mathcal{E}|)$. ■

Compared with Algorithm 3.1, Algorithm 3.3 not only reduces the time complexity from $O(|\mathcal{E}|)$ to $O(\log|\mathcal{E}|)$ but also does not need to compute the weight of each hyperedge. Using a similar approach to that in [16], if we assume that in each session all sinks are only one hop away from the source, we can show the following theorem on the performance of the randomized algorithm.

Theorem 3.11 *If in each session all sinks are only one hop away from the source, then Algorithm 3.3 stabilizes the system for any rate vector within* $\frac{1}{K}\mathbf{\Lambda}$.

Algorithm 3.3 can be readily turned into a constant-time algorithm by executing the while loop in Algorithm 3.3 only M times. We call this algorithm **Algorithm 3.4**.

As maximal matching plays an important role in many scheduling algorithms, see, e.g., [61, 74], we expect that Algorithm 3.3 can also serve as a basis for other scheduling algorithms for our problem. Note that the approach in [61] cannot be trivially adopted as the connected components in the union of the new hypergraph matching and the old hypergraph matching may be very large. Also, the connected components are not simply cycles or paths as in [61].

3.5.4 Hybrid Algorithm

Algorithms 3.1 and 3.2 perform well but with high complexity, while Algorithms 3.3 and 3.4 have low complexity but have a poor performance guarantee as they do not take into account the weight of hyperedge. We next consider combining these two types of algorithms to take advantage of the strengths of both.

In the hybrid algorithm, we first run Algorithm 3.1 for T_{th} time slots. To speed up Algorithm 3.1, we execute the while loop of Algorithm 3.3 once at the end of T_{th} time slots. We then continue running Algorithm 3.1. The process continues until there does not exist a node i such that $\Gamma_i \neq \emptyset$. Clearly, if $T_{th} = 0$, the hybrid algorithm reduces to Algorithm 3.3, while if $T_{th} = \infty$, the hybrid algorithm reduces to Algorithm 3.1. Thus T_{th} is used to control the tradeoff between complexity and performance. Similarly, Algorithm 3.2 can also be combined with Algorithm 3.3. We call this algorithm **Algorithm 3.5**. In Algorithm 3.1, each node needs to wait until all its neighbors are included in some local heaviest hyperarc or its neighbors response to the matching request. By running Algorithm 3.3, each node can directly construct a hyperarc with its neighbors without waiting for its neighbors' decisions on the locally heaviest hyperarc. Thus, the hybrid algorithm can accelerate Algorithm 3.1. The running time of Algorithm 3.5 is between Algorithm 3.1 and Algorithm 3.3. Algorithm 3.5 also returns a maximal hypergraph matching, and thus Theorem 3.11 still applies.

Alternatively, we can apply the algorithms in [66] to find a maximum weighted matching first and then add the unmatched nodes into the hypergraph matching randomly.

Remark: In the previous discussion of the scheduling algorithms, we do not consider possible collision of coordinating/signalling messages in carrying out these algorithms. This issue is particularly relevant when we come to the implementation of the scheduling algorithm in real systems. We usually assume the existence of a separate control channel to do message passing. Or we divide a time slot into control mini-slots and data slot and message passing happens in control slots. There are basically two ways to coordinate message passing and resolve collisions. The first one is to have a "reservation" protocol to pre-specify who talks and in which order. The more common strategy is to use a random access scheme such as Aloha to coordinate message passing over the control channel or mini-slots.

3.6 Simulation Results

In this section, we provide numerical examples to complement the analysis in previous sections. We assume that node i's signal power is attenuated by a factor of $\rho_{i,j}^{-2}$ when the signal is received by node j, where $\rho_{i,j}$ is the Euclidean distance between i and j. All nodes have unit signal power and identical noise power 0.02. We assume the use of orthogonal spreading sequences and white Gaussian noise channels and compute r_{iJ} using

$$r_{iJ}(\underline{P}, \underline{S}) = \log\left(1 + \min_{j \in J} \mathrm{SNR}_{i,j}\right), \tag{3.38}$$

where $\text{SNR}_{i,j} = P_i^{\text{tot}} \frac{|h_{i,j}|^2}{\sigma_j^2}$ is the effective SNR from node i to node j, σ_j^2 is additive white Gaussian noise power at node j, and $h_{i,j}$ is the channel fading coefficient from node i to node j. We have neglected the spreading factor in (3.38). Two nodes i and j are considered to be connected if and only if the SNR is at least 1 over link (i, j) (i.e., SNR threshold $\lambda = 1$) or the distance between i and j is less than 7.07 meters. We choose log utility function $\log(x)$ for each source in all the experiments.

3.6.1 Wireless Butterfly Network

We first consider the wireless butterfly network in Fig. 3.1 with two sources s_1, s_2, two sinks t_1, t_2, and one relay node r. Each source multicasts data to both sinks. We thus only consider a single multicast session. We compare our cross-layer design with different scheduling algorithms in Section 3.5 to hypergraph matching scheduling without network coding, that in [18] with maximum weighted matching algorithm in [30] and local greedy matching algorithm in [66]. As the network is small, we also show the performance of our cross-layer design with maximum weighted hypergraph matching by formulating the matching problem as an integer programming and solving it exactly.

Fig. 3.2 shows the evolution of source s_1's rate versus the number of iterations with fixed stepsize $\gamma = 0.01$, where our cross-layer design with maximum weighted hypergraph matching, Algorithm 3.1 and Algorithm 3.2, the algorithm without network coding, the algorithm in [18] with maximum weighted graph matching and local greedy matching are compared. We observe that the rates of all algorithms converge to within a small neighborhood of the steady values after 500 steps as we have chosen a constant stepsize. Fig. 3.3 shows the evolution of source s_1's rate versus the number of iterations with fixed stepsize $\gamma = 0.01$, where our cross-layer design with maximum weighted hypergraph matching, Algorithm 3.3, Algorithm 3.4, and Algorithm 3.5, and the algorithm in [18] with maximum weighted graph matching are compared. Compared with Fig. 3.2, the rates of Algorithm 3.3, Algorithm 3.4, and Algorithm 3.5 oscillate more severely as all the algorithms use randomized mechanism, which

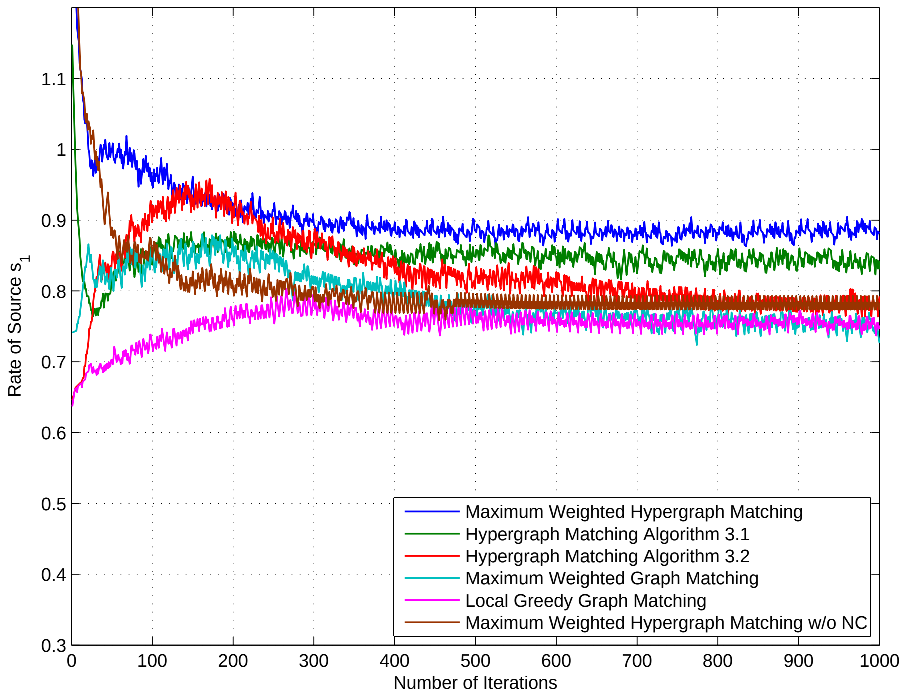


Figure 3.2: The evolution of source s_1's rate versus the number of iterations with fixed stepsize $\gamma = 0.01$ for the network in Fig. 3.1, where our cross-layer design with maximum weighted hypergraph matching, Algorithm 3.1 and Algorithm 3.2, and the algorithm in [18] with maximum weighted graph matching and local greedy matching are compared.

only guarantees the queue size at each node is finite all the time. We quantify the performance of different algorithms in Table 3.6.1, where $\mathrm{HM}_{\mathrm{opt}}$ denotes the maximum weighted hypergraph matching, $\mathrm{HM}_{\mathrm{alg}i}$ denotes Algorithm 3.i in Section 3.5, $\mathrm{HM}_{\mathrm{alg4},m}$ denotes Algorithm 3.4 with m time-slots, $\mathrm{HM}_{\mathrm{alg5},t}$ denotes Algorithm 3.5 with $T_{th} = t$, $\mathrm{HM}_{\mathrm{w/onc}}$ denotes the hypergraph matching algorithm without network coding, $\mathrm{M}_{\mathrm{opt}}$ denotes maximum weighted graph matching, and $\mathrm{M}_{\mathrm{lgd}}$ denotes local greedy graph matching. The first row shows the average rate by averaging the rate of different algorithms in Figs. 3.2 and 3.3 from 700th step to 1000th step. Row two shows rate gains of different algorithms over the maximum weighted graph matching.

We can see that our design with broadcast advantage and $\mathrm{HM}_{\mathrm{opt}}$ has about 17% gain over that without using broadcast advantage. Even with Algorithm 3.1, about 13% gain can still be achieved. The loss by using Algorithm 3.3, the randomized algorithm, is only 3.08% gain. A 11.81% gain can be realized by Algorithm 3.5. The

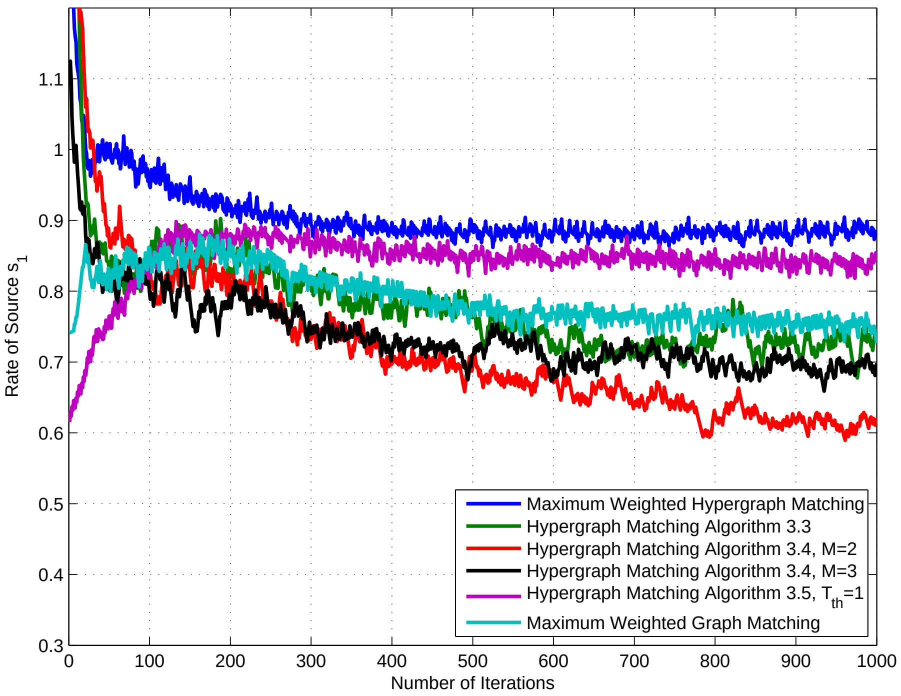


Figure 3.3: The evolution of source s_1's rate versus the number of iterations with fixed stepsize $\gamma = 0.01$ for the network in Fig. 3.1, where our cross-layer design with maximum weighted hypergraph matching, Algorithm 3.3, Algorithm 3.4, and Algorithm 3.5, and the algorithm in [18] with maximum weighted graph matching.

third row compares expected ratio between the weight of different algorithms and that of $\mathrm{HM_{opt}}$. It can be seen that $\mathrm{HM_{alg2}}$ has a greater ratio than both $\mathrm{M_{opt}}$ and $\mathrm{M_{lgd}}$ but they have the same throughput. This indicates that an algorithm that can return a heavier weight does not necessary achieve a higher throughput. Without network coding, the throughput gain over $\mathrm{M_{lgd}}$ is small, which is only 3.43%. Row four shows the average number of required time-slots by different algorithms. Surprisingly, both $\mathrm{HM_{alg1}}$ and $\mathrm{HM_{alg2}}$ require less time-slots than $\mathrm{M_{lgd}}$ does, but the former two have higher rates than the latter. This is because the broadcast advantage is exploited during scheduling, where one matching or drop message can reach several nodes. Also note that each hyperedge contains several nodes, which means that nodes are added faster into the hypergraph matching than graph matching. $\mathrm{HM_{alg5,1}}$ has a less number of time-slots than $\mathrm{M_{lgd}}$ but with a rate gain.

Table 3.1: Comparison of different algorithms in the wireless butterfly network

	$\mathrm{HM}_{\mathrm{opt}}$	$\mathrm{HM}_{\mathrm{alg1}}$	$\mathrm{HM}_{\mathrm{alg2}}$	$\mathrm{HM}_{\mathrm{alg3}}$	$\mathrm{HM}_{\mathrm{alg4,2}}$	$\mathrm{HM}_{\mathrm{alg4,3}}$	$\mathrm{HM}_{\mathrm{alg5,1}}$	$\mathrm{HM}_{\mathrm{w/onc}}$	$\mathrm{M}_{\mathrm{opt}}$	$\mathrm{M}_{\mathrm{lgd}}$
Average rate (bits/s)	0.8831	0.8486	0.7972	0.7327	0.6411	0.7062	0.8453	0.7819	0.7594	0.7560
Rate gain over $\mathrm{M}_{\mathrm{lgd}}$	0.1681	0.1225	0.0545	-0.0308	-0.1520	-0.0659	0.1181	0.0343	0.0045	0
Average $w/w_{\mathrm{HM}_{\mathrm{opt}}}$	1	0.9409	0.8928	0.8765	0.8829	0.8690	0.9250	0.9056	0.8666	0.8578
Average time-slots	-	4	3.9920	4.5500	3.3760	3.8532	5.0300	-	-	5

3.6.2 Random Networks

We next show the results on random networks. We assume N nodes are randomly and uniformly placed on a 20 meter by 20 meter square. Both source and sinks are randomly chosen from the 10 nodes. We consider only a single multicast session with one source and various number of sinks.

Tables 3.2-3.4 show the simulation results with 2, 4, and 6 sinks and $N = 10$ nodes in the network, and Table 3.5 shows the simulation results with 3 sinks and $N = 15$ nodes in the network. 1000 feasible network realizations are generated. Since the number of hyperedges becomes large as the size of network increases, it is hard to find the maximum weighted hypergraph matching by solving the integer programming directly. For comparison purposes, we take a suboptimal approach by computing the linear programming relaxation of the integer program first. In the next phase, we only consider the hyperedges with nonzero solution by the linear programming and solve the integer program with only those hyperedges. This method is denoted as $\mathrm{HM}_{\mathrm{sub}}$. From the tables, we can see that even with this suboptimal solution we can achieve a rate gain from 9.45% to 23.47%. Gain increases as the number of sinks increases. The same observation holds for all the other algorithms. On average, Algorithm 3.1 performs better than Algorithm 3.2. Algorithm 3.5 performs close to Algorithm 3.1 but with reduced complexity. Algorithm 3.3 has the worst performance but with the lowest complexity among all our proposed algorithms and a comparable throughput as the matching solution. The average number of edges in the connectivity graph is 21.52. The number of time-slots required by Algorithms 3.1, 3.2 and 3.5 is on the order of this number. The locally optimal matching algorithm performs close to the optimal matching algorithm, and has a lower complexity than the proposed hypergraph matching algorithms. However, hypergraph matching provides a throughput gain that increases with the number of sinks. Our results suggest that it is more advantageous to use hypergraph matching when the multicast group is large.

Table 3.2: Comparison of different algorithms in random networks with 10 nodes, 1 source and 2 sinks

	HM_{sub}	HM_{alg1}	HM_{alg2}	HM_{alg3}	$\text{HM}_{\text{alg5,1}}$	M_{opt}	M_{lgd}
Rate gain over M_{lgd}	9.45%	5.74%	4.32%	-6.46%	5.44%	2.30%	-
Average $w/w_{\text{HM}_{\text{sub}}}$	1	0.9749	0.9593	0.8584	0.9714	0.9623	0.9593
Average time-slots	-	13.38	13.46	5.88	9.63	-	6.15

Table 3.3: Comparison of different algorithms in random networks with 10 nodes, 1 source and 4 sinks

	HM_{sub}	HM_{alg1}	HM_{alg2}	HM_{alg3}	$\text{HM}_{\text{alg5,1}}$	M_{opt}	M_{lgd}
Rate gain over M_{lgd}	15.13%	8.78%	4.99%	-1.42%	8.23%	4.49%	-
Average $w/w_{\text{HM}_{\text{sub}}}$	1	0.9737	0.9697	0.9310	0.9729	0.9707	0.9697
Average time-slots	-	10.98	11.83	5.90	9.33	-	6.26

Table 3.4: Comparison of different algorithms in random networks with 10 nodes, 1 source and 6 sinks

	HM_{sub}	HM_{alg1}	HM_{alg2}	HM_{alg3}	$\text{HM}_{\text{alg5,1}}$	M_{opt}	M_{lgd}
Rate gain over M_{lgd}	23.47%	13.05%	5.50%	1.18%	12.15%	6.47%	-
Average $w/w_{\text{HM}_{\text{sub}}}$	1	0.9787	0.9776	0.9459	0.9785	0.9760	0.9776
Average time-slots	-	10.40	11.95	5.91	9.10	-	6.57

3.7 Conclusion

We have studied the cross-layer optimization for multicasting in wireless networks with wireless broadcast advantage. By designing distributed approximation algorithms for broadcast link scheduling, we gave fully distributed algorithms for joint rate control, network coding and scheduling. Numerical results have shown promising throughput gain by using the proposed algorithms, and surprisingly, in some cases with even lower complexity than the cross-layer design without the broadcast advantage. Our results suggest that broadcast link scheduling may be a promising avenue of further research.

Table 3.5: Comparison of different algorithms in random networks with 15 nodes, 1 source and 3 sinks

	$\mathrm{HM}_{\mathrm{sub}}$	$\mathrm{HM}_{\mathrm{alg1}}$	$\mathrm{HM}_{\mathrm{alg2}}$	$\mathrm{HM}_{\mathrm{alg3}}$	$\mathrm{HM}_{\mathrm{alg5,1}}$	$\mathrm{M}_{\mathrm{opt}}$	$\mathrm{M}_{\mathrm{lgd}}$
Rate gain over $\mathrm{M}_{\mathrm{lgd}}$	13.07%	7.79%	3.89%	1.39%	6.45%	4.73%	-
Average $w/w_{\mathrm{HM}_{\mathrm{sub}}}$	1	0.9813	0.9809	0.9769	0.9544	0.9806	0.9689
Average time-slots	-	15.75	15.41	6.99	11.73	-	7.32

Chapter 4

Multiple Access Random Medium Access Control

In this chapter, we develop a new class of random medium access control protocol, which allows each user to transmit at multiple data rates. By using successive interference cancellation, multiple packets can be received simultaneously. To achieve the desired throughput optimal equilibrium in a distributed fashion, a game-theoretic framework is proposed. We investigate the design of random access games, characterize their equilibria, study their dynamics, and propose distributed algorithms to achieve the equilibria.

4.1 Introduction

The medium access control (MAC) layer decides when competing nodes may access the shared medium. Different from schedule-based medium access requiring a central authority, multiple nodes share the medium by using random access in contention based MAC. Most conventional random access protocols such as Aloha [6] and carrier sense multiple access (CSMA) [49] assume simple collision models, where the channel is noiseless, and reception failure is caused by collisions among users. Though the analysis and protocol design are simple in the collision model, the maximum achievable throughput of this model is limited. With more sophisticated physical layer approaches, simultaneous reception of multiple packets is possible, for example, by using code division multiple access (CDMA) and multiuser detection. In order to represent such random access systems, a model for a channel with multipacket reception capability (MPR) with its stability property has been developed in [31]. A decentralized MAC protocol is proposed in [32]. In both works [31, 32], it is shown that the achievable throughput by using MPR is higher than that by using Aloha.

In MPR, each node transmits only at a single rate. On the other hand, a multiple access system with N users and one base station can be considered as a multiple access channel (mac). In the Gaussian noise case, if each user transmits with power P and the noise power at the base station is σ^2, the maximum information theoretic sum rate of all users is $\frac{1}{2}\log\left(1+\frac{NP}{\sigma^2}\right)$, which can be achieved with multirate transmission capability and successive interference cancelation (SIC) [22].

In this chapter, we develop a new class of MAC protocol by applying a SIC based approach at the MAC layer. The MPR model in [31, 32] is generalized by allowing each user to transmit at different data rates chosen randomly from an appropriately determined set of rates. By using SIC, multiple packets can be received simultaneously.

In slotted Aloha type networks with Gaussian channels, we show that the achievable sum rate of the new protocol using decentralized control is at least a constant fraction of that achievable by using centralized control, i.e., $\frac{C}{2}\log\left(1+\frac{NP}{\sigma^2}\right)$, $0 < C < 1$, where C can be interpreted as the distributive loss due to contention and lack of cooperation between users. This result suggests that the total throughput increases with N as opposed to Aloha where the total throughput decreases with N.

The proposed protocol is also extended to CSMA. We consider a half duplex single cell wireless LAN. By maximizing the achievable sum rate, we can obtain a desired transmission probability for each data rate, which depends on the number of users in the network.

In order to adaptively adjust the channel access probability as nodes join and leave, we consider a general game-theoretic model, called random access game, whose equilibrium is the desired throughput maximizing point. Dynamic algorithms such as best response and gradient play are proposed to achieve the equilibrium distributively without the knowledge of the number of nodes N. We show that under mild conditions

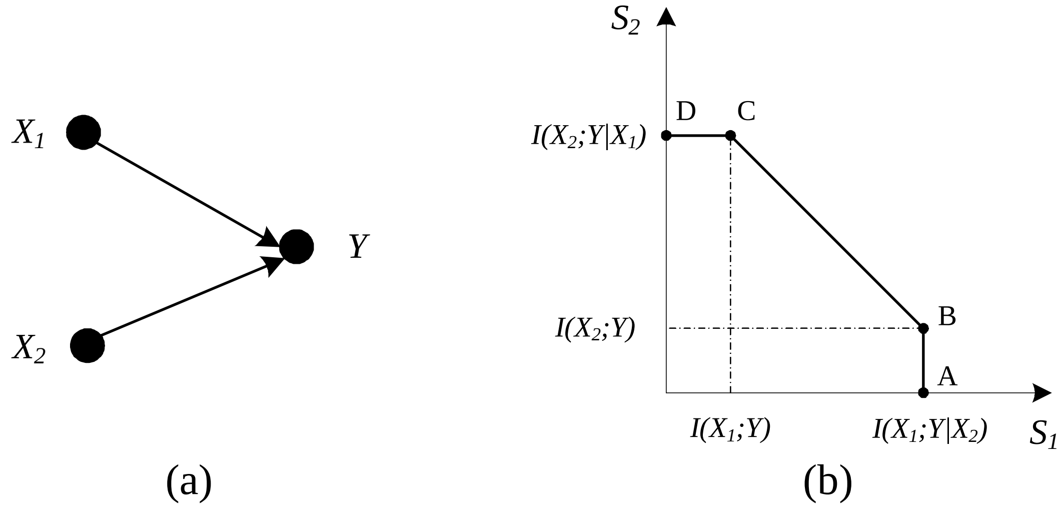


Figure 4.1: An illustration example.

all algorithms converge to the unique equilibrium. We also establish the convergence of gradient play under propagation delay and estimation error.

Finally, we consider extension of our multiple access scheme to rate splitting [69]. Rate splitting has been applied to Aloha in [14, 59]. We propose a new class of rate splitting algorithm where each virtual user can transmit at multiple potential transmission rates, which improves the achievable throughput.

Our simulation results support our analysis and show that the proposed protocol achieves a significant throughput gain over conventional Aloha. In a single cell WLAN, the proposed protocol not only achieves a higher throughput over the standard IEEE 802.11 DCF but also provides a better short term fairness.

4.2 A Motivating Example

We first consider a simple example to motivate this new MAC model. There are $N = 2$ users in the network, where user i's transmitted signal is X_i, $i = 1, 2$. Both users are saturated, i.e., they always have packets to send. The received signal at the base station is

$$Y = X_1 + X_2 + W, \tag{4.1}$$

where the average power of X_i, $i = 1, 2$ is P and the additive white Gaussian noise (AWGN) W is of zero mean and variance σ^2. Let S_i be the data sending rate of user i. From [22], the capacity region of the multiple access channel (4.1) is

$$\begin{aligned} S_1 <& I(X_1; Y|X_2) = \frac{1}{2}\log\left(1 + \frac{P}{\sigma^2}\right), \\ S_2 <& I(X_2; Y|X_1) = \frac{1}{2}\log\left(1 + \frac{P}{\sigma^2}\right), \\ S_1 + S_2 <& I(X_1, X_2; Y) = \frac{1}{2}\log\left(1 + \frac{2P}{\sigma^2}\right), \end{aligned} \tag{4.2}$$

where $I(X; Y)$ is the mutual information between random variables X and Y [22]. With a centralized controller, the maximum achievable sum rate is $R_c = \frac{1}{2}\log\left(1 + \frac{2P}{\sigma^2}\right)$. The corner point can be achieved by decoding user 2's signal first, treating user 1's signal as noise. The base station can then subtract the decoded signal from Y and decode user 1's signal. This is called successive interference cancelation. Similarly, corner point C can be achieved. The points on the line connecting B and C can be achieved by time sharing between B and C or by using rate splitting [69].

Using Aloha, we assume that each user transmits at rate $\frac{1}{2}\log\left(1 + \frac{P}{\sigma^2}\right)$ with probability p and remains idle with probability $1 - p$. The achievable sum rate of Aloha is

$$R_{\text{Aloha}}(p) = p(1-p)\log\left(1 + \frac{P}{\sigma^2}\right), \tag{4.3}$$

whose maximum is attained at $p = \frac{1}{2}$. The maximum achievable throughput is

$$R^*_{\text{Aloha}} = \frac{1}{4}\log\left(1 + \frac{P}{\sigma^2}\right). \tag{4.4}$$

To achieve the maximum sum rate R_c without using rate splitting, the two users should operate cooperatively at corner point B or C, i.e., one of the users should transmit at rate $R_1 = \frac{1}{2}\log\left(1 + \frac{P}{\sigma^2}\right)$ and the other one at rate $R_2 = \frac{1}{2}\log\left(1 + \frac{P}{P+\sigma^2}\right)$.

Without coordination, each user transmits at rate R_1 with probability p and at rate R_2 with probability $1 - p$. When both users transmit at rate R_1, the rate pair is outside the capacity region (4.2). Thus, the receiver cannot decode both packets. In all other cases, by using SIC, both users' packets can be decoded. The average achievable rate by using the decentralized mac is

$$R_{\mathrm{d}}(p) = 2(R_1 + R_2)p(1-p) + 2R_2(1-p)^2. \tag{4.5}$$

$R_{\mathrm{mac-SIC}}$ is maximized when $p = \frac{R_1 - R_2}{2R_1}$ and the maximum throughput is

$$R_{\mathrm{d}}^* = \frac{(R_1 + R_2)^2}{2R_1} > \frac{1}{2}(R_1 + R_2). \tag{4.6}$$

We thus have

$$R_c > R_{\mathrm{d}}^* > \frac{1}{2}R_c, \quad R_{\mathrm{d}}^* > R_{\mathrm{Aloha}}^*. \tag{4.7}$$

Therefore, the proposed new MAC protocol can achieve at least 50% of the throughput using a centralized controller and its throughput is always greater than Aloha.

We next show that the proposed strategy is actually optimal over all possible transmission strategies when rate splitting is not used and both users adopt the same transmission strategy for fairness. The transmission strategy is defined by a probability density function $f(r)$ for a transmission rate r. Without rate splitting, each user's packet can always be decoded if its transmission rate is less than R_2. When it transmits at a rate above R_2, its packet can be decoded if and only if the other user's packet can be decoded. The average throughput of user 1 is

$$\begin{aligned} \bar{R} &= \int_0^{R_2} r_1 f(r_1) dr_1 + \int_{R_2}^{R_1} r_1 f(r_1) \left(\int_0^{R_2} f(r_2) dr_2 \right) dr_1 \\ &= \int_0^{R_2} r f(r) dr + \int_0^{R_2} f(r) dr \int_{R_2}^{R_1} r f(r) dr. \end{aligned} \tag{4.8}$$

When $\int_0^{R_2} f(r)dr$ is fixed to be $1-p$ and $\int_{R_2}^{R_1} f(r)dr = p$, it is easy to see that the maximum of $\int_0^{R_2} rf(r)dr$ is $(1-p)R_2$ and the maximum of $\int_{R_2}^{R_1} rf(r)dr$ is pR_1. Therefore, the proposed strategy is optimal for the case of decentralized control and without using rate splitting.

Note that using the MPR model both users randomly attempt to transmit at a single fixed rate. When both users transmit at R_1, the MPR model reduces to Aloha. When the rate is R_2, the two users can transmit simultaneously. However, the achievable throughput is less than Aloha when $P \gg \sigma^2$. Therefore, by enabling multirate data transmission at each node and using SIC, the proposed protocol outperforms both Aloha and MPR and has comparable performance with a centralized controller.

4.3 Multiple Access MAC in Aloha Type Networks

4.3.1 MAC on AWGN Multiple Access Channels

Let X_i be the transmitted signal by user i and Y be the signal received by the receiver. We use the model

$$Y = \sum_{i=1}^{N} h_i X_i + W, \tag{4.9}$$

where the power of user i is P_i and the AWGN W is of zero mean and variance σ^2. We first consider a homogenous system with $P_i = P$ and $h_i = 1$, $\forall i \in \{1, \ldots, N\}$. Let S_i be the data rate of user i. From [22], the capacity region of mac with a centralized controller is

$$\sum_{i \in \mathcal{U}} S_i \leq I\left(X\left(\mathcal{U}\right); Y | X\left(\mathcal{U}^c\right)\right), \forall \mathcal{U} \subseteq \{1, 2, \ldots, N\}, \tag{4.10}$$

for some product distribution $p_1(x_1)p_2(x_2)\cdots p_N(x_N)$, where $X\left(\mathcal{U}\right) = \{X_i : i \in \mathcal{U}\}$ and $\mathcal{U}^c$ denotes the complement of $\mathcal{U}$ in $\{1, 2, \ldots, N\}$. The capacity region (4.10) constitutes a polytope, which contains $N!$ corner points. Each corner point corresponds to a permutation π of the N users. The receiver decodes using SIC. User $\pi(i)$ is decoded by treating users $\pi(1), \ldots, \pi(i-1)$ as noise. After decoding, the contribution of user $\pi(i)$ in Y is removed. The process continues until user $\pi(1)$'s packet is decoded. The maximum achievable sum rate with a central controller is

$$R_{\mathrm{c}}(N) = \frac{1}{2} \log\left(1 + \frac{NP}{\sigma^2}\right). \tag{4.11}$$

As in Section 4.2, users want to reach a corner point distributedly to attain the maximum sum rate. In our multiple access MAC, we assume that each user is capable of transmitting at one of N rates, where the k-th rate is

$$R_k = \frac{1}{2} \log\left(1 + \frac{P}{(k-1)P + \sigma^2}\right), \; k = 1, \ldots, N. \tag{4.12}$$

Note that $\sum_{k=1}^{N} R_k = R_{\mathrm{c}}(N)$. When each user chooses a different rate from $\{R_k : k = 1, \ldots, N\}$, a corner point is attained. Let n_k be the number of users that transmit at rate R_k. Using SIC, the packets of rate R_k can be decoded if and only if the packets of rate less than R_k are decoded correctly so that their contribution can be cancelled from Y, and the number of users transmitting at rate greater than or equal to R_k is at most k, i.e., $\sum_{l=1}^{j} n_l \leq j,\ j = k, \ldots, N$, because from (4.12) the user at R_k can tolerate interference level $(k-1)P$.

We can use pseudo random variables with random seeds to choose the transmission rate at each node, so that with the random seed each receiver knows each user's transmission rate at each time slot and thus the order in which to decode the users. In the absence of random seeds or other information on the transmitters' sending rates, the receiver can first attempt to decode the lowest rate packets for each source. After cancelling the contribution of decoded signal from the received signal, the receiver tries to decode the second lowest rate packets for each source. The process repeats until the highest rate is reached. The decoding complexity in this case is N times higher.

4.3.2 Achievable Results

In this subsection, we study the average achievable throughput of our model using the set of transmission rates (4.12) and compare it with $R_{\mathrm{c}}(N)$ in (4.11). Let $\mathcal{S}_{-i} = (S_1, \ldots, S_{i-1}, S_{i+1}, \ldots, S_N)$ be the state of all the nodes other than node i, where S_i is the transmission rate of node i. The average throughput of the network attained by the distributed strategy is

$$\begin{aligned}
R_{\mathrm{d}}(N) = E\left\{\sum_{i=1}^{N} b_i S_i\right\} &\overset{(a)}{=} N E\{b_i S_i\} = N E_{\mathcal{S}_{-i}}\{E\{b_i S_i | \mathcal{S}_{-i}\}\} \\
=& N \sum_{k=1}^{N} p_k R_k \sum_{\mathcal{S}_{-i}} Pr\left(b_i = 1 | (S_i = R_k, \mathcal{S}_{-i}\right) Pr(\mathcal{S}_{-i}) \\
=& N \sum_{k=1}^{N} p_k R_k \underbrace{\sum_{n'_1, \ldots, n'_N} Pr\left(\sum_{l=1}^{j} n'_l \leq j-1,\ j = k, \ldots, N\right)}_{q_k},
\end{aligned} \tag{4.13}$$

where (a) is due to symmetry, $b_i \in \{0, 1\}$ indicates whether user i's packet is decoded correctly at the receiver, and n'_k denotes the number of users transmitting at rate R_l other than user i with $n'_k = n_k - 1$ and $n'_l = n_l$ for $l \neq k$. Note that (4.13) only requires that packets with rate less than R_k are decoded, where packets with rate higher than R_k may be decoded incorrectly. Let $m_k = \sum_{l=1}^{k} n'_l$. We can write q_k as

$$q_k = \sum_{\substack{m_k, n'_{k+1}, \ldots, n'_N, \\ m_k + \sum_{l=k+1}^{j} n'_l \le j-1, \\ j=k,\ldots,N}} \binom{N-1}{m_k, n'_{k+1}, \ldots, n'_N} \prod_{l=k+1}^{N} p_l^{n'_l} \left(\sum_{l=1}^{k} p_l \right)^{m_k}. \tag{4.14}$$

Given p_k, it is complex to compute q_k through (4.14). To circumvent this problem, we find a recursive relationship between q_k and q_{k-1}. Let A_k denote the event that the rates of all users excluding user i are such that if i were to send at rate R_k, its packets would be decoded correctly. Then let $q_k = Pr(A_k)$. By the total probability theorem, we obtain

$$Pr(A_k) = Pr(A_k|A_{k-1})Pr(A_{k-1}) + Pr(A_k|A_{k-1}^c)Pr(A_{k-1}^c). \tag{4.15}$$

Given $\mathcal{S}_{-i}$, if $S_i = R_{k-1}$ and user i's packet can be decoded at the receiver, when $S_i = R_k$ user i's packet can still be decoded because $R_k < R_{k-1}$, which gives $Pr(A_k|A_{k-1}) = 1$. On the other hand, $A_k \cap A_{k-1}^c$ means that if user i were to send at rate R_k its packets would be decoded correctly but its packets cannot be decoded if it were to send at rate R_{k-1}. This event occurs if and only if among the remaining $N-1$ users, $k-1$ of them transmit at rates above R_k so that transmitting at R_k is admissible but not at R_{k-1}, $N-k$ of them transmit at rate below R_k, and the $N-k$ users' packets can be decoded correctly. Let $\eta_{j,k}$ denote the probability that j given users transmit at a rate less than or equal to R_k and their packets can be decoded correctly at the receiver. From the definition of R_k in (4.12), if a user transmits at rate R_N, the receiver can decode its packet regardless of other users' transmissions, which means $q_N = 1$. We can thus establish the recursive relation

$$q_{k-1} = q_k - \binom{N-1}{k-1} \left(\sum_{j=1}^{k-1} p_j \right)^{k-1} \eta_{N-k,k+1}, \; q_N = 1. \tag{4.16}$$

To compute $\eta_{j,k}$, note that there are $\binom{j}{l}$ ways to choose l users out of j users. Suppose that these l users transmit at rate R_k and the remaining $j-l$ users transmit at rates less than R_k. Assuming that $N-j$ other users transmit at rate greater than R_k, all the j users' packets can be decoded if and only if $N-j+l \leq k$ and the remaining $j-l$ users' packets can be decoded. We thus have the recursive equation

$$\eta_{j,k} = \sum_{l=0}^{k-N+j} \binom{j}{l} p_k^l \eta_{j-l,k+1}, \text{ and } \eta_{j,N} = p_N^j. \tag{4.17}$$

Given p_k, by using these two recurrences (4.16) and (4.17), we can evaluate the throughput efficiently by first creating a table for $\eta_{j,k}$ using (4.17) and then using (4.16) to compute q_k, which enables fast computation of achievable rate.

To find the maximum achievable asymptotic rate of the proposed scheme, we need to find the optimal p_k by maximizing $R_{\mathrm{d}}(N)$ in (4.13) for each N, which is hard to obtain in closed form. Instead, we find a lower bound on $R_{\mathrm{d}}(N)$ by choosing a suboptimal p_k based on the following result.

Theorem 4.1 *The maximum achievable sum rate by using distributed mac and the set of rates in (4.12) is* $R_{\mathrm{d}}(N) = \Theta\left(\log\left(1+\frac{NP}{\sigma^2}\right)\right)$. *Furthermore, there exists a constant* $C>0$ *such that* $R_{\mathrm{d}}(N) \geq CR_{\mathrm{c}}(N)$.

Proof. We consider $p_k = \frac{\alpha}{N-1}$, $k=1,\ldots,N-1$ and $p_N = 1-\alpha$, where $0<\alpha<1$ is a constant to be determined later. Note that $q_N = 1$ and

$$\binom{N-1}{k-1}\left(\sum_{j=1}^{k-1} p_j\right)^{k-1} \eta_{N-k,k+1} \leq \frac{(k-1)^{k-1}}{(k-1)!}\alpha^{k-1}. \tag{4.18}$$

where we have used the fact that $\eta_{j,k} \leq 1$ and $\frac{(N-1)!}{(N-k)!} < (N-1)^{k-1}$. Applying induction on (4.16), we obtain

$$q_1 \geq 1 - \sum_{k=2}^{N} \frac{(k-1)^{k-1}}{(k-1)!}\alpha^{k-1} > 1 - \sum_{k=1}^{+\infty} \frac{k^k}{k!}\alpha^k. \tag{4.19}$$

To show the convergence of the series $\sum_{k=1}^{+\infty}\frac{k^k}{k!}\alpha^k$, we use ratio test and compute

$$L = \lim_{k\to\infty}\frac{(k+1)^{k+1}}{(k+1)!}\alpha^{k+1}\frac{k!}{k^k}\alpha^{-k} = \alpha e. \tag{4.20}$$

Therefore, if $L < 1$ or $\alpha < e^{-1}$, $\sum_{k=1}^{+\infty}\frac{k^k}{k!}\alpha^k$ converges to a number $B(\alpha)$. It is easy to see that $B(0) = 0$ and $B(\alpha)$ is a continuous and increasing function in α. Therefore, there exists a threshold γ such that $B(\alpha) < 1$ when $0 < \alpha < \gamma$. Note that $q_1 \le q_2 \le \cdots \le q_N$ and $p_N > p_k$, $k = 1, \dots, N-1$ when N is large. Thus, we have

$$\begin{aligned} R_{\mathrm{d}}(N) =& N\sum_{k=1}^{N} p_k q_k R_k \ge \alpha(1 - B(\alpha))\sum_{k=1}^{N} R_k \\ =& \frac{\alpha}{2}(1 - B(\alpha))\log\left(1 + \frac{NP}{\sigma^2}\right). \end{aligned} \tag{4.21}$$

We can find the optimal α by maximizing $\alpha(1 - B(\alpha))$, which is the solution of

$$\sum_{k=1}^{+\infty}\frac{k^k}{k!}(k+1)\alpha^k = 1. \tag{4.22}$$

By solving (4.22) numerically, we find that $\alpha = 0.2011$ and

$$R_{\mathrm{d}}(N) \ge \frac{0.13}{2}\log\left(1 + \frac{NP}{\sigma^2}\right) = 0.13R_{\mathrm{c}}(N). \tag{4.23}$$

On the other hand, $R_{\mathrm{d}}(N)$ is less than $R_{\mathrm{c}}(N)$, the achievable rate of a centralized controller. Therefore, we show that $R_{\mathrm{d}}(N) = \Theta\left(\log\left(1 + \frac{NP}{\sigma^2}\right)\right)$. ■

The constant C in Theorem 4.1 can be interpreted as the distributive loss due to contention and lack of cooperation between users. By choosing $p_1 = \frac{1}{N}$, $p_2 =, \cdots, = p_{N-1} = 0$, and $p_N = \frac{N-1}{N}$, it is easy to see that the throughput of the proposed model is greater than that of Aloha. We thus obtain the following corollary.

Corollary 4.2 *Let $R_{\mathrm{Aloha}}(N)$ be the throughput of Aloha. We have $R_{\mathrm{d}}(N) > R_{\mathrm{Aloha}}(N)$.*

Theorem 4.1 suggests that the total throughput of the new MAC model increases with increasing N as opposed to Aloha where the total throughput decreases in N. Aloha can be considered to be a distributed implementation of TDMA while our approach is a distributed implementation of mac.

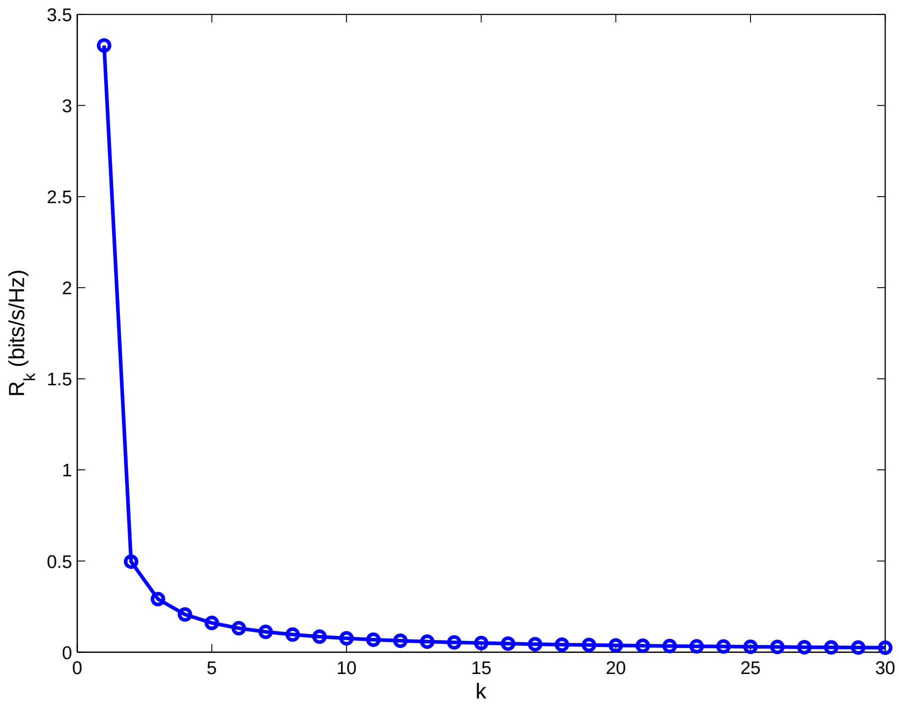


Figure 4.2: R_k versus k in a network with $N = 30$ users and $P = 10$, $\sigma^2 = 1$.

4.3.3 Fixed Number of Transmission Rates

The N-rate model can achieve a fraction of the achievable rate by a centralized controller. However, in practice, the MAC layer is built into firmware and the set of rates cannot be altered as the number of users in the network varies. Furthermore, the first few R_k's in (4.12) are significantly larger than the other rates in practical scenarios. This is illustrated in Fig. 4.2, which shows R_k in (4.12) versus k in a network with $N = 30$ users and $P = 10$, $\sigma^2 = 1$. Motivated by these two factors, we thus generalize the N-rate model to the case with K rates, where K is a fixed number that does not vary with N. Each node is capable of transmitting at one of K rates, $R_1, \ldots, R_K$ and $R_1 > R_2 > \cdots > R_K$. Assuming that n_k nodes transmit at rate R_k, the packets of rate R_k can be decoded if and only if $\sum_{l=1}^{j} n_l \leq \omega_j$, $j = k, \ldots, K$, where ω_k is the maximum number of users with transmission rates higher than R_k such that users with transmission rate R_k can decode their packets. The decoding complexity is proportional to NK.

In this subsection, we consider $R_k = \frac{1}{2}\log\left(1 + \frac{P}{(k-1)P+\sigma^2}\right)$, $k = 1, \ldots, K-1$ and

$R_K = \frac{1}{2}\log\left(1 + \frac{P}{(N-1)P+\sigma^2}\right)$ for simplicity, which gives $\omega_k = k$, $k = 1, \ldots, K-1$ and $\omega_K = N$. Even though the optimal p_k maximizing the average throughput may depend on R_k and N in a complicated way which does not lead to simple practical protocol design, the following theorem shows that the optimal p_k has a simple form asymptotically as $N \to +\infty$.

Theorem 4.3 *When $R_k = \frac{1}{2}\log\left(1 + \frac{P}{(k-1)P+\sigma^2}\right)$, $k = 1, \ldots, K-1$ and $R_K = \frac{1}{2}\log\left(1 + \frac{P}{(N-1)P+\sigma^2}\right)$, and $\omega_k = k$, $k = 1, \ldots, K-1$ and $\omega_K = N$, the optimal p_k maximizing the average throughput satisfies $\lim_{N\to+\infty} Np_k = \xi_k$, $k = 1, \ldots, K-1$, where ξ_k are constants depending only on $R_1, \ldots, R_K$. In other words, $p_k = \frac{\xi_k}{N}$ maximizes the average throughput asymptotically.*

The proof of Theorem 4.3 can be found in the appendix. Theorem 4.3 removes the dependence on N in optimizing p_k, which facilities distributed dynamic algorithm design, e.g., optimization on ξ_k is done only once and the resulting ξ_k can be applied for any N. The game theoretic framework in Section 4.4 further removes the need to know N.

When K is small, we can obtain ξ_k in closed form. We give an example in the following.

Example 4.1 ($K = 2$): When $K = 2$, the average throughput is

$$NR_1 p(1-p)^{N-1} + NR_2(1-p), \tag{4.24}$$

where p is the probability of choosing R_1 and $\omega_1 = 1$, $\omega_2 = N$. Maximizing (4.24), we obtain the optimal p, whose closed form does not exist in general. When $R_2 = 0$, it reduces to Aloha, whose throughput is maximized when $p = \frac{1}{N}$ in this case.

We could also choose $R_1 = \frac{1}{2}\log\left(1 + \frac{P}{(k-1)P+\sigma^2}\right)$ and $\omega_1 = k$, where k is an integer in $\{1, \ldots, N\}$. The average throughput is

$$NR_1 p \sum_{l=0}^{k-1} \binom{N-1}{l} p^l (1-p)^{N-1-l} + NR_2(1-p), \tag{4.25}$$

where p is the probability of choosing R_1. When $k \ll N$ and $N \to +\infty$, (4.25) can

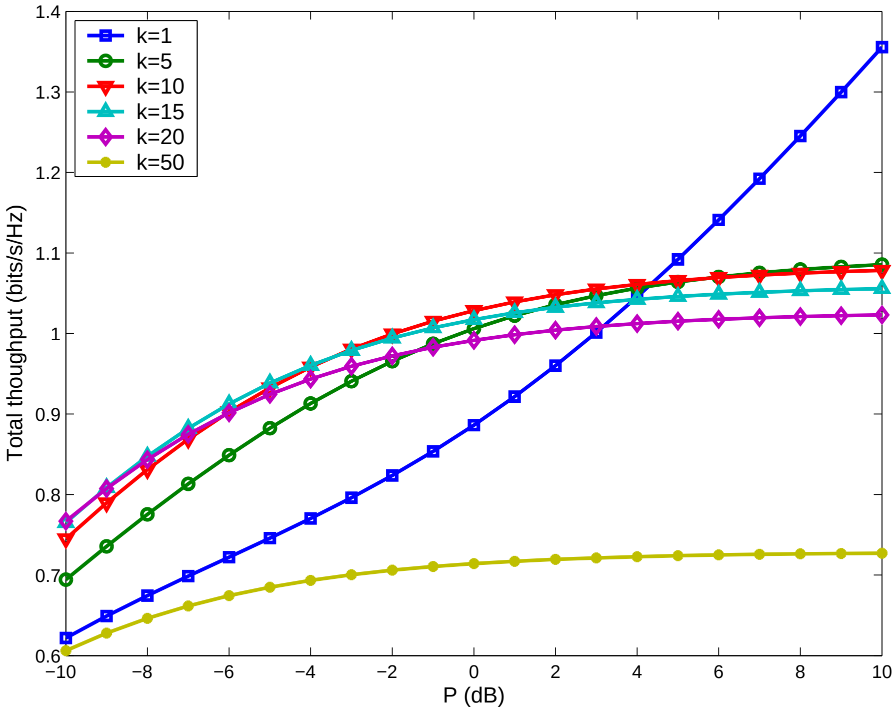


Figure 4.3: Comparison of the total throughput versus P with different k in a network with $N = 50$ users and $\sigma^2 = 1$.

be approximated as

$$\xi e^{-\xi} R_1 \sum_{l=0}^{k-1} \frac{1}{l!} \xi^l + NR_2, \tag{4.26}$$

where $\xi = Np$.

Fig. 4.3 shows the total throughput with different k in a network with $N = 50$ users and $\sigma^2 = 1$ as a function of P. We can see that the k achieving the highest rate depends on P or SNR. When $P > 4.37$ dB or in high SNR, R_1 with $k = 1$ achieves the highest throughput. When 4.37 dB$> P > -3.4$ dB, R_1 with $k = 10$ achieves the highest throughput. When -3.4 dB$> P > -10$ dB, R_1 with $k = 15$ achieves the highest throughput. Compared with conventional Aloha, which achieves a throughput of only 0.0137 bits/s/Hz (not shown in Fig. 4.3), the proposed scheme achieves a much higher throughput. In practice, R_1 can be adapted according to SNR to achieve the highest throughput.

Example 4.2 ($K = 3$): When $K = 3$, we have 3 rates: R_1, R_2, R_3 with $\omega_1 = 1$,

$\omega_2 = 2$, $\omega_3 = N$. The average throughput is

$$\begin{aligned}&NR_1p_1\left((1-p_1-p_2)^{N-1}+(N-1)p_2(1-p_1-p_2)^{N-2}\right)\\&+NR_2p_2\left((N-1)p_1(1-p_1-p_2)^{N-2}+(N-1)p_2(1-p_1-p_2)^{N-2}+(1-p_1-p_2)^{N-1}\right)\\&+NR_3(1-p_1-p_2),\end{aligned} \tag{4.27}$$

where p_1 and p_2 are the probabilities of choosing R_1 and R_2. Maximizing (4.27), we obtain the optimal p_1, p_2. Consider the special case when $R_3 = 0$. The optimal solution when $N \rightarrow +\infty$ is

$$p_1 = \frac{(R_1-R_2)\left({R_1}^2+4R_1R_2+\sqrt{4{R_1}^3R_2+10{R_1}^2{R_2}^2+{R_1}^4+{R_2}^4+4R_1{R_2}^3}+{R_2}^2\right)}{2N\left(R_1+R_2\right)^2R_1} \tag{4.28}$$

and

$$p_2 = \frac{{R_2}^2-{R_1}^2+2R_1R_2+\sqrt{4{R_1}^3R_2+10{R_1}^2{R_2}^2+{R_1}^4+{R_2}^4+4R_1{R_2}^3}}{2N\left(R_1+R_2\right)R_1}, \tag{4.29}$$

where ξ_1 and ξ_2 can be easily recognized.

4.4 Multiple Access MAC in WLAN

In this section, we extend the multiple access MAC in Aloha type networks in Section 4.3 to WLAN using CSMA. There two main differences between the Aloha type networks and WLAN. First, all the nodes transmit to a common receiver in Aloha type networks, while each node can be transmitter or receiver in WLAN. Second, different from Aloha where nodes cannot perform carrier sensing, in CSMA, nodes listen before data transmission to reduce collision. We consider single-cell wireless LANs, where every wireless node can hear every other node in the network. We assume that each packet has the same length and each packet takes the same time to transmit. Different data rate is reflected by the amount of data information contained in each packet after removing redundant error protection bits.

We consider half-duplex network where each node can only transmit or receive at any given time slot. For centralized scheme, at any time slot, we divide N nodes into two parts, each of which contains $N/2$ nodes (assuming N is even. If N is odd, each part contains $\lfloor N/2 \rfloor$ nodes and one node is idle). Every node in the first part transmits to a node in the second part. The network becomes a $N/2$ transmitter and $N/2$ receiver interference channel. When a symmetric network is considered, from [15] the capacity region of this interference channel is the same as that of an $N/2$ user mac. By time sharing between all possible partitions, we obtain an achievable rate region of the centralized scheme.

In decentralized scheme, we consider each node is able to transmit at one of K rates. The first $K-1$ rates are the same as those in Section 4.3.3, while the K-th rate should be zero because if each node always transmits it cannot receive information due to half duplex constraint. Let T_t denote the average transmission duration, T_c be the collision duration, and T_{SLOT} denote the slot duration. The average throughput of the network is

$$R_{\text{hd}-\text{mac}}(N) = \frac{N s_d}{P_s T_t + (1 - P_s - p_K^N) T_c + p_K^N T_{\text{SLOT}}} = \frac{N\bar{R}}{P_s \eta_1 + (1 - P_s - p_K^N)\eta_2 + p_K^N \eta_3}, \tag{4.30}$$

where s_d is the average number of data bits successfully transmitted in a time slot by one node, P_s is probability of at least one node with successful transmission, and $\bar{R}$ is the average throughput of each node in a half-duplex slotted network, $\eta_1 = \frac{T_t}{T_d}$, $\eta_2 = \frac{T_c}{T_d}$ and $\eta_3 = \frac{T_{\text{SLOT}}}{T_d}$ where $s_d = \bar{R} T_d$ and T_d is the effective data transmission time. As we use variable rate transmission, we assume that T_d or T_t is the same for all transmission rates. Due to half duplex constraint, when a node transmits, its packet's destination must be idle. The probability that the destination node is idle is p_K. Similar to (4.13), $\bar{R}$ can be derived as

$$\bar{R} = p_K \sum_{k=1}^{K-1} p_k R_k \sum_{\substack{m_k + \sum_{l=k+1}^{j} n'_l \le j, \\ j=k,\dots,K-1}} \binom{N-2}{m_k, n'_{k+1}, \dots, n'_K} \prod_{j=k+1}^{K-1} p_j^{n'_j} \left(1 - \sum_{j=1}^{K-1} p_j\right)^{n'_K} \left(\sum_{j=1}^{k} p_j\right)^{m_k}, \tag{4.31}$$

where $n'_k = n_k - 1$, $n'_K = n_K - 1$ and $n'_l = n_l$ for $l \neq k, K$ as in (4.13) and $m_k = \sum_{j=1}^{k} n'_j$. By following the proof of Theorem 4.3, we can also show that as $N \to +\infty$ $p_k = \frac{\xi_k}{N}$ for $k = 1, \ldots, K-1$ and $p_K = 1 - \sum_{k=1}^{K-1} p_k$ maximize $R_{\text{hd}-\text{mac}}$ asymptotically. In the following, we give a simple example on optimizing p_k.

Example 4.3 ($K = 3$): When $K = 3$, we have 2 nonzero rates R_1, R_2 with $\omega_1 = 1$ and $\omega_2 = 2$. The average throughput is

$$\begin{aligned}
&\frac{1}{\eta_1 P_s + \eta_2(1 - P_s - (1-p_1-p_2)^N) + \eta_3(1-p_1-p_2)^N} \\
&\times \Big(N R_1 p_1 \left((1-p_1-p_2)^{N-1} + (N-2)p_2(1-p_1-p_2)^{N-2}\right) \\
&\quad + N R_2 p_2 \left((N-2)p_1(1-p_1-p_2)^{N-2} + (N-2)p_2(1-p_1-p_2)^{N-2} + (1-p_1-p_2)^{N-1}\right) \Big),
\end{aligned} \tag{4.32}$$

where

$$\begin{aligned}
P_s \approx & N p_1 \left((1-p_1-p_2)^{N-1} + (N-2)p_2(1-p_1-p_2)^{N-2}\right) \\
& + N p_2 \left((N-2)p_1(1-p_1-p_2)^{N-2} + (N-2)p_2(1-p_1-p_2)^{N-2} + (1-p_1-p_2)^{N-1}\right).
\end{aligned} \tag{4.33}$$

When $N \to +\infty$, we can approximate (4.32) as

$$\begin{aligned}
&\frac{e^{-(\xi_1+\xi_2)}}{\eta_1 e^{-(\xi_1+\xi_2)}(\xi_1 + \xi_2 + 2\xi_1\xi_2 + \xi_2^2) + \eta_2(1 - e^{-(\xi_1+\xi_2)}(1 + \xi_1 + \xi_2 + 2\xi_1\xi_2 + \xi_2^2)) + \eta_3 e^{-(\xi_1+\xi_2)}} \\
&\times \Big(R_1 \xi_1 \left(1 + \xi_2\right) + R_2 \xi_2 (1 + \xi_1 + \xi_2) \Big).
\end{aligned} \tag{4.34}$$

Given R_1, R_2 and η_1, η_2, η_3, we could solve ξ_1, ξ_2 by maximizing (4.34).

Fig. 4.4 compares the optimal achievable rate by maximizing (4.32) for each N with the achievable rate by choosing $p_k = \frac{\xi_k}{N}$, $k = 1, 2$, where ξ_k is obtained by maximizing (4.34). In Fig. 4.4, we choose $P = 1$, $\sigma^2 = 1$, $\eta_1 = \eta_2 = 2$ and $\eta_3 = 0.25$. In this case, we find that $\xi_1 = 0.3971$ and $\xi_2 = 0.5165$.

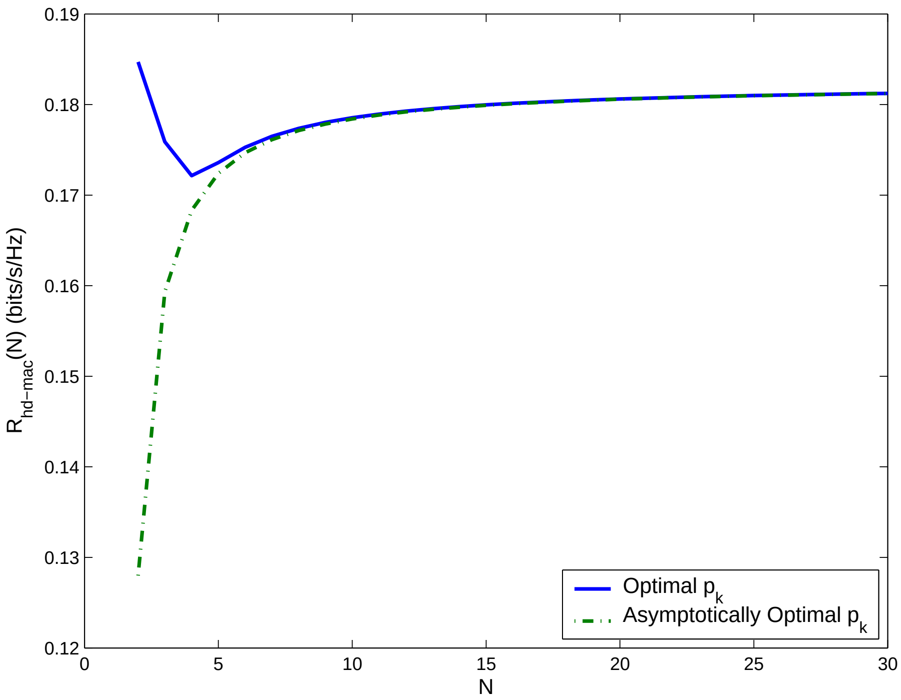


Figure 4.4: Comparison of throughput by using optimal p_k and asymptotically optimal p_k with $P = 1$, $\sigma^2 = 1$, $\eta_1 = 2$ and $\eta_2 = 0.25$.

4.5 Game-Theoretic Model of Contention Control

In Section 4.3 and Section 4.4, we consider the case where each node can transmit at one of K rates. The optimal p_k can be found by maximizing the throughput of the network. In practice, nodes may join or leave a network. Each node should therefore adjust its probability p_k independently. In this section, we present a general game-theoretic framework for designing contention based medium access control. The framework extends the random access game model in [20]. In [20], this problem is treated as a random access game with a given utility function. In contrast to [20], we consider how to design utility functions to achieve the desired equilibrium and to reverse engineer a given protocol. For example, we will give a utility function with the throughput optimal channel access probability in Section 4.4 as the equilibrium. We also propose several dynamic algorithms to achieve the equilibrium in a distributed fashion and characterize their convergence under asychronousness and estimation error.

4.5.1 Random Access Game

Consider a set $\mathcal{N}$ of wireless nodes in a wireless LAN with contention-based medium access. In this subsection, we mainly consider *single-cell* wireless LANs with a single transmission rate at each node. We assume all nodes always have a frame to transmit. The network is noise free and packet loss is only due to collision. We will mainly present our theory and analysis in terms of "channel access probability." If a backoff mechanism is implemented, the channel access probability p is related to the contention window W according to $p = \frac{2}{W+1}$, which is derived under the decoupling approximation with constant contention windows, see, e.g., [11]. Contention control is an iterative feedback system described mathematically as:

$$p_i(t+1) = \mathcal{F}_i(p_i(t), \mathbf{q}_i(t)), \quad \mathbf{q}_i(t+1) = \mathcal{G}_i(\mathbf{p}(t)), \tag{4.35}$$

where $p_i(t)$ is the channel access probability of node i, $\mathbf{p}(\mathbf{t}) = \{p_i(t)\}$ is the corresponding vector, and $\mathbf{q}_i(t)$ is certain measure of contention observed by node i that depends on the vector $\mathbf{p}(t)$.

In practice, it is hard for wireless nodes to learn the exact channel access probabilities of others. Each node infers the contention of the wireless network through observing several contention measure signals $\mathbf{q}_i(\mathbf{p})$, which are functions of other nodes' channel access probabilities. We model the interaction among wireless nodes as a non-cooperative game. Formally, we define a random access game [20] as follows.

Definition 4.4 *A* random access game $\mathcal{G}$ *is defined as a triple* $\mathcal{G} := \{\mathcal{N}, (\mathcal{S}_i)_{i\in\mathcal{N}}, (u_i)_{i\in\mathcal{N}}\}$, *where* $\mathcal{N}$ *is a set of players (wireless nodes), player* $i \in \mathcal{N}$ *strategy* $\mathcal{S}_i := \{p_i | p_i \in [\nu_i, \omega_i]\}$ *with* $0 \le \nu_i < \omega_i \le 1$, *and payoff function* $u_i(\mathbf{p}) = U_i(p_i) - p_i C_i(\mathbf{q}_i)$ *with utility function* $U_i(p_i)$ *and price function* $C_i(\mathbf{q}_i)$.

The payoff function can be interpreted as the net gain of utility from channel access discounted by the contention "cost." One property of this random access game is that the computation of the payoff function does not require explicit exchange of channel access probabilities between nodes. Thus, this game can be played and implemented distributedly. Random access game is a rather general model for contention control, as the payoff function can be reverse-engineered from (4.35). The equilibrium point of (4.35) defines an implicit relation between channel access probability p_i and contention measure $\mathbf{q}_i$. If this relation can be written as

$$C_i(\mathbf{q}_i) = F_i(p_i), \tag{4.36}$$

the utility function of each node i is defined as

$$U_i(p_i) = \int F_i(p_i) \mathrm{d}p_i. \tag{4.37}$$

Therefore, we can reverse engineer medium access control protocols and study them in game theoretic framework: medium access control can be interpreted as a distributed strategy update algorithm to achieve the equilibrium of the random access game.

In random access game, one of the most important questions is whether a Nash equilibrium exists or not. Denote the channel access probability for all nodes but i by $\mathbf{p}_{-i} := (p_1, \ldots, p_{i-1}, p_{i+1}, \ \ldots, p_{|\mathcal{N}|})$, and write $(p_i, \mathbf{p}_{-i}) := \mathbf{p}$. We have the following definition of Nash equilibrium [29].

Definition 4.5 *A channel access probability vector* $\mathbf{p}^*$ *is said to be a* Nash equilibrium *if no node can improve its payoff by unilaterally changing its probability of transmission, i.e.,* $u_i(p_i^*, \mathbf{p}_{-i}^*) \geq u_i(p_i, \mathbf{p}_{-i}^*)$, $\forall p_i \in \mathcal{S}_i$. *A Nash equilibrium* $\mathbf{p}^*$ *is a* nontrivial equilibrium *if* p_i^* *satisfies*

$$\frac{\partial}{\partial p_i} u_i(p_i^*, \mathbf{p}_{-i}^*) = 0, \; \forall i \in \mathcal{N}. \tag{4.38}$$

To facilitate analysis in the following, we list the assumptions that will be used in this section.

A1: The utility function $U_i(\cdot)$ is twice continuously differentiable, strictly concave, and with finite curvatures that are bounded away from zero, i.e., there exist some positive constants μ and χ such that $1/\mu \geq -1/U_i''(p_i) \geq 1/\chi > 0$.

A2: The inverse function $\left(U_i'\right)^{-1}\left(C_i\left(\mathbf{q}_i\right)\right)$ maps any $\mathbf{q}_i$ into a point in $\mathcal{S}_i$ for all $i \in \mathcal{N}$.

A3: At a nontrivial Nash equilibrium $\mathbf{p}^*$, there exists a function $\Phi_i(p_i)$ for each node i such that $\Phi_i(p_i^*) = \Phi_j(p_j^*)$, $\forall i, j \in \mathcal{N}$ and $\Phi_i(p_i)$ is monotone in $\mathcal{S}_i$, $\forall i \in \mathcal{N}$.

By [29, Theorem 1.2] and Brouwer's fixed point theorem [12], the following two theorems are immediate.

Theorem 4.6 *Under assumption A1, there exists a Nash equilibrium for any random access game $\mathcal{G}$.*

Theorem 4.7 *Suppose A2 holds. Random access game $\mathcal{G}$ has a nontrivial Nash equilibrium.*

Since the equilibrium determines the operating point of medium access control, it is desired to have a unique nontrivial Nash equilibrium. One way to show this is to use Banach fixed point theorem [12] by showing that $G_i(\mathbf{p}) := \left(U_i'\right)^{-1}\left(C_i\left(\mathbf{q}_i(\mathbf{p})\right)\right)$ is a contraction mapping [5]. However, the conditions obtained using this approach are sometimes restrictive. Another way to show uniqueness is to apply the following theorem.

Theorem 4.8 *Suppose that A1 and A3 hold and random access game $\mathcal{G}$ has a nontrivial Nash equilibrium. If additionally for all $i \in \mathcal{N}$, $\Phi_i(p_i)$ is a strictly monotone function in S_i and $C_i(\mathbf{q}_i(\mathbf{p}))$ is strictly increasing in $\mathbf{p}$, then $\mathcal{G}$ has a unique nontrivial Nash equilibrium.*

Proof. Since $U_i(p_i)$ is a continuously differentiable concave function, $U_i'(p_i)$ is a continuous, decreasing function. Without loss of generality, we consider the case that $\Phi_i(p_i)$ is strictly increasing. Suppose that there are two nontrivial Nash equilibria $\bar{\mathbf{p}}$ and $\hat{\mathbf{p}}$. By A3, there exist $\gamma_1, \gamma_2 > 0$ such that, for all i, $\Phi_i(\bar{p}_i) = \gamma_1$, $\Phi_i(\hat{p}_i) = \gamma_2$. Since $\Phi_i(p_i)$ is strictly increasing, $\gamma_1 \neq \gamma_2$. Without loss of generality, assume $\gamma_1 > \gamma_2$. Thus, $\bar{p}_i > \hat{p}_i$ for all i. By equation (4.38), $U_i'(\bar{p}_i) = C_i(\mathbf{q}_i(\bar{\mathbf{p}})) > C_i(\mathbf{q}_i(\hat{\mathbf{p}})) = U_i'(\hat{p}_i)$, which contradicts the fact that $U_i'(p_i)$ is a decreasing function. Thus, if $\mathcal{G}$ has a nontrivial Nash equilibrium, it is unique. ■

4.5.2 Utility Function Design

In the following, we give several examples to show how to interpret existing medium access algorithms within random access game framework and design utility functions to achieve desired equilibrium properties. There are basically three ways to design utility functions.

4.5.2.1 Reverse Engineering from Existing Protocols

Take 802.11 DCF as an example. Different from [53], which reverse engineer exponential backoff type of protocols from the dynamic, we reverse engineer 802.11 DCF from the equilibrium point. Let $q_i := 1 - \prod_{j \in \mathcal{N}/\{i\}}(1 - p_j)$ denote the conditional collision probability of node i. It is well established that for a single-cell wireless LAN at steady state, channel access probability p_i relates to conditional collision probability q_i as follows [11]:

$$p_i = \frac{2(1-2q_i)}{(1-2q_i)(a+1) + q_i a(1-(2q_i)^m)}, \tag{4.39}$$

where $a = CW_{\min}$ is the base contention window and m is the maximum backoff stage. Note that (4.39) defined an implicit function $q_i = C_i(q_i) = F_i(p_i)$. Following procedures (4.36)–(4.37), we can derive a utility function $U_i(p_i)$. When $0 \leq q_i \leq 1$, $m \geq 1$, and $a \geq 1$, it can be verified that $U_i''(p_i) < 0$. Also, it can be readily checked that $F_i^{-1}(q_i)$ maps any $q_i \in [0, 1]$ into a point $p_i \in [0, 1]$. From Theorem 4.7, the random access game $\mathcal{G}$ with the derived utility function has a nontrivial Nash equilibrium. To show the uniqueness of equilibrium, we define $\Phi_i(p_i) = (1 - p_i)(1 - U_i'(p_i))$. At equilibrium $\mathbf{p}^*$, we have $\Phi_i(p_i^*) = \Pi_{i \in \mathcal{N}}(1 - p_i^*) = \Phi_j(p_j^*)$, $\forall i, j \in \mathcal{N}$. As $F_i(p_i)$ is an implicit function, we define $\tilde{\Phi}_i(q_i) = (1 - F_i^{-1}(q_i))(1 - q_i)$, where $F_i^{-1}(q_i)$ is given in (4.39). It is easy to show that $\tilde{\Phi}_i(q_i)$ is a strictly decreasing function in q_i and $F_i^{-1}(q_i)$ is also a strictly decreasing function in q_i. Therefore, $\Phi_i(p_i)$ is a strictly increasing function. Also, $C_i(q_i) = q_i$ is strictly increasing. By Theorem 4.8, the random access game $\mathcal{G}$ has a unique nontrivial Nash equilibrium.

4.5.2.2 Reverse Engineering from Desired Operating Points

Single-Rate Protocol without SIC: In [38], a medium access control method is proposed by using the mean number of idle slots between transmission attempts. Let T_c denote the average collision duration and T_{SLOT} denote the slot duration. It is derived in [38] that when the number of users in the network $|\mathcal{N}| \to \infty$, the throughput-optimal number of idle slots between two transmission attempts is

$$\bar{n}_{i\infty}^{\text{opt}} = \frac{e^{-\xi}}{1 - e^{-\xi}}, \tag{4.40}$$

where ξ is the solution to $1 - \xi = \eta e^{-\xi}$ and $\eta = 1 - T_{\text{SLOT}}/T_c$. Note that $\bar{n}_{i\infty}^{\text{opt}}$ is completely determined by the protocol parameters but not by the number of nodes in the network. Let $q_i := 1 - \prod_{j \in \mathcal{N}/\{i\}}(1 - p_j)$. The probability of an idle slot is

$$(1 - p_i)(1 - q_i) = \frac{\bar{n}_{i\infty}^{\text{opt}}}{\bar{n}_{i\infty}^{\text{opt}} + 1} = e^{-\xi}. \tag{4.41}$$

Let n denote the number of consecutive idle slots between two transmissions. Since n has the geometric distribution with parameter $\gamma(\mathbf{p}) = \prod_{i \in \mathcal{N}}(1 - p_i)$, its mean $\bar{n}$ is given by $\bar{n} = \frac{\gamma(\mathbf{p})}{1 - \gamma(\mathbf{p})}$, which can be estimated by averaging over *ntrans* occurrences of this event. At every step, $\bar{n}$ is updated according to $\bar{n} \leftarrow \beta \bar{n} + (1 - \beta) isum/ntrans$, where *isum* is the total number of idle slots during *ntrans* occurrences. Thus, each node can estimate its conditional collision probability according to

$$q_i = 1 - \frac{\gamma(\mathbf{p})}{1 - p_i} = \frac{1 - (\bar{n} + 1)p_i}{(\bar{n} + 1)(1 - p_i)}. \tag{4.42}$$

Applying (4.37) with $C_i(q_i) = q_i$, we obtain the utility function as

$$U_i(p_i) = p_i + e^{-\xi} \log(1 - p_i). \tag{4.43}$$

Note that $U_i(p_i)$ does not satisfy A2 but it is clear that the random access game with utility (4.43) has a nontrivial Nash equilibrium. This also shows the limitation of Theorem 4.7. Utility (4.43) does not satisfy Theorem 4.8. In fact, there exist infinite number of equilibria in the game with (4.43). To design a game with unique equilibrium, we note that when $|\mathcal{N}|$ is large the optimal attempt probability that maximizes the throughput is very small as shown in [38]. We thus have

$$(1-p_i)^{\alpha}(1-q_i) = (1-p_i)^{\alpha-1}e^{-\xi} \approx e^{-\xi}, \tag{4.44}$$

where $\alpha > 1$ and the approximation holds when α is not very large. Applying (4.37), we obtain the utility function as

$$U_i(p_i) = p_i + \frac{e^{-\xi}}{1-\alpha}(1-p_i)^{1-\alpha}. \tag{4.45}$$

Note that (4.45) still does not satisfy A2 and we cannot use Theorem 4.7 and contraction mapping to show existence and uniqueness of nontrivial Nash equilibrium. But at least one nontrivial Nash equilibrium exists, i.e., $p_i^* = 1 - e^{-\xi/(\alpha+|\mathcal{N}|-1)}$. Define $\Phi_i(p_i) = (1-p_i)(1-U_i'(p_i)) = \frac{e^{-\xi}}{(1-p_i)^{\alpha-1}}$, which is strictly increasing in p_i when $\alpha > 1$. Also $q_i(\mathbf{p})$ is strictly increasing in $\mathbf{p}$. By Theorem 4.8, the random access game $\mathcal{G}$ has a unique nontrivial Nash equilibrium. Note that due to the approximation in (4.44) the equilibrium point obtained by (4.45) may not achieve the optimal number of idle slots $\bar{n}_{i\infty}^{\text{opt}}$. We will discuss in Section 4.6.4 how to design equilibrium selection algorithm such that the equilibrium point by using (4.45) can actually hit $\bar{n}_{i\infty}^{\text{opt}}$.

Multiple-Rate Protocol with SIC: From Theorem 4.3, we know that $p_k = \frac{\xi_k}{N}$, $k = 1, \ldots, K-1$ is asymptotically optimal when N is large for the multiple-rate protocol with SIC in Section 4.3 and Section 4.4. We thus define $\xi = \sum_{k=1}^{K-1} \xi_k$ and $\tilde{p} = \sum_{k=1}^{K-1} p_k = \frac{\xi}{N}$. We only need to determine $\tilde{p}$ because we can find $p_k = \frac{\xi_k}{\xi}\tilde{p}$. The probability that all the nodes transmit at rate R_K in a given time slot is $(1-\tilde{p})^N$, which converges to $e^{-\xi}$ as $N \to +\infty$. For example, in half-duplex WLAN, this probability corresponds to the probability of an idle time slot. Let $\tilde{p}_i$ be node i's local estimate of $\tilde{p}$, which is the probability that at least one other user transmits at a rate higher than R_K. Each node could infer the contention of the network through the contention measure signal $\tilde{q}_i = 1 - \prod_{j \neq i}(1-\tilde{p}_j)$. Note that the number of consecutive time slots where all nodes transmit at rate R_K, m_K, follows a geometric distribution with parameter $\prod_i(1-\tilde{p}_i)$. The expected value of m_K is thus $\bar{m}_K = \frac{\prod_i(1-\tilde{p}_i)}{1-\prod_i(1-\tilde{p}_i)}$, which can be estimated at each node. $\tilde{q}_i$ can then be estimated as $\frac{1-(\bar{m}_K+1)\tilde{p}_i}{(\bar{m}_K+1)(1-\tilde{p}_i)}$.

To design a random access game with $U_i(\tilde{p}_i)$ such that its Nash equilibrium maximizes the total throughput, we use the similar approach as in (4.45). When N is large, we obtain the utility function as

$$U_i(\tilde{p}_i) = \tilde{p}_i + \frac{e^{-\xi}}{1-\alpha}(1-\tilde{p}_i)^{1-\alpha}. \tag{4.46}$$

Similarly, when $\alpha > 1$ the random access game with utility function (4.46) has a unique nontrivial Nash equilibrium.

4.5.2.3 Forward Engineering by Heuristics

Consider random access game with the following payoff function

$$u_i(\mathbf{p}) := U_i(p_i) - p_i \prod_{j\neq i}(1-p_j) = U_i(p_i) - p_i q_i, \tag{4.47}$$

where $q_i = C_i(q_i) = \prod_{j\neq i}(1-p_j)$ is the contention measurement signal representing the probability that all nodes except node i do not transmit. This payoff function is motivated by the heuristic that each wireless node should be "charged" according to the throughput it achieves.

It turns out that the random access game with payoff (4.47) is a supermodular game. Supermodularity was introduced into the game theory by Topkis [76]. Supermodular games are of particular interest since they have many nice properties such as the existence of Nash equilibria and the convergence of the equilibria under different dynamics. The simplicity of supermodular games makes concavity and differentiability assumptions as presumed in Theorems 4.6 and 4.7 unnecessary. For one dimensional user strategy spaces as in random access game, the definition of supermodular game simplifies to the following.

Definition 4.9 $u_i(p_i, \mathbf{p}_{-i})$ *has* nondecreasing differences *in* $(p_i, \mathbf{p}_{-i})$ *if for all* $\mathbf{p}_{-i} \geq$

$\mathbf{p}'_{-i}$ *the quantity* $u_i(p_i, \mathbf{p}_{-i}) - u_i(p_i, \mathbf{p}'_{-i})$ *is nondecreasing in* p_i*. For continuous and twice differentiable payoffs,* $u_i(p_i, \mathbf{p}_{-i})$ *has nondecreasing differences is equivalent to* $\frac{\partial^2 u_i(\mathbf{p})}{\partial p_i \partial p_j} \geq 0$*, for all* $j \neq i$*.*

Definition 4.10 *A random access game* $\mathcal{G}$ *is* supermodular *if, for each node* $i \in \mathcal{N}$*,* $u_i(p_i, \mathbf{p}_{-i})$ *has nondecreasing differences in* $(p_i, \mathbf{p}_{-i})$*.*

It is easy to check that $\partial^2 u_i(\mathbf{p})/\partial p_i \partial p_j = \prod_{j' \neq i, j' \neq j}(1 - p_{j'}) \geq 0$. From Definition 4.10, we have the following theorem.

Theorem 4.11 *A random access game* $\mathcal{G}$ *with payoff function (4.47) is a supermodular game and the set of Nash equilibria for* $\mathcal{G}$ *is nonempty.*

As indicated by Theorem 4.11, no concavity assumption on utility function is required to guarantee the existence of Nash equilibria as in non-supermodular games. By following the same proof as Theorem 4.8, we have the following corollary on the uniqueness of equilibrium for supermodular games.

Corollary 4.12 *Suppose that supermodular game* $\mathcal{G}$ *has a nontrivial Nash equilibrium and the utility function* $U_i(\cdot)$ *is twice continuously differentiable, strictly convex. If* $\Phi_i(p_i) = (1 - p_i)U_i'(p_i)$ *is a strictly decreasing function in* S_i*,* $\mathcal{G}$ *has a unique nontrivial Nash equilibrium.*

As an example, we consider the following utility function given in [20]

$$U_i(p_i) := \frac{1}{a_i}\left(\frac{(a_i - 1)b_i}{a_i}\ln\left(a_i p_i - b_i\right) - p_i\right), \tag{4.48}$$

where $0 < b_i < 1$, $a_i < 1$, and $p_i \in \left(b_i/a_i, \frac{b_i + \sqrt{b_i^2 + a_i(a_i b_i - b_i^2 - b_i)}}{a_i}\right)$. It is easy to check that $U_i(p_i)$ is strictly convex and $\Phi_i'(p_i) < 0$ when $p_i < \frac{b_i + \sqrt{b_i^2 + a_i(a_i b_i - b_i^2 - b_i)}}{a_i}$. From Corollary 4.12, the supermodular game with utility function (4.48) has a unique nontrivial Nash equilibrium.

There are many ways to design utility functions and random access games. We only show a few specific examples in this section. The key point of this section is that the random access game model is general enough to include, if not all, most of existing medium access control algorithms. Most of algorithms can be reverse engineered to be a random access game with specific utility function.

4.6 Dynamics of Random Access Game

The dynamic of game studies how players could converge to a Nash equilibrium. It is a difficult problem in general. In distributed random access games, wireless nodes can observe the outcome (in terms of some contention measure) of the actions of others, but do not have direct knowledge of other nodes' actions and payoffs. We consider repeated play of random access game, and look for strategy update mechanism in which nodes repeatedly adjust channel access probabilities in response to observations of other players' actions so as to achieve the Nash equilibrium.

4.6.1 Basic Dynamic Algorithms

4.6.1.1 Best Response

The simplest update mechanism is the best response strategy: at each stage, every node chooses the best response to the actions of all the other nodes in the previous stage. Let $\mathbf{p}(0)$ be the largest vector in the strategy space $(\mathcal{S}_i)_{i\in\mathcal{N}}$. At stage $t+1$, node $i \in \mathcal{N}$ chooses a channel access probability

$$p_i(t+1) = B_i(\mathbf{p}(t)) := \max\left\{\arg\max_{p\in\mathcal{S}_i} u_i\left(p, \mathbf{p}_{-i}(t)\right)\right\}. \tag{4.49}$$

At each stage, more than one probability may be a best response to a given $\mathbf{p}_{-i}(t)$. In this case, (4.49) always chooses the largest probability. Clearly, if the above dynamics reaches a steady state, this state is a Nash equilibrium. We restrict our discussion on supermodular games. The significance of supermodularity is the fact that the best response strategy converges to a Nash equilibrium. We have the following theorem.

Theorem 4.13 *The best response strategy converges to a Nash equilibrium of random access game $\mathcal{G}$. Furthermore, it is the largest equilibrium in the set of Nash equilibria.*

The proof follows [76, Lemma 4.1]. If we always choose the smallest probability in (4.49) and $\mathbf{p}(0)$ is the smallest vector in the strategy space, the best response strategy will converge to the smallest equilibrium. When there exist multiple equilibria, the following theorem indicates that the equilibrium attained by (4.49) yields the highest aggregate payoff.

Theorem 4.14 *The best response strategy converges to a Pareto dominant equilibrium, i.e., $u_i(\overline{\mathbf{p}}) \geq u_i(\mathbf{p})$ for all $\mathbf{p}$ in the strategy space.*

Denote the mapping $\tilde{B}_i(\mathbf{p}(t)) := \arg\max_p u_i\left(p, \mathbf{p}_{-i}(t)\right)$. The following theorem guarantees that (4.49) converges to a nontrivial equilibrium.

Theorem 4.15 *If $\tilde{B}_i(\mathbf{p}(0))$ belongs to the strategy space and $\tilde{B}_i(\mathbf{p}(0)) \leq p_i(0)$, $\forall i \in \mathcal{N}$, the best response strategy converges to the largest nontrivial Nash equilibrium.*

Similar theorem can also be obtained if $\mathbf{p}(0)$ is the smallest vector in the strategy space. By using Theorem 4.15, it is easy to obtain conditions on a_i and b_i in (4.48) such that the best response strategy converges to a nontrivial equilibrium of the corresponding game. Without using Corollary 4.12, the uniqueness of nontrivial equilibrium can also be obtained by showing that $B_i(\mathbf{p}(t))$ in (4.49) is a contraction mapping. Note that a condition for convergence of best response strategy is given in [20], which is strict and hard to verify. Supermodularity greatly simplifies the conditions for the convergence of best response strategy.

4.6.1.2 Gradient Play

An alternative update mechanism is gradient play [28]. Compared to "best response" strategy, gradient play can be viewed as a "better response". In gradient play, every node adjusts its channel access probability gradually in a gradient direction suggested by contention measurements. Mathematically, each node $i \in \mathcal{N}$ updates its strategy according to

$$p_i(t+1) = [p_i(t) + \epsilon_i(t)(U_i'(p_i(t)) - C_i(\mathbf{q}_i(\mathbf{p}(t))))]^{\mathcal{S}_i}, \tag{4.50}$$

where the stepsize $\epsilon_i(\cdot) > 0$ is a function in time, $[\cdot]^{\mathcal{S}_i}$ denotes the projection onto node i's strategy space. From (4.50), if the marginal utility $U_i'(p_i(t))$ is greater than the contention price $C_i(\mathbf{q}_i(\mathbf{p}(t)))$, we increase the access probability, and if the marginal utility is less than the contention price, we decrease the access probability. In the following, we assume that all nodes have the same stepsize $\epsilon_i(t) = \epsilon(t)$, $\forall i \in \mathcal{N}$.

Theorem 4.16 *Let $C(\mathbf{p}) = (C_i(\mathbf{q}_i(\mathbf{p})))$ be a mapping and $\mathbf{J}^C = (J_{ij}^C)$ be the Jacobian of $C(\mathbf{p})$. Suppose that the smallest eigenvalue of $\mathbf{J}^C$, $\lambda_{\min}(\mathbf{J}^C)$, satisfies $\mu + \lambda_{\min}(\mathbf{J}^C) > 0$, $\max_j \left|J_{ij}^C\right|^2 \leq M$, and the strategy space $(\mathcal{S}_i)_{i\in\mathcal{N}}$ contains a unique nontrivial Nash equilibrium $\mathbf{p}^*$, the gradient play (4.50) converges geometrically to $\mathbf{p}^*$ if the stepsize $\epsilon(t) < \frac{\mu+\lambda_{\min}(\mathbf{J}^C)}{\chi^2+|\mathcal{N}|M}$.*

The proof of Theorem 4.16 is given in the appendix. Theorem 4.16 also shows the convergence rate of gradient play. As an example of using Theorem 4.16, we consider the utility function defined in (4.45). By assuming that all nodes' strategy spaces are identical, i.e., $\mathcal{S} = [\nu, \omega]$. In this case, we have

$$\mu = \frac{\alpha e^{-\xi}}{(1-\nu)^{\alpha+1}}, \; \chi = \frac{\alpha e^{-\xi}}{(1-\omega)^{\alpha+1}}. \tag{4.51}$$

To find $\lambda_{\min}(\mathbf{J}^C)$, we note that

$$\mathbf{J}^C(\mathbf{p}) = -\left(\prod_i (1-p_i)\right)\left(\operatorname{diag}(\mathbf{x})^2 - \mathbf{x}\mathbf{x}^T\right), \tag{4.52}$$

where $\mathbf{x} = \left[\frac{1}{1-p_1}, \ldots, \frac{1}{1-p_{|\mathcal{N}|}}\right]^T$. Note that each entry of $\mathbf{x}$ is less than $\frac{1}{1-\omega}$. By using Rayleigh quotient [43], it is easy to show that the maximum eigenvalue of

$\text{diag}(\mathbf{x})^2 - \mathbf{x}\mathbf{x}^T$ is less than $\frac{1}{(1-\omega)^2}$. Thus, Theorem 4.16 requires that

$$\lambda_{\min}(\mathbf{J}^C) + \mu \geq -\frac{(1-\upsilon)^{|\mathcal{N}|}}{(1-\omega)^2} + \frac{\alpha e^{-\xi}}{(1-\nu)^{\alpha+1}} > 0. \tag{4.53}$$

Condition (4.53) is mild. For example, if we take $\omega = 2/33$ and $\alpha = 2$, all $\nu \in [0, 1]$ satisfy (4.53). We see that a larger α indicates a larger μ, which means a greater convergence rate by (4.84).

Note that $\|\mathbf{p}(t+1) - \mathbf{p}^*\|_2^2$ can be considered as a universal Lyapunov function for all random access games. Theorem 4.16 also implies that if multiple equilibria exist, gradient play converges to one equilibrium but we do not know which one.

4.6.2 Asynchronous Dynamic Algorithms

Due to propagation delay and that all the nodes may not enter the network at the same time, nodes may not update their channel access probability at the same time. In this subsection, we discuss asynchronous counterparts of the algorithms in Section 4.6.1. We assume that the contention measurement signals that node i uses to update its channel access probability result from the vector

$$\mathbf{p}(\tau^i(t)) = \left(p_1(\tau_1^i(t)), p_2(\tau_2^i(t)), \ldots, p_{|\mathcal{N}|}(\tau_{|\mathcal{N}|}^i(t))\right), \tag{4.54}$$

where $0 \leq \tau_j^i(t) \leq t$ denotes the most recent time that node j's action affects node i's observation, and $\tau_i^i(t) = t$.

4.6.2.1 Best Response

The best response strategy (4.49) is modified to

$$p_i(t+1) = B_i(\mathbf{p}(\tau^i(t))) := \max\left\{\arg\max_{p \in \mathcal{S}_i} u_i\left(p, \mathbf{p}_{-i}(\tau^i(t))\right)\right\}. \tag{4.55}$$

Parallel to Theorem 4.13, we have the following theorem on the convergence of asynchronous best response for supermodular games.

Theorem 4.17 *The asynchronous best response strategy (4.55) converges to a Nash equilibrium of the random access game $\mathcal{G}$. Furthermore, it is the largest equilibrium in the set of Nash equilibria.*

Proof. We show this by induction. Suppose that $\mathbf{p}(\tau+1) \leq \mathbf{p}(\tau)$, $\forall \tau \in \{0, \ldots, t-1\}$. It is true when $t = 0$ as $\mathbf{p}(0)$ is the largest vector in the strategy space. As $\tau_j^i(t+1) \geq \tau_j^i(t)$, we have $p_j(\tau_j^i(t+1)) \leq p_j(\tau_j^i(t))$. By induction hypothesis, we get $\mathbf{p}_{-i}(\tau_j^i(t+1)) \leq \mathbf{p}_{-i}(\tau_j^i(t))$. By supermodularity and [76, Lemma 4.1], we can show that $p_i(t+1) \leq p_i(t)$. Therefore, the hypothesis is also true when $\tau = t$. By induction, we have

$$\mathbf{p}(0) \geq \mathbf{p}(1) \geq \cdots \geq \mathbf{p}(t) \geq \cdots, \tag{4.56}$$

or $\{\mathbf{p}(t)\}$ is a nonincreasing sequence. The remainder of proof follows that of Theorem 4.13. ■

All other results in Section 4.6.1 for best response also hold in the asynchronous case.

4.6.2.2 Gradient Play

The gradient play (4.50) is modified to

$$p_i(t+1) = \left[p_i(t) + \epsilon_i(t)\left(U_i'(p_i(t)) - C_i(\mathbf{q}_i(\mathbf{p}(\tau^i(t))))\right)\right]^{\mathcal{S}_i}. \tag{4.57}$$

Since at each step, nodes update channel access probabilities by a small amount, gradient play is expected to converge if $\tau_j^i(t)$ is not far away from t, $\forall j \in \mathcal{N}$. The following result verifies this intuition.

Theorem 4.18 *Let $C(\mathbf{p}) = (C_i(\mathbf{q}_i(\mathbf{p})))$ be a mapping and $\mathbf{J}^C = (J_{ij}^C)$ be the Jacobian of $C(\mathbf{p})$. Assume a constant stepsize $\epsilon_i(t) = \epsilon$ in (4.57). Suppose that $\|\mathbf{J}^C\|_1 \leq M_1$, $\max_j \left|J_{ij}^C\right|^2 \leq M_2$, and the strategy space $(\mathcal{S}_i)_{i\in\mathcal{N}}$ contains a unique nontrivial Nash equilibrium $\mathbf{p}^*$, and $t - \tau_j^i(t) \leq B$, where $B \geq 0$ is a constant, the asynchronous gradient play (4.57) geometrically converges to $\mathbf{p}^*$ if there exists $\epsilon > 0$ and $0 < \gamma < 1$*

such that

$$\gamma = 1 - 2\epsilon \left(\mu - M_1 \sqrt{\frac{|\mathcal{N}|}{\gamma^B}} + \epsilon \left(\chi^2 + \frac{M_2 |\mathcal{N}|^2}{\gamma^B} \right) \right). \tag{4.58}$$

The proof is given in the appendix.

4.6.3 Dynamic Algorithms under Estimation Uncertainties

In this subsection, we consider dynamic algorithms under estimation uncertainties. The dynamic algorithms require the knowledge of contention measure signals. In practice, contention measure signals can be estimated via the observation of the wireless medium over several time slots as in Section 4.5.2.2. Due to the use of estimated contention measure signals, the algorithms in Section 4.6.1 are in fact stochastic algorithms. In the following, we consider gradient play. We assume that $C_i(\mathbf{q}_i(\mathbf{p}(t)))$ is replaced by $\hat{C}_i(\mathbf{q}_i(\mathbf{p}(t))) = C_i(\mathbf{q}_i(\mathbf{p}(t))) + w_i(t)$ in (4.50), where $w_i(t)$ is the error. Let $\mathcal{F}_t$ be an increasing sequence of σ-fields. Without loss generality, we write $w_i(t)$ as $w_i(t) = \bar{w}_i(t) + \tilde{w}_i(t)$, where $\bar{w}_i(t) = E\{w_i(t)|\mathcal{F}_t\}$ can be considered as the deterministic error and $\tilde{w}_i(t) = w_i(t) - \bar{w}_i(t)$ is the stochastic error with zero mean. We further assume that $\lim_{t\to\infty} \bar{w}_i(t) = \bar{w}_i$. The deterministic error may be caused by the bias of signal estimation and carrier sense error due to fading and background noise. For ease of understanding, in the following, we discuss deterministic and stochastic errors separately. The proof of the following theorems can be found in the appendix.

Theorem 4.19 *Let $\lambda_{\min}(\mathbf{J}^C)$ denote the smallest eigenvalue of $\mathbf{J}^C$ and $\max_j \left|J_{ij}^C\right|^2 \leq M$. Let $\mathbf{p}^*$ denote the equilibrium defined by*

$$U_i'(p_i^*) = C_i(\mathbf{q}_i(\mathbf{p}^*)) + \bar{w}_i. \tag{4.59}$$

If $\mathbf{p}^*$ is within the strategy space and it is the unique equilibrium defined by (4.59), the gradient play converges to $\mathbf{p}^*$ provided $\mu + \lambda_{\min}(\mathbf{J}^C) > 0$ and $\epsilon(t) < \frac{\mu+\lambda_{\min}(\mathbf{J}^C)}{\chi^2+4|\mathcal{N}|M}$.

The uniqueness of $\mathbf{p}^*$ can be obtained by using Theorem 4.8. Note that under certain conditions, by implicit function theorem [5], (4.59) defines an implicit function

$\mathbf{p}^*(\bar{\mathbf{w}})$ at the neighborhood of $\bar{\mathbf{w}} = \mathbf{0}$. Therefore, for any $\epsilon > 0$, there exists a $\delta > 0$ such that if $\|\bar{\mathbf{w}}\|_2 < \delta$, $\|\mathbf{p}^*(\bar{\mathbf{w}}) - \mathbf{p}^*(\mathbf{0})\|_2 < \epsilon$. So the gradient play converges to a neighborhood of the equilibrium point without errors.

For the stochastic error, we consider gradient play with variable stepsize and constant stepsize, respectively.

Theorem 4.20 *Let $\lambda_{\min}(\mathbf{J}^C)$ denote the smallest eigenvalue of $\mathbf{J}^C$. Suppose that $E\{w_i(t)|\mathcal{F}_t\} = 0$, $E\{w_i^2(t)|\mathcal{F}_t\} \leq B$, and*

$$\sum_{t=0}^{\infty} \epsilon(t) = \infty, \sum_{t=0}^{\infty} \epsilon^2(t) < \infty, \text{ e.g., } \epsilon(t) = 1/t. \tag{4.60}$$

If $\mathbf{p}^*$ is the unique nontrivial Nash equilibrium, the gradient play converges to $\mathbf{p}^*$ with probability 1 provided $\mu + \lambda_{\min}(\mathbf{J}^C) > 0$.

Theorem 4.21 *Let $\lambda_{\min}(\mathbf{J}^C)$ denote the smallest eigenvalue of $\mathbf{J}^C$ and $\max_j \left|J_{ij}^C\right|^2 \leq M$. Suppose that $E\{w_i(t)|\mathcal{F}_t\} = 0$, $E\{w_i^2(t)|\mathcal{F}_t\} \leq B$, and $\epsilon(t) = \epsilon$, $\forall t$. If $\mathbf{p}^*$ is the unique nontrivial Nash equilibrium, there exists a constant $D(B, \epsilon) > 0$ such that*

$$\limsup_{t\to\infty} \|\mathbf{p}(t) - \mathbf{p}^*\|_2 \leq D(B, \epsilon) \tag{4.61}$$

provided $\mu + \lambda_{\min}(\mathbf{J}^C) > 0$ and $\epsilon < \frac{\mu+\lambda_{\min}(\mathbf{J}^C)}{\chi^2+4|\mathcal{N}|M}$.

By combining Theorems 4.19 and 4.21, we can conclude that with constant stepsize, the stochastic gradient play converges to a neighborhood of the equilibrium point.

4.6.4 Equilibrium Selection

The equilibrium attained by using the dynamic algorithms in Section 4.6.1 does not necessarily converge to the desired operating point when the utility functions in Section 4.5.2.2 are considered. This is because the approximation used in (4.44). One approach of equilibrium selection is to estimate the number of users via $\hat{N} =$

$\log(1-q_i)/\log(1-p_i)+1$ at equilibrium and to set the channel access probability to be the optimal value computed by using $\hat{N}$. However, as commented in [44], this approach may not converge due to open loop control. The other approach is to use an outer loop iteration and treat the algorithms in Section 4.6.1 as the inner loop iteration. Take utility function (4.45) for example. Let τ denote the counter of outer loop iteration and define the utility function at the τ-th outer iteration as

$$U_i(p_i) = p_i + \frac{\eta(\tau)}{1-\alpha}(1-p_i)^{1-\alpha}, \tag{4.62}$$

where $\eta(0) = e^{-\xi}$ and the equilibrium point of this utility function as $p_i(\tau)$. To cancel the effect of neglecting $(1-p_i)^{\alpha-1}$ in (4.44), we do the outer iteration

$$\eta(\tau+1) = (1-p_i(\tau))^{\alpha-1}e^{-\xi}. \tag{4.63}$$

At equilibrium, all nodes have the same access probability, denoted as $p(\tau)$. By (4.63), we obtain

$$p(\tau+1) = 1 - \sqrt[|\mathcal{N}|+\alpha-1]{(1-p(\tau))^{\alpha-1}e^{-\xi}}. \tag{4.64}$$

Let $\mathcal{M}(p)$ be the mapping defined by (4.64). By mean value theorem, it is easy to see

$$|\mathcal{M}(p_1) - \mathcal{M}(p_2)| \leq \frac{e^{-\frac{\xi}{|\mathcal{N}|+\alpha-1}}(\alpha-1)(1-\omega)^{\frac{\alpha-1}{|\mathcal{N}|+\alpha-1}-1}}{|\mathcal{N}|+\alpha-1}|p_1-p_2| \tag{4.65}$$

Thus, if $\frac{e^{-\frac{\xi}{|\mathcal{N}|+\alpha-1}}(\alpha-1)(1-\omega)^{\frac{\alpha-1}{|\mathcal{N}|+\alpha-1}-1}}{|\mathcal{N}|+\alpha-1} < 1$, $\mathcal{M}(p)$ is a contraction mapping [5] and (4.64) converges to the unique fixed point of $\mathcal{M}(p)$, which is the desired operating point. Note that in Section 4.6.1-Section 4.6.3, we have established the convergence of inner loop under different conditions. Therefore, the whole algorithm converges to the desired operating point.

From (4.65), we can see that a larger α indicates a smaller outer loop convergence rate, while a larger α results a greater inner loop convergence rate as suggested in Theorem 4.16. Therefore, there exists an optimal α to achieve the least overall convergence rate. In practice, when exact $p(\tau)$ is not available, we can use the average probability over a long duration. Also, outer loop iteration can be executed without waiting for the convergence of the inner loop iteration.

4.7 Rate Splitting

Rate splitting has been applied to interference channels in [36] and multiple access channel in [34, 69]. In mac, it is shown in [34, 69] that rate splitting provides another way to achieve arbitrary point within the capacity region of mac defined in (4.10) besides time sharing. Rate splitting has been applied to Aloha in [14, 59]. By using rate splitting, each user is split into several virtual users sharing the total power of the actual user. The same successive interference cancellation method is used to decode each virtual user's packets. It is shown in [14] that as the number of virtual users at each node goes to infinity, the achievable sum rate of the distributed rate splitting converges to the maximum possible sum rate of the mac. The algorithm in [14] requires perfect knowledge of the number of users in the network and each user needs to change its virtual users' transmission rates whenever the total number of users in the network changes. In this section, we propose an alternative rate splitting scheme with a finite number of virtual users, each with multiple possible transmission rates, by extending our multiple access MAC scheme from Section 4.3. The rate splitting scheme improves throughput at the expense of additional complexity.

We begin by considering the two user network as in Section 4.2. Two virtual users, denoted as U_i' and U_i'' are created at node i, $i = 1, 2$, with power αP and $(1-\alpha)P$, respectively, where α is a parameter to be optimized. We take a suboptimal layering approach as in [14], where U_i'''s packet is always decoded before U_i''s packet. Thus, each virtual user as in Section 4.2 only needs to consider two rates, i.e., U_i' takes

$$R_k'(\alpha) = \frac{1}{2}\log\left(1 + \frac{\alpha P}{(k-1)\alpha P + \sigma^2}\right), \; k = 1, 2, \tag{4.66}$$

with probability p'_k and U''_i takes

$$R''_k(\alpha) = \frac{1}{2}\log\left(1 + \frac{(1-\alpha)P}{(k-1)(1-\alpha)P + 2\alpha P + \sigma^2}\right),\ k = 1, 2, \tag{4.67}$$

with probability p''_k. We choose $p'_1 = p'$, $p'_2 = 1 - p'$, and $p''_1 = p''$, $p''_2 = 1 - p''$. Note that our strategy is different from [14] where U'_i only transmits at rate $R'_2(\alpha)$ and U''_i only transmits at rate $R''_2(\alpha)$. The approach in [14] can be considered as a special case of our strategy by choosing $p' = p'' = 0$.

As in Section 4.2, the average throughput of multiple access MAC with rate splitting can be obtained as

$$\begin{aligned} R_{\text{drs}-\text{mac}}(p', p'', \alpha) =& 2\Big(\left(1 - p''^2\right) \left(\left(R'_1(\alpha) + R'_2(\alpha)\right) p'(1-p') + R'_2(\alpha)(1-p')^2\right) \\ & + \left(R''_1(\alpha) + R''_2(\alpha)\right) p''(1-p'') + R''_2(\alpha)(1-p'')^2\Big). \end{aligned} \tag{4.68}$$

Given α and p'', we first maximize (4.68) over p' and obtain

$$\begin{aligned} R^*_{\text{drs}-\text{mac}}(p'', \alpha) = 2\Big(& \left(1 - p''^2\right) \frac{\left(R'_1(\alpha) + R'_2(\alpha)\right)^2}{4R'_1(\alpha)} \\ & + \left(R''_1(\alpha) + R''_2(\alpha)\right) p''(1-p'') + R''_2(\alpha)(1-p'')^2\Big). \end{aligned} \tag{4.69}$$

Let $A = \frac{\left(R'_1(\alpha)+R'_2(\alpha)\right)^2}{4R'_1(\alpha)}$. Maximizing (4.69) over p'' we obtain

$$R^*_{\text{drs}-\text{mac}}(\alpha) = \frac{\left(R''_1(\alpha) + R''_2(\alpha) + 2A\right)^2}{2(R''_1(\alpha) + A)}. \tag{4.70}$$

By performing a linear search over α, we obtain the maximum total throughput $R^*_{\text{drs}-\text{mac}}$.

Fig. 4.5 compares the throughput of the proposed algorithm with that in [14] and Aloha with $\sigma^2 = 1$ and different P for two virtual users. We can see that the proposed schemes perform better than both Aloha and the one in [14]. The achievable rate of [14] is saturated when P is large due to the lack of contention resolution mechanism. By using rate splitting, an additional performance gain is attained by using the proposed protocol compared with that without using rate splitting.

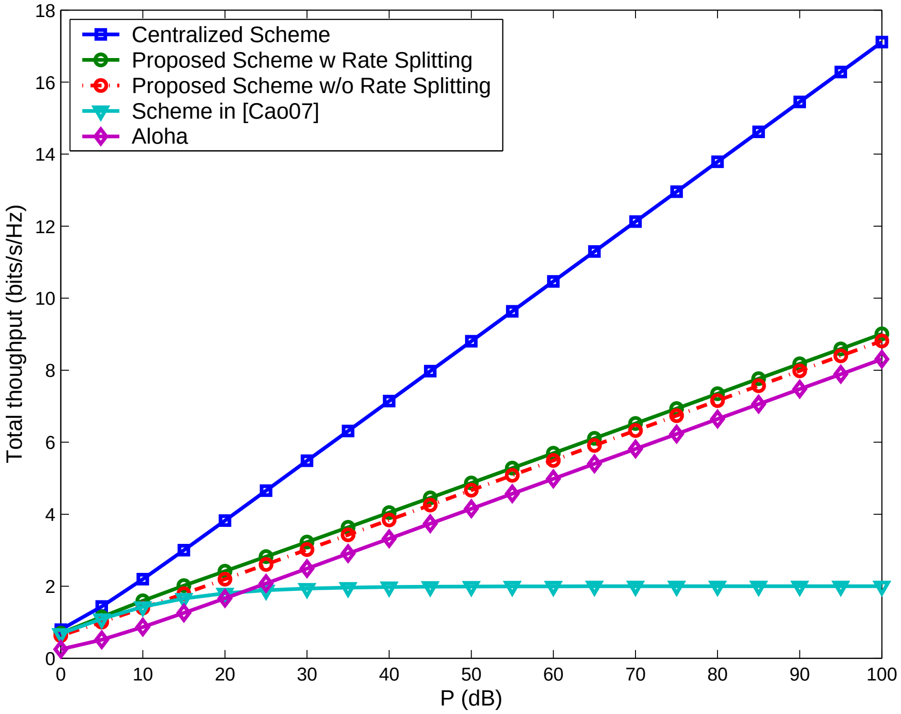


Figure 4.5: Comparison of different schemes with $\sigma^2 = 1$ and different P in a network with 2 users.

The proposed approach can be readily generalized to the case of choosing M virtual users at each node. We still assume a layered decoding approach at the receiver, where the packets from virtual users at layer m are decoded before the packets from virtual users at layer $m-1$. The users at layer m are assigned power $\tilde{P}_m$ such that $\sum_{m=1}^{M} \tilde{P}_m = P$. Let ϕ_m be the probability that all layer m users' packets are decoded correctly conditioned on all the packets at layers less than m being decoded successfully, which can be computed by a similar approach as in Section 4.3. Let R_m be the total throughput of users at layer m given all lower layers' packets are decoded correctly. The total throughput of all virtual users can be written as

$$R_{\text{drs-mac}} = \sum_{m=1}^{M} R_m \Pi_{l=1}^{m-1} \phi_l. \tag{4.71}$$

The optimization of $\tilde{P}_m$ and the rate selection probabilities at each layer are coupled and are complicated to optimize. We thus take a suboptimal approach by decoupling the two optimizations. We first optimize over $\tilde{P}_m$ by ignoring each layer's impact on upper layers or assuming $\phi_m = 1$, $m = 1, \dots, M$. Note that the decoding process at each layer is similar to that in Section 4.3. According to Theorem 4.1, the achievable throughput of the users at layer m can be approximated as

$$R_m = \frac{C}{2} \log \left(1 + \frac{N\tilde{P}_m}{N \sum_{l=m+1}^{M} \tilde{P}_l + \sigma^2} \right). \tag{4.72}$$

By following the approach in [14], we can show that $\tilde{P}_m = \frac{\sigma^2}{N} \left(1 + \frac{NP}{\sigma^2}\right)^{\frac{M-m}{M}} \left(\left(1 + \frac{NP}{\sigma^2}\right)^{\frac{1}{M}} - 1 \right)$, $m = 1, \dots, M$, maximizes $\sum_{m=1}^{M} R_m$. After obtaining $\tilde{P}_m$, we optimize the rate assignment probability of layer m backward from $m = M$ to $m = 1$. When it comes to layer m, we need to maximize $\sum_{i=m}^{M} R_i \Pi_{l=m}^{i-1} \phi_l$, which can be solved similarly as in Section 4.3.

4.8 Simulation Results

In this section, we present simulation results on the proposed multiple access MAC protocol in both Aloha type networks and WLAN. We will discuss the benefits of the proposed protocol over existing protocols.

4.8.1 Aloha Type Networks

We first study the achievable throughput of the proposed protocol with SIC in Aloha type networks. We only consider the maximum achievable throughput without protocol overhead. The performance of the dynamic algorithms to achieve such throughput will be presented in the next subsection.

Fig. 4.6 compares the achievable throughput of different strategies as a function of N when $P = 10$ and $\sigma^2 = 1$ (SNR=10 dB). We compare the proposed protocol with centralized scheme and conventional Aloha. In our proposed protocol, we set the number of transmission rates at each user to be N. We also include the lower bound (4.23) in Theorem 4.1. The "Equal Probability" throughput is obtained by $p_k = \frac{\alpha}{N-1}$,

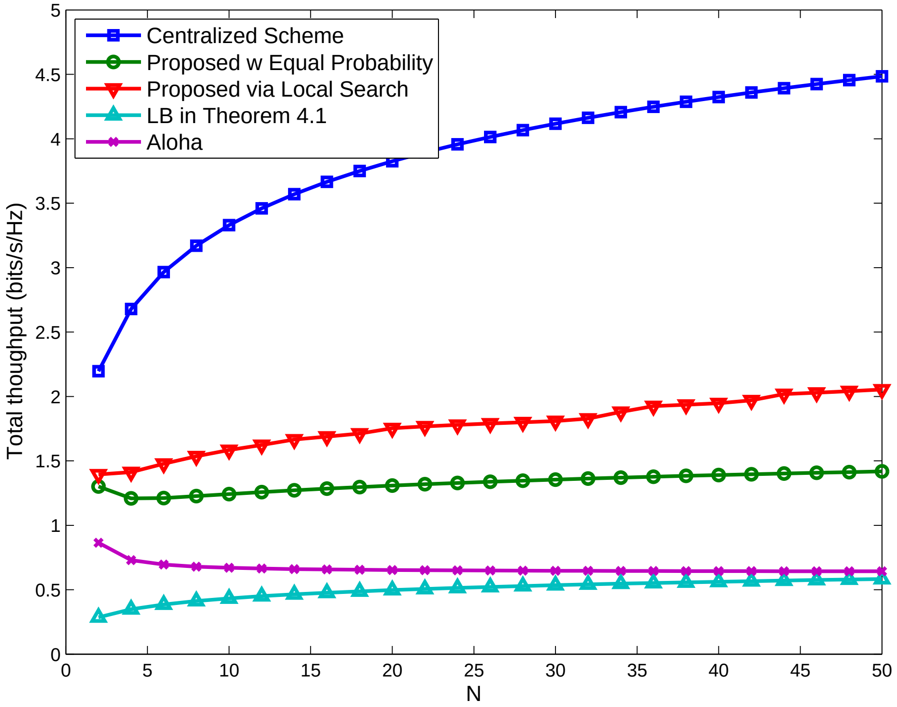


Figure 4.6: Achievable throughput comparison of different strategies as a function of N when $P = 10$ and $\sigma^2 = 1$ (SNR=10 dB).

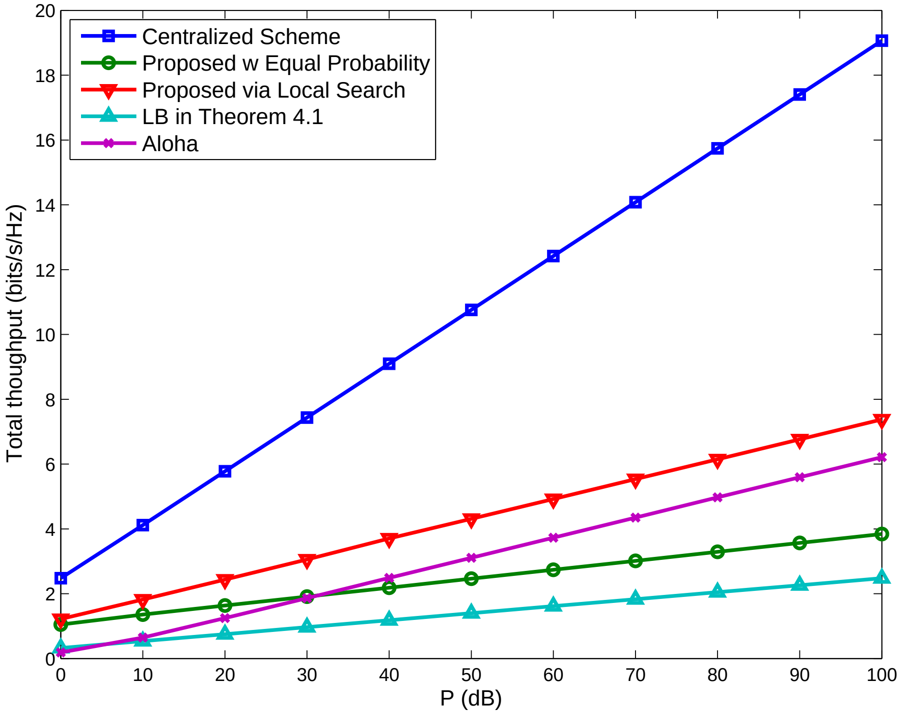


Figure 4.7: Achievable throughput comparison of different strategies as a function of P in a network with $N = 30$ users and $\sigma^2 = 1$.

$k = 1, \ldots, N-1$ and $p_N = 1 - \alpha$, where $\alpha = 0.2011$ as in the proof of Theorem 4.1. The optimized throughput of the proposed strategy is obtained by maximizing (4.13) via a local search around the "Equal Probability", which does not necessarily achieve the maximum throughput. The throughput of Aloha decreases as N increases while that of the proposed protocol increases as N increases. When $N = 50$, the proposed protocol with local search achieves a 3.1951 times throughput over Aloha. Even with equal probability, the proposed protocol has a 2.2064 times throughput over Aloha at $N = 50$. We also find that the lower bound (4.23) in Theorem 4.1 is very loose. The ratio between the centralized scheme and the proposed strategy with local search decreases as N increases. When $N = 50$, the proposed strategy with local search attains 0.4580 throughput of the centralized scheme.

Fig. 4.7 compares the achievable throughput of different protocols as a function of P or SNR in a network with $N = 30$ users and $\sigma^2 = 1$. Let the slope of each curve be denoted as ρ. The throughput of each protocol can be written as $B(N) + \rho \log \frac{P}{\sigma^2}$ in high SNR, where $B(N)$ is a function of N. From Fig. 4.7, we find that slope ρ of the proposed protocol with local search is the same as that of Aloha, which is $\frac{1}{e}$. It seems that the throughput difference between the proposed protocol and Aloha lies in that $B(N) = 0$ in Aloha while $B(N)$ is an increasing function in N in the proposed protocol.

Fig. 4.8 compares the achievable throughput of the proposed protocol with a finite number K transmission rates as in Section 4.3.3. The proposed protocol with local search with N transmission rates is also compared. The other settings are similar to those in Fig. 4.6. The throughput of the proposed protocol increases by increasing K. Even with $K = 2$, the proposed protocol achieves a 3.4167 times throughput gain over Aloha at $N = 50$. However, unlike that using N transmission rates whose throughput strictly increases as N increases, the throughput by using a finite number of transmission rates converges to a value as $N \to +\infty$ like Aloha.

Fig. 4.9 compares the achievable throughput of the proposed protocol with only K transmission rates as a function of P or SNR in a network with $N = 30$ users and $\sigma^2 = 1$. The proposed protocol with local search with N transmission rates is

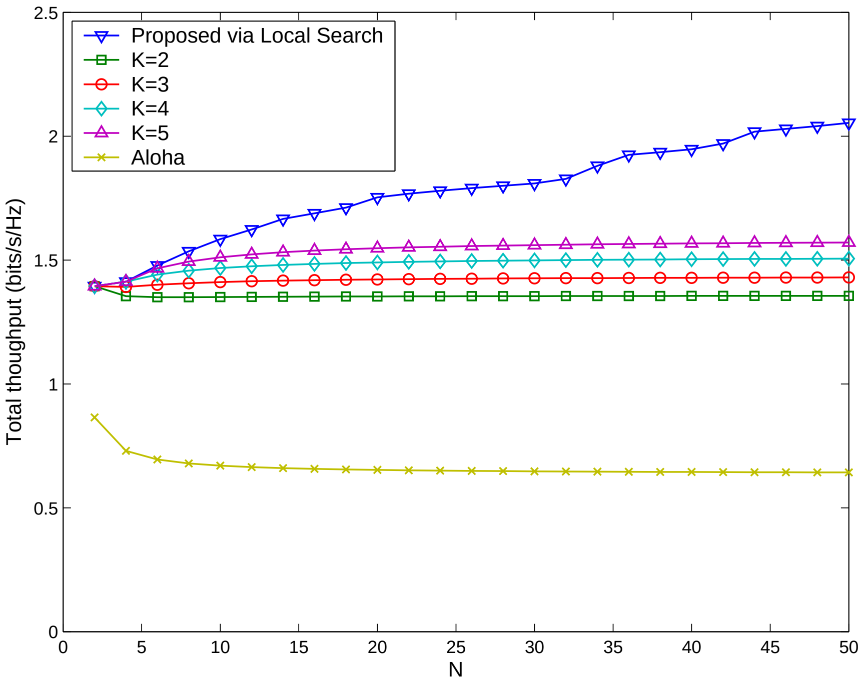


Figure 4.8: Achievable throughput comparison of different strategies as a function of N when $P = 10$ and $\sigma^2 = 1$ (SNR=10 dB). A finite number K of transmission rates is chosen in the proposed scheme.

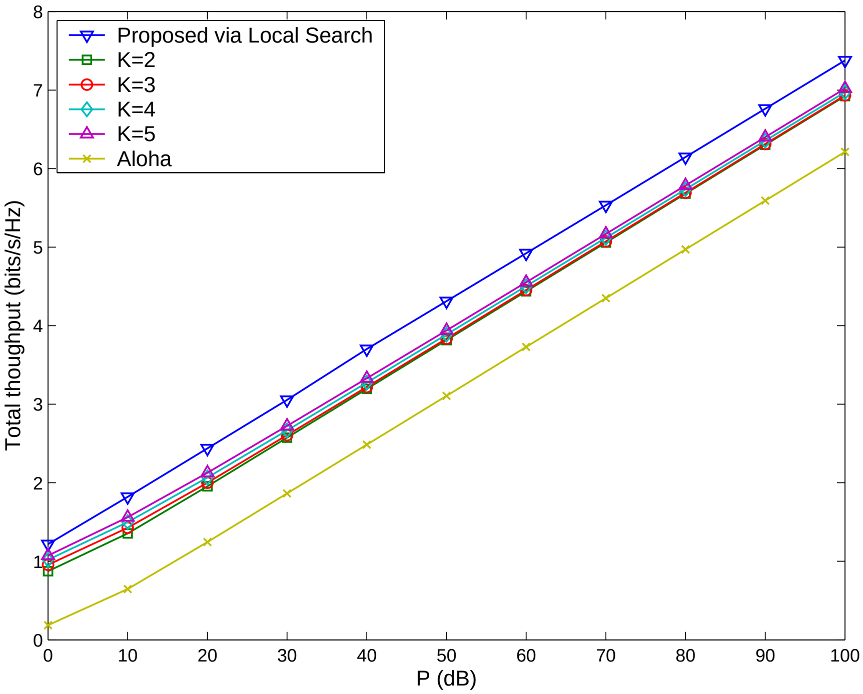


Figure 4.9: Achievable throughput comparison of different strategies as a function of P in a network with $N = 30$ users and $\sigma^2 = 1$. A finite number K of transmission rates is chosen in the proposed scheme.

also compared. As Fig. 4.7, the throughput of the proposed protocol increases as K increases. But different K's have the same slope ρ as Aloha and that with N transmission rates. Their difference lies only in $B(N)$.

4.8.2 Half Duplex WLAN

In this section, we compare the performance of different medium access protocols using a packet-level simulator. The system parameters are those specified in the 802.11a standard with DSSS PHY layer [2], where the values of parameters are summarized in Table 4.1.

Table 4.1: Parameters in Half Duplex WLAN Simulations

Slot Time (T_{SLOT})	9 μs
SIFS	16 μs
DIFS	34 μs
Propagation Delay	1 μs
Header	20 μs
ACK	4 μs

4.8.2.1 Throughput Comparison with Capacity Formula

We first consider the information theoretic result using capacity formula (4.12), which assumes ideal error correcting codes. The system is allocated 20 MHz bandwidth. The effective data transmission time T_d is 37.9259 μs, which corresponds to transmitting 256 bytes data using 54 Mbps in 802.11a.

Fig. 4.10 compares the achievable throughput of the proposed MAC protocol without SIC using the utility function (4.45), the one with SIC using the utility function (4.46) and IEEE 802.11 DCF when $P = 1$ and $\sigma^2 = 1$ (SNR=0 dB). Similar phenomena as in Fig. 4.1 are observed. The throughput of the protocol with SIC is optimized via a local search around the "Equal Probability". The throughput of the protocol without SIC decreases as N increases while that of the protocol with SIC increases as N increases. When $N = 50$, the protocol with SIC achieves a 1.8912

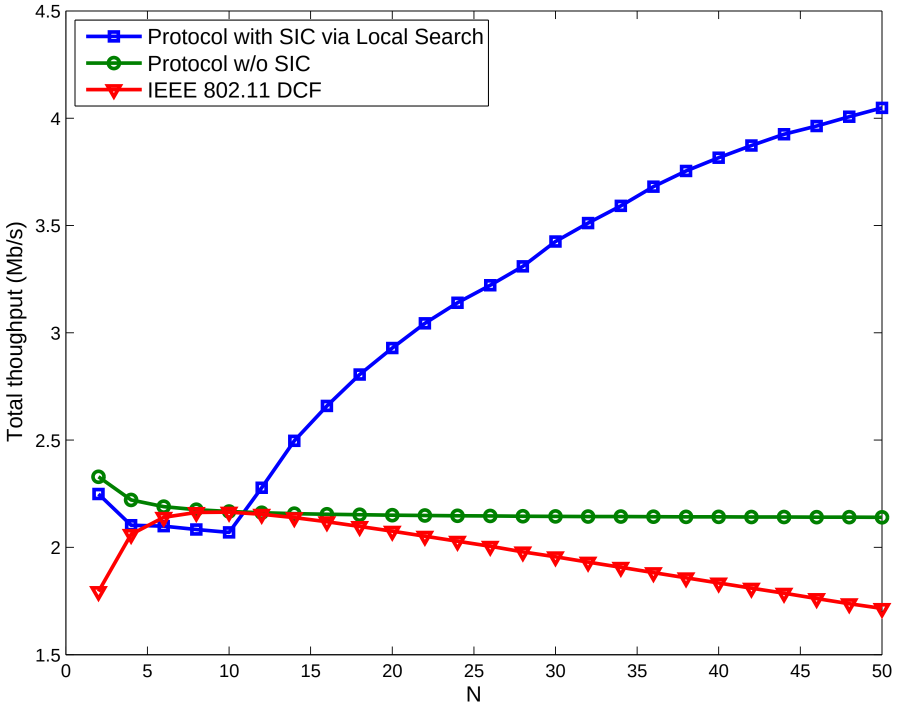


Figure 4.10: Achievable throughput comparison of different strategies as a function of N in WLAN when $P = 10$ and $\sigma^2 = 1$ (SNR=10 dB).

times throughput over the protocol without SIC and a 2.3615 throughput gain over IEEE 802.11 DCF.

Fig. 4.11 compares the achievable throughput of different protocols as a function of P or SNR in a network with $N = 30$ nodes and $\sigma^2 = 1$. We can see that the protocol with SIC performs better than that without SIC when SNR$<$ 2.5 or in low SNR, while the latter performs almost the same as the former in high SNR. This phenomenon can be understood from (4.34), where it is easy to check that when $\eta_2/\eta_1 = \frac{T_{\text{SLOT}}}{T_t}$ is small and $R_1 \gg R_2$ we have $\xi_2 = 0$. In WLAN, $\frac{T_{\text{SLOT}}}{T_t}$ is usually very small due to carrier sensing. In high SNR, the protocol without SIC using existing 802.11 parameters has a close to optimum performance without using multirate and rate splitting. On the other hand, in slotted Aloha networks, $\eta = 1$ and the protocol with SIC performs better than conventional Aloha. From Fig. 4.10 and Fig. 4.11, we find that the protocol with SIC outperforms existing protocols in a network with a large number of users or when SNR is low.

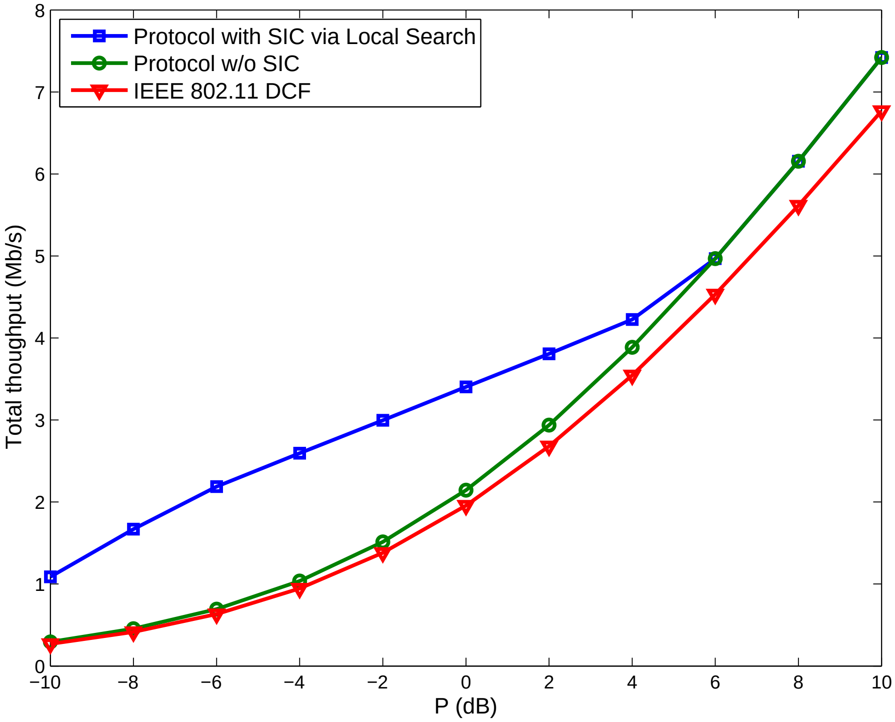
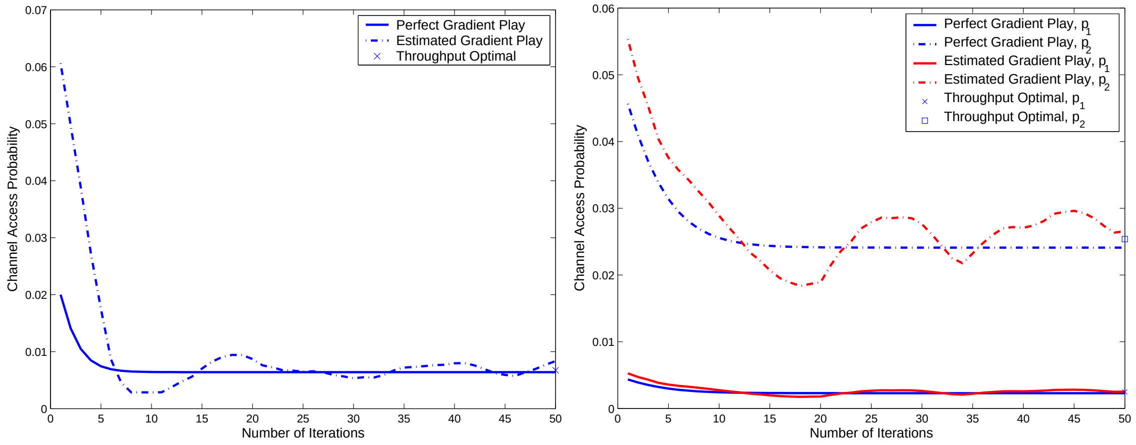


Figure 4.11: Achievable throughput comparison of different strategies as a function of P in a WLAN with $N = 30$ users and $\sigma^2 = 1$.

4.8.2.2 Throughput Comparison with Convolutional Codes

Next, we consider a more practical scenario, where 1/2 rate convolutional code with BPSK modulation in IEEE 802.11a is assumed. We choose $R_1 = R_2 = 6$ Mb/s and $\omega_1 = 1$, $\omega_2 = 2$ as in Example 4.3. We assume that the SNR is 10 dB. When only one node transmits at R_1, the packet delivery probability can be approximated as $\lambda_1 = 1$, while when two nodes transmits at rate R_2 the packet delivery probability can be approximated to be $\lambda_2 = 0.8738$ at signal to interference plus noise ratio (SINR) $\frac{P}{P+\sigma^2}$. We compare the protocol with SIC with that without SIC and IEEE 802.11 DCF.

Fig. 4.12 shows the protocol without SIC and the proposed protocol in a network with 30 nodes, where both protocols use gradient play and the stepsize is chosen to be $\epsilon_i(t) = 0.02$ in (4.50). By maximizing (4.34), we find that $\xi_1 = 0.0722$ and $\xi_2 = 0.7574$. We choose $\xi = 0.8296$ and $\alpha = 2$ in (4.46). The dynamic with perfect contention measurement signal is denoted as "Perfect" while that with estimated



(a) Protocol without SIC

(b) Protocol with SIC

Figure 4.12: Dynamics of the protocol without SIC and the proposed protocol in a network with $N = 30$ nodes. We choose $\xi_1 = 0.0722$ and $\xi_2 = 0.7574$ in (4.34), $\xi = 0.8296$ and $\alpha = 2$ in (4.46), and the stepsize $\epsilon_i(t) = 0.05$ in (4.50).

contention measurement signal is denoted as “Estimated”. With perfect signal, both protocols converges to the equilibrium only after 5 iterations. Even with estimated signal, we can see that both protocols oscillate around the equilibrium after less than 10 iterations, which agrees with Theorem 4.19 that with estimated signal the protocol converges to within a small neighborhood of the desired equilibrium. In Fig. 4.12, we also show the optimal channel access probability to achieve the maximum throughput. The equilibrium of game model is close to the optimal value but not equal due to unknown number of users and the approximation in (4.44). The equilibrium selection algorithm in Section 4.6.4 can be used to achieve the desired equilibrium exactly.

Fig. 4.13 compares the achievable throughput of different protocols as a function of N. In both the protocol with SIC and the one without SIC, we include both the maximum achievable throughput by maximizing the throughput expression directly (denoted as “Perfect”) and the achievable throughput using the game model in Section 4.5 (denoted as “Estimated”). We can see that the throughput of both the protocol without SIC and IEEE 802.11 DCF decreases as N increases while that of the protocol with SIC increases as N increases. The throughput of both the protocol without SIC and the protocol with SIC converges to a constant value, while the throughput of

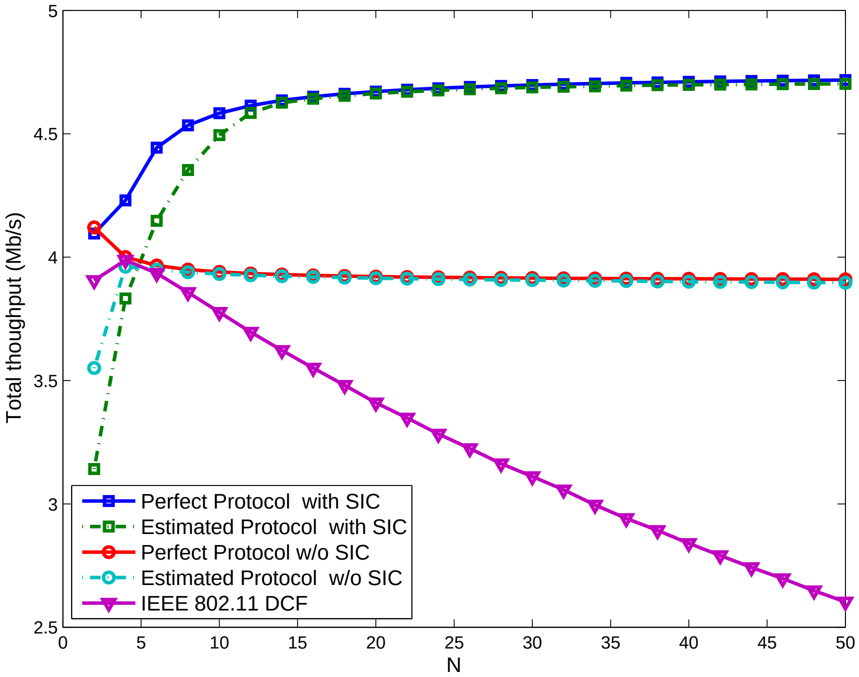


Figure 4.13: Throughput comparison of different protocols as a function of N when $P = 10$ and $\sigma^2 = 1$ (SNR=10 dB). 1/2 rate convolutional code with BPSK is assumed.

IEEE 802.11 DCF strictly decreases in N. The protocol with SIC has a 17.60% throughput gain over that without SIC at $N = 50$, while the gain is 81.31% over IEEE 802.11 DCF. In both the protocol with SIC and the one without SIC, the throughput loss due to use of the game model in Section 4.5 is negligible when N is large.

Fig. 4.14 compares short-term fairness of different protocols using Jain fairness index [46] in a network with $N = 30$ nodes for normalized window sizes that are multiples of the number of wireless nodes. All other parameters are the same as in Fig. 4.13. We can see that both the protocol without SIC and the protocol with SIC provide better short-term fairness than IEEE 802.11 DCF as in both protocols wireless nodes have roughly the same contention window size. Interestingly, besides achieving a higher throughput, the protocol with SIC provides even a better short-term fairness than the protocol without SIC because the protocol with SIC allows multiple nodes transmit simultaneously.

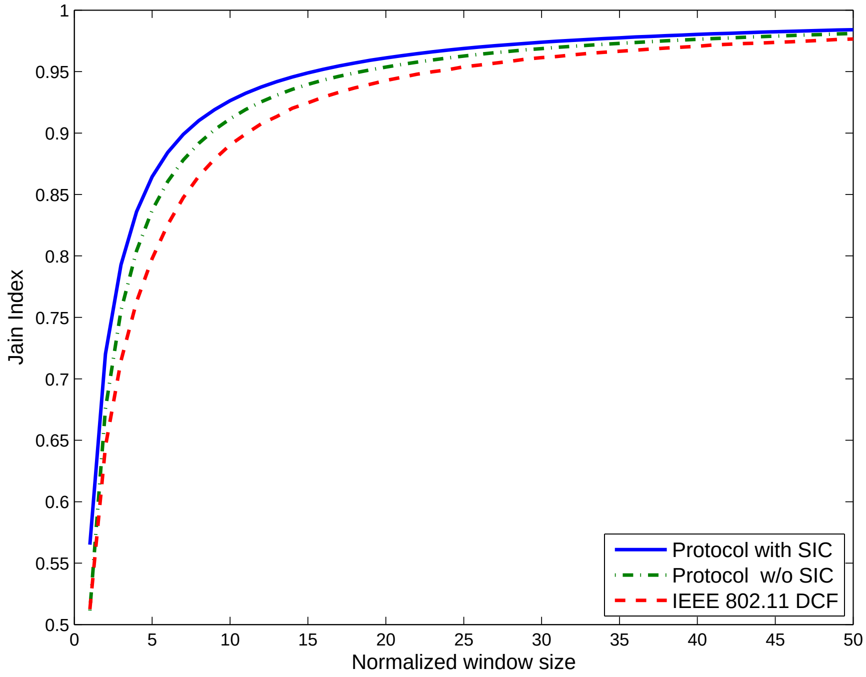


Figure 4.14: Fairness comparison of different protocols in a network with $N = 30$ nodes, $P = 10$ and $\sigma^2 = 1$ (SNR=10 dB). 1/2 rate convolutional code with BPSK is assumed.

4.9 Conclusion

In this chapter, we have developed a new class of random access protocols. These protocols allow each user to transmit at multiple potential data rates. By using successive interference cancellation, multiple packets can be received simultaneously. In slotted Aloha type networks with Gaussian channels, we showed that the achievable sum rate of the new protocol is at least a constant fraction of the information theoretic limit. The proposed protocol was also extended to wireless LAN with half duplex nodes. To achieve the desired throughput optimal equilibrium in a distributed fashion without the knowledge of N, we have designed a random access game and provided dynamic algorithms, whose convergence to the equilibrium is established. Generalization to rate splitting was discussed in the end.

4.10 Appendix

4.10.1 Proof of Theorem 4.3

Proof. As in (4.13), the total throughput of all the users where each user can transmit at one of K rates, $R_{\mathrm{d},K}(N)$, can be derived as

$$\begin{aligned} & R_{\mathrm{d},K}(N) \\ =& N\sum_{k=1}^{K-1} p_k R_k \sum_{\substack{m_k+\sum_{l=k+1}^{j} n'_l \le j,\\ j=k,\ldots,K-1}} \binom{N-1}{m_k, n'_{k+1}, \ldots, n'_K} \prod_{j=k+1}^{K-1} p_j^{n'_j} \left(1-\sum_{j=1}^{K-1} p_j\right)^{n'_K} \left(\sum_{j=1}^{k} p_j\right)^{m_k} \\ & + Np_K R_K, \end{aligned} \tag{4.73}$$

where n'_k denotes the number of users transmitting at rate R_l other than user i with $n'_l = n_l - 1$ if $l = k$ and $n'_l = n_l$ if $l \neq k$, and $m_k = \sum_{j=1}^{k} n'_j$. We next show that by choosing $p_k = \frac{\xi_k}{N}$, $k = 1, \cdots, K-1$, the asymptotic optimal rate can be achieved. We can write (4.73) as

$$\begin{aligned} & R_{\mathrm{d},K}(N) \\ =& N\sum_{k=1}^{K-1} p_k R_k \sum_{\substack{m_k+\sum_{l=k+1}^{j} n'_l \le j,\\ j=k,\ldots,K-1}} \frac{(N-1)(N-2)\cdots n'_K}{m_k! n'_{k+1}! \cdots n'_{K-1}!} \prod_{j=k+1}^{K-1} p_j^{n'_j} \left(1-\sum_{j=1}^{K-1} p_j\right)^{n'_K} \left(\sum_{j=1}^{k} p_j\right)^{m_k} \\ & + Np_K R_K. \end{aligned} \tag{4.74}$$

Note that $N - K + 1 \le n'_K \le N$. Define

$$\begin{aligned}
&\hat{R}_{\mathrm{d},K}(N) \\
&= \sum_{k=1}^{K-1} N p_k R_k \sum_{\substack{m_k + \sum_{l=k+1}^{j} n'_l \le j, \\ j=k,\dots,K-1}} \frac{N^{m_k + \sum_{l=k+1}^{K-1} n'_l}}{m_k! n'_{k+1}! \cdots n'_{K-1}!} \prod_{j=k+1}^{K-1} p_j^{n'_j} \left(1 - \sum_{j=1}^{K-1} p_j\right)^{N-K} \left(\sum_{j=1}^{k} p_j\right)^{m_k} \\
&\qquad + N p_K R_K \\
&= \sum_{k=1}^{K-1} \xi_k R_k \sum_{\substack{m_k + \sum_{l=k+1}^{j} n'_l \le j, \\ j=k,\dots,K-1}} \frac{1}{m_k! n'_{k+1}! \cdots n'_{K-1}!} \prod_{j=k+1}^{K-1} \xi_j^{n'_j} \left(1 - \frac{1}{N} \sum_{j=1}^{K-1} \xi_j\right)^{N-K} \left(\sum_{j=1}^{k} \xi_j\right)^{m_k} \\
&\qquad + \left(N - \sum_{k=1}^{K} \xi_k\right) R_K,
\end{aligned} \tag{4.75}$$

and

$$\begin{aligned}
&\check{R}_{\mathrm{d},K}(N) \\
&= \sum_{k=1}^{K-1} (N-K) p_k R_k \sum_{\substack{m_k + \sum_{l=k+1}^{j} n'_l \le j, \\ j=k,\dots,K-1}} \frac{(N-K)^{m_k + \sum_{l=k+1}^{K-1} n'_l}}{m_k! n'_{k+1}! \cdots n'_{K-1}!} \prod_{j=k+1}^{K-1} p_j^{n'_j} \left(1 - \sum_{j=1}^{K-1} p_j\right)^{N} \left(\sum_{j=1}^{k} p_j\right)^{m_k} \\
&\qquad + (N-K) p_K R_K \\
&= \sum_{k=1}^{K-1} \check{\xi}_k R_k \sum_{\substack{m_k + \sum_{l=k+1}^{j} n'_l \le j, \\ j=k,\dots,K-1}} \frac{1}{m_k! n'_{k+1}! \cdots n'_{K-1}!} \prod_{j=k+1}^{K-1} \check{\xi}_j^{n'_j} \left(1 - \frac{1}{N-K} \sum_{j=1}^{K-1} \check{\xi}_j\right)^{N} \left(\sum_{j=1}^{k} \tilde{\xi}_j\right)^{m_k} \\
&\qquad + \left(N - K - \sum_{k=1}^{K} \tilde{\xi}_k\right) R_K,
\end{aligned} \tag{4.76}$$

where $N p_k = \xi_k$ and $(N-K) p_k = \tilde{\xi}_k$. We thus have

$$\max_{\tilde{\xi}_k} \check{R}_{\mathrm{d},K}(N) \le \max_{p_k} R_{\mathrm{d},K}(N) \le \max_{\xi_k} \hat{R}_{\mathrm{d},K}(N). \tag{4.77}$$

Since K is a finite number, we obtain

$$\begin{aligned}
&\lim_{N\to+\infty} \check{R}_{\mathrm{d},K}(N) = \lim_{N\to+\infty} \hat{R}_{\mathrm{d},K}(N) \\
&= \sum_{k=1}^{K-1} \xi_k R_k \sum_{\substack{m_k+\sum_{l=k+1}^{j} n'_l \le j, \\ j=k,\dots,K-1}} \frac{1}{m_k! n'_{k+1}! \cdots n'_{K-1}!} \prod_{j=k+1}^{K-1} \xi_j^{n'_j} e^{-\sum_{j=1}^{K-1} \xi_j} \left(\sum_{j=1}^{k} \xi_j \right)^{m_k} \\
&\qquad + \log e - \sum_{k=1}^{K} \xi_k R_K
\end{aligned} \tag{4.78}$$

by setting $\xi_k = \tilde{\xi}_k$. By using the squeeze rule in calculus, we obtain

$$\lim_{N\to+\infty} \max_{p_k} R_{\mathrm{d},K}(N) = \lim_{N\to+\infty} \max_{\xi_k} \hat{R}_{\mathrm{d},K}(N). \tag{4.79}$$

From (4.79), we can see that $p_k = \frac{\xi_k}{N}$ achieves the optimal throughput asymptotically. Furthermore, we can deduce from (4.78) that the optimal ξ_k maximizing (4.78) only depends on $R_1, \dots, R_K$. ■

4.10.2 Proof of Theorem 4.16

Proof. By equation (4.50), we have

$$\begin{aligned}
&\|\mathbf{p}(t+1) - \mathbf{p}^*\|_2^2 = \sum_{i\in\mathcal{N}} \left| \left[p_i(t) + \epsilon(t) \left(U'_i(p_i(t)) - C_i(\mathbf{q}_i(\mathbf{p}(t))) \right) \right]^{\mathcal{S}_i} - p_i^* \right|^2 \\
&\le \sum_{i\in\mathcal{N}} \left| p_i(t) + \epsilon(t) \left(U'_i(p_i(t)) - C_i(\mathbf{p}(t)) \right) - p_i^* \right|^2 \\
&\le \|\mathbf{p}(t) - \mathbf{p}^*\|_2^2 + 2\epsilon(t) \sum_i (p_i(t) - p_i^*) \left(U'_i(p_i(t)) - C_i(\mathbf{p}(t)) \right) + \epsilon^2(t) \sum_i \left(U'_i(p_i(t)) - C_i(\mathbf{p}(t)) \right)^2 \\
&\overset{(a)}{\le} \|\mathbf{p}(t) - \mathbf{p}^*\|_2^2 + 2\epsilon(t) \sum_i (p_i(t) - p_i^*) \left(U'_i(p_i(t)) - U'_i(p_i^*) \right) \\
&\quad - 2\epsilon(t) \sum_i (p_i(t) - p_i^*) \left(C_i(\mathbf{p}(t)) - C_i(\mathbf{p}^*) \right) + \epsilon^2(t) \sum_i \left(U'_i(p_i(t)) - C_i(\mathbf{p}(t)) \right)^2,
\end{aligned} \tag{4.80}$$

where we have used $C_i(\mathbf{p}(t))$ to denote $C_i(\mathbf{q}_i(\mathbf{p}(t)))$. In (a), we use the fact that $U_i'(p_i^*) = C_i(\mathbf{p}^*)$ at the nontrivial Nash equilibrium. By mean value theorem, we find

$$\sum_i (p_i(t) - p_i^*)\left(U_i'(p_i(t)) - U_i'(p_i^*)\right) = \sum_i U_i'(\tilde{p}_i)(p_i(t) - p_i^*)^2 \leq -\mu\|\mathbf{p}(t) - \mathbf{p}^*\|_2^2, \quad (4.81)$$

where $\tilde{p}_i = \gamma p_i(t) + (1-\gamma)p_i^*$, $0 \leq \gamma \leq 1$. Define a scalar function $f(\mathbf{p}) = (\mathbf{p}(t) - \mathbf{p}^*)^T C(\mathbf{p})$. By mean value theorem, we have

$$f(\mathbf{p}(t)) - f(\mathbf{p}^*) = (\mathbf{p}(t) - \mathbf{p}^*)^T \mathbf{J}^C(\tilde{\mathbf{p}})(\mathbf{p}(t) - \mathbf{p}^*) \geq \lambda_{\min}(\mathbf{J}^C)\|\mathbf{p}(t) - \mathbf{p}^*\|_2^2. \quad (4.82)$$

We also have

$$\begin{aligned}
&\sum_i \left(U_i'(p_i(t)) - C_i(\mathbf{p}(t))\right)^2 = \sum_i \left(U_i'(p_i(t)) - U_i'(p_i^*) + C_i(\mathbf{p}^*) - C_i(\mathbf{p}(t))\right)^2 \\
\leq& 2\sum_i \left(U_i'(p_i(t)) - U_i'(p_i^*)\right)^2 + 2\sum_i \left(C_i(\mathbf{p}(t)) - C_i(\mathbf{p}^*)\right)^2 \\
\overset{(a)}{\leq}& 2\chi^2\|\mathbf{p}(t) - \mathbf{p}^*\|_2^2 + 2\sum_i (\mathbf{J}_i^C(\tilde{\mathbf{p}}^i)(\mathbf{p}(t) - \mathbf{p}^*))^2 \qquad (4.83) \\
\leq& 2\chi^2\|\mathbf{p}(t) - \mathbf{p}^*\|_2^2 + 2\left(\sum_i \max_j \left|J_{ij}^C(\tilde{\mathbf{p}}^i)\right|^2\right)\|\mathbf{p}(t) - \mathbf{p}^*\|_2^2 \\
\leq& 2(\chi^2 + |\mathcal{N}|M)\|\mathbf{p}(t) - \mathbf{p}^*\|_2^2,
\end{aligned}$$

where (a) comes from mean value theorem. Substituting (4.81)-(4.83) into (4.80), we obtain

$$\|\mathbf{p}(t+1) - \mathbf{p}^*\|_2^2 \leq \left(1 - 2\epsilon(t)\left(\mu + \lambda_{\min}(\mathbf{J}^C) - \epsilon(t)(\chi^2 + |\mathcal{N}|M)\right)\right)\|\mathbf{p}(t) - \mathbf{p}^*\|_2^2. \quad (4.84)$$

Therefore, if $\mu + \lambda_{\min}(\mathbf{J}^C) > 0$ and $\epsilon(t) < \frac{\mu + \lambda_{\min}(\mathbf{J}^C)}{\chi^2 + |\mathcal{N}|M}$, $\mathbf{p}(t)$ converges to $\mathbf{p}^*$ geometrically. ■

4.10.3 Proof of Theorem 4.18

Proof. We show this by induction. The proof basically follows that of Theorem 4.16. For brevity, we omit several immediate steps. Suppose that

$$\|\mathbf{p}(\tau+1)-\mathbf{p}^*\|_2^2 \leq \gamma\|\mathbf{p}(\tau)-\mathbf{p}^*\|_2^2,\ \forall \tau \in \{0,\ldots,t-1\}, \tag{4.85}$$

where $0<\gamma<1$ is a constant. When $\tau = t$, by equation (4.57), we have

$$\begin{aligned}\|\mathbf{p}(t+1)-\mathbf{p}^*\|_2^2 \leq& \sum_{i\in\mathcal{N}} \left|p_i(t)+\epsilon\left(U_i'(p_i(t))-C_i(\mathbf{p}(\tau^i(t)))\right)-p_i^*\right|^2\\ \leq&\|\mathbf{p}(t)-\mathbf{p}^*\|_2^2+2\epsilon\sum_i (p_i(t)-p_i^*)\left(U_i'(p_i(t))-U_i'(p_i^*)\right)\\ &-2\epsilon\sum_i (p_i(t)-p_i^*)\left(C_i(\mathbf{p}(\tau^i(t)))-C_i(\mathbf{p}^*)\right)+\epsilon^2\sum_i\left(U_i'(p_i(t))-C_i(\mathbf{p}(\tau^i(t)))\right)^2.\end{aligned} \tag{4.86}$$

By mean value theorem, we have

$$\begin{aligned}&f(\mathbf{p}(\tau^i(t)))-f(\mathbf{p}^*)=(\mathbf{p}(t)-\mathbf{p}^*)^T\mathbf{J}^C(\tilde{\mathbf{p}})(\mathbf{p}(\tau^i(t))-\mathbf{p}^*)\\ \geq&-\|\mathbf{J}^C(\tilde{\mathbf{p}})\|_1\|\mathbf{p}(t)-\mathbf{p}^*\|_2\|\mathbf{p}(\tau^i(t))-\mathbf{p}^*\|_2 \geq -M_1\|\mathbf{p}(t)-\mathbf{p}^*\|_2\|\mathbf{p}(\tau^i(t))-\mathbf{p}^*\|_2.\end{aligned} \tag{4.87}$$

Note that

$$\begin{aligned}\|\mathbf{p}(\tau^i(t))-\mathbf{p}^*\|_2^2 =& \sum_{j\in\mathcal{N}}|p_j(\tau_j^i(t))-p_j^*|^2 \leq \sum_{j\in\mathcal{N}}\|\mathbf{p}(\tau_j^i(t))-\mathbf{p}^*\|^2\\ \leq&\sum_{j\in\mathcal{N}}\gamma^{\tau_j^i(t)-t}\|\mathbf{p}(t)-\mathbf{p}^*\|^2 \leq \frac{|\mathcal{N}|}{\gamma^B}\|\mathbf{p}(t)-\mathbf{p}^*\|^2.\end{aligned} \tag{4.88}$$

Similar to (4.84), we obtain

$$\|\mathbf{p}(t+1)-\mathbf{p}^*\|_2^2 \leq \left(1-2\epsilon\left(\mu-M_1\sqrt{\frac{|\mathcal{N}|}{\gamma^B}}+\epsilon\left(\chi^2+\frac{M_2|\mathcal{N}|^2}{\gamma^B}\right)\right)\right)\|\mathbf{p}(t)-\mathbf{p}^*\|_2^2. \tag{4.89}$$

Therefore, if there exists $\epsilon > 0$ and $0 < \gamma < 1$ such that (4.58) holds, the induction hypothesis is true for $\tau = t$. ■

4.10.4 Proof of Theorem 4.19

Proof. By following (4.80), we obtain

$$\begin{aligned}\|\mathbf{p}(t+1) - \mathbf{p}^*\|_2^2 \le & \|\mathbf{p}(t) - \mathbf{p}^*\|_2^2 + \epsilon^2(t) \sum_i \left(U_i'(p_i(t)) - C_i(\mathbf{p}(t)) - \bar{w}_i(t)\right)^2 \\ & + 2\epsilon(t) \sum_i (p_i(t) - p_i^*) \left(U_i'(p_i(t)) - C_i(\mathbf{p}(t) - \bar{w}_i(t))\right) \\ \le & \gamma \|\mathbf{p}(t) - \mathbf{p}^*\|_2^2 + 4\epsilon\omega \sum_i |\bar{w}_i(t) - \bar{w}_i| + 2\epsilon^2 \sum_i (\bar{w}_i(t) - \bar{w}_i)^2,\end{aligned} \tag{4.90}$$

where

$$\gamma = 1 - 2\epsilon \left(\mu + \lambda_{\min}(\mathbf{J}^C) - \epsilon(\chi^2 + 4M)\right), \tag{4.91}$$

and

$$\epsilon < \frac{\mu + \lambda_{\min}(\mathbf{J}^C)}{\chi^2 + 4|\mathcal{N}|M}. \tag{4.92}$$

By assumption $\lim_{t\to\infty} \bar{w}_i(t) = \bar{w}_i$, for any $\delta > 0$, there exists a t_0 such that if $t \ge t_0$ $|\bar{w}_i(t) - \bar{w}_i| < \delta$, $\forall i$. Applying (4.90) recursively, we obtain

$$\begin{aligned}\|\mathbf{p}(t+1) - \mathbf{p}^*\|_2^2 \le & \gamma^{t-t_0} \|\mathbf{p}(t_0) - \mathbf{p}^*\|_2^2 + 4\epsilon\omega|\mathcal{N}|\delta \sum_{\tau=0}^{t-t_0} \gamma^\tau + 2\epsilon|\mathcal{N}|\delta^2 \sum_{\tau=0}^{t-t_0} \gamma^{2\tau} \\ \le & \gamma^{t-t_0} \|\mathbf{p}(t_0) - \mathbf{p}^*\|_2^2 + \frac{4\epsilon\omega|\mathcal{N}|\delta}{1-\gamma} + \frac{2\epsilon|\mathcal{N}|\delta^2}{1-\gamma^2}.\end{aligned} \tag{4.93}$$

By taking $\delta \to 0$ and $t \to \infty$, we obtain

$$\limsup_{t\to\infty} \|\mathbf{p}(t) - \mathbf{p}^*\|_2^2 = 0. \tag{4.94}$$

Therefore, $\mathbf{p}(t)$ converges to $\mathbf{p}^*$.

■

4.10.5 Proof of Theorem 4.20

Proof. By following (4.80), we obtain

$$\begin{aligned}
&E\left\{\|\mathbf{p}(t+1)-\mathbf{p}^*\|_2^2|\mathcal{F}_t\right\}\\
\leq&\|\mathbf{p}(t)-\mathbf{p}^*\|_2^2+\epsilon^2(t)\sum_i\left(U_i'(p_i(t))-C_i(\mathbf{p}(t))-\tilde{w}_i(t)\right)^2\\
&+2\epsilon(t)\sum_i(p_i(t)-p_i^*)\left(U_i'(p_i(t))-C_i(\mathbf{p}(t)-\tilde{w}_i(t))\right)\\
\leq&\|\mathbf{p}(t)-\mathbf{p}^*\|_2^2-\epsilon(t)\kappa\|\mathbf{p}(t)-\mathbf{p}^*\|_2^2+2\epsilon^2(t)E\left\{\sum_i\tilde{w}_i^2(t)|\mathcal{F}_t\right\}-2\epsilon(t)E\left\{\sum_i(p_i(t)-p_i^*)\tilde{w}_i(t)|\mathcal{F}_t\right\}\\
\leq&\|\mathbf{p}(t)-\mathbf{p}^*\|_2^2-\epsilon(t)\kappa\|\mathbf{p}(t)-\mathbf{p}^*\|_2^2+2\epsilon^2(t)|\mathcal{N}|B,
\end{aligned}\tag{4.95}$$

where

$$2\left(\mu+\lambda_{\min}(\mathbf{J}^C)-\epsilon(t)(\chi^2+4|\mathcal{N}|M)\right)>\kappa>0.\tag{4.96}$$

From (4.60), $\exists t_0,\kappa$ such that for all $t\geq t_0$, (4.96) holds. Taking expectation both sides of (4.95) over $\mathcal{F}_t$ and applying the resulting equation recursively,

$$\begin{aligned}
E\left\{\|\mathbf{p}(t+1)-\mathbf{p}^*\|_2^2\right\}\leq&E\left\{\|\mathbf{p}(t_0)-\mathbf{p}^*\|_2^2\right\}-\kappa\sum_{t=t_0}^{t}\epsilon(t)E\left\{\|\mathbf{p}(t)-\mathbf{p}^*\|_2^2\right\}\\
&+2|\mathcal{N}|B\sum_{t=t_0}^{t}\epsilon^2(t),
\end{aligned}\tag{4.97}$$

from which we get

$$\sum_{t=t_0}^{\infty}\epsilon(t)E\left\{\|\mathbf{p}(t)-\mathbf{p}^*\|_2^2\right\}<\infty.\tag{4.98}$$

Since $\sum_{t=0}^{\infty}\epsilon(t)=\infty$ and $E\left\{\|\mathbf{p}(t)-\mathbf{p}^*\|_2^2\right\}\geq 0$, $\mathbf{p}(t)$ converges to $\mathbf{p}^*$ with probability 1.

■

4.10.6 Proof of Theorem 4.21

Proof. By following (4.90), we obtain

$$\begin{aligned}
&\|\mathbf{p}(t+1)-\mathbf{p}^*\|_2^2\\
\leq&\|\mathbf{p}(t)-\mathbf{p}^*\|_2^2+\epsilon^2(t)\sum_i\left(U_i'(p_i(t))-C_i(\mathbf{p}(t))-\tilde{w}_i(t)\right)^2\\
&+2\epsilon(t)\sum_i(p_i(t)-p_i^*)\left(U_i'(p_i(t))-C_i(\mathbf{p}(t)-\tilde{w}_i(t))\right)\\
\leq&\gamma\|\mathbf{p}(t)-\mathbf{p}^*\|_2^2-2\epsilon\sum_i(p_i(t)-p_i^*)\tilde{w}_i(t)+2\epsilon^2\sum_i\tilde{w}_i^2(t),
\end{aligned}\tag{4.99}$$

where γ is defined in (4.91). Applying (4.99) recursively, we obtain

$$\begin{aligned}
\|\mathbf{p}(t+1)-\mathbf{p}^*\|_2^2\leq&\gamma^t\|\mathbf{p}(0)-\mathbf{p}^*\|_2^2+2\epsilon^2\sum_{\tau=0}^{t}\gamma^{t-\tau}\sum_i\tilde{w}_i^2(\tau)\\
&-2\epsilon\sum_{\tau=0}^{t}\gamma^{t-\tau}\sum_i(p_i(\tau)-p_i^*)\tilde{w}_i(\tau).
\end{aligned}\tag{4.100}$$

As $E\{w_i(t)|\mathcal{F}_t\}=0$ and $E\left\{\sum_i\tilde{w}_i^2(\tau)|\mathcal{F}_t\right\}\leq B$, by using [10, Lemma 2], there exists a constant $D(B,\epsilon)>0$ such that

$$\liminf_{t\to\infty}2\epsilon^2\sum_{\tau=0}^{t}\gamma^{t-\tau}\sum_i\tilde{w}_i^2(\tau)-2\epsilon\sum_{\tau=0}^{t}\gamma^{t-\tau}\sum_i(p_i(\tau)-p_i^*)\tilde{w}_i(\tau)\leq D(B,\epsilon).\tag{4.101}$$

Therefore, we get (4.61). ■

Chapter 5

Secure Network Communications

In this chapter, we consider secure communications over networks with erasures and networks with unequal link capacities in the presence of a wiretapper that can wiretap any subset of k links.

5.1 Introduction

Information-theoretic security is a principle that can strengthen the security of wireless networks at the physical layer by using coding to guarantee that the messages sent cannot be decoded by a malicious eavesdropper. The theoretical basis for this information-theoretic approach was first studied in the seminal paper by Wyner [80] using Shannon's notion of perfect secrecy [71], where a coset coding scheme based on a linear maximum distance separable code is proposed to provide security for a wiretap channel. Recently, information-theoretic security is studied in more complicated networks, see e.g., [13, 27, 60]. The secure network coding problem was introduced in [13] for the case of multicast in wireline networks where each link has equal capacity. In the presence of a wiretapper that can look at most k links in the network, constructions of information-theoretically secure linear network codes are proposed in e.g. [13, 27]. For this network model, trade-offs between security, code alphabet size, and multicast rate of secure linear network codes are considered in [27]. In [60], secure communication is considered for wireless erasure networks. The given results extend the capacity result of wireless erasure networks in [25], where an outer bound on the secrecy unicast rate is derived when the wiretapper can wiretap at most k links.

In this chapter, we consider secure communication over more general classes of wireline networks: networks with unequal link capacities and networks with erasures. The former generalizes the equal link capacity secure network coding problem formulation of [13], while the latter generalizes the wiretap channel problem formulation of [80] from a single channel to a network of erasure links.

In the case of throughput optimization without security requirements, the assumption that all links have unit capacity is made without loss of generality, since links of larger capacity can be modeled as multiple unit capacity links in parallel. However, in the secure communication problem, such an assumption cannot be made without loss of generality. Indeed, we show that there are significant differences between the equal capacity and unequal capacity cases. For the case of equal (unit) link capacities, the secrecy capacity is given by the cut set bound, whether or not the location of the wiretapped links is known. This capacity can be achieved by injecting k random keys at the source which are decoded at the sink along with the message [13]. We refer to this approach as the global key strategy. In contrast, we show that for unequal link capacities, the secrecy capacity is not the same in general when the location of the wiretapped links is known or unknown. We give new achievable strategies that can outperform the global key strategy. In these new strategies, random keys are canceled at intermediate non-sink nodes, or injected at intermediate non-source nodes. Finally, we show that determining the secrecy capacity is an NP-complete problem.

5.2 Network Model and Problem Formulation

In this chapter we focus on acyclic graphs for simplicity; we expect that our results can be generalized to cyclic networks using the approach in [40, 50] of working over fields of rational functions in an indeterminate delay variable. We model a wireline network by a directed acyclic graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the vertex set and $\mathcal{E}$ is the directed edge set.

For each node $i \in \mathcal{V}$, $\mathcal{N}_{\mathcal{O}}(i)$ and $\mathcal{N}_{\mathcal{I}}(i)$ denote the set of in-neighbors and out-neighbors of i, i.e.,

$$\mathcal{N}_{\mathcal{I}}(i) = \{j | (j,i) \in \mathcal{E}\}, \quad \mathcal{N}_{\mathcal{O}}(i) = \{j | (i,j) \in \mathcal{E}\}. \tag{5.1}$$

A cut for $x, y \in \mathcal{V}$ is a partition of $\mathcal{V}$ into two sets $\mathcal{V}_x$ and $\mathcal{V}_y = \mathcal{V}_x^c$ such that $x \in \mathcal{V}_x$ and $y \in \mathcal{V}_y$. For the $x - y$ cut given by $\mathcal{V}_x$, the cut-set $[\mathcal{V}_x, \mathcal{V}_y]$ is the set of edges going from $\mathcal{V}_x$ to $\mathcal{V}_y$, i.e.,

$$[\mathcal{V}_x, \mathcal{V}_y] = \{(u,v) | (u,v) \in \mathcal{E},\, u \in \mathcal{V}_x,\, v \in \mathcal{V}_y\}. \tag{5.2}$$

In the most general network model that we consider, each edge $(i,j) \in \mathcal{E}$ represents a memoryless erasure channel from node i to node j with erasure probability $p_{i,j}$. As in [13], there is an eavesdropper, who can wiretap any k edges of this network. For any wiretapped edge $(i,j) \in \mathcal{E}$, the wiretapper can receive the symbols sent by node i to node j via another memoryless erasure channel with erasure probability $q_{i,j}$. Note that our model includes those in [13, 60] as special cases. When $p_{i,j} = q_{i,j} = 0$ for all links (i,j), our model reduces to that in [13]. When $p_{i,j} = q_{i,j}$ takes different values for different links (i,j), with appropriate capacity scaling, the network is equivalent to an network with unequal capacity links where the wiretapper fully observes transmissions on the links it wiretaps.

We consider multicast problems, where a set of sinks $\mathcal{D} = \{d_1, \ldots, d_{|\mathcal{D}|}\} \subset \mathcal{V}$ demands all of the information from a source $s \in \mathcal{V}$. This includes the unicast problem with a single sink d as a special case.

The secrecy requirement is that the message communicated from the source to the sinks must have zero mutual information with the wiretapper's observations. The secrecy capacity is the highest possible communication rate such that the wiretapper gets no information about the message being multicast.

In the following, we consider four related scenarios.

1. First, we consider a wireline erasure network with $p_{i,j} \neq q_{i,j}$ in general and the

location of the wiretapped links is known. We treat this as an optimization problem from the point of view of the wiretapper, who chooses which links to wiretap so as to minimize the achievable secrecy rate of the source.

2. The network interdiction problem [79] is to minimize the maximum flow of the network when k links are removed from the entire network. This is equivalent to the first scenario when $p_{i,j} = q_{i,j}$.

3. Our third scenario is a wireline network with equal link capacities, where the wiretapper can wiretap an unknown subset of k links from a known collection of vulnerable network links. We consider an optimization problem from the point of view of the communicating users who seek to maximize their communication rates subject to the requirement that the message is secret regardless of the choice of wiretapped links.

4. Our final scenario is a wireline network with unequal link capacities, where the wiretapper can wiretap an unknown subset of k links from the entire network. Scenario 3 can be considered as a bridge for studying scenario 4. In the following, scenario 3 is usually discussed first. We then convert a network considered in scenario 3 to a corresponding network for scenario 4 such that the same result holds.

5.3 Secrecy Capacity Region When the Location of the Wiretapped Links is Known

We first consider scenario 1, where the location of the wiretapped links is known. The wiretapper chooses the set of wiretapped links such that the resulting secrecy capacity is minimized. Our main result of this section is the following cut-set expression of the secrecy capacity region in Theorem 5.1.

Before stating the theorem, we briefly review the results in [80], where a wiretap channel with one source, one sink and one wiretapper is considered. Let X be the secret message sent by the source, and let Y and Z be the received signal at the sink and wiretapper, respectively. By using a coset coding scheme based on a linear maximum distance separable code, Wyner showed that the secrecy capacity of the wiretap channel is

$$C_s = \max_{p_X(x)} I(Y;X) - I(Z;X), \tag{5.3}$$

with $H(X|Z) = H(X)$, where $p_X(x)$ is the pdf of X.

Theorem 5.1 *Consider a single source and single sink wireline erasure network in which a secret message M is delivered from source s to destination d. There exists a wiretapper in the network that can wiretap at most k links and the wiretapped messages are denoted as $\mathbf{Z}$. Assuming that the destination has complete knowledge of the erasure locations on each link of the network and the locations of the wiretapped links, the secrecy capacity is given by*

$$C_s = \min_{\{\mathcal{V}_s:\mathcal{V}_s \textit{ is an } s-d \textit{ cut}\}} \min_{\{\mathcal{A}|\mathcal{A}\subseteq[\mathcal{V}_s,\mathcal{V}_s^c],\,|\mathcal{A}|\le k\}} \sum_{(i,j)\in[\mathcal{V}_s,\mathcal{V}_s^c]-\mathcal{A}} (1-p_{i,j}) + \sum_{(i,j)\in\mathcal{A}} \max\left(q_{i,j}-p_{i,j},0\right), \tag{5.4}$$

where

$$H(M|\mathbf{Z}) = H(M). \tag{5.5}$$

Proof. Achievability: We show the achievability of (5.4) by applying the coding scheme of [80] on each link individually. Let $X_{i,j}$, $Y_{i,j}$ and $Z_{i,j}$ be the local message, channel output, and wiretapper's output on link $(i,j)\in\mathcal{A}$. From (5.3), we know that as long as the rate of $X_{i,j}$ is less than

$$\max_{P_{i,j}(x_{i,j})} I(X_{i,j};Y_{i,j}) - I(X_{i,j};Z_{i,j}) = \max_{\pi}(q_{i,j}-p_{i,j})H(\pi) = \max\left(q_{i,j}-p_{i,j},0\right), \tag{5.6}$$

node j can receive $X_{i,j}$ securely, i.e., $I(X_{i,j};Z_{i,j}) = 0$. As $M \to \mathbf{X} \to \mathbf{Z}$ forms a Markov chain, we have

$$\begin{aligned} I(M;\mathbf{Z}) \le& I(\mathbf{X};\mathbf{Z}) = H(\mathbf{Z}) - H(\mathbf{Z}|\mathbf{X}) \\ &\le \sum_{(i,j)\in\mathcal{A}} H(Z_{i,j}) - \sum_{(i,j)\in\mathcal{A}} H(Z_{i,j}|X_{i,j}) = \sum_{(i,j)\in\mathcal{A}} I(X_{i,j};Z_{i,j}) = 0, \end{aligned} \tag{5.7}$$

where the second inequality follows since conditioning reduces entropy [22] and that $Z_{i,j}$ is conditionally independent of the local messages and wiretapped observations at other nodes given $X_{i,j}$. As mutual information is nonnegative, we have $I(M;\mathbf{Z}) = 0$ and perfect secrecy is achieved. Therefore, given the wiretapping set $\mathcal{A}$, we can decouple the secrecy coding from the routing or network coding, i.e., routing or network coding is oblivious to the secrecy coding. We simply replace the capacity of each link with the secrecy capacity of each link. Therefore, the following cut-set bound is achievable

$$\min_{\{\mathcal{V}_s:\,\mathcal{V}_s \text{ is an } s-d \text{ cut}\}} \sum_{(i,j)\in[\mathcal{V}_s,\mathcal{V}_s^c]-\mathcal{A}} (1-p_{i,j}) + \sum_{(i,j)\in\mathcal{A}} \max\left(q_{i,j}-p_{i,j},0\right). \tag{5.8}$$

The wiretapper chooses the set $\mathcal{A}$ to minimize the secrecy rate in (5.8), which gives (5.4). This concludes the achievability part.

Converse: Let $\mathcal{V}_s$ be a cut of the network and $\mathcal{A} \subseteq [\mathcal{V}_s,\mathcal{V}_s^c]$, $|\mathcal{A}| \leq k$ be the set of wiretapping edges. Denote by $\mathbf{X}$ the transmitted signals from nodes in $\mathcal{V}_s$ over links in $[\mathcal{V}_s,\mathcal{V}_s^c]$ and denote by $\mathbf{Z}$ and $\mathbf{Y}$ the observed signals from links in $\mathcal{A}$ and in $[\mathcal{V}_s,\mathcal{V}_s^c]$, respectively. Let $\mathcal{A}_h$ be the set of links (i,j) such that $p_{i,j} \geq q_{i,j}$, and let $\mathbf{Y}_h$ and $\mathbf{Y}_d$ contain the observations from links in $\mathcal{A}_h$ and $[\mathcal{V}_s,\mathcal{V}_s^c]-\mathcal{A}_h$, respectively. $\mathbf{Z}_h$ and $\mathbf{Z}_d$ are defined similarly, where $\mathbf{Z}_d$ is a degraded version of $\mathbf{Y}_d$ while $\mathbf{Y}_h$ is a degraded version of $\mathbf{Z}_h$. We consider block coding with block length n. We have

$$
\begin{aligned}
nR_s \le & H(M|\mathbf{Z}^n) \\
& \overset{(a)}{\le} H(M|\mathbf{Z}^n) - H(M|\mathbf{Y}^n) + n\epsilon_n \\
& \overset{(b)}{=} H(M|\mathbf{Z}_d^n, \mathbf{Z}_h^n) - H(M|\mathbf{Y}_d^n, \mathbf{Y}_h^n) + n\epsilon_n \\
& \overset{(c)}{\le} H(M|\mathbf{Z}_d^n, \mathbf{Y}_h^n) - H(M|\mathbf{Y}_d^n, \mathbf{Y}_h^n) + n\epsilon_n \\
& \overset{(d)}{\le} H(M|\mathbf{Z}_d^n, \mathbf{Y}_h^n) - H(M|\mathbf{Z}_d^n, \mathbf{Y}_d^n, \mathbf{Y}_h^n) + n\epsilon_n \\
& = I(M; \mathbf{Y}_d^n|\mathbf{Z}_d^n, \mathbf{Y}_h^n) + n\epsilon_n \\
& \overset{(e)}{\le} I(\mathbf{X}^n; \mathbf{Y}_d^n|\mathbf{Z}_d^n, \mathbf{Y}_h^n) + n\epsilon_n, \\
& = \sum_{i=1}^{n} H(\mathbf{Y}_{d,i}|\mathbf{Z}_d^n, \mathbf{Y}_h^n) - \sum_{i=1}^{n} H(\mathbf{Y}_{d,i}|\mathbf{X}^n, \mathbf{Z}_d^n, \mathbf{Y}_h^n) + n\epsilon_n, \\
& \overset{(f)}{\le} \sum_{i=1}^{n} H(\mathbf{Y}_{d,i}|\mathbf{Z}_{d,i}, \mathbf{Y}_{h,i}) - \sum_{i=1}^{n} H(\mathbf{Y}_{d,i}|\mathbf{X}_i, \mathbf{Z}_{d,i}, \mathbf{Y}_{h,i}) + n\epsilon_n, \\
& = nI(\mathbf{X}; \mathbf{Y}_d|\mathbf{Z}_d, \mathbf{Y}_h) + n\epsilon_n, \\
& = n\left(I(\mathbf{X}; \mathbf{Y}_d, \mathbf{Y}_h) - I(\mathbf{X}; \mathbf{Z}_d, \mathbf{Y}_h)\right) + n\epsilon_n, \\
& \le n \max_{p(\mathbf{X})} \left(I(\mathbf{X}; \mathbf{Y}_d, \mathbf{Y}_h) - I(\mathbf{X}; \mathbf{Z}_d, \mathbf{Y}_h)\right) + n\epsilon_n, \\
& \overset{(g)}{=} n\left(\sum_{(i,j)\in[\mathcal{V}_s, \mathcal{V}_s^c]} (1-p_{i,j}) - \sum_{(i,j)\in\mathcal{A}_d} (1-q_{i,j}) - \sum_{(i,j)\in\mathcal{A}_h} (1-p_{i,j})\right) + n\epsilon_n, \\
& = n\left(\sum_{(i,j)\in[\mathcal{V}_s, \mathcal{V}_s^c]-\mathcal{A}} (1-p_{i,j}) + \sum_{(i,j)\in\mathcal{A}} \max(q_{i,j}-p_{i,j}, 0)\right) + n\epsilon_n,
\end{aligned}
\tag{5.9}
$$

where $\epsilon_n \to 0$ as $n \to +\infty$ and

(a) comes from Fano's inequality.

(b) follows from the definition of $\mathbf{Y}_d, \mathbf{Y}_h, \mathbf{Z}_d, \mathbf{Z}_h$.

(c) comes from the fact that $M \to \mathbf{X}^n \to (\mathbf{Z}_d^n, \mathbf{Z}_h^n) \to (\mathbf{Z}_d^n, \mathbf{Y}_h^n)$ forms a Markov chain.

(d) follows from conditioning reduces entropy.

(e) comes from the fact that $M \to \mathbf{X}^n \to (\mathbf{Y}_d^n, \mathbf{Y}_h^n) \to (\mathbf{Z}_d^n, \mathbf{Y}_h^n)$ forms a Markov chain and $A \to B \to C \to D \Rightarrow I(A; C|D) \le I(B; C|D)$. To show this inequality, we have

$$\begin{aligned} I(A;C|D)-I(B;C|D) =& I(A;C,D)-I(A;D)-I(B;C,D)+I(B;D) \\ =& I(B;D|A)-I(B;C|A)=-I(B;C|A,D)\leq 0. \end{aligned}$$

(f) follows from the fact that conditioning reduces entropy and that $\mathbf{Y}_{d,i}$ is independent of other variables given $\mathbf{X}_i, \mathbf{Z}_{d,i}, \mathbf{Y}_{h,i}$.

(g) is because both $I(\mathbf{X};\mathbf{Y}_d,\mathbf{Y}_h)$ and $I(\mathbf{X};\mathbf{Z}_d,\mathbf{Y}_h)$ are maximized when the entries of $\mathbf{X}$ are i.i.d. Bernoulli(1/2).

■

By decoupling the secrecy coding from the routing or network coding as in the achievability proof of Theorem 5.1, Theorem 5.1 can be readily extended to the multicast case. The proof is similar to the unicast case. We thus give the following theorem without proof.

Theorem 5.2 *Consider a multicast problem in a wireline erasure network $\mathcal{G}=(\mathcal{V},\mathcal{E})$ with a single source $s\in\mathcal{V}$ and a set of destinations $\mathcal{D}\subseteq\mathcal{V}$. A secret message M is multicast from s to all nodes in $\mathcal{D}$. There exists a wiretapper in the network that can wiretap at most k links, and the wiretapped messages are denoted as $\mathbf{Z}$. Assuming that the destination has complete knowledge of the erasure locations on each link of the network and the locations of the wiretapped links, the secrecy multicast capacity of the network is given by*

$$C_s=\min_{d\in\mathcal{D}}\ \min_{\{\mathcal{V}_s:\,\mathcal{V}_s \text{ is an } s-d \text{ cut}\}}\ \min_{\{\mathcal{A}|\mathcal{A}\subseteq[\mathcal{V}_s,\mathcal{V}_s^c],\,|\mathcal{A}|\leq k\}} \sum_{(i,j)\in[\mathcal{V}_s,\mathcal{V}_s^c]-\mathcal{A}}(1-p_{i,j})+\sum_{(i,j)\in\mathcal{A}}\max\left(q_{i,j}-p_{i,j},0\right), \tag{5.10}$$

where

$$H(M|\mathbf{Z})=H(M). \tag{5.11}$$

5.4 Achievable Strategies when the Location of the Wiretapped Links is Unknown

Next, we consider scenarios 3 and 4, where the source does not know the location of the wiretapped links. In the case of unit link capacities, the secrecy capacity can be achieved using global keys generated at the source and decoded at the sink [13]. The source transmits r secret information symbols and k random key symbols, where $r+k$ is equal to the min-cut of the network. This scheme does not achieve capacity in general erasure networks when link capacities are unequal. Intuitively, this is because the total rate of random keys is limited by the min cut from the source to the sink, and cannot fully utilize capacity on large capacity cuts with large links.

Capacity can be improved by using a combination of local and global random keys. A local key is injected at a non-source node and/or canceled at a non-sink node. However, it is complicated to optimize over all possible combinations of nodes at which keys are injected and canceled. Thus, we propose the following more tractable family of constructions. In this section, we focus on the case of a single source and a single sink, and we assume that $q_{i,j} = p_{i,j}$ in this section. Let $z_{i,j}$ be the actual flow on link (i, j). Let $\mathcal{W}$ be the set of all possible maximal subsets of links that the wiretapper may access simultaneously. For scenario 4, $\mathcal{W} = \{\mathcal{A} | \mathcal{A} \subseteq \mathcal{E}, \, |\mathcal{A}| = k\}$.

Strategy 1: Random Keys Injected by Source and Possibly Canceled at Intermediate Nodes

Connect each subset of links $\mathcal{A} \in \mathcal{W}$ in the network $\mathcal{G}$ to a virtual node $t^{\mathcal{A}}$, and connect both $t^{\mathcal{A}}$ and the actual sink to a virtual sink $d^{\mathcal{A}}$. Let $R_{s\to\mathcal{A}}$ be the total flow between s and $t^{\mathcal{A}}$. The virtual link between $t^{\mathcal{A}}$ and $d^{\mathcal{A}}$ has capacity $R_{s\to\mathcal{A}}$, and the virtual link between the actual sink and $d^{\mathcal{A}}$ has capacity R_s. This is illustrated in Fig. 5.1. The source multicasts a secret message $\mathbf{v} = [v_1, \ldots, v_{R_s}]^T$ with R_s symbols plus R_w random key symbols $\mathbf{w} = [w_1, \ldots, w_{R_w}]$. We want to choose the secrecy rate R_s and the random key rate R_w such that the virtual receiver $d^{\mathcal{A}}$ can decode $R_s + R_{s\to\mathcal{A}}$ message and key symbols from the source, and the original receiver can decode the R_s message symbols.

If the rate $R_s + R_{s\to\mathcal{A}}$ satisfies the min-cut between the source and the virtual receiver $d^{\mathcal{A}}$ and $R_{s\to\mathcal{A}} \leq R_w$, by using [82, Corollary 19.21], there exists a network code such that $d^{\mathcal{A}}$ receives $R_s + R_{s\to\mathcal{A}}$ linearly independent combinations of $\mathbf{v}$ and $\mathbf{w}$ when the finite field size is sufficiently large. Let the signals received at a particular virtual sink $d^{\mathcal{B}}$ be denoted as $\mathbf{M}_{\mathcal{B}}[\mathbf{v}^T, \mathbf{w}^T]^T$, where $\mathbf{M}_{\mathcal{B}}$ is an $R_s + R_{s\to\mathcal{A}}$ by $R_s + R_w$ received coding matrix with full row rank[1]. We can add $R_w - R_{s\to\mathcal{B}}$ rows to $\mathbf{M}_{\mathcal{B}}$ to get a full rank $(R_s + R_w) \times (R_s + R_w)$ square matrix $\tilde{\mathbf{M}}_{\mathcal{B}}$. We thus precode the secret message and keys using $\tilde{\mathbf{M}}_{\mathcal{B}}^{-1}$, i.e., the source transmits $\tilde{\mathbf{M}}_{\mathcal{B}}^{-1}[\mathbf{v}^T, \mathbf{w}^T]^T$. This results in the actual sink receiving the message $\mathbf{v}$, which is transmitted to each virtual sink by the corresponding virtual link.

For any virtual sink $d^{\mathcal{A}}$, the received coding matrix after precoding is $\mathbf{M}_{\mathcal{A}}\tilde{\mathbf{M}}_{\mathcal{B}}^{-1}$, which is a full row rank matrix. As $\mathbf{M}_{\mathcal{A}}\tilde{\mathbf{M}}_{\mathcal{B}}^{-1}$ is a full row rank matrix, the coding vectors of received signals from the set $\mathcal{A}$ of wiretapping links span a rank $R_{s\to\mathcal{A}}$ subspace that is linearly independent of the coding vectors of message $\mathbf{v}$ which is received from the actual sink d. Therefore, perfect secrecy rate R_s can be achieved provided that the finite field size $q > \binom{|\mathcal{E}|}{k}$. Note that by applying $\tilde{\mathbf{M}}_{\mathcal{B}}^{-1}$, the random keys injected by the source are either implicitly canceled at intermediate nodes or decoded by the sink.

Since computing $R_{s\to\mathcal{A}}$ involves a linear optimization in $z_{i,j}$, to simplify the computation, we replace $R_{s\to\mathcal{A}}$ with an upper bound $\sum_{(i,j)\in\mathcal{A}} z_{i,j}$, which gives a lower bound on the achievable rate using Strategy 1. We can write an LP for this key cancelation strategy as follows:

$$\begin{aligned}
&\max\ R_s \\
&\text{subject to} \sum_{(i,j)\in\mathcal{E}} f_{i,j}^{\mathcal{A}} - \sum_{(i,j)\in\mathcal{E}} f_{j,i}^{\mathcal{A}} = \begin{cases} R_s + \sum_{(i,j)\in\mathcal{A}} z_{i,j}, & \text{if } i = s, \\ -R_s - \sum_{(i,j)\in\mathcal{A}} z_{i,j}, & \text{if } i = d^{\mathcal{A}}, \\ 0, & \text{otherwise,} \end{cases} \qquad (5.12) \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \forall \mathcal{A} \in \mathcal{W}, \\
&\qquad f_{i,j}^{\mathcal{A}} \leq z_{i,j} \leq 1 - p_{i,j},\ \forall (i,j) \in \mathcal{E},
\end{aligned}$$

[1] We assume that R_s and $R_{s\to\mathcal{B}}$ are integers, which can be approximated arbitrarily closely by scaling the capacity of all links by the same factor.

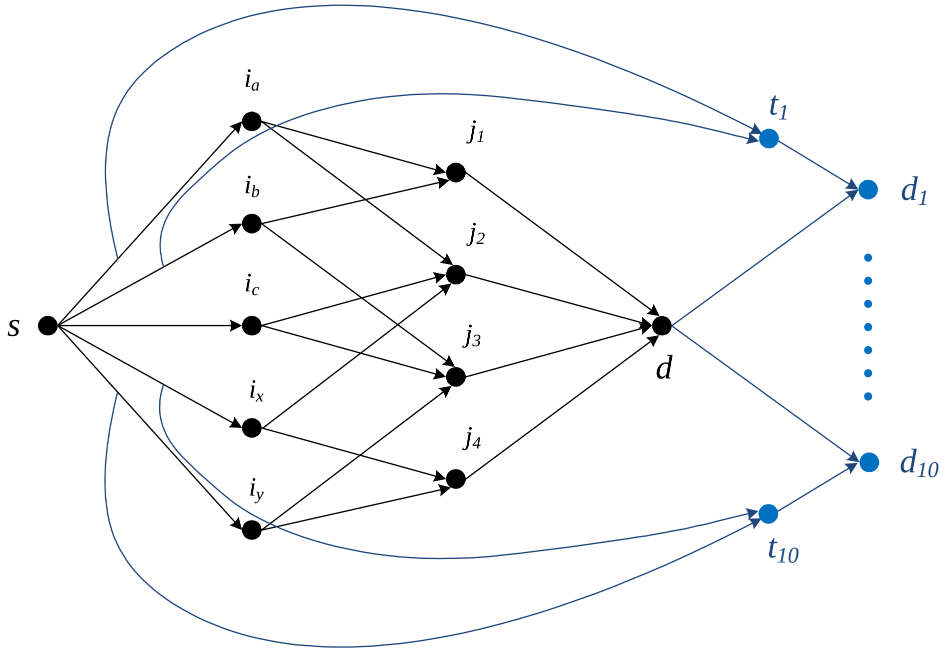


Figure 5.1: Illustration of strategy 1, an achievable construction where random keys are injected by the source and possibly canceled at intermediate nodes. In this figure, $k = 2$ and only the 5 links in the first layer can be wiretapped.

where $f_{i,j}^{\mathcal{A}}$ is the virtual flow on link (i, j) for the virtual sink corresponding to wiretapping set $\mathcal{A}$ and $z_{i,j}$ is the actual flow on link (i, j). Note that when all links can be wiretapped the number of variables in (5.12) is exponential in k. The optimal value of (5.12) gives an achievable rate for scenarios 3 and 4 where the source does not know the location of the wiretapped links.

An illustration of the strategy 1 construction is given in Fig. 5.1 where the number of wiretapped links is $k = 2$, and only the first layer of the three layer network is allowed to be wiretapped. Each link in the network has unit capacity. Let c denote the minimum cut after deleting any k links in the first layer of the graph. As the wiretapped links are connected to the source directly, the min-cut between each virtual sink and the source is at least $c+k$. Since c is the cut-set upper bound on the secrecy rate, by using the key cancelation scheme the secrecy rate c is achievable, which is equal to the secrecy rate when the location of wiretap links is known. For the example in Fig. 5.1, the secrecy rate $c = 3$ is achievable. When key cancelation is not applied, let r and w be the secrecy rate and the random key rate at the source, respectively.

Let x be the total actual flow on the first layer. To achieve secrecy, we must have $w \geq \frac{2}{5}x$, where the min-cut condition on the first layer requires $r + w \leq x$. Since the sink needs to decode both message and random key symbols from the source, the min-cut condition on the last layer requires $r + w \leq 4$. Combining these we obtain $r \leq \frac{12}{5}$, which is strictly less than 3.

Strategy 2: Random Keys Injected by Source and/or Intermediate Nodes and Decoded at Sink

Connect each subset of links $\mathcal{A} \in \mathcal{W}$ in the network $\mathcal{G}$ to a virtual receiver $d^{\mathcal{A}}$. If the number of linearly independent keys received at $d^{\mathcal{A}}$ is greater than or equal to the total amount of data received through the corresponding wiretap links, perfect secrecy can be achieved. Let $R_{w,v}$ be the secret key injection rate at node v and R_s be the secrecy rate at the source. We want to maximize R_s subject to the condition that the sink can decode the random keys injected at all nodes plus the message, and each wiretap set gets total key rate greater than or equal to its received flow. We then have a linear program:

$$\begin{aligned}
&\max \;\; R_s \\
&\text{subject to} \; \sum_j f^{\mathcal{A}}_{i,j} - \sum_j f^{\mathcal{A}}_{j,i} = \begin{cases} -\sum_{v\in\mathcal{V}} \kappa^{\mathcal{A}}_v, & \text{if } i = d^{\mathcal{A}}, \\ \kappa^{\mathcal{A}}_i, & \text{otherwise,} \end{cases} \forall \mathcal{A} \in \mathcal{W}, \\
&\sum_j g_{i,j} - \sum_j g_{j,i} = \begin{cases} R_s + R_{w,s}, & \text{if } i = s, \\ -\left(R_s + \sum_{v\in\mathcal{V}, v\neq d} R_{w,v}\right), & \text{if } i = d, \\ R_{w,i}, & \text{otherwise,} \end{cases} \\
&\sum_{v\in\mathcal{V}} \kappa^{\mathcal{A}}_v \geq \sum_{(i,j)\in\mathcal{A}} g_{i,j}, \; \forall \mathcal{A} \in \mathcal{W}, \\
&\kappa^{\mathcal{A}}_i \leq R_{w,i}, \quad f^{\mathcal{A}}_{i,j} \leq g_{i,j}, \quad g_{i,j} \leq 1 - p_{i,j}, \; \forall (i,j) \in \mathcal{E},
\end{aligned} \tag{5.13}$$

where the first equality is the flow conservation for the random keys intended to the virtual sink $d^{\mathcal{A}}$, $\kappa^{\mathcal{A}}_v$ is the random key injection rate at node v intended to $d^{\mathcal{A}}$ and $f^{\mathcal{A}}_{i,j}$ is the random key flow on link (i,j) for $d^{\mathcal{A}}$; the second equality is the flow conservation for the secret data and random keys and $g_{i,j}$ is the actual flow on link (i,j); the third

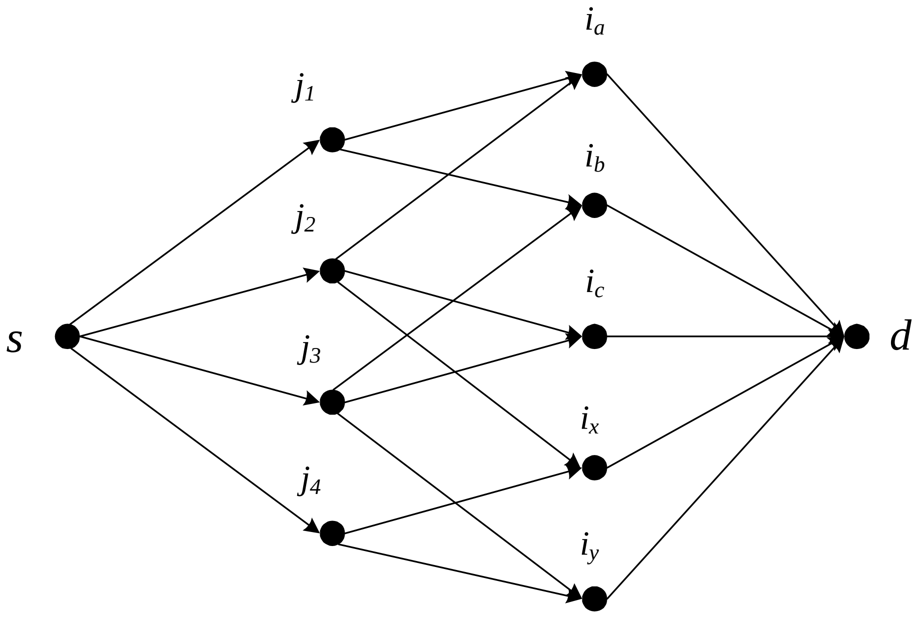


Figure 5.2: Example of usefulness of strategy 2.

inequality requires that the total amount of keys for wiretapping set $\mathcal{A}$ is greater than or equal to the total amount of data that the wiretapper receives on $\mathcal{A}$; the fourth set of inequalities requires that the amount of keys injected at each node intended to the wiretapping set $\mathcal{A}$ is less than or equal to the total amount of keys injected at that node, and the key flow on each link is less than or equal to the actual flow. The actual flow on each link is constrained by the capacity of each link. The number of wiretapping subsets $\mathcal{A}$ is exponential in k, but it is not so bad if the number of links that can be wiretapped is small. Note that under the assumption that different nodes do not have common randomness we cannot apply the key cancelation and precoding idea in Strategy 1, as after applying the precoding matrix each node may potentially be required to transmit a mixture of all the random keys in the network. If different nodes can share random keys, this is equivalent to having a virtual key source in the network and the precoding idea can be applied.

An example where this strategy is useful is given in Fig. 5.2, which is obtained by interchanging the source and the sink as well as reversing all the links in Fig. 5.1. At most three links in the last layer can be wiretapped. By injecting one local key at node j_2 and two global keys at the source, strategy 2 can achieve secrecy rate 2.

On the other hand, if random keys are only injected at the source, the secrecy rate is at most $\frac{8}{5}$. Let r and w be the secrecy rate and the random key rate at the source, respectively. Let x be the total actual flow on the last layer. To achieve secrecy, we must have $w \geq \frac{3}{5}x$, where the min-cut condition on the last layer requires $r + w \leq x$. Since the source injects all the random keys, the min-cut condition on the first layer requires $r + w \leq 4$. Combining these we obtain $r \leq \frac{8}{5}$, which is strictly less than 2.

From the proposed two strategies, we can see that strategy 1 seems to be useful if the wiretapped links are upstream of the min-cut while strategy 2 is useful if the wiretapped links are downstream of the min-cut. In general, these two strategies can be combined to obtain a higher secrecy rate.

5.5 Unachievability of Cut Set Bound

From Theorem 5.1, when the location of the wiretap links is known, the secrecy capacity is equal to the erasure capacity where the wiretap links are erased, which is given by the cut set bound. As the case when the location of the wiretap links is unknown is more restricted than that when the location is known, the cut set bound is also an outer bound for the former case. In the case of unrestricted wiretapping sets and unit link capacities, the secrecy capacity is the same whether or not the locations of the wiretap links are known, i.e. the cut set bound [13]. In the following we will show that the cut set bound does not hold in general for the case when the locations of the wiretap links are unknown, by considering the example in Fig. 5.3, where the set of wiretappable links is restricted (scenario 3). We give an explicit proof that the cut set bound is not achievable. We also use the program Information Theoretic Inequalities Prover (Xitip) [3] to show that the secrecy capacity is bounded away from the cut set bound. We then convert the example into one with unequal link capacities (scenario 4), and show the unachievability of the cut set bound for this case also.

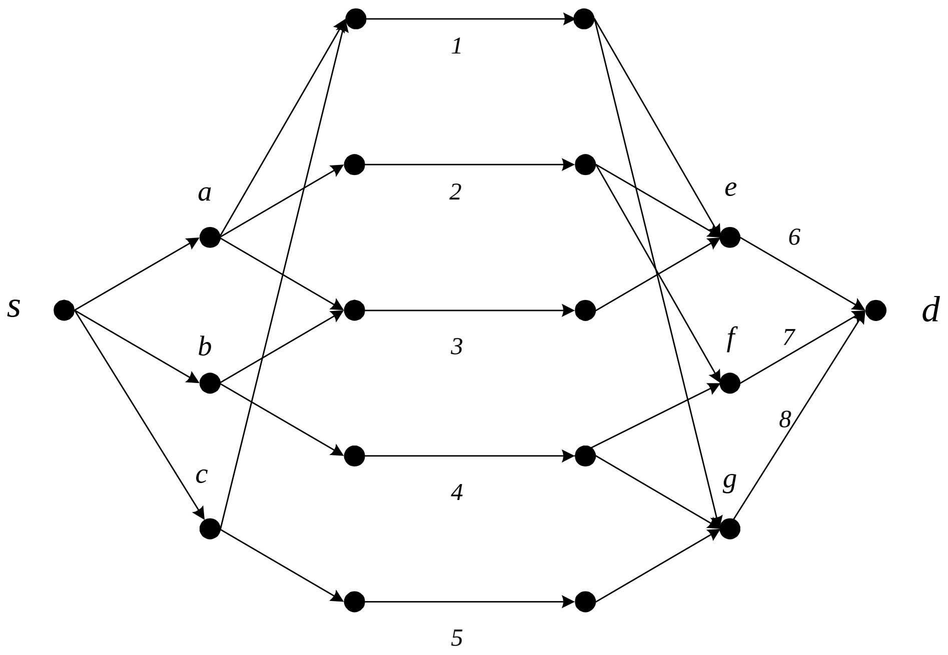


Figure 5.3: An example to show that the secrecy rate without knowledge of wiretapping set is smaller than that with such knowledge. The wiretapper can wiretap any three of the five links in the middle layer.

5.5.1 Restricted Wiretap Set (Scenario 3)

Let the middle layer links be 1-5 (from top to bottom) and the last layer links be 6-8 (from top to bottom). All links have unit capacity. Let the signal carried by link i be called signal i, or S_i. Let the source information be denoted X. For this example, the secrecy rate is two if any three of the five links in the middle layer are deleted, i.e., the number of wiretapped links is three.

The constraints required are that the source information is a function of the signals on the sink's incoming links, and that there is zero mutual information between the source information and the signals on the links in each adversarial subset.

In this example, the cut set bound is 2. To provide intuition, we first show that secrecy rate 2 cannot be achieved by using linear coding. Then, the argument is converted to an information theoretic proof that secrecy rate 2 cannot be achieved using any coding schemes.

Suppose secrecy rate 2 is achievable with a linear network code. First note that the source cannot inject more than unit amount of random key, otherwise the first layer cannot carry two units of source data. Let the random key injected by the source be denoted K. For the case when the source injects a unit amount of secret key, we first have the following observations. Signal 6 must be a function of signal 1, otherwise if the adversary sees the signals 2-4 then he knows signals 6-7. Also, signal 8 must be a function of signal 5, otherwise if the adversary sees signals 1, 2 and 4, then he knows signals 7-8. Similarly we can show that signal 8 must be a function of signal 1, and signal 7 must be a function of signal 2. We consider the following two cases.

Case 1: signal 5 is a linear combination of signals present at the source node. To achieve the full key rank condition on links 1, 2 and 5, the top second layer node must put independent local keys k_1 and k_2 on links 1 and 2 respectively. Link 7, whose other input is independent of k_2, is then a function of k_2. Similarly, Link 8 is a function of k_1. This means that the last layer has two independent local keys on it.

Case 2: signal 5 is a linear combination of signals present at the source node as well as a local key k injected by the bottom second layer node.

Case 2a: k is also present in signal 1. Then k is present in signal 6, and is independent of the key present in signal 7.

Case 2b: k is not present in signal 1. Then k is present in signal 8, and is independent of the key present in signal 7.

From Cases 1, 2a, and 2b, we conclude that the secrecy rate without knowledge of the wiretapping set by using only linear network coding is less than two.

Next, we convert this argument into an information theoretic proof that secrecy rate 2 is not achievable with any coding scheme. Suppose rate 2 is achievable. Then each triple of links in the middle layer has zero mutual information with the source data, and each pair of links in the middle layer has joint conditional entropy 2 given the other three links.

Since the last layer has to carry two units of source data, there is at most 1 unit of random key on the last layer. Since $I(S_1, S_2, S_3; X) = 0$, by the data processing inequality $I(S_6; X) = 0$. Similarly, $I(S_7; X) = I(S_8; X) = 0$. Therefore the last layer links must have joint entropy 3, and no additional random keys can be injected after the middle layer. Then the adversary can know the signal on any one of the links in the last layer, so there must be one unit of random key on the last layer. Therefore, the coding scheme must ensure that the mutual information between the adversary's observations and the information on the last layer is 1. Then, the mutual information between signal 6 and signals 2-3 is 0, otherwise if the adversary sees signals 2-4 his mutual information with signals 6-7 is greater than 1. The mutual information between signal 8 and signals 1, 4 is 0, otherwise if the adversary sees signals 1, 2, 4 his mutual information with signals 7-8 is greater than 1. The mutual information between signal 8 and signals 4-5 must be 0, otherwise if the adversary sees signals 2, 4, 5 his mutual information with signals 7-8 is greater than 1. The mutual information between signal 7 and signals 4-5 must be 0, otherwise if the adversary sees signals 1, 4, 5, his mutual information with signals 7-8 is greater than 1.

Case 1: signal 5 is a function of only signals present at the source node, i.e., $H(S_5|X,K) = 0$. By the zero mutual information condition for links 1, 2 and 5, $H(S_1,S_2,S_5|X)$=3, so $H(S_1,S_2,S_5|X,K) = H(S_1,S_2|X,K,S_5) = 2$. Since S_4 is conditionally independent of S_1, S_2 given X and K, we have $H(S_1,S_2|X,K,S_4,S_5) = 2$, $I(S_1,S_2;X,K,S_4,S_5) = 0$ and $I(S_1,S_2;X,K|S_4,S_5) = 0$. Now

$$\begin{aligned}I(S_1,S_2,S_7,S_8;X,K|S_4,S_5) =& I(S_7,S_8;X,K|S_4,S_5) + I(S_1,S_2;X,K|S_7,S_8,S_4,S_5)\\ =& I(S_1,S_2;X,K|S_4,S_5) + I(S_7,S_8;X,K|S_1,S_2,S_4,S_5).\end{aligned}$$

Since S_7, S_8 is conditionally independent of X, K given S_1, S_2, S_4, S_5, we have

$$I(S_7,S_8;X,K|S_1,S_2,S_4,S_5) = 0.$$

Then by the non-negativity of conditional mutual information, $I(S_7,S_8;X,K|S_4,S_5) \leq I(S_1,S_2;X,K|S_4,S_5) = 0$. Next, note that S_1 and S_2 are conditionally independent given S_4 and S_5, since $H(S_1|S_4,S_5) = H(S_2|S_1,S_4,S_5) = 1$. Therefore S_7 and S_8 are conditionally independent given S_4 and S_5, i.e. $I(S_7;S_8|S_4,S_5) = 0$. Since $H(S_7|S_4,S_5) = H(S_7) - I(S_7;S_4,S_5) = 1$, it follows that $H(S_7|S_8,S_4,S_5) = 1$. Then we have

$$\begin{aligned} I(S_7, S_8; S_4, S_5) =& I(S_8; S_4, S_5) + I(S_7; S_4, S_5|S_8) \\ =& I(S_8; S_4, S_5) + H(S_7|S_8) - H(S_7|S_4, S_5, S_8) = 0 + 1 - 1 = 0. \end{aligned}$$

So, $I(S_7, S_8; X, K, S_4, S_5) = I(S_7, S_8; X, K|S_4, S_5) + I(S_7, S_8; S_4, S_5) = 0$, and therefore $H(S_7, S_8|X) \geq H(S_7, S_8|X, K, S_4, S_5) = 2$, which contradicts the requirement that there is at most 1 unit of secret key on the last layer.

Case 2: signal 5 is not a function only of signals present at the source

Case 2a: signal 1 has nonzero mutual information with some secret key injected at node c. Then $H(S_1|X, K, S_2, S_3, S_4) > 0$. For brevity, let $A = (S_2, S_3)$ and $Y = (X, K, S_4)$. Since $I(S_6; A) = 0$ and $H(S_6|S_1, A) = 0$, we have $H(A) + H(S_6) = H(A, S_6) \leq H(A, S_1) = H(S_1) + H(A|S_1)$. Since $H(S_6) = H(S_1)$, we have $H(A) = H(A|S_1)$ and so $H(S_1|A) = H(S_1)$. Then from $H(S_1, S_6|A) = H(S_1|A, S_6) + H(S_6|A) = H(S_6|A, S_1) + H(S_1|A)$, we have $H(S_1|A, S_6) = 0$. Since $H(S_1|A, Y, S_6) \leq H(S_1|A, S_6) = 0$ and $H(S_6|A, Y, S_1) \leq H(S_6|A, S_1) = 0$, from $I(S_1; S_6|Y, A) = H(S_1|A, Y) - H(S_1|A, Y, S_6) = H(S_6|A, Y) - H(S_6|A, Y, S_1)$ we have $H(S_6|A, Y) = H(S_1|A, Y) > 0$. Then since $H(S_7|S_2, S_4) = 0$, we have $H(S_6|S_7, X) > 0$. Also, since $H(S_7|X) = 1$, we have $H(S_6, S_7|X) > 1$.

Case 2b: signal 1 has zero mutual information with any random key injected at node c. Then $H(S_5|X, K, S_1, S_2, S_4) > 0$. Similar reasoning as for case 2a applies with $A = (S_1, S_4)$, $Y = (X, K, S_2)$, S_5 in place of S_1, and S_8 in place of S_6.

From Cases 1, 2a, and 2b, we conclude that the secrecy rate without knowledge of the wiretapping set by using any nonlinear or linear coding strategy is smaller than two obtained for the case where such knowledge is present at the source.

We can also show that the secrecy rate is bounded away from 2 by using the framework for linear information inequalities [81]. Let X be the message sent from the source and Z_i, $i = 1, \ldots, 3$ be the signals on the links adjacent to the source. We want to check whether $H(X) \leq \omega$ is implied by

$$\begin{array}{rl}
(1) & H(Z_i) \leq 1,\ H(S_j) \leq 1,\ i = 1, \ldots, 3,\ j = 1, \ldots, 8, \\
(2) & H(X|S_6, S_7, S_8) = 0, \\
(3) & I(X, Z_1, Z_2, Z_3, S_4, S_5, S_7, S_8; S_6|S_1, S_2, S_3) = 0, \\
(4) & I(X, Z_1, Z_2, Z_3, S_1, S_3, S_5, S_6, S_8; S_7|S_2, S_4) = 0, \\
(5) & I(X, Z_1, Z_2, Z_3, S_2, S_3, S_6, S_7; S_8|S_1, S_4, S_5) = 0, \\
(6) & I(X; S_1, S_2, S_3) = 0,\ I(X; S_1, S_2, S_4) = 0, \\
(7) & I(X; S_1, S_2, S_5) = 0,\ I(X; S_1, S_3, S_4) = 0, \\
(8) & I(X; S_1, S_3, S_5) = 0,\ I(X; S_1, S_4, S_5) = 0, \\
(9) & I(X; S_2, S_3, S_4) = 0,\ I(X; S_2, S_3, S_5) = 0, \\
(10) & I(X; S_2, S_4, S_5) = 0,\ I(X; S_3, S_4, S_5) = 0, \\
(11) & I(S_1; Z_2|Z_1, Z_3) = 0,\ I(S_2; Z_2, Z_3|Z_1) = 0, \\
(12) & I(S_3; Z_3|Z_1, Z_2) = 0,\ I(S_4; Z_1, Z_3|Z_2) = 0, \\
(13) & I(S_5; Z_1, Z_2|Z_3) = 0,\ I(S_1; S_4|Z_1, Z_2, Z_3) = 0, \\
(14) & I(S_2; S_4, S_5|Z_1, Z_2, Z_3) = 0,\ I(S_3; S_5|Z_1, Z_2, Z_3) = 0, \\
(15) & I(S_4; S_1, S_2, S_5|Z_1, Z_2, Z_3) = 0,\ I(S_5; S_2, S_3, S_4|Z_1, Z_2, Z_3) = 0, \\
(16) & I(S_1, S_2, S_3, S_4, S_5; X|Z_1, Z_2, Z_3) = 0,
\end{array} \tag{5.14}$$

where the first inequality is the capacity constraint, the second constraint shows that the sink can decode X, constraints 3 to 5 mean that the signals in the last layer are independent of other signals given the incoming signals from the middle layer, constraints 6 to 10 represent that the secrecy constraints when any 3 links in the middle layer are wiretapped, and constraints 11 to 16 represent the conditional independence between the signals in the first layer and those in the middle layer. Note that constraints 3 to 5 and 11 to 16 implicitly allow some randomness to be injected at the corresponding nodes. We use the Xitip program [3], which relies on the framework in [81], to show that $H(X) \leq 5/3$ is implied by the set of equalities

(5.14). Therefore, 5/3 is an upper bound on the secrecy rate when the location of wiretapper is unknown, which is less than the secrecy rate 2 achievable when such information is known. Therefore, there is a strict gap between the secrecy capacity and the cut set bound.

5.5.2 Unequal Link Capacities (Scenario 4)

We have restricted the wiretapped links to be in the middle layer in Fig. 5.3. We next show that the unachievability of the cut-set bound also holds for the secure network coding problem with unequal link capacities (scenario 4). We convert the example of Fig. 5.3 by partitioning each non-middle layer link into $\frac{1}{\epsilon}$ parallel small links each of which has capacity ϵ. Any three links can be wiretapped in the transformed graph.

For the case where the location of the wiretap links is known, if $\epsilon < \frac{1}{3}$, deleting any k' ($k' \leq 3$) non-middle layer links reduces the max flow by at most $\frac{k'}{3} \leq 1$. When $k' \geq 1$ or at most 2 middle layer links are wiretapped, the min-cut between the source and the sink is at least 3 after deleting these wiretapped links. Therefore, the min-cut is at least $3 - \frac{k'}{3} \geq 2$ when $k' \geq 1$. Thus, the wiretapper does not have incentive to wiretap the non-middle layer links in the transformed graph. This shows the secrecy rate in the original graph is the same as that in the transformed graph when the location of the wiretap links is known.

For the case where the location of the wiretap links is unknown, we prove the unachievability of the cut-set bound in the transformed network. First, consider the transformed network with the restriction that the wiretapper can only wiretap any 3 links in the middle layer. The optimal solution is exactly the same as for the original network of the previous subsection, and achieves secrecy rate at most 5/3. Now, consider the transformed network without the restriction on wiretapping set, i.e., the wiretapper can wiretap any 3 links in the entire network. As wiretapping only the middle layer links is a subset of all possible strategies that the wiretapper can have, the secrecy rate in the transformed network is less than or equal to that in the former case, which is strictly smaller than the cut-set bound 2. Therefore, the cut-set bound is still unachievable when the wiretap links are unrestricted in the transformed graph.

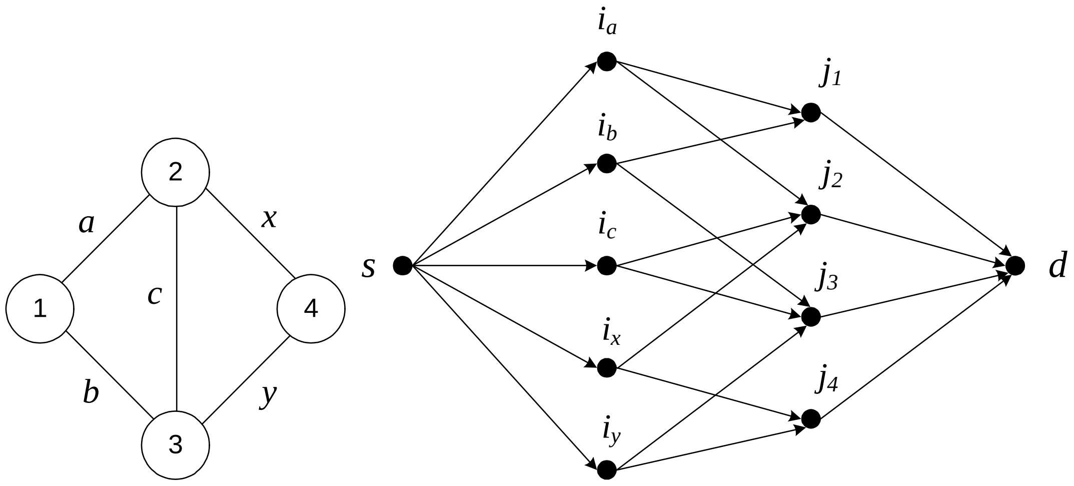
5.6 NP-Completeness

We show in the following that determining the secrecy capacity is NP-complete when the location of the wiretap links is known or unknown by reduction from the clique problem, which determines whether a graph contains a clique[2] of at least a given size r.

The case when the location of the wiretap links is known by the source and the wiretapper can choose the wiretapping set to minimize the secrecy rate is closely related to the network interdiction problem [79]. The network interdiction problem is to minimize the maximum flow of the network when a certain number of links in the network is removed, which is a special case of the secrecy communication problem when the location of the wiretap links is known and $q_{i,j} = p_{i,j}$, $\forall (i,j) \in \mathcal{E}$. It is shown in [79] that the network interdiction problem is NP-complete. Therefore, the case where the location of the wiretap links is known is also NP-complete. To show that the case where the location of the wiretap links is unknown is NP-complete, we use the construction in [79] showing that for any clique problem on a given graph $\mathcal{H}$, there exists a corresponding network $\mathcal{G}^{\mathcal{H}}$ whose secrecy capacity is r when the location of the wiretap links is known if and only if $\mathcal{H}$ contains a clique of size r. We then show that for all such networks the secrecy rate for the case when the location of the wiretap links is unknown is equal to that for the case when such information is known, which shows that there is a one-to-one correspondence between clique problem and the secrecy capacity problem.

We briefly describe the approach in [79] in the following. Given an undirected graph $\mathcal{H} = (\mathcal{V}_h, \mathcal{E}_h)$, we will define a capacitated directed network $\hat{\mathcal{G}}^{\mathcal{H}}$ such that there exists a set of links $\hat{\mathcal{A}}'$ in $\hat{\mathcal{G}}^{\mathcal{H}}$ containing less than or equal to $|\mathcal{E}_h| - \binom{r}{2}$ links such that $\hat{\mathcal{G}}^{\mathcal{H}} - \hat{\mathcal{A}}'$ has a maximum flow of r if and only if $\mathcal{H}$ contains a clique of size r.

[2] A clique in a graph is a set of pairwise adjacent vertices, or in other words, an induced subgraph which is a complete graph.



(a) Original Graph (b) Transformed Graph

Figure 5.4: Example of NP-completeness proof for the case with knowledge of wiretapping set

For a given undirected graph $\mathcal{H} = (\mathcal{V}_h, \mathcal{E}_h)$ without parallel edges and self loops, we create a capacitated, directed graph $\mathcal{G}^{\mathcal{H}} = (\mathcal{N}, \mathcal{A})$ as follows: For each edge $e \in \mathcal{E}_h$ create a node i_e in a node set $\mathcal{N}_1$ and for each vertex $v \in \mathcal{V}_h$ create a node j_v in a node set $\mathcal{N}_2$. In addition, create source node s and destination node d. For each edge $e \in \mathcal{E}_h$, direct an arc in $\mathcal{G}^{\mathcal{H}}$ from s to i_e with capacity 2 and call this set of arcs $\mathcal{A}_1$. For each edge $e = (u, v) \in \mathcal{E}_h$, direct two arcs in $\mathcal{G}^{\mathcal{H}}$ from i_e to j_v and j_u with capacity 1, respectively and call this set of arcs $\mathcal{A}_2$. For each vertex $v \in \mathcal{V}_h$, direct an arc with capacity 1 from j_v to d. Let this be the set of arcs $\mathcal{A}_3$. This completes the construction of $\mathcal{G}^{\mathcal{H}} = (\mathcal{N}, \mathcal{A}) = (\{s\} \cup \{d\} \cup \mathcal{N}_1 \cup \mathcal{N}_2, \mathcal{A}_1 \cup \mathcal{A}_2 \cup \mathcal{A}_3)$. In Fig. 5.4, we give an example of the graph transformation, where $\mathcal{H} = (\{1, 2, 3, 4\}, \{a, b, c, x, y\})$. We replicate [79, Lemma 2] as follows.

Lemma 5.3 *Let $\mathcal{G}^{\mathcal{H}}$ be constructed from $\mathcal{H}$ as above. Then, there exists a set of arcs $\mathcal{A}'_1 \subseteq \mathcal{A}_1$ with $|\mathcal{A}'_1| = |\mathcal{E}_h| - \binom{r}{2}$ such that the maximum flow from s to d in $\mathcal{G}^{\mathcal{H}} - \mathcal{A}'_1$ is r if and only if $\mathcal{H}$ contains a clique of size r.*

After obtaining $\mathcal{G}^{\mathcal{H}}$, we generate $\hat{\mathcal{G}}^{\mathcal{H}}$ by replacing each arc (i_e, j_v) with $|\mathcal{E}_h|$ parallel arcs each with capacity $1/|\mathcal{E}_h|$ and call this arc set $\hat{\mathcal{A}}_2$. We carry out the same procedure for arcs (j_v, d_l) and call this arc set $\hat{\mathcal{A}}_3$. Then $\hat{\mathcal{G}}^{\mathcal{H}} = (\mathcal{N}, \mathcal{A}) = (\{s\} \cup \{d\} \cup$

$\mathcal{N}_1 \cup \mathcal{N}_2, \mathcal{A}_1 \cup \hat{\mathcal{A}}_2 \cup \hat{\mathcal{A}}_3)$. For the case when the location of wiretap links is known, it is shown in [79] that the worst case wiretapping set $\hat{\mathcal{A}}'$ must be a subset of $\mathcal{A}_1$. By using Lemma 5.3, this case is NP-complete.

Now, we consider the case where the wiretapping set is unknown, and show that the secrecy capacity of $\hat{\mathcal{G}}^{\mathcal{H}}$ when the wiretapper accesses any unknown subset of $k = |\mathcal{E}_h| - \binom{r}{2}$ links is r if and only if $\mathcal{H}$ contains a clique of size r. From Lemma 5.3, the condition that $\mathcal{H}$ contains a clique of size r is equivalent to the condition that the max-flow to the sink in $\mathcal{G}^{\mathcal{H}}$ after removing any k links from $\mathcal{A}_1$ is r. We now show that the latter condition is equivalent to the condition that the secrecy capacity of $\mathcal{G}^{\mathcal{H}}$ when the wiretapper accesses any unknown subset of k links from $\mathcal{A}_1$ is r. We create a virtual sink connecting each subset of k links from $\mathcal{A}_1$ and the actual sink. As the wiretapped links are connected to the source directly, the min-cut between each virtual sink and the source is at least $2k + r$. Since r is the cut-set upper bound on the secrecy rate, by using the key cancelation scheme (Strategy 1) in Section 5.4 with $2k$ global keys the secrecy rate r is achievable, which is equal to the secrecy rate when the location of wiretap links is known.

Finally, we show that the secrecy capacity of $\mathcal{G}^{\mathcal{H}}$ when any k links of $\mathcal{A}_1$ are wiretapped is equal to the secrecy capacity of $\hat{\mathcal{G}}^{\mathcal{H}}$ when any k links are wiretapped. Since each second layer link has a single first layer link as its only input, wiretapping a second layer link yields no more information to the wiretapper than wiretapping a first layer link. When some links in the third layer are wiretapped, let the wiretapping set be $\hat{\mathcal{A}}' = \hat{\mathcal{A}}'_1 \cup \hat{\mathcal{A}}'_3$ where $\hat{\mathcal{A}}'_3 \neq \emptyset$ or $|\hat{\mathcal{A}}'_3| \geq 1$ and $|\hat{\mathcal{A}}'_1| \leq k - 1$. Thus $\mathcal{A}_1 - \hat{\mathcal{A}}'_1$ contains at least $\binom{r}{2} + 1$ arcs. As in Section 5.4, we create a node $t^{\hat{\mathcal{A}}'}$ by connecting all wiretapping links in $\hat{\mathcal{A}}'$ and a virtual sink $d^{\hat{\mathcal{A}}'}$. Connect the actual sink to $d^{\hat{\mathcal{A}}'}$ with a link of capacity r and connect $t^{\hat{\mathcal{A}}'}$ to $d^{\hat{\mathcal{A}}'}$ with a link of capacity $R_{s \to \hat{\mathcal{A}}'}$, where $R_{s \to \hat{\mathcal{A}}'}$ is the min-cut between the source and $t^{\hat{\mathcal{A}}'}$. As removing links in $\mathcal{A}_1$ is equivalent to removing links in $\mathcal{H}$, after removing links in $\mathcal{H}$ corresponding to $\hat{\mathcal{A}}'_1$, $\mathcal{H}$ contains a subgraph $\mathcal{H}_1$ containing $\binom{r}{2}$ edges plus at least an edge $e = (u, v)$. We consider two cases.

Case 1: $\mathcal{H}_1$ is a clique of size r. In this case, the number of vertices with degree greater than 0 in $\mathcal{H}_1 \cup e$ is $r+2$.

Case 2: $\mathcal{H}_1$ is not a clique. $\mathcal{H}_1$ contains at least $r+1$ vertices with degree greater than 0.

According to [79, Lemma 1], the max-flow in $\mathcal{G}^{\mathcal{H}}$ is equal to the number of vertices in $\mathcal{H}$ with degree greater than 0. In both cases, the max-flow of $\mathcal{G}^{\mathcal{H}}$ after removing links in $\hat{\mathcal{A}}'_1$ is at least $r+1$. Let $R_{s\rightarrow\hat{\mathcal{A}}'_3}$ be the max-flow capacity from the source to $\hat{\mathcal{A}}'_3$ in $\mathcal{G}^{\mathcal{H}} - \hat{\mathcal{A}}'_1$ and $\mathcal{D}$ a corresponding max-flow subgraph. After removing $\mathcal{D}$ from $\mathcal{G}^{\mathcal{H}} - \hat{\mathcal{A}}'_1$, the min-cut between the source and the actual sink is at least $r+1-|\hat{\mathcal{A}}'_3|/|\mathcal{E}_h| > r+1-(|\mathcal{E}_h|-1)/|\mathcal{E}_h| > r$. Therefore, the min-cut between the source and $d^{\hat{\mathcal{A}}'}$ in $\mathcal{G}^{\mathcal{H}} - \hat{\mathcal{A}}'_1 - \mathcal{D}$ is r, and the min-cut between the source and $d^{\hat{\mathcal{A}}'}$ in $\mathcal{G}^{\mathcal{H}}$ is $r + R_{s\rightarrow\hat{\mathcal{A}}'}$. On the other hand, the total amount of wiretapped data is at most $2|\hat{\mathcal{A}}'_1| + |\hat{\mathcal{A}}'_3|/|\mathcal{E}_h| \leq 2|\hat{\mathcal{A}}'_1| + (|\mathcal{E}_h|-1)/|\mathcal{E}_h| \leq 2k-2+(|\mathcal{E}_h|-1)/|\mathcal{E}_h| < 2k$. By using the precoding scheme in Strategy 1 in Section 5.4, if the source sends r message symbols and $2k$ random keys, perfect secrecy is achieved when $\hat{\mathcal{A}}'$ is wiretapped and the size of finite field is large enough. Thus, the secrecy rate for the case when the location of the wiretap links is unknown is equal to that for the case when such information is known with unrestricted wiretapping set. We have the following theorem.

Theorem 5.4 *Computing the secrecy capacity no matter whether the location of the wiretap links is known is NP-complete.*

5.7 Conclusion

In this chapter, we have considered secrecy capacity of wireline erasure networks where different links have different capacities. We have shown that the secrecy capacity is not the same in general when the location of the wiretapped links is known or unknown. We gave achievable strategies where random keys are canceled at intermediate non-sink nodes, or injected at intermediate non-source nodes. We showed that determining the secrecy capacity is a NP-complete problem no matter whether the location of the wiretapped links is known or unknown.

Chapter 6

Conclusions and Future Research

In this thesis, we have considered wireless broadcast at different layers of wireless networks and studied network coding for secure communications. Our results have demonstrated the usefulness of wireless broadcast and network coding for network design. At the physical layer, we have studied physical layer network coding in two-way relay channels, where network coding is considered to be a mapping from the relay's received signal to its transmitted signal. We analyzed the symbol-error performance of several relay strategies such as amplify and forward, detect and forward, and estimate and forward. Furthermore, the relay function was also optimized via functional analysis such that the average probability of error is minimized. The optimized function was shown to behave like AF at low SNR and like DF at high SNR, respectively. These results suggest that the interference caused by wireless broadcast can be exploited to improve the spectrum efficiency.

We have also integrated wireless broadcast into cross-layer optimization and designed cross-layer protocols using dual decomposition. Under the primary interference model, the link scheduling problem was shown to be the maximum weighted hypergraph matching problem that is NP-complete. Several distributed approximation algorithms were proposed, whose worst case performance was also bounded. With random network coding, we obtained a fully distributed cross-layer design. Our results show that wireless broadcast is potentially useful in multicast scenarios.

We have further developed a new class of random medium access control protocol by allowing each user to transmit at different data rates chosen randomly from an appropriately determined set of rates, which uses successive interference cancelation to resolve packet collision due to wireless broadcast. When the number of transmission rates at each node is equal to the number of users, the achievable total throughput was shown to be at least a constant fraction of the centralized multiple access channel sum rate in slotted Aloha type networks. To facilitate practical protocol design, we also studied the case when only a limited number of transmission rates is available at each node. A game-theoretic framework was proposed to achieve the desired throughput optimal equilibrium in the absence of centralized knowledge of the total number of users. We studied the design of random access games, characterized their equilibria, studied their dynamics, and proposed distributed algorithms to achieve the equilibria.

Lastly, we considered secure communications in networks with erasure and unequal link capacities in the presence of a wiretapper. For the case when the location of the wiretapped links is known, we have derived the secrecy capacity region. For the case when the location of the wiretapped links is unknown, we proposed several achievable strategies. We showed that unlike the case of equal link capacities, the secrecy capacity when the location of wiretapped links is known and unknown are generally unequal. We also showed that computing the secrecy capacity for both cases are NP-complete.

With the increasing demand for wireless multimedia services and high-speed Internet access, we expect to see increasing interest in exploiting wireless broadcast and network coding in network design, which offers both theoretical and practical benefits. The study in this thesis only scratches the tip of the iceberg and many important problems remain to be answered.

In chapter 2, we have considered memoryless operations in two-way relay channels. When the relay has a large memory, more complicated signal processing can be performed at the relay. One open problem is determining the capacity region of Gaussian two-way relay channel. We hope that the relay strategies in chapter 2 can motivate capacity achieving schemes. For applications such as wireless teleconferencing, the N-way relay network is a more general wireless network architecture than the two-way relay channel, where there are N source terminals in the network and each source terminal needs to exchange information between all other terminals with the help of a relay node. It is interesting to extend the approaches in chapter 2 to the N-way relay network. The relaying strategies in chapter 2 assume perfect knowledge of channel state information at all the nodes. Such information is hard to obtain especially in TWRC where two channel coefficients are required to be estimated each way. In this case, non-coherent schemes, which do not rely on instant channel state information, become a preferred choice.

As to the cross-layer optimization with wireless broadcast in chapter 3, it is interesting to investigate the achievable rate ratio between multicasting with and without wireless broadcast or with and without network coding. From our experimental results, it seems that this ratio depends on the size of multicast group. Although we only proposed distributed scheduling algorithms for primary interference model, it is of interest to extend the proposed scheduling policy and approximation algorithms to other interference models such as two-hop interference model. In the proposed algorithms, we have assumed a control channel to facilitate hypergraph matching. A possible future work is to eliminate the use of control channel. Finally, it would be interesting to develop practical protocols based on the proposed algorithms.

Regarding the multiple access MAC in chapter 4, other advanced signal processing and information theory techniques besides successive interference cancelation can also be used to design new protocols. An important question is whether there exists a simple strategy that can achieve the capacity of centralized multiple access channel distributedly. We have been focusing on laying out a theoretical framework. In parallel, much work remains to take it from a promising design framework to a full-fledged medium access control protocol. It is also interesting to investigate the coexistence of new protocols and 802.11 DCF that use different contention signals: how the resource is allocated to and shared among wireless nodes using different medium access methods. This issue is important for the deployment of the new protocols.

In chapter 5, we have only given several achievable strategies when the location of the wiretapped links is unknown. The secrecy capacity region in this scenario in both wired and wireless networks are unknown. Combinations of the proposed strategies or new strategies are needed. We have not bounded the secrecy capacity gap between the case when the location of the wiretapped links is known and the case when such information is unknown. Finally, it is interesting to derive distributed and polynomial time algorithms to achieve the secrecy capacity in practice.

Bibliography

[1] Evolved universal terrestrial radio access (E-UTRA); multiplexing and channel coding. The 3rd Generation Partnership Project, TS 36.212, Jun. 8, 2009.

[2] Wireless LAN media access control (MAC) and physical layer (PHY) specifications. IEEE Standard 802.11, June 1999.

[3] Xitip - information theoretic inequalities prover. `http://xitip.epfl.ch/`.

[4] I. Abou-Faycal and M. Médard. Optimal uncoded regeneration for binary antipodal signaling. In *Proc. of IEEE ICC*, pages 742–746, June 2004.

[5] R. Abraham, J. E. Marsden, and T. Ratiu. *Manifolds, Tensors, Analysis, and Applications*. Springer-Verlag, 2 edition, 1988.

[6] N. Abramson. The Aloha system–Another alternative for computer communications. In *Proc. of Fall Joint Comput. Conf.*, pages 281–285, Apr. 1970.

[7] R. Ahlswede, N. Cai, S. Y. R. Li, and R. W. Yeung. Network information flow. *IEEE Trans. Inform. Theory*, 46(4):1204–1216, Jul. 2000.

[8] C. Berrou, A. Glavieux, and P. Thitimajshima. Near shannon limit error-correcting coding and decoding:turbo-codes. In *Proc. of IEEE ICC*, pages 1064–1070, May 1993.

[9] D. P. Bertsekas and R. Gallager. *Data Networks*. Prentice Hall, 2nd edition, 1992.

[10] Dimitri P. Bertsekas and John N. Tsitsiklis. Gradient convergence in gradient methods with errors. *SIAM J. Optim.*, 10(3):627–642, May 1999.

[11] G. Bianchi. Performance analysis of the IEEE 802.11 distributed coordination function. *IEEE J. Select. Areas Commun.*, 18(3):535–547, Mar. 2000.

[12] K. C. Border. *Fixed Point Theorems with Applications to Economics and Game Theory.* Cambridge University Press, 1985.

[13] Ning Cai and R.W. Yeung. Secure network coding. In *Proc. of IEEE ISIT*, page 323, June 2002.

[14] J. Cao and E. M. Yeh. Asymptotically optimal multiple-access communication via distributed rate splitting. *IEEE Trans. Inform. Theory*, 53(1):304–319, Jan. 2007.

[15] A. B. Carleial. Interference channels. *IEEE Trans. Inform. Theory*, 24(1):60–70, Jan. 1978.

[16] Prasanna Chaporkar, Koushik Kar, and Saswati Sarkar. Fairness and throughput guarantees with maximal scheduling in multi-hop wireless networks. In *Proc. of Allerton Conf. on Comm., Contr. and Comput.*, Sept. 2005.

[17] L. Chen, S. H. Low, M. Chiang, and J. C. Doyle. Cross-layer congestion control, routing and scheduling design in ad hoc wireless networks. In *Proc. of IEEE Infocom*, Apr. 2006.

[18] Lijun Chen, Tracey Ho, Steven H. Low, Mung Chiang, and John C. Doyle. Optimization based rate control for multicast with network coding. in *Proc. IEEE Infocom*, 2007.

[19] Lijun Chen, Steven H. Low, and John C. Doyle. Joint congestion control and media access control design for wireless ad hoc networks. In *Proc. of IEEE Infocom*, Mar. 2005.

[20] Lijun Chen, Steven H. Low, and John C. Doyle. Random access game and medium access control design. Caltech CDS, Tech. Rep., Mar. 2006. `http://www.cds.caltech.edu/~chen/papers/ramac.pdf`.

[21] M. Chiang, S. H. Low, A. R. Calderbank, and J. C. Doyle. Layering as optimization decomposition. *Proc. of IEEE*, Jan. 2007.

[22] Thomas Cover and Joy Thomas. *Elements of Information Theory.* 1991.

[23] T. Cui, F. Gao, and A. Nallanathan. Optimal training design for channel estimation in amplify and forward relay networks. In *Proc. of IEEE Globecom*, 2007.

[24] Tao Cui, Feifei Gao, Tracey Ho, and Arumugam Nallanathan. Distributed space-time coding for two-way wireless relay networks. In *Proc. of IEEE ICC*, May 2008.

[25] F. A. Dana, R. Gowaikar, R. Palanki, B. Hassibi, and M. Effros. Capacity of wireless erasure networks. *IEEE Trans. Inform. Theory*, 52(3):789–804, Mar. 2006.

[26] A. Eryilmaz and D. S. Lun. Control for inter-session network coding. In *Proc. of the Workshop on Network Coding, Theory and Applications*, 2007.

[27] J. Feldman, T. Malkin, R. Servedio, and C. Stein. On the capacity of secure network coding. In *Proc. of Allerton Conference on Communication, Control, and Computing*, Sept. 2004.

[28] S. D. Flam. Equilibrium, evolutionary stability and gradient dynamics. *International Game Theory Review*, 4(4):357–370, Dec. 2002.

[29] D. Fudenburg and J. Tirole. *Game Theory.* MIT Press, 1991.

[30] Harold N. Gabow. Data structures for weighted matching and nearest common ancestors with linking. In *Proc. of ACM-SIAM Symposium on Discrete algorithms*, pages 434–443, 1990.

[31] S. Ghez, S. Verdu, and S. C. Schwartz. Stability properties of slotted Aloha with multipacket reception capability. *IEEE Trans. Automat. Contr.*, 33(7):640–649, Jul. 1988.

[32] S. Ghez, S. Verdu, and S. C. Schwartz. Optimal decentralized control in the random access multipacket channel. *IEEE Trans. Automat. Contr.*, 34(11):1153–1163, Nov. 1989.

[33] Krishna Srikanth Gomadam and Syed Ali Jafar. Optimal relay functionality for SNR maximization in memoryless relay networks. *IEEE J. Select. Areas Commun.*, 25(2):390–401, Feb. 2007.

[34] A.J. Grant, B. Rimoldi, R.L. Urbanke, and P.A. Whiting. Rate-splitting multiple access for discrete memoryless channels. *IEEE Trans. Inform. Theory*, 47(3):873–890, Mar. 2001.

[35] B. Hajek and G. Sasaki. Link scheduling in polynomial time. *IEEE Trans. Inform. Theory*, 34(5):910–917, Sept. 1988.

[36] T. S. Han and K. Kobayashi. A new achievable rate region for the interference channe. *IEEE Trans. Inform. Theory*, 27(1):49–60, Jan. 1981.

[37] C. Hausl and J. Hagenauer. Iterative network and channel decoding for the two-way relay channel. In *Proc. of IEEE ICC*, pages 1568–1573, June 2006.

[38] M. Heusse, F. Rousseau, R. Guillier, and A. Dula. Idle sense: An optimal access method for high throughput and fairness in rate diverse wireless LANs. In *Proc. of ACM Sigcomm*, pages 121–132, Aug. 2005.

[39] T. Ho, M. Médard, R. Koetter, D. Karger, M. Effros, J. Shi, and B. Leong. A random linear network coding approach to multicast. *IEEE Trans. Inform. Theory*, 52(10):4413–4430, Oct. 2006.

[40] T. Ho, M. Médard, J. Shi, M. Effros, and D. R. Karger. On randomized network coding. In *Proc. of Allerton Conference on Communication, Control, and Computing*, Sept. 2003.

[41] T. Ho and H. Viswanathan. Dynamic algorithms for multicast with intra-session network coding. submitted to *IEEE Trans. Inform. Theory*, 2006.

[42] T. Ho and H. Viswanathan. Dynamic algorithms for multicast with intra-session network coding. In *Proc. of Allerton Conf. on Comm., Contr. and Comput.*, Sept. 2005.

[43] R. A. Horn and C. R. Johnson. *Matrix Analysis.* Cambridge University Press, 1985.

[44] C. Hu and J. C. Hou. A novel approach to contention control in IEEE 802.11e-operated WLANs. In *Proc. of IEEE Infocom*, pages 1190–1198, May 2007.

[45] S. Jaggi, P. Sanders, P.A. Chou, M. Effros, S. Egner, K. Jain, and L. Tolhuizen. Polynomial time algorithms for multicast network code construction. *IEEE Trans. Inform. Theory*, 51(6):1973–1982, June 2005.

[46] R. Jain, D. Chiu, and W. Hawe. A quantitative measure of fairness and discrimination for resource allocation in shared computer systems. DEC Research Report TR-301, Sept. 1984.

[47] Sachin Katti, Shyamnath Gollakota, and Dina Katabi. Embracing wireless interference: Analog network coding. In *Proc. of ACM SIGCOMM*, 2007.

[48] Sachin Katti, Hariharan Rahul, Wenjun Hu, Dina Katabi, Muriel Médard, and Jon Crowcroft. XORs in the air: Practical wireless network coding. In *Proc. of ACM SIGCOMM*, pages 243–254, Oct. 2006.

[49] L. Kleinrock and F. Tobagi. Packet switching in radio channels: Part I–carrier sense multiple-access modes and their throughput-delay characteristics. *IEEE Trans. Commun.*, 23(12):1400–1416, Dec. 1975.

[50] Ralf Koetter and Muriel Médard. An algebraic approach to network coding. *IEEE/ACM Trans. Networking*, 11(5):782–795, Oct. 2003.

[51] 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*, 50(12):3062–3080, Dec. 2004.

[52] P. Larsson, N. Johansson, and K.-E. Sunell. Coded bi-directional relaying. In *Proc. of IEEE VTC-Spring*, pages 851–855, May 2006.

[53] J. W. Lee, A. Tang, J. Huang, M. Chiang, and A. R. Calderbank. Reverse-engineering MAC: A non-cooperative game model. *IEEE J. Select. Areas Commun.*, 25(6):1135–1147, Aug. 2007.

[54] S. Y. R. Li, R. W. Yeung, and Ning Cai. Linear network coding. *IEEE Trans. Inform. Theory*, 49(2):371 – 381, Feb. 2003.

[55] Xiaojun Lin and N.B. Shroff. The impact of imperfect scheduling on cross-layer congestion control in wireless networks. *IEEE/ACM Trans. Networking*, 14(2):302–315, April 2006.

[56] L. Lovasz and M.D. Plummer. *Matching Theory*. North Holland, 1986.

[57] D. S. Lun, M. Médard, and R. Koetter. Efficient operation of wireless packet networks using network coding. In *Proc. International Workshop on Convergent Technologies*, Jun. 2005.

[58] D. S. Lun, N. Ratnakar, M. Médard, R. Koetter, D. R. Karger, T. Ho, E. Ahmed, and F. Zhao. Minimum-cost multicast over coded packet networks. *IEEE Trans. Inform. Theory*, 52(6):2608–2623, June 2006.

[59] M. Medard, Jianyi Huang, A.J. Goldsmith, S.P. Meyn, and T.P. Coleman. Capacity of time-slotted ALOHA packetized multiple-access systems over the AWGN channel. *IEEE Trans. Wireless Commun.*, 3(2):486–499, Mar. 2004.

[60] A. Mills, B. Smith, T.C. Clancy, E. Soljanin, and S. Vishwanath. On secure communication over wireless erasure networks. In *Proc. of IEEE ISIT*, pages 161–165, July 2008.

[61] E. Modiano, D. Shah, and G. Zussman. Maximizing throughput in wireless networks via gossiping. *ACM SIGMETRICS Performance Evaluation Review*, 34(1):27–38, June 2006.

[62] Rajeev Motwani and Prabhakar Raghavan. *Randomized Algorithms.* Cambridge University Press, 1995.

[63] M. Neely, E. Modiano, and C. Li. Fairness and optimal stochastic control for heterogeneous networks. In *Proc. IEEE Infocom*, 2005.

[64] M.J. Neely, E. Modiano, and C.E. Rohrs. Dynamic power allocation and routing for time-varying wireless networks. *IEEE J. Select. Areas Commun.*, 23(1):89–103, Jan. 2005.

[65] P. Popovski and H. Yomo. Physical network coding in two-way wireless relay channels. In *IEEE International Conference on Communications*, pages 707–712, Glasgow, U.K., June 2007.

[66] R. Preis. Linear time 1/2-approximation algorithm for maximum weighted matching in general graphs. In *Proc. of the Symposium on Theoretical Aspects of Computer Science (STACS)*, pages 259–269, 1999.

[67] B. Rankov and A. Wittneben. Achievable rate regions for the two-way relay channel. In *Proc. of IEEE ISIT*, pages 1668–1672, July 2006.

[68] B. Rankov and A. Wittneben. Spectral efficient protocols for half-duplex fading relay channels. 25(2):379–389, February 2007.

[69] B. Rimoldi and R. Urbanke. A rate-splitting approach to the Gaussian multiple-access channel. *IEEE Trans. Inform. Theory*, 42(2):364–375, Mar. 1996.

[70] Y. E. Sagduyu and A.; Ephremides. Crosslayer design for distributed MAC and network coding in wireless ad hoc networks. In *Proc. of ISIT*, pages 1863 – 1867, Sept. 2005.

[71] C. E. Shannon. Communication theory of secrecy systems. *Bell Syst.Tech. J.*, 28:656–715, 1948.

[72] C. E. Shannon. Two-way communication channels. In *Proc. 4th Berkeley Symp. Math. Stat. Prob.*, pages 611–644, 1961.

[73] N. Z. Shor. *Monimization Methods for Non-Differentiable Functions.* Springer-Verlag, 1985.

[74] L. Tassiulas. Linear complexity algorithms for maximum throughput in radionetworks and input queued switches. In *Proc. of Infocom*, pages 533–539, March 1998.

[75] L. Tassiulas and A. Ephremides. Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks. *IEEE Trans. Automat. Contr.*, 37(12):1936–1948, Dec. 1992.

[76] D. M. Topkis. Equilibrium points in nonzero-sum n-person submodular games. *SIAM J. of Contr. and Optim.*, 17(6):773–787, 1979.

[77] Danail Traskov, Niranjan Ratnakar, Desmond S. Lun, Ralf Koetter, and Muriel Médard. Network coding for multiple unicasts: An approach based on linear optimization. In *Proc. of IEEE ISIT*, pages 1758–1762, July 2006.

[78] J.E. Wieselthier, G.D. Nguyen, and A. Ephremides. On the construction of energy-efficient broadcast and multicast trees in wireless networks. In *Proc. of Infocom*, pages 585–594, March 2000.

[79] R. Kevin Wood. Deterministic network interdiction. *Mathematical and Computer Modeling*, 17(2):1–18, 1993.

[80] AD Wyner. The wire-tap channel. *Bell Systems Technical Journal*, 54(8):1355–1387, Oct. 1975.

[81] Raymond W. Yeung. A framework for linear information inequalities. *IEEE Trans. Inform. Theory*, 43(6):1924–1934, Nov. 1997.

[82] Raymond W. Yeung. *Information Theory and Network Coding.* Springer, August 2008.

[83] Shengli Zhang, Soung Chang Liew, and Patrick P. Lam. Hot topic: physical-layer network coding. In *Proc. of ACM Mobicom*, pages 358–365, 2006.