Optimal Resource Management in Wireless Access Networks

by

Amir-Hamed Mohsenian-Rad

B.Sc., Electrical Engineering, Amir-Kabir University of Technology, Iran, 2002
M.Sc., Electrical Engineering, Sharif University of Technology, Iran, 2004

A THESIS SUBMITTED IN PARTIAL FULFILMENT OF
THE REQUIREMENTS FOR THE DEGREE OF

Doctor of Philosophy

in

The Faculty of Graduate Studies

(Electrical and Computer Engineering)

The University of British Columbia

(Vancouver)

August, 2008

Abstract

This thesis presents several simple, robust, and optimal resource management schemes for multi-hop wireless access networks with the main focus on multi-channel wireless mesh networks (MC-WMNs). In this regard, various resource management optimization problems are formulated and efficient algorithms are proposed to solve each problem. First, we consider the channel assignment problem in MC-WMNs and formulate different resource management problems within the general framework of network utility maximization (NUM). Unlike most of the previously proposed channel assignment schemes, our algorithms can not only assign the orthogonal (i.e., non-overlapped) channels, but also partially overlapped channels. This better utilizes the available frequency spectrum as a critical resource in MC-WMNs. Second, we propose two distributed random medium access control (MAC) algorithms to solve a non-convex NUM problem at the MAC layer. The first algorithm is fast, optimal, and robust to message loss and delay. It also only requires a limited message passing among the wireless nodes. Using distributed learning techniques, we then propose another NUM-based MAC algorithm which achieves the optimal performance without frequent message exchange. Third, based on our results on random MAC, we develop a distributed multi-interface multi-channel random access algorithm to solve the NUM problem in MC-WMNs. Different from most of the previous channel assignment schemes in the literature, where channel assignment is intuitively modeled in the form of combinatorial and dis-

crete optimization problems, our scheme is based on formulating a novel continuous optimization model. This makes the analysis and implementation significantly easier. Finally, we consider the problem of pricing and monetary exchange in multi-hop wireless access networks, where each intermediate node receives a payment to compensate for its offered packet forwarding service. In this regard, we propose a market-based wireless access network model with two-fold pricing. It uses relay-pricing to encourage collaboration among the access points. It also uses interference-pricing to leverage optimal resource management. In general, this thesis widely benefits from several mathematical techniques as both modeling and solution tools to achieve simple, robust, optimal, and practical resource management strategies for future wireless access networks.

Table of Contents

List of Tables

List of Figures

Abbreviations

ACK	Acknowledgement
ACM	Association for Computing Machinery
CDMA	Code division multiple access
CLICA	Connected low interference channel assignment
CTS	Clear to send
DCF	Distributed coordination function
DMMRA	Distributed multi-interface multi-channel random access
EED	End-to-end delay
FF	Fast fading
IEEE	Institute of Electrical and Electronics Engineers
ILS	Iterated local search
KKT	Karush-Kuhn-Tucker (optimality conditions)
MAC	Medium access control

MCR	Multiple channel reception
MC-WMN	Multi-channel wireless mesh network
NIC	Network interface card
NUM	Network utility maximization
OCICA	Optimal combinatorial interface and channel assignment
PDR	Packet delivery ratio
RTS	Request to send
SCR	Single channel reception
SF	Slow fading
SFP	Single fold pricing
TCP	Transmission control protocol
TFP	Two fold pricing
UDP	User datagram protocol
WMN	Wireless mesh network

Acknowledgements

It is difficult to overstate my gratitude to my supervisor Dr. Vincent Wong for his guidance, patience, and support. I sincerely appreciate the amount of time and effort he invested for teaching me technical writing and introducing me to several exciting research areas.

I am very grateful to Dr. Mung Chiang, Dr. Jianwei Huang, and Dr. Victor Leung for their collaborations and insightful comments and suggestions that helped me during some research directions for my degree. I would also like to express my sincere thanks to Dr. Maurice Queyranne, Dr. Robert Schober, Dr. Vikram Krishnamurthy, Dr. Junshan Zhang, Dr. Lijun Chen, Dr. Dahai Xu, Dr. Yung Yi, Amir Nasri, Enrique Stevense-Navarro, and Shahab Moradi for their helpful suggestions. I also thank my colleagues and friends who assisted and encouraged me along the way. I am especially grateful to Mohammad Mahmoodi Ghidari, Ehsan Bayaki, Mohammad Hossein Bateni, Majid Khabbazian, Vahid Shah-Mansouri, Keivan Ronasi, and Mani Malekesmaeili.

Lastly, and most importantly, I wish to thank my lovely wife, Hoda, my parents, Mahdi and Alieh Mohsenian-Rad, my parents-in-law, Bagher and Fatemeh Mojazi-Amiri, my sister, Neda, my brother and his wife, Amir-Farzad and Lida, and my sister-in-law, Elham, for their love, support, and encouragement. To them I dedicate this thesis.

This work was supported by the Natural Sciences and Engineering Research Council (NSERC) of Canada, Bell Canada, Pacific Century Graduate Scholorship from the Ministry of Advanced

Education of British Columbia Provincial Government, and the University of British Columbia Graduate Fellowship.

Chapter 1

Introduction

Optimization-based approaches have been used extensively over the past several years to study various resource management problems in communications and computer networks. Example includes the Internet congestion control which is modeled in the general form of a distributed system to solve a network-wide optimization problem to maximize the aggregate network utility. These approaches can systematically model the complicated issues in the networking systems and result in deep understandings of the existing network protocols such as the transmission control protocol (TCP) [1].

The main focus of this thesis is on formulating various resource management schemes for wireless access networks in the form of tractable optimization problems within the general framework of network utility maximization and also proposing simple and practical algorithms to solve the formulated problems. The rest of this chapter is organized as follows. We first provide an overview of the structure and the existing resource management problems in wireless access networks with the main focus on multi-channel wireless mesh networks in Section 1.1. We then list most of the mathematical and analytical tools that we use throughout this thesis in Section 1.2. Next, we summarize the our contributions and outline the results in Section 1.3. The list of related publications is given in Section 1.4. Finally, we describe the organization of the thesis in Section 1.5.

1.1 Wireless Access Networks

1.1.1 Structure

Wireless access networks have recently received an increasing attention, especially under the context of wireless mesh networks (WMNs), where the wireless network is *multi-hop* and offers ubiquitous and inexpensive Internet access to various wireless users within a wide coverage area [2, 3]. A sample wireless mesh network is shown in Fig. 1.1. The network consists of a number of wireless stationary access points and several wireless users. In the context of WMNs, the access points are called wireless mesh routers while the wireless users are called the wireless mesh clients. The access points may also be referred to as wireless relay nodes. The wireless access points form a fully wireless backbone network to provide the connectivity service to the users. Some of them also act as gateways to the Internet via high-speed wired links. Users first transfer data to their associated access points, and these data are then transferred to the Internet or other users via the intermediate access points in a multi-hop manner.

The research and development of wireless access and mesh networks are motivated by several applications including broadband home networking, community and neighborhood networking, enterprise networking, metropolitan area networking, and building automation [2]. Some vendors have recently begun to offer products in this area [4, 5, 6, 7]. The IEEE has also set up the 802.11s task group for mesh networking [8].

For the wireless access networks, interference is a major issue which can significantly limit the aggregate capacity of the network. As a practical solution, the capacity of the IEEE 802.11 a/b/g-based WMNs [9, 10] can be substantially improved via the use of multiple network interface cards (NICs) and multiple orthogonal frequency channels [11]. In this scenario, each access point

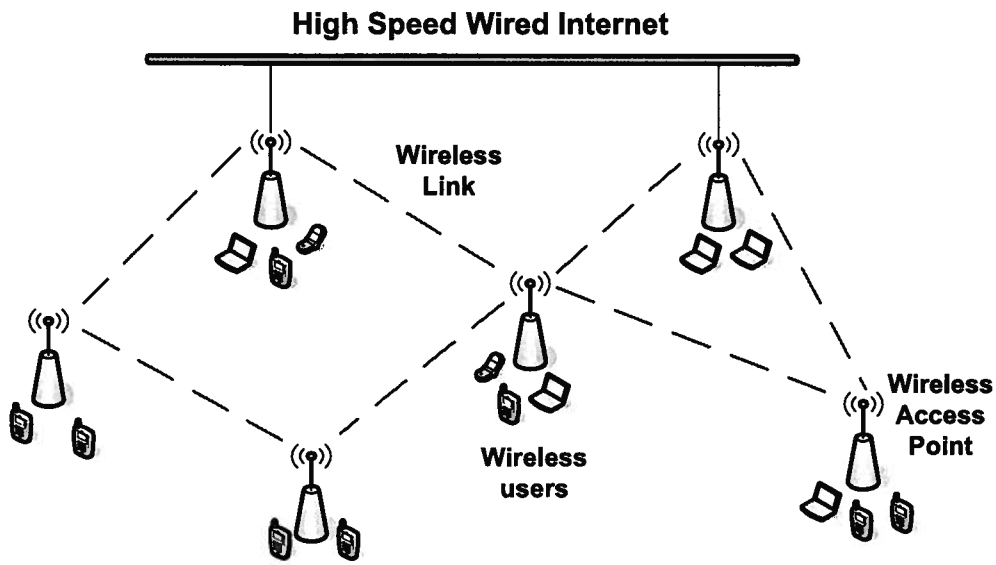


Figure 1.1: A sample multi-hop wireless access network with six wireless access points (also called wireless relay nodes or wireless mesh routers), and 15 wireless users (also called wireless mesh clients).

is equipped with multiple NICs. Each NIC is then assigned to a distinct frequency channel. A pair of neighboring access points can communicate with each other as long as one of their NICs uses the same channel. On the other hand, those transmissions which take place over two different (orthogonal) frequency channels do not interfere with each other.

1.1.2 Resource Management

The number of available channels depends on the frequency band. For example, the 802.11 b/g standards have 11 channels within the 2.4 GHz frequency band, of which 3 channels are orthogonal (non-overlapping). The 802.11a standard has 79 operating channels in North America, of which 12 channels are orthogonal. Notice that the key benefit of using multiple NICs is to provide independent simultaneous transmissions via different NICs. The number of NICs per each access point is usually limited to 2 or 3.

The limited number of frequency channels implies that some transmissions may be required to operate over the same channels. The limited number of NICs also avoids having several simultaneous transmissions. These limit the performance gain of the multi-channel deployments and make it critical to design intelligent interface assignment and channel allocation (also called channel assignment) schemes. Interface assignment determines which wireless links should operate on each NIC. Channel allocation determines which frequency channel should be assigned to each wireless link. Other resource management problems include medium access control, transmission power control, routing, and congestion control.

In this thesis, we apply optimization theory to solve the mentioned resource management problems. Our main focus is on cross-layer optimization where joint designs are given for multiple resource management problems which affect different layers in the network layering model [12]. A review of the work related to each resource management problem of interest is given later in each chapter. Tutorial papers on cross-layer design can be found in [13] and [14].

1.2 Mathematical Foundation

In this thesis, we extensively employed several powerful mathematical tools from *optimization theory*, *theory of parallel and distributed computing*, and *utility theory*. Optimization theory is a mature field that has experienced major development in the last twenty years [15, 16]. In particular, the theory of convex optimization has been widely accepted as both modeling language and solution tool in complex communication and networking problems [17]. The theory of parallel and distributed computing is also a promising field that can be brought to bear on many important problems in both parallel computers and distributed data communication networks

[18]. It can particularly be helpful in various convergence and robustness analysis studies. This thesis also benefits from the recent developments in using utility theory in networking problems which is indeed a promising research area towards establishing a mathematical theory for network architectures [19]. In this section, we provide an overview of each of these three areas.

1.2.1 Optimization Theory

Consider the following problem:

$$\begin{aligned} \underset{\boldsymbol{x}\in\mathcal{X}}{\text{minimize}} \quad & f_0(\boldsymbol{x}) \\ \text{subject to} \quad & f_i(\boldsymbol{x}) \leq 0, \quad \forall i = 1, \ldots, N_\text{i} \\ & h_i(\boldsymbol{x}) = 0, \quad \forall i = 1, \ldots, N_\text{e} \end{aligned} \tag{1.1}$$

which describes the problem of finding an $\boldsymbol{x}$ that minimizes function $f_0(\boldsymbol{x})$ among all *feasible points*; i.e., all $\boldsymbol{x} \in \mathcal{X}$ that satisfy the conditions $f_i(\boldsymbol{x}) \leq 0$ for all $i = 1, \ldots, N_\text{i}$, and $h_i(\boldsymbol{x}) = 0$, for all $i = 1, \ldots, N_\text{e}$. Here $\boldsymbol{x} = (x_i, \forall\, i = 1, \ldots, N) \in \mathcal{X}$ denotes the vector of *optimization variables* where $\mathcal{X} \subseteq \boldsymbol{R}^N$. The function $f_0 : \boldsymbol{R}^N \to \boldsymbol{R}$ is called the *objective function* or *cost function.* The inequalities $f_i(\boldsymbol{x}) \leq 0$ for $i = 1, \ldots, N_\text{i}$ are called the *inequality constraints*, and the corresponding functions $f_i : \boldsymbol{R}^N \to \boldsymbol{R}$ for $i = 1, \ldots, N_\text{i}$ are called the *inequality constraint functions.* The equations $h_i(\boldsymbol{x}) = 0$ for $i = 1, \ldots, N_\text{e}$ are called the *equality constraints*, and the functions $h_i : \boldsymbol{R}^N \to \boldsymbol{R}$ for $i = 1, \ldots, N_\text{e}$ are called the *equality constraint functions.* If there are no constraints (i.e., $N_\text{i} = N_\text{e} = 0$), then we say that problem (1.1) is *unconstrained.* Generally speaking, optimization theory provides the mathematical tools to model and solve *optimization problems* in the general form of problem (1.1). Notice that any maximization problem can also

be written in the form of (1.1) if we multiply its objective function by -1. Thus, we only focus on minimization problems in this section. The results can be extended to maximization problems accordingly. Next, we describe four important classes of optimization problems, namely linear, convex, integer, and mixed-integer optimization problems.

Linear Optimization

An important class of optimization problems is the *linear* optimization problems. Problem (1.1) reduces to a linear optimization problem if $\mathcal{X} = \boldsymbol{R}^N$ and for each vector $\boldsymbol{x} \in \mathcal{X}$, we have:

$$f_i(\boldsymbol{x}) = \boldsymbol{a}^T\boldsymbol{x} + b, \qquad \forall i = 0, \ldots, N_{\mathrm{i}}, \tag{1.2}$$

$$h_i(\boldsymbol{x}) = \boldsymbol{c}^T\boldsymbol{x} + d, \qquad \forall i = 1, \ldots, N_{\mathrm{e}}, \tag{1.3}$$

where $\boldsymbol{a}, \boldsymbol{c} \in \boldsymbol{R}^N$ and $b, d \in \boldsymbol{R}$. Linear optimization problems can be solved in *polynomial time.* That is, the run time is no greater than a polynomial function of the problem size $N + N_{\mathrm{i}} + N_{\mathrm{e}}$ and the number of bits used to represent each variable. Various techniques can be used to solve linear problems such as *simplex method* [20] and *interior point method* [21]. Recent textbooks on linear optimization theory include [22, 23, 24].

Convex Optimization

The optimization problem (1.1) is called a *convex* optimization problem if $\mathcal{X} = \boldsymbol{R}^N$, condition (1.3) holds, and for any choice of vectors $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{X}$ and any scalar $0 \leq \theta \leq 1$, we have:

$$f_i(\theta\boldsymbol{x} + (1-\theta)\boldsymbol{y}) \leq \theta f_i(\boldsymbol{x}) + (1-\theta) f_i(\boldsymbol{y}), \qquad \forall i = 0, \ldots, N_{\mathrm{i}}. \tag{1.4}$$

The above inequality implies that the objective function and all the inequality constraint functions are indeed *convex functions* [16, pp. 67]. Notice that any linear function is also a convex function. Thus, all linear optimization problems are also convex optimization problems. If problem (1.1) is a maximization problem, then it is a convex problem as long as (1.4) holds for $-f_0$; rather than f_0. That is, if the objective function f_0 is *concave.* Some example convex and concave functions are listed in [16, Chapter 3].

In a minimization problem as in (1.1), a vector $\boldsymbol{x}^\star \in \mathcal{X}$ is called a *local minimum* if it is no worse (i.e., greater) than its *neighbors* feasible points. The vector $\boldsymbol{x}^\star \in \mathcal{X}$ is also *global minimum* if it is less than *all* feasible points. That is, $f_i(\boldsymbol{x}^\star) \leq f_i(\boldsymbol{x})$ for all $\boldsymbol{x} \in \mathcal{X}$ that satisfy the conditions $f_i(\boldsymbol{x}) \leq 0$ for all $i = 1, \ldots, N_\mathrm{i}$, and $h_i(\boldsymbol{x}) = 0$, for all $i = 1, \ldots, N_\mathrm{e}$. Recent textbooks on convex optimization theory include [15, 16, 25].

Theorem 1 *In a convex optimization problem, any locally optimal point $\boldsymbol{x}^\star \in \mathcal{X}$ is indeed globally optimal [16, pp. 138].*

We define the *Lagrangian* $L : \boldsymbol{R}^N \times \boldsymbol{R}^{N_\mathrm{i}} \times \boldsymbol{R}^{N_\mathrm{e}} \rightarrow \boldsymbol{R}$ associated with problem (1.1) as:

$$L(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\nu}) = f_0(\boldsymbol{x}) + \sum_{i=1}^{N_\mathrm{i}} \lambda_i f_i(\boldsymbol{x}) + \sum_{i=1}^{N_\mathrm{e}} \nu_i h_i(\boldsymbol{x}), \tag{1.5}$$

where λ_i is defined as the *Lagrange multiplier* associated with the ith inequality constraint $f_i(\boldsymbol{x}) \leq 0$ for any $i = 0, \ldots, N_\mathrm{i}$ and ν_i is defined as the Lagrange multiplier associated with the ith equality constraint $h_i(\boldsymbol{x}) = 0$ for any $i = 0, \ldots, N_\mathrm{e}$. The vectors $\boldsymbol{\lambda} = (\lambda_i,\ i = 0, \ldots, N_\mathrm{i})$ and $\boldsymbol{\nu} = (\nu_i,\ i = 0, \ldots, N_\mathrm{e})$ are called the *dual variables* or *Lagrange multiplier vectors* associated with the optimization problem in (1.1). We can also define the *dual function* $g : \boldsymbol{R}^{N_\mathrm{i}} \times \boldsymbol{R}^{N_\mathrm{e}} \rightarrow \boldsymbol{R}$

as the minimum value of the Lagrangian $L(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\nu})$ over $\boldsymbol{x}$. We have:

$$g(\boldsymbol{\lambda}, \boldsymbol{\nu}) = \inf_{\boldsymbol{x} \in \mathcal{X}} L(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\nu}), \quad \forall \boldsymbol{\lambda} \in \boldsymbol{R}^{N_\mathrm{i}},\ \boldsymbol{\nu} \in \boldsymbol{R}^{N_\mathrm{e}}. \tag{1.6}$$

The *dual problem* associated with problem (1.1) is now defined as:

$$\begin{aligned} \underset{\boldsymbol{\lambda} \in \boldsymbol{R}^{N_\mathrm{i}}, \boldsymbol{\nu} \in \boldsymbol{R}^{N_\mathrm{e}}}{\text{maximize}} \quad & g(\boldsymbol{\lambda}, \boldsymbol{\nu}) \\ \text{subject to} \quad & \lambda_i \geq 0, \quad \forall i = 1, \ldots, N_\mathrm{i}. \end{aligned} \tag{1.7}$$

Notice that since the dual function $g(\boldsymbol{\lambda}, \boldsymbol{\nu})$ is the pointwise infimum of a family of affine (Lagrangian) functions of $(\boldsymbol{\lambda}, \boldsymbol{\nu})$, it is a concave function. Since the inequality functions $\lambda_i \geq 0$ for any $i = 1, \ldots, N_\mathrm{i}$ are linear, the dual problem in (1.7) is indeed a convex optimization problem even if the *primal problem* in (1.1) is *not* convex.

We can show the following key result.

Theorem 2 *Let $p^\star = f_0(\boldsymbol{x}^\star)$ denote the global minimum of the primal problem (1.1). Also let $d^\star$ denote the global maximum of the dual problem (1.7). We have [16, pp. 225]:*

$$d^\star \leq p^\star. \tag{1.8}$$

The inequality in (1.8) is called weak duality. *It holds even if the primal problem in (1.1) is* not *convex. On the other hand, if problem (1.1) is convex, then we have [16, pp. 226]:*

$$d^\star = p^\star. \tag{1.9}$$

The equality in (1.9) is called strong duality.

The difference $p^\star - d^\star$ is always non-negative and is called the *duality gap.* From Theorem 2, the duality gap is *zero* for convex optimization problems. Theorems 1 and 2 show two fundamental properties of the convex optimization problems. Using these properties, various schemes can be used to solve convex optimization problems, including *gradient projection methods*, *interior point method*, and *primal-dual method* [16].

There are also some important optimization problems, called *geometric optimization problems*, which are *not* convex; but can be *converted* to equivalent convex optimization problems using *logarithmic* change of variables. Details are given in [26] and [27]. Some other non-convex optimization problems which can be converted to convex problems are described in [28, 29, 30].

Integer and Mixed-Integer Optimization

Linear and convex optimization problems are two important classes of *continuous* optimization problems over *real-valued* optimization variables. In this section, we summarize two other important classes of optimization problems called *integer optimization problems* and *mixed-integer optimization problems.* Problem (1.1) reduces to an integer optimization problem if $\mathcal{X} \subseteq \boldsymbol{Z}^N$ where $\boldsymbol{Z}^N$ is the set of all $N \times 1$ integer vectors; i.e., vectors with integer entries. Similarly, a mixed-integer optimization problem is an optimization problem in which *some* of the optimization variables are integer-valued, while the rest are real-valued. Integer and mixed-integer optimization problems are *linear* if conditions (1.2) and (1.3) hold. Integer and mixed-integer optimization problems are usually difficult to solve. In particular, they cannot be solved within polynomial time. In fact, the complexity of integer and mixed-integer optimization problems is *NP*-hard. That is, finding the optimal solution may require examining all the feasible points

using *exhaustive search.* Nevertheless, there are efficient commercial computer codes such as CPLEX [31] and MOSEK [32] that can solve integer and mixed-integer problems using *branch-and-bound* [33] and *branch-and-cut* methods [34]. In general, solving *linear* integer and *linear* mixed-integer optimization problems is computationally easier compared to solving *non-linear* integer/mixed-integer problems.

Binary Linearization

An important class of integer problems is *binary optimization problems* where the optimization variables can only take the values of 0 and 1. That is,

$$x_i \in \{0, 1\}, \quad \forall i = 1, \ldots, N. \tag{1.10}$$

The binary optimization problems are especially important as they can be used to model various *decision making problems*: optimization variable $x_i = 1$ if we should perform an action and $x_i = 0$ if we should not perform the action. In this section, we explain how we can convert several non-linear binary and mixed-binary optimization problems to the *equivalent* linear problems using the *exact binary optimization schemes.*

Theorem 3 *Consider two binary variables x_1 and x_2. Their product (i.e., the non-linear quadratic term $x_1 x_2$) can be replaced by a new binary auxiliary variable π, such that its value corresponds*

to the values of x_1 and x_2 as follows:

$$\pi = \begin{cases} 0, & \text{if } x_1 = 0,\ x_2 = 0, \\ 0, & \text{if } x_1 = 0,\ x_2 = 1, \\ 0, & \text{if } x_1 = 1,\ x_2 = 0, \\ 1, & \text{if } x_1 = 1,\ x_2 = 1. \end{cases} \tag{1.11}$$

The desired correspondence is obtained by requiring that $\pi \in \{0, 1\}$ and we have [35]:

$$\begin{aligned} x_1 + x_2 - \pi &\leq 1, \\ -x_1 - x_2 + 2\pi &\leq 0. \end{aligned} \tag{1.12}$$

We can also show the following for mixed-binary products.

Theorem 4 *Consider a binary variable x_b and a non-negative real variable x_r. Assume that r_{max} is an upper bound for the real variable x_r. The quadratic term $x_b x_r$ can be replaced by a new non-negative real auxiliary variable υ, such that its value corresponds to the values of x_b and x_r as follows:*

$$\upsilon = \begin{cases} 0, & \text{if } x_b = 0, \\ x_r, & \text{if } x_b = 1. \end{cases} \tag{1.13}$$

The desired correspondence is obtained by simply requiring that [36]:

$$\begin{aligned} 0 \leq \upsilon \leq x_r &\leq r_{max}, \\ x_r - r_{max}\,(1 - x_b) \leq \upsilon &\leq r_{max}\, x_b. \end{aligned} \tag{1.14}$$

Theorem 5 *Now consider two binary variables x_b and y_b and a non-negative real variable x_r. Assume that r_{max} is an upper bound for the real variable x_r. We define $\sigma = x_r x_b y_b$. The desired correspondence between σ and variables x_b, y_b, and x_r is obtained by simply requiring that [37]:*

$$\begin{aligned} 0 \leq \sigma \leq r_{max} x_b, \\ 0 \leq \sigma \leq r_{max} y_b, \\ r_{max}(x_b + y_b - 2) + x_r \leq \sigma \leq r_{max}(2 - x_b - y_b) + x_r. \end{aligned} \tag{1.15}$$

Theorems 3, 4 and 5, together provide a toolkit of various linearization techniques which can help us to convert any non-linear binary or mixed-binary optimization problem with *polynomial non-linearity* to the equivalent linear optimization problems. Further binary linearization techniques can be found in [38, 39, 40, 41, 42, 43].

1.2.2 Network Utility Maximization

Historically, most of the network protocols in the layered architectures have been designed based on a heuristic or ad-hoc basis, making the analysis sometimes difficult. A decade ago, Kelly *et al.* [44] proposed a new approach of optimization-based modeling and decomposition-based solutions using *utility theory* [45] to simplify the understanding of the complex interactions in network congestion control. Since then, this approach has been extended substantially in many ways, forming a promising direction towards a mathematical theory of network architectures. In this regard, the overall network is modeled as a *network utility maximization* (NUM) problem, where each layer corresponds to a decomposed subproblem, and the interfaces among layers are quantified as functions of the optimization variables coordinating the subproblems. An interesting

survey on recent developments in formulating and solving various NUM problems is given in [19].

NUM problem is usually formulated either at the *network layer* (cf. [44, 46, 47, 48, 49, 50, 51, 52] or at the *link layer* (cf. [53, 54, 55, 56, 57]). The former is mostly used to design congestion control algorithms while the latter is considered to design medium access control (MAC) protocols. Assume a network with $\mathcal{N}$ as the set of nodes and $\mathcal{L}$ as the set of links. For each link $l \in \mathcal{L}$, we define $c_l(\boldsymbol{x})$ as the capacity of link l, where $\boldsymbol{x}$ denotes the vector of all resource management variables. For example, $\boldsymbol{x}$ may include the vector of all transmission powers, persistent probabilities, routing settings, etc. Let $\mathcal{S}$ denote the set of all end-to-end flows. For each flow $s \in \mathcal{S}$, let $r_s(\boldsymbol{x})$ denote the transmission rate of flow s. We also define $\mathcal{L}_s(\boldsymbol{x})$ as the set of links along the routing path of flow s. The NUM problem at the network layer can be defined as [44, 58]:

$$\begin{aligned} \underset{\boldsymbol{x}\in\mathcal{X}}{\text{maximize}} \quad & \sum_{s\in\mathcal{S}} u_s(r_s(\boldsymbol{x})) \\ \text{subject to} \quad & \sum_{s:l\in\mathcal{L}_s} r_s(\boldsymbol{x}) \leq c_l(\boldsymbol{x}), \quad \forall\, l \in \mathcal{L}, \end{aligned} \tag{1.16}$$

where for each flow $s \in \mathcal{S}$, the *utility* u_s is an increasing and concave function of the data rate r_s and indicates flow s's degree of satisfaction on its data rate. Notice that since the data rates are functions of the resource management variables $\boldsymbol{x}$, the utilities $u_s(r_s(\boldsymbol{x}))$ are also functions of $\boldsymbol{x}$ for any $s \in \mathcal{S}$. Different utility functions can be considered to achieve different design objectives. In particular, the utility functions can be α-fair [50]:

$$u_s(r_s(\boldsymbol{x})) = \begin{cases} (1-\alpha)^{-1}\, r_s(\boldsymbol{x})^{1-\alpha}, & \text{if } \alpha \neq 1, \\ \log r_s(\boldsymbol{x}), & \text{if } \alpha = 1, \end{cases} \qquad \forall s \in \mathcal{S}, \tag{1.17}$$

where $\alpha > 0$ is a design parameter, called *fairness index*. Using (1.17), a wide range of efficient

and fair resource management objectives can be modeled. In particular, problem (1.16) reduces to throughput maximization with $\alpha \to 0$, to proportional fair allocation with $\alpha = 1$, to harmonic mean fair allocation with $\alpha = 2$, and to max-min fairness with $\alpha \to \infty$.

Similarly, we can define the NUM problem at the link layer as follows [53]:

$$\underset{\boldsymbol{x} \in \mathcal{X}}{\text{maximize}} \quad \sum_{l \in \mathcal{L}} u_l(c_l(\boldsymbol{x})). \tag{1.18}$$

Assuming that the α-fair utility functions are being used, similar to problem (1.16), problem (1.18) can be solved to obtain various fair resource management objectives among the link-layer flows. Notice that both problems (1.16) and (1.18) can be written in the general form of the constrained optimization problem in (1.1). For example, considering the NUM problem in the network layer, we have:

$$f_0(\boldsymbol{x}) = \textstyle\sum_{s \in \mathcal{S}} u_s(r_s(\boldsymbol{x})), \tag{1.19}$$

$$f_1(\boldsymbol{x}) = \textstyle\sum_{s: l \in \mathcal{L}_s} r_s(\boldsymbol{x}) - c_l(\boldsymbol{x}). \tag{1.20}$$

Almost any network resource management problem can be formulated as a NUM problem in form of either (1.16) or (1.18). Various decomposition techniques can then be used to decompose the formulated NUM problem into several small subproblems to be solved in a distributive fashion at each network node. The *decomposition techniques* include *primal decomposition*, *dual decomposition*, and *indirect decomposition*. Survey papers on decomposition techniques can be found in [59] and [60].

1.2.3 Distributed Systems and Convergence Analysis

The theory of parallel and distributed computing is also a promising field that can be brought to bear on many important problems in both parallel computers and distributed data communication networks [18]. In particular, it can be helpful in various convergence and robustness analysis studies in wired as well as wireless networking systems. As in Section 1.2.2, let $\boldsymbol{x}$ denote the vector of all resource management variables. Also let $\{t_1, t_2, t_3, \ldots\}$ denote the set of time instances at which the resource management variables are being updated. We consider the following general *update formulation*:

$$x_i(t_{k+1}) = g_i(\boldsymbol{x}(t_k)), \qquad \begin{array}{l} \forall i = 1, \ldots, N, \\ \forall k = 1, 2, 3, \ldots, \end{array} \tag{1.21}$$

where for any $i = 1, \ldots, N$, function g_i is called the *mapping function* associated with resource management variable x_i. In general, we are interested in those mappings $\boldsymbol{g} = (g_i, \forall i = 1, \ldots, N)$ such that the update equations in (1.21) converge towards the optimal solution of the NUM problems in (1.16) or (1.18) or any other resource management optimization problem in the general form of problem (1.1).

Important Mappings

A mapping function $\boldsymbol{g} : \mathcal{X} \to \mathcal{X}$ is a *contraction mapping* if we have [18, pp. 181]:

$$\|\boldsymbol{g}(\boldsymbol{x}) - \boldsymbol{g}(\boldsymbol{y})\| \leq \gamma \|\boldsymbol{x} - \boldsymbol{y}\|, \quad \forall \boldsymbol{x}, \boldsymbol{y} \in \mathcal{X}, \tag{1.22}$$

where $\|\cdot\|$ is some *norm* and scalar $\gamma \in [0,1)$ is called *modulus* of mapping $\boldsymbol{g}$. On the other hand, mapping $\boldsymbol{g}$ is a *monotone increasing mapping* if we have [18, pp. 191]:

$$\boldsymbol{x} \preceq \boldsymbol{y} \quad \Rightarrow \quad \boldsymbol{g}(\boldsymbol{x}) \preceq \boldsymbol{g}(\boldsymbol{y}), \qquad \forall\, \boldsymbol{x}, \boldsymbol{y} \in \mathcal{X}, \tag{1.23}$$

and is a *monotone decreasing mapping* if,

$$\boldsymbol{x} \preceq \boldsymbol{y} \quad \Rightarrow \quad \boldsymbol{g}(\boldsymbol{x}) \succeq \boldsymbol{g}(\boldsymbol{y}), \qquad \forall\, \boldsymbol{x}, \boldsymbol{y} \in \mathcal{X}. \tag{1.24}$$

Here $\preceq$ and *succeq* are interpreted coordinatewise.

Convergence Theorems

For each mapping $\boldsymbol{g} : \mathcal{X} \to \mathcal{X}$, the vector $\boldsymbol{x}^* \in \mathcal{X}$ is a *fixed point* if we have [18, pp. 181]:

$$\boldsymbol{g}(\boldsymbol{x}^*) = \boldsymbol{x}^*. \tag{1.25}$$

In general, we are interested to know (a) whether there exists any fixed point (*existence*), (b) whether the fixed point is unique (*uniqueness*), and (c) whether the mapping can reach its fixed point starting from any arbitrary initial point (*convergence*). It is then ideal if we have: $\boldsymbol{x}^* = \boldsymbol{x}^\star$, where $\boldsymbol{x}^\star$ is the optimal solution for the optimization problem of interest.

Theorem 6 *Assume that $\boldsymbol{g} : \mathcal{X} \to \mathcal{X}$ is a contraction mapping. Then, the mapping $\boldsymbol{g}$ has a unique fixed point and for every initial vector $\boldsymbol{x}(0) \in \mathcal{X}$, it synchronously converges to its unique fixed point $\boldsymbol{x}^*$ geometrically fast [18, Proposition 1.1, Chapter 3].*

Theorem 6 provides results on *synchronous* convergence of contraction mapping. We can also show the following results on *asynchronous convergence.*

Theorem 7 *Assume that $\boldsymbol{g} : \mathcal{X} \to \mathcal{X}$ is a contraction mapping with respect to any weighted*

maximum norm:

$$\|\boldsymbol{x}\|_{\infty}^{w} = \max_{i} \frac{|x_i|}{w_i}, \tag{1.26}$$

where $\boldsymbol{w} \in \boldsymbol{R}^N$ *is an* $N \times 1$ *vector with positive entries. Then, the mapping* $\boldsymbol{g}$ *has a unique fixed point and for every initial vector* $\boldsymbol{x}(0) \in \mathcal{X}$*, it asynchronously converges to its unique fixed point* $\boldsymbol{x}^*$ *geometrically fast [18, pp. 434].*

The asynchronous convergence is of special importance as it implies *robustness* and less *co-ordination overhead.* Finally, we have the following results on the asynchronous convergence of monotone mappings.

Theorem 8 *Assume that* $\boldsymbol{g} : \mathcal{X} \rightarrow \mathcal{X}$ *is a monotone mapping (i.e., either increasing or decreasing). Set* $\mathcal{X}$ *is assumed to be bounded. Then, for every initial vector* $\boldsymbol{x}(0) \in \mathcal{X}$*, the mapping* $\boldsymbol{g}$ *will asynchronously converge to one of its fixed points [18, pp. 445-446].*

Similar to Theorem 7, Theorem 8 provides convergence results for asynchronous updates. However, monotone mappings may have *multiple fixed points.* Depending on the initial point $\boldsymbol{x}(0)$, the mapping may converge to different fixed points. Combining Theorems 6 and 8, a monotone mapping which is also a contraction mapping with respect to an *arbitrary* norm is guaranteed to *asynchronously* converge to its *unique* fixed point.

1.3 Summary of Results and Contributions

This thesis covers several resource management problems in wireless access networks with special focus on MC-WMNs. The results are divided into seven chapters. The contributions in each chapter are as follows.

- Chapter 2 considers the problem of joint optimal design of topology formation, interface assignment, channel allocation, and routing for MC-WMNs as a unified a cross-layer linear optimization problem to maximize the network performance in a centralized and permanent fashion. Structural results and numerical algorithms are derived. Extensive simulations show that the proposed joint design results in significantly higher aggregate network throughput and lower transmission delays compared to two recently proposed algorithms in the literature.

- Chapter 3 formulates the interface channel assignment problems jointly with MAC in the NUM framework. The formulated problems, with α-fair utility functions, achieve various fair and efficient resource management objectives. An optimal design, based on exact binary linearization techniques, is proposed which leads to a global maximum. A near-optimal design, based on approximate dual decomposition techniques, is also proposed which is practical for distributed deployment. Numerical results show that our proposed designs can lead to MC-WMNs which are more efficient and fair compared to their single-channel counterparts. The performance gain on both efficiency and fairness increase as the number of available NICs per router or the number of available channels increases.

- Chapter 4 addresses the problem of assigning partially-overlapped channels in MC-WMNs and provides the required modeling tools accordingly. The proposed modeling is different from the previous channel assignment models in the literature, where only the orthogonal (i.e., non-overlapped) channels are being used. Simulation results show that the network capacity can increase up to 90% when all partially overlapped channels are being used.

- Chapter 5 is different from the previous chapters and addresses the problem of designing

simple, robust, and optimal random access for multi-hop wireless networks. In this chapter, we propose two distributed contention-based MAC algorithms to solve a NUM problem. Our algorithms overcome four important performance bottlenecks in the previous NUM-based random access algorithms in the literature. First, only limited message passing among nodes is required. The complexity reduction is in the order of 10. Second, fully asynchronous updates of contention probabilities are allowed. Furthermore, our algorithms are robust to arbitrary large message passing delays and loss. Third, we do not utilize any stepsize during updates, thus our algorithms can achieve faster convergence. Finally, our proposed algorithms have provable convergence, optimality, and robustness properties under a wider range of utility functions, even if the NUM problem is non-convex. The analysis techniques we propose in this chapter are general and can be used to tackle other non-convex optimization problems in communications and networking. Simulation results show the optimality and fast convergence of our algorithms, performance improvements compared with the subgradient-based MAC, and better efficiency-fairness tradeoff compared with the IEEE 802.11 distributed coordination function.

- Chapter 6 gives a positive answer to the following question: *Is it possible to design a NUM-based MAC algorithm that can achieve the optimal performance without frequent explicit message passing?* Compared with the related algorithms in the literature, our proposed algorithm in this chapter achieves the optimal performance without frequent explicit message passing among wireless users. This is of critical importance in practice, since any explicit message passing among wireless users will lead to further contentions in the network and reduce the network performance. We prove the convergence of our algorithm under the as-

sumption that the users can estimate the required information through local observation of the shared wireless medium with asymptotically converging estimation errors. This includes the important case where the underlying channel is lossy and thus not every transmission can be correctly decoded. When the channel is perfect, our algorithm converges to the global optimal solution of the NUM problem. Simulation results show the optimality and fast convergence of our algorithm, and better efficiency-fairness tradeoff compared with the IEEE 802.11 distributed coordination function.

- Chapter 7 considers the problem of designing a distributed and optimal multi-interface multi-channel random access algorithm based on the results from the previous chapters. In general, most of the recently proposed channel and interface assignment algorithms are based on formulating combinatorial optimization problems and discrete optimization. The key is to assign exactly one channel to each NIC. However, as we show in this chapter, combinatorial channel assignment models may result in computationally complicated algorithms and inefficient utilization of the available frequency spectrum. In this chapter, we revisit channel assignment problem by formulating a novel continuous multi-interface multi-channel random access model. This includes elaborate modeling of the link data rates for various multi-interface multi-channel networking scenarios. We then propose a fast, fully distributed and easy to implement multi-interface multi-channel random access algorithm. Simulation results show that our proposed algorithm significantly outperforms combinatorial channel assignment algorithms in terms of achieved network utility and throughput.

- Chapter 8 considers the problem of pricing in multi-hop wireless access networks. In a multi-hop wireless access network, where each node is an independent self-interested com-

mercial entity, pricing is helpful not only to encourage collaboration but also to utilize the network resources efficiently. In this chapter, we propose a market-based model with *two-fold pricing* (TFP) for wireless access networks. In our model, the *relay-pricing* is used to encourage nodes to forward each other's packets. That is, each node receives a payment as a compensation for the relay service it provides. We also consider *interference-pricing* to leverage optimal resource allocation. Together, the relay and the interference prices incorporate both cooperative and competitive interactions among the nodes. We prove that TFP guarantees *positive* profit for each individual wireless node for a wide range of pricing functions. The profit increases as the node forwards more packets. Thus, the cooperative nodes are well rewarded. We then determine the relay and the interference pricing functions such that the network social welfare and the aggregate network utility are maximized. Simulation results show that, compared to a recently proposed *single-fold pricing* (SFP) model where only the relay prices are considered, our proposed TFP scheme significantly increases the total network profit and the network throughput. TFP also leads to more fair revenue sharing and profit distribution among the wireless nodes.

1.4 List of Publications

The following publications have been completed based on the work in this thesis. In some cases the conference papers contain materials overlapping with the journal papers.

Journal Papers

- Amir-Hamed Mohsenian-Rad, Vincent W.S. Wong, "Joint Logical Topology Design, Interface Assignment, Channel Allocation, and Routing for Multi-Channel Wireless Mesh

Networks," *IEEE Trans. on Wireless Communications*, vol. 6, no. 12, pp. 4432-4440, December 2007.

- Amir-Hamed Mohsenian-Rad, Vincent W.S. Wong, "Cross-layer Fair Bandwidth Sharing for Multi-Channel Wireless Mesh Networks," accepted for publication in *IEEE Trans. on Wireless Communications*, 2008.

- Amir-Hamed Mohsenian-Rad, Jianwei Huang, Mung Chiang, Vincent W.S Wong, "Utility Optimal Random Access: Reduced Complexity, Fast Convergence, and Robust Performance," accepted (pending to minor revision) for publication in *IEEE Trans. on Wireless Communications*, 2008.

- Amir-Hamed Mohsenian-Rad, Jianwei Huang, Mung Chiang, Vincent W.S Wong, "Utility Optimal Random Access: Optimal Performance without Frequent Explicit Message Passing," accepted (pending to minor revision) for publication in *IEEE Trans. on Wireless Communications*, 2008.

- Amir-Hamed Mohsenian-Rad, Vincent W.S. Wong, Victor C.M. Leung, "Two-Fold Pricing to Guarantee Individual Profits and Maximum Social Welfare in Wireless Access Networks," submitted to *IEEE Trans. on Wireless Communications*, 2008.

- Amir-Hamed Mohsenian-Rad, Vincent W.S. Wong, "Distributed Multi-Interface Multi-Channel Random Access Using Convex Optimization," submitted to *IEEE Trans. on Mobile Computing*, 2008.

- Amir-Hamed Mohsenian-Rad, Vincent W.S. Wong, "Congestion-Aware Channel Assignment for Multi-Channel Wireless Mesh Networks," submitted to *Computer Networks*, 2008.

Conference Papers

- Amir-Hamed Mohsenian-Rad, Vincent W.S. Wong, "Joint Optimal Channel Assignment and Congestion Control in Multi-Channel Wireless Mesh Networks," in *Proc. of IEEE International Conference on Communications (ICC)*, Istanbul, Turkey, June 2006.

- A. Hamed Mohsenian-Rad, Vincent W.S. Wong, "Logical Topology Design and Interface Assignment for Multi-Channel Wireless Mesh Networks," in *Proc. of IEEE Global Telecommunication Conference (GLOBECOM)*, San Francisco, CA, November 2006.

- Amir-Hamed Mohsenian-Rad, Vincent W.S. Wong, "Joint Channel Allocation, Interface Assignment, and MAC Design for Multi-Channel Wireless Mesh Networks," in *Proc. of IEEE Conference on Computer Communications (INFOCOM)*, Anchorage, AK, May 2007.

- Amir-Hamed Mohsenian-Rad, Vincent W.S. Wong, "Partially Overlapped Channel Assignment for Multi-Channel Wireless Mesh Networks," in *Proc. of IEEE International Conference on Communications (ICC)*, Glasgow, UK, June 2007.

- Amir-Hamed Mohsenian-Rad, Jianwei Huang, Mung Chiang, Vincent W.S Wong, "Simple, Optimal, and Robust Random Access," accepted for publication *in Proc. of Military Communications Conference (MILCOM)*, San Diego, CA, November 2008.

- Amir-Hamed Mohsenian-Rad, Vincent W.S. Wong, "Optimal Multi-Interface Multi-Channel Random Access," accepted for publication *in Proc. of IEEE Global Telecommunications Conference (GLOBECOM)*, New Orleans, LA, November 2008.

- Amir-Hamed Mohsenian-Rad, Vincent W.S. Wong, Victor C.M. Leung, "Two-Fold Pricing to Guarantee Individual Profits and Maximum Social Welfare in Wireless Access Net-

works," accepted for publication *in Proc. of IEEE Global Telecommunications Conference (GLOBECOM)*, New Orleans, LA, November 2008.

1.5 Thesis Organization

The remainder of the thesis is organizes as follows. Our proposed static and centralized joint logical topology design, interface assignment channel allocation, and routing algorithm to achieve max-min resource allocation in MC-WMNs is described in Chapter 2. We then extend our design to dynamic and distributed channel allocation in Chapter 3, where we consider some design objectives such as proportional fair and harmonic fair resource allocations. The formulations for assigning partially overlapped frequency channels are given in Chapter 4. Our low-complexity, robust, fast, and optimal distributed random access algorithm is described in Chapter 5. We then extend our model and propose a random access algorithm which does not require any explicit message passing among the wireless nodes in Chapter 6. In Chapter 7, we combine the results from Chapters 2 to 6 and propose a distributed multi-interface multi-channel random access using convex optimization techniques. Our proposed optimal two-fold pricing scheme for multi-hop wireless access networks in described in Chapter 8, where we also analytically prove its key properties. Finally, Chapter 9 contains discussions of main results, conclusions and proposals of future research directions. Each of the main chapters in this thesis is self-contained and included in separate journal articles and conference papers. A review of the related work for each optimization context is given for each chapter accordingly. The notations are separately defined for each chapter, but have been consistent throughout the thesis.

Chapter 2

Joint Logical Topology Design, Interface Assignment, Channel Allocation, and Routing for MC-WMNs

The aggregate capacity of WMNs can be increased by the use of multiple frequency channels as explained in Section 1.1.1. Within the IEEE 802.11 a/b/g frequency bands, the number of available channels is limited. The 802.11b/g bands and the 802.11a band provide 3 and 12 non-overlapping frequency channels, respectively. This implies that some logical links may operate on the same channel. In addition, the number of NICs is also limited. In the experimental MC-WMN test-beds in [11] and [61], each router is equipped with two NICs. A small number of NICs implies that some logical links in a router may need to share an NIC to transmit and receive data packets. Two nearby links that operate on the same channel or share the same NIC cannot be active simultaneously. In general, given the physical topology (i.e., the physical locations of the wireless mesh routers), four key issues should be addressed in MC-WMNs:

1. *Logical Topology Formation*: Given the physical topology, how many logical links (if any) should be assigned between a pair of neighboring routers?

2. *Interface Assignment*: Given the logical topology, how should the logical links be assigned to each NIC in a wireless mesh router?

3. *Channel Allocation*: Given the logical topology and interface assignment, how should a frequency channel be allocated on each logical link?

4. *Routing*: Given the logical topology, interface assignment, and channel allocation, through which logical links should the packets be forwarded?

In this chapter, we mathematically formulate the logical topology formation, interface assignment, channel allocation, and routing as a joint mixed-integer linear optimization problem. Our goal is to better understand the channel and interface assignment problems and evaluate the performance of MC-WMNs at optimal operation. We call our proposed MC-WMN architecture *TiMesh*. Our contributions are as follows:

- Our model formulation takes into account several design parameters such as the number of available NICs in each wireless mesh router, the number of available frequency channels, the communication range and the interference range of routers, and the expected traffic load between different source and destination pairs.

- Unlike most of the previously proposed channel and interface assignment strategies, our model formulation allows having multiple logical links between the same pair of routers. This further increases the effective data transmission rate between the two routers and more efficiently utilizes the available frequency channels.

- Our proposed algorithm guarantees the network connectivity. It also supports both *internal traffic* among the wireless routers and *external traffic* to the Internet.
- Simulation results show that *TiMesh* achieves higher throughput and lower end-to-end delay than the recently proposed *Hyacinth* [61] and *CLICA* [62] MC-WMN architectures.

The rest of this section is organized as follows. We present related work in Section 2.1. Our proposed joint design is described in Section 2.2. Performance evaluation and comparison are given in Section 2.3. A summary of the chapter is given in Section 2.4.

2.1 Related Work

Several logical topology formation, interface assignment, channel allocation, and routing algorithms have been recently proposed for MC-WMNs. Raniwala *et al.* [61] proposed a logical tree topology architecture for MC-WMNs, called *Hyacinth.* The tree construction mechanism is similar to the IEEE 802.1D [63] spanning tree formation. The gateways are the roots. Each router uses an up-NIC to exclusively connect to its parent and uses several (probably shared) down-NICs to connect to its children. Each parent router provides the Internet connectivity to its children routers. As a result, each wireless mesh router can access the Internet through the shortest available routing path. In the *Hyacinth* architecture, each router allocates the channels that are less used by its neighboring routers to its down-NICs. Marina *et al.* [62] also proposed the *Connected Low Interference Channel Assignment* (*CLICA*) algorithm for topology formation in MC-WMNs. The interference among logical links is modeled by a weighted conflict graph. The weight of the edge between two vertices in the weighted conflict graph indicates the extent of interference between their corresponding logical links using the protocol interference model

[64]. The proposed heuristic algorithm determines the logical links and assigns their channels permanently so that the average interference weight in the resultant conflict graph is minimized while the network connectivity is maintained. In [65], the channels are allocated so as to minimize the maximum number of interfering links within each neighborhood, subject to the connectivity constraint of the logical topology graph. A bandwidth-aware routing algorithm is also proposed to facilitate the path finding operation.

Various centralized and distributed channel allocation algorithms have also recently been proposed for MC-WMNs. The centralized schemes (e.g., [66, 67]) require a network controller to collect the topology information and assign the channels. In the distributed schemes (e.g., [68, 69]), some of the routers are responsible for channel assignment for a subset of interfaces. The channel allocation algorithms can also be classified as static and dynamic. The static algorithms (e.g., [70, 71, 72, 72, 73]) assign a frequency channel to each NIC permanently, while dynamic algorithms allow each NIC to change its channel either in a *short-term* (e.g., packet-by-packet [69, 74, 75]) or a *long-term* basis (e.g., every several minutes [66, 67, 76, 77]). Unlike the static algorithms, the dynamic channel allocation requires a coordination mechanism to ensure that the sending and the receiving routers/NICs use the same channel at the same time.

Various joint designs for MC-WMNs have also been proposed. Some recent work include joint channel assignment and routing [67, 71, 78], joint routing and interface assignment [79], and joint channel assignment and congestion control [77].

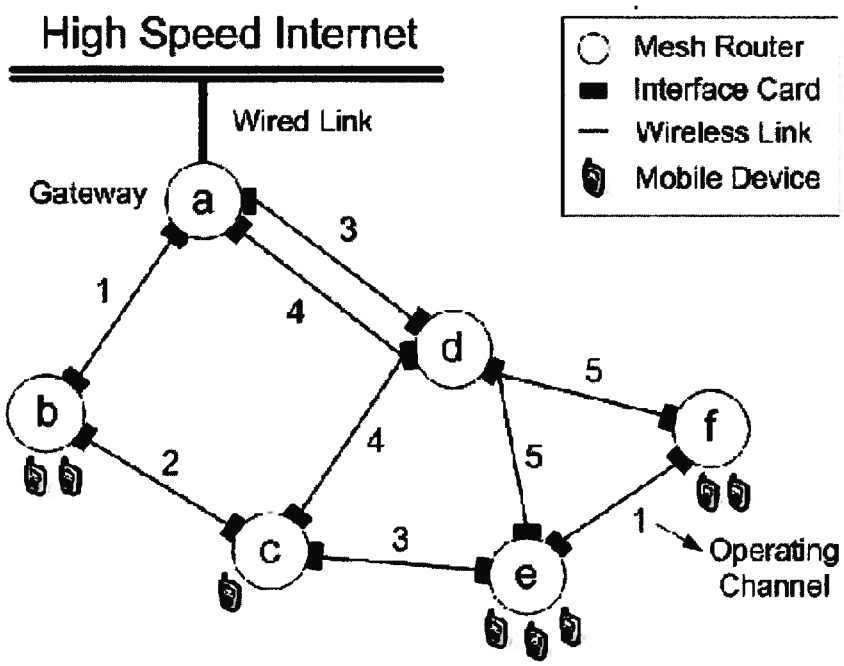


Figure 2.1: An MC-WMN with six wireless mesh routers, five frequency channels, and three NICs per router. The number on each link indicates the operating channel number.

2.2 Joint Logical Topology Design, Interface Assignment, Channel Allocation, and Routing

In this section, we formulate the logical topology formation, interface assignment, channel allocation, and routing as a joint linear optimization problem. For the rest of this chapter, the terms routers and nodes are used interchangeably.

2.2.1 Problem Formulation

We first model an MC-WMN by a *physical topology graph* $G(\mathcal{N}, \mathcal{E})$ where $\mathcal{N}$ denotes the set of all vertices and $\mathcal{E}$ denotes the set of all unidirectional edges. Each vertex $n \in \mathcal{N}$ represents a stationary wireless mesh router. For simplicity, we assume that $\mathcal{N} = \{1, 2, \ldots, N\}$. Notice that

$N = |\mathcal{N}|$. Similarly, we define $E = |\mathcal{E}|$. For the network in Fig. 2.1, we have: $N = 6$ and $E = 16$. For any two nodes $m, n \in \mathcal{N}$, if node n is within the *communication range* of node m, then there is an edge or link from node m to node n in set $\mathcal{E}$. The link from node m to node n is denoted by $e_{mn} \in \mathcal{E}$. We assume the connectivity to be symmetric. That is, link $e_{mn} \in \mathcal{E}$ if and only if $e_{nm} \in \mathcal{E}$. Each mesh router is equipped with I network interface cards. There are C orthogonal frequency channels available.

For any two nodes m and n such that $e_{mn} \in \mathcal{E}$, and any channel $i \in \{1, \ldots, C\}$, we define a *link channel allocation variable* x^i_{mn}. In the logical topology, if node m communicates with node n over the ith frequency channel, then x^i_{mn} is equal to 1; otherwise, it is equal to zero. For the MC-WMN in Fig. 2.1 with $C = 5$, we have $x^1_{ab} = 1$ and $x^2_{ab} = x^3_{ab} = x^4_{ab} = x^5_{ab} = 0$. In general, two nodes may communicate with each other over multiple distinct frequency channels. For example, consider the MC-WMN in Fig. 2.1, we have $x^3_{ad} = x^4_{ad} = 1$ and $x^1_{ad} = x^2_{ad} = x^5_{ad} = 0$. As a result, the neighboring nodes a and d can communicate with each other over both channels 3 and 4 at the same time.

To establish the logical links, nodes m and n should assign the same frequency channels to communicate with each other. This requires that,

$$x^i_{mn} = x^i_{nm}, \qquad \forall\, m, n \in \mathcal{N},\ e_{mn} \in \mathcal{E},\ \forall\, i = 1, \ldots, C. \tag{2.1}$$

The link channel allocation variables implicitly provide the required information to create the logical topology. Due to traffic and interference constraints, it is possible that there is a link between nodes m and n in the physical topology graph (i.e., $e_{mn} \in \mathcal{E}$), but there is no logical link between them in the logical topology. In that case, we have $x^i_{mn} = 0$ for all $i = 1, \ldots, C$. Note

that we allow multiple logical links between the same pair of nodes in the logical topology. They operate independently over distinct channels and can significantly increase the effective capacity between two neighboring nodes.

For any node $m \in \mathcal{N}$ and any channel $i \in \{1, \ldots, C\}$, we define y_m^i to be as follows:

$$y_m^i = \begin{cases} 1, & \text{if } \exists n \in \mathcal{N} \text{ and } e_{mn} \in \mathcal{E}, \text{ such that } x_{mn}^i = 1 \\ 0, & \text{otherwise.} \end{cases} \tag{2.2}$$

We refer to y_m^i as the *node channel allocation variable* corresponding to node m and channel i. For node a in Fig. 2.1 with $C = 5$ and $I = 3$, we have $y_a^1 = y_a^3 = y_a^4 = 1$ and $y_a^2 = y_a^5 = 0$. From (2.2), $\sum_{i=1}^{C} y_m^i$ indicates the total number of channels that are being used by node m to establish logical links with its neighboring nodes. Since each NIC operates on a distinct frequency channel, $\sum_{i=1}^{C} y_m^i$ cannot be larger than the total number of available NICs on node m. That is,

$$\sum_{i=1}^{C} y_m^i \leq I, \qquad \forall\, m \in \mathcal{N}. \tag{2.3}$$

The link and node channel allocation variables implicitly provide the required information for interface assignment. For example, given $y_a^1 = y_a^3 = y_a^4 = 1$, we assign channel 1 to the first NIC, channel 3 to the second NIC, and channel 4 to the third NIC of node a. Since $x_{ab}^1 = x_{ad}^3 = x_{ad}^4 = 1$, node a uses its first NIC to communicate with node b and its second and third NICs to communicate with node d.

Lemma 1 *The desired correspondence in (2.2) is obtained by having y_m^i be a* continuous real

variable *for all nodes* $m \in \mathcal{N}$ *and all channels* $i \in \{1, \ldots, C\}$ *and also requiring that:*

$$0 \leq y_m^i \leq \sum_{n \in \mathcal{N}, e_{mn} \in \mathcal{E}} x_{mn}^i \tag{2.4a}$$

and

$$x_{mn}^i \leq y_m^i \leq 1, \qquad \forall\, n \in \mathcal{N},\ e_{mn} \in \mathcal{E}. \tag{2.4b}$$

2.2.2 Effective Capacity

Let c^0 denote the *nominal* link-layer data rate in the corresponding 802.11 standard (e.g., 54 Mbps in 802.11a). Also let $0 \leq c_{mn}^i \leq c^0$ denote the effective capacity of the logical link (m, n) in the direction from node m to node n over frequency channel i. We have:

$$c_{mn}^i \leq x_{mn}^i c^0, \quad \forall m, n \in \mathcal{N},\ e_{mn} \in E,\ \forall i = 1, \ldots, C. \tag{2.5}$$

From (2.5), if node m does not allocate frequency channel i to communicate with node n (i.e., channel allocation variable $x_{mn}^i = 0$), then node m cannot transmit any packet to node n over channel i (i.e., $c_{mn}^i = 0$).

For any two nodes m and n such that $e_{mn} \in \mathcal{E}$, we define a set of potential interfering links $\mathcal{F}_{mn} \subset \mathcal{E}$. $\mathcal{F}_{mn}$ includes all $e_{pq} \in \mathcal{E}$ such that nodes p or q (or both) are within the *interference range* of nodes m or n (or both). Note that we always have $e_{nm} \in \mathcal{F}_{mn}$. Considering the IEEE

802.11 based RTS-CTS-DATA-ACK model, we have [67, 76]:

$$\frac{c^i_{mn}}{c^0} + \sum_{p,q,\ e_{pq} \in F_{mn}} \frac{c^i_{pq}}{c^0} \leq 1, \ \forall\, m,n \in \mathcal{N}, \ \ e_{mn} \in \mathcal{E}, \qquad \forall\, i = 1, \ldots, C \tag{2.6}$$

where c^i_{mn}/c^0 denotes the fraction of time that logical link (m,n) can be active in the direction from node m to node n over frequency channel i.

2.2.3 Total Flows on a Logical Link

For efficient network planning, a statistical model for network traffic needs to be available. Let γ^{sd} denote the expected traffic rate to be delivered between source and destination pair (s,d), where $s,d \in \mathcal{N}$. We assume that the information γ^{sd} for all source and destination pairs is given. For any source and destination pair (s,d), any nodes $m,n \in \mathcal{N}$ such that $e_{mn} \in \mathcal{E}$, and any channel $i \in \{1,\ldots,C\}$, we define a binary *routing variable* $a^{sd}_{mn,i}$. The variable $a^{sd}_{mn,i}$ is equal to 1 if the traffic from source s to destination d is being routed via link (m,n) in the direction from node m to node n over channel i, and is equal to 0 otherwise. Note that $a^{sd}_{mn,i} \neq a^{sd}_{nm,i}$ in general. Multiple links between a pair of nodes can provide more than one path between them. Since each of the multiple links is operating over a distinct channel, packets that are forwarded on different links experience different latencies. Thus, if packets that belong to the same flow use parallel links between a pair of neighboring nodes, this can cause packets to arrive out of order. To avoid this issue, only one of the available logical links between each pair of neighboring nodes is used to route packets of each flow. That is,

$$\sum_{i=1}^{C} a^{sd}_{mn,i} \leq 1, \qquad \forall\, s,d,m,n \in \mathcal{N}, \ e_{mn} \in \mathcal{E}. \tag{2.7}$$

Let λ^i_{mn} denote the aggregate traffic from all source and destination pairs that is routed on link (m, n) in the direction from node m to node n over channel i. We have:

$$\lambda^i_{mn} = \sum_{s,d \in \mathcal{N}} a^{sd}_{mn,i} \gamma^{sd}, \qquad \forall\, m, n \in \mathcal{N},\ \ e_{mn} \in \mathcal{E},\ \ i = 1, \ldots, C. \tag{2.8}$$

The aggregate traffic λ^i_{mn} cannot be more than the effective capacity c^i_{mn} for all nodes $m, n \in \mathcal{N}$ such that $e_{mn} \in \mathcal{E}$, and all channels $i \in \{1, \ldots, C\}$. Consider the following constraint:

$$\lambda^i_{mn} \leq \Lambda\, c^i_{mn}, \quad \forall\, m, n \in \mathcal{N},\ e_{mn} \in \mathcal{E},\ \forall\, i = 1, \ldots, C. \tag{2.9}$$

where $\Lambda \leq 1$ is a positive parameter. From (2.9), the parameter Λ imposes an upper bound on the expected *link utilization* λ^i_{mn}/c^i_{mn}. The higher the link utilization, the higher the queueing delay [80]. In the current Internet, an access link is considered overloaded when its average utilization is greater than 80% [81, 82]. Thus, we set $\Lambda = 0.8$.

2.2.4 Flow Conservation at Each Node

The flow conservation requires that for $s, d, m \in \mathcal{N}$,

$$\sum_{\substack{n \in \mathcal{N}, \\ e_{mn} \in \mathcal{E}}} \sum_{i=1}^{C} a^{sd}_{mn,i} \gamma^{sd} - \sum_{\substack{n \in \mathcal{N}, \\ e_{nm} \in \mathcal{E}}} \sum_{i=1}^{C} a^{sd}_{nm,i} \gamma^{sd} = \begin{cases} \gamma^{sd}, & \text{if } s = m, \\ -\gamma^{sd}, & \text{if } d = m, \\ 0, & \text{otherwise.} \end{cases} \tag{2.10}$$

In (2.10), the term on the left-hand side is the *net flow* out of node m for the flow from source s to destination d. The net flow is the difference between the outgoing flow and the incoming flow.

The term on the right-hand side is equal to 0 if node m is neither the source nor the destination for that specific flow. If node m is the source (i.e., $s = m$), then the net flow is equal to γ^{sd}. If node m is the destination (i.e., $d = m$), then the net flow is equal to $-\gamma^{sd}$. Note that both sides in (2.10) can be divided by the common factor γ^{sd}. We also notice that (2.10) implies (2.7).

The constraint in (2.10) also guarantees that there is at least one routing path available between each source and destination pair (s, d). In practice, all nodes have traffic to or from the Internet; thus we can make the valid assumption that $\gamma^{sd} > 0$, if either s or d is a gateway node. The constraint in (2.10) and the aforementioned assumption guarantee that the obtained topology is connected. That is, there is neither an isolated node nor an isolated group of nodes.

2.2.5 Feasible Region and the Objective Function

Given the expected traffic demand γ and the network resources C, I, and c^0, the constraints in (2.1)-(2.10) form the feasible region for all logical topologies that can properly support the expected traffic demand γ. The feasible region could be empty. We can enlarge the feasible region by choosing a higher value for parameter Λ; however, even for $\Lambda = 1$, the feasible region can be empty if and only if the network resources cannot support the expected traffic demand. In that case, we need to either increase the available resources or limit the traffic demand.

From constraint (2.9), the difference $(\Lambda c^i_{mn} - \lambda^i_{mn})$ is always non-negative. As λ^i_{mn} approaches Λc^i_{mn}, the difference $(\Lambda c^i_{mn} - \lambda^i_{mn})$ tends to 0 and the corresponding logical link becomes more prone to congestion. Let $\delta_{\min}$ denote the *minimum* difference $(\Lambda c^i_{mn} - \lambda^i_{mn})$ across all channels

and all links that exist in the logical topology. That is,

$$\delta_{\min} = \min_{\substack{m,n \in \mathcal{N},\ e_{mn} \in \mathcal{E}, \\ i \in \{1,\ldots,C\},\ x^i_{mn}=1}} (\Lambda\, c^i_{mn} - \lambda^i_{mn})\ . \tag{2.11}$$

Note that $\delta_{\min}$ corresponds to the most congested (i.e., the bottleneck) logical link across the network. Our objective is to maximize the variable $\delta_{\min}$. It can be achieved by decreasing the aggregate traffic load or increasing the effective capacity (or both) on the network's bottleneck link. The former implies *load balancing*: balancing the traffic load among different logical links using proper logical topology formation and routing schemes; while the latter implies *congestion-aware capacity planning*: providing higher effective capacity for more congested logical links using proper logical topology formation, interface assignment, and channel allocation schemes. Load balancing is shown to be a proper objective for joint topology control and routing algorithms in optical networks [83]. Congestion aware capacity planning is also proposed for cross-layer congestion control designs in wireless ad-hoc [49] and mesh networks [77].

Maximizing δ_{min} can also be justified in terms of providing fairness among the existing logical links. In fact, it leads to achieving *max-min* or *bottleneck-optimal* fairness [50, 80, 84]. This is equivalent to solving the NUM problem in (1.18) for α-fair utility functions with $\alpha \to \infty$. Note that the system is fair in the sense that all the links experience similar level of congestion.

Lemma 2 *The desired correspondence in (2.11) can be obtained by requiring that:*

$$\delta_{min} \leq (\Lambda c^i_{mn} - \lambda^i_{mn}) + \Lambda c^0 (1 - x^i_{mn}), \quad \forall m,n \in \mathcal{N},\ e_{mn} \in \mathcal{E},\ \forall i = 1,\ldots,C. \tag{2.12}$$

2.2.6 Hop Count Constraint

Load balancing avoids highly loaded links and prevents congestion; however, it may lead to assigning long routing paths. For each source and destination pair (s, d), the hop count along the assigned routing path is obtained as $\sum_{m,n\in\mathcal{N},\, e_{mn}\in\mathcal{E}} \sum_{i=1}^{C} a_{mn,i}^{sd}$. Let h_G^{sd} denote the hop count for the *minimum* hop path between source and destination pair (s, d) in the physical topology graph $G(\mathcal{N}, \mathcal{E})$. The ratio $\sum_{m,n\in\mathcal{N}, e_{mn}\in\mathcal{E}} \sum_{i=1}^{C} a_{mn,i}^{sd}/h_G^{sd}$ is always greater than or equal to 1, and is defined as the *stretch factor* for the routing path from the source node s to the destination node d. We define the hop count constraint to be:

$$\sum_{m,n\in\mathcal{N},\, e_{mn}\in\mathcal{E}} \sum_{i=1}^{C} a_{mn,i}^{sd} \leq \Gamma\, h_G^{sd}, \qquad \forall\, s, d \in \mathcal{N}. \tag{2.13}$$

where $\Gamma \geq 1$ is a tunable parameter to set an upper bound on the routing stretch factor. Note that there is always a *trade off* between load balancing and shortest path routing [85]. This trade off can be controlled by using the tunable parameter Γ. By assigning $\Gamma = 1$, the routing part of the algorithm becomes the shortest path routing. If $\Gamma = \infty$, the hop-count constraint (2.13) is relaxed. In general, the greater the tunable parameter Γ, the larger the feasible region.

2.2.7 Optimization Problem

Given the physical topology graph $G(\mathcal{N}, \mathcal{E})$ and all parameters $C, I, \Gamma, \Lambda, c^0, \mathcal{F}_{mn}, \gamma^{sd}, h_G^{sd}$,

$$
\begin{aligned}
\underset{x,y,c,a,\lambda,\delta_{\min}}{\textbf{maximize}} \quad & \delta_{\min} \\
\textbf{subject to} \quad & x_{mn}^i = x_{nm}^i, \\
& x_{mn}^i \leq y_m^i, \\
& y_m^i \leq \textstyle\sum_{n\in\mathcal{N},\, e_{mn}\in\mathcal{E}} x_{mn}^i, \\
& \textstyle\sum_{i=1}^{C} y_m^i \leq I, \\
& c_{mn}^i \leq x_{mn}^i c^0, \\
& c_{mn}^i + \textstyle\sum_{p,q,\, e_{pq}\in\mathcal{F}_{mn}} c_{pq}^i \leq c^0, \\
& \textstyle\sum_{n\in\mathcal{N},e_{mn}\in\mathcal{E}} \sum_{i=1}^{C} a_{mn,i}^{sd} - \sum_{n\in\mathcal{N},e_{nm}\in\mathcal{E}} \sum_{i=1}^{C} a_{nm,i}^{sd} = \begin{cases} 1, & \text{if } s = m, \\ -1, & \text{if } d = m, \\ 0, & \text{otherwise,} \end{cases} \\
& \lambda_{mn}^i = \textstyle\sum_{s,d\in\mathcal{N}} a_{mn,i}^{sd}\, \gamma^{sd} \leq \Lambda\, c_{mn}^i, \\
& \delta_{\min} \leq (\Lambda c_{mn}^i - \lambda_{mn}^i) + \Lambda c^0 (1 - x_{mn}^i), \\
& \textstyle\sum_{m,n\in\mathcal{N},\, e_{mn}\in\mathcal{E}} \sum_{i=1}^{C} a_{mn,i}^{sd} \leq \Gamma\, h_G^{sd}, \\
\textbf{where} \quad & x_{mn}^i, a_{mn,i}^{sd} \in \{0,1\}, \quad y_m^i, c_{mn}^i, \lambda_{mn}^i, \delta_{\min} \geq 0, y_m^i \leq 1, \\
& c_{mn}^i \leq c^0, \ \forall\, m,n,s,d \in \mathcal{N}, \ \ e_{mn} \in \mathcal{E}, \ \ \forall\, i = 1,\ldots,C.
\end{aligned}
\tag{2.14}
$$

Let W denote the number of source and destination pairs. The linear mixed-integer problem (2.14) has $EC(1+W)$ integer variables and $C(2E+N)+1$ real variables. It also has $1.5EC+NW$ equality and $E(5C+W)+N(C+1)+W$ inequality constraints.

Algorithm 1 *Iterated Local Search Algorithm to Solve Problem (2.14) - Centralized*

1: **set** K = the maximum number of iterations.
2: **set** $x^1_{mn}[1] = 1,\ \forall\, m,n \in \mathcal{N},\ e_{mn} \in \mathcal{E}$.
3: **set** $x^2_{mn}[1] = \cdots = x^I_{mn}[1] = 0,\ \forall\, m,n \in \mathcal{N},\ e_{mn} \in \mathcal{E}$.
4: **for** $k = 1$ **to** K **do**
5: Randomly choose $p, q \in \mathcal{N}$ such that $e_{mn} \in \mathcal{E}$.
6: **solve** problem (2.14) **subject to**
$0 \le a^{sd}_{mn,i} \le 1,\quad \forall\, s,d,m,n \in \mathcal{N},\ e_{mn} \in \mathcal{E},\ \forall\, i \in \{1,\ldots,C\}$
$x^i_{mn} = x^i_{mn}[k],\quad \forall m,n \in \mathcal{N},\ e_{mn} \in \mathcal{E},\ m,n \notin \{p,q\}$.
7: **set** $x^i_{mn}[k+1] = x^i_{mn},\ \forall\, m,n \in \mathcal{N},\ e_{mn} \in \mathcal{E}\ \forall\, i \in \{1,\cdots,C\}$.
8: **end**
9: **for** each source and destination pair (s,d) **do**
10: **set** m = s.
11: **while** $m \neq d$ **do**
12: **set** $a^{sd}_{mn,i} = 1$ **where** $\{i,n\} = \mathbf{arg\ max}_{i,n}\ a^{sd}_{mn,i}$.
13: **set** $m = n$.
14: **end**
15: **for** all $m,n \in \mathcal{N}$ and all channels $i \in \{1,\ldots,C\}$ **do**
16: **if** $a^{sd}_{mn,i} \neq 1$, **then set** $a^{sd}_{mn,i} = 0$.
17: **end**
18: **end**

2.2.8 Algorithm

There are efficient commercial software (e.g., CPLEX [31]) to solve linear mixed-integer programs. Most of them use the branch-and-cut algorithm [86]. Problem (2.14) can easily be solved for small-scale MC-WMNs. However, finding the optimal solutions are not trivial for large-scale networks. An alternative is to use some simple and efficient *metaheuristic* methods to find the sub-optimal solutions [87]. In this chapter, we use the Iterated Local Search (ILS) [88] which is a powerful metaheuristic algorithm. We will investigate the sub-optimality of the ILS algorithm in comparison with the optimal branch-and-cut algorithm in Section 2.4.

The pseudo-code for the proposed ILS algorithm is provided in Algorithm 1. In line 2, a fully connected single-channel logical topology is selected as the starting point. At each iteration, lines 4 and 5 are used to randomly select a pair of nodes $p, q \in \mathcal{N}$ (e.g., with probability that is

proportional to the worst congestion status among the current logical links between them). From the two additional constraints in line 6, the integer constraint on the routing variable a is relaxed. Most of the variables are also set as constants in the current iteration. Therefore, the modified problem only has a few integer variables and can be solved easily. Given the sub-optimal topology formation, interface assignment and channel allocation solutions, the routing path from source s to destination d is assigned by traversing the logical topology from source s to destination d, and by choosing the next hop based on the maximum observed value for routing variable a (lines 10-18). The intuitive justification is that if the relaxed $a^{sd}_{mn,i}$ is close to 1, it indicates that it is better to forward the packets from source s to destination d on the logical link (m, n) over channel i. On the other hand, if the relaxed $a^{sd}_{mn,i}$ is close to 0, it implies that it is better to avoid forwarding packets on logical link (m, n) over channel i.

2.3 Simulation Results

In this section, we evaluate the performance of our proposed *TiMesh* MC-WMN architecture and compare it with the *Hyacinth* [61] and *CLICA* [62] architectures using *ns-2* simulations. We consider both UDP (User Datagram Protocol) and TCP (Transmission Control Protocol) traffic. The *default* simulation model is as follows. The size of the network field is 1000 m $\times$ 800 m. Ten sample MC-WMNs are generated. Each MC-WMN consists of 30 routers. Four of them serve as gateways. The gateways are located at the four corners of the field. The communication and the interference ranges are 250 m and 450 m, respectively. Each router is equipped with three NICs. Six channels are available. The parameter Γ is set to two. The IEEE 802.11a standard with 54 Mbps data rate is being used. In each topology, there are 30 flows: 15 flows are internal, and 15

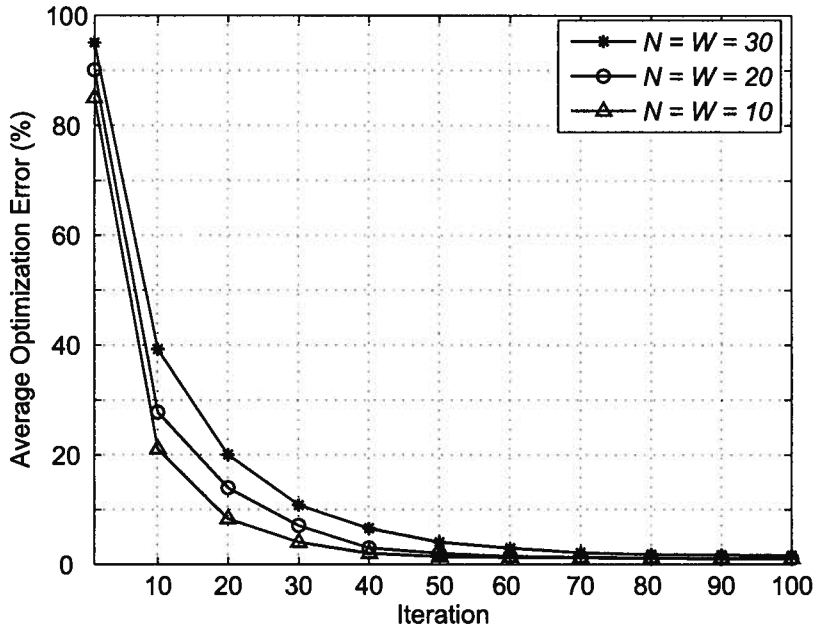


Figure 2.2: Average optimization error versus the iteration number using the ILS algorithm when the number of nodes/flows vary from 10 to 30.

flows are external. For each internal flow, two non-gateway nodes are randomly selected to be the source and destination nodes. Each external flow is established between a randomly selected node and a gateway. For UDP traffic, the packet size is 1000 bytes and the transmission rate is 500 kbps. For TCP traffic, the packet size is 1020 bytes and the transmission rate is set by the TCP Vegas. The simulation time (i.e., the duration of the simulation) is 300 sec.

For UDP traffic, the performance metrics are: 1) *packet delivery ratio*: the total number of packets received by all destinations divided by the total number of packets transmitted by all sources; 2) *end-to-end delay*: the time takes for a packet to traverse the network from a source to a destination. For TCP traffic, the metrics are: 1) *aggregate throughput*: the total number of correctly received packets (in bits) divided by the simulation time; 2) *round-trip time*: the time delay between sending a TCP segment and receiving its acknowledgement.

2.3.1 Optimal and Sub-optimal Solutions

In this section, we compare the solutions for problem (2.14) obtained from the ILS algorithm with those obtained from the optimal branch-and-cut solver. Let $\delta^*_{\min}$ denote the optimal value. Also let $\delta_{\min}[k]$ denote the value obtained from the ILS algorithm after the kth iteration. We define $(\delta^*_{\min} - \delta_{\min}[k])/\delta^*_{\min}$ as the *optimization error*. Fig. 2.2 shows the average optimization error across all ten topologies versus the iteration number when the number of nodes and flows vary from 10 to 30. We can see that the optimization error decreases when the number of iterations in ILS algorithm increases. After 50 iterations, the error is 1.4%, 1.9%, and 4% for 10, 20, and 30 nodes/flows, respectively. These results show that a near optimal solution can be achieved within a limited number of iterations. For the results presented in the subsequent sections, the near optimal solutions of problem (2.14) are obtained using ILS with 50 iterations.

2.3.2 Sample Logical Topology

Fig. 2.3(a) shows a sample physical topology. The corresponding logical topology, interface assignment, and channel allocation are shown in Fig. 2.3(b). For the physical topology graph in Fig. 2.3(a), 72 pairs of neighboring nodes are within the communication range of each other. The logical topology in Fig. 2.3(b) includes at least one logical link between 43 pairs of neighboring nodes. There are two links between nodes G_1 and a, G_3 and i, G_4 and w, as well as s and n. These links operate over distinct channels. In Fig. 2.3(b), there are 12 logical links that share an NIC with some others. The sharing of the logical links do not happen often as it reduces the corresponding link capacities. In the obtained routing solution, the 2-hop route $\{o, t, G_2\}$ is replaced by the 3-hop route $\{o, t, x, G_2\}$ to take the advantage of the unused logical link (t, x).

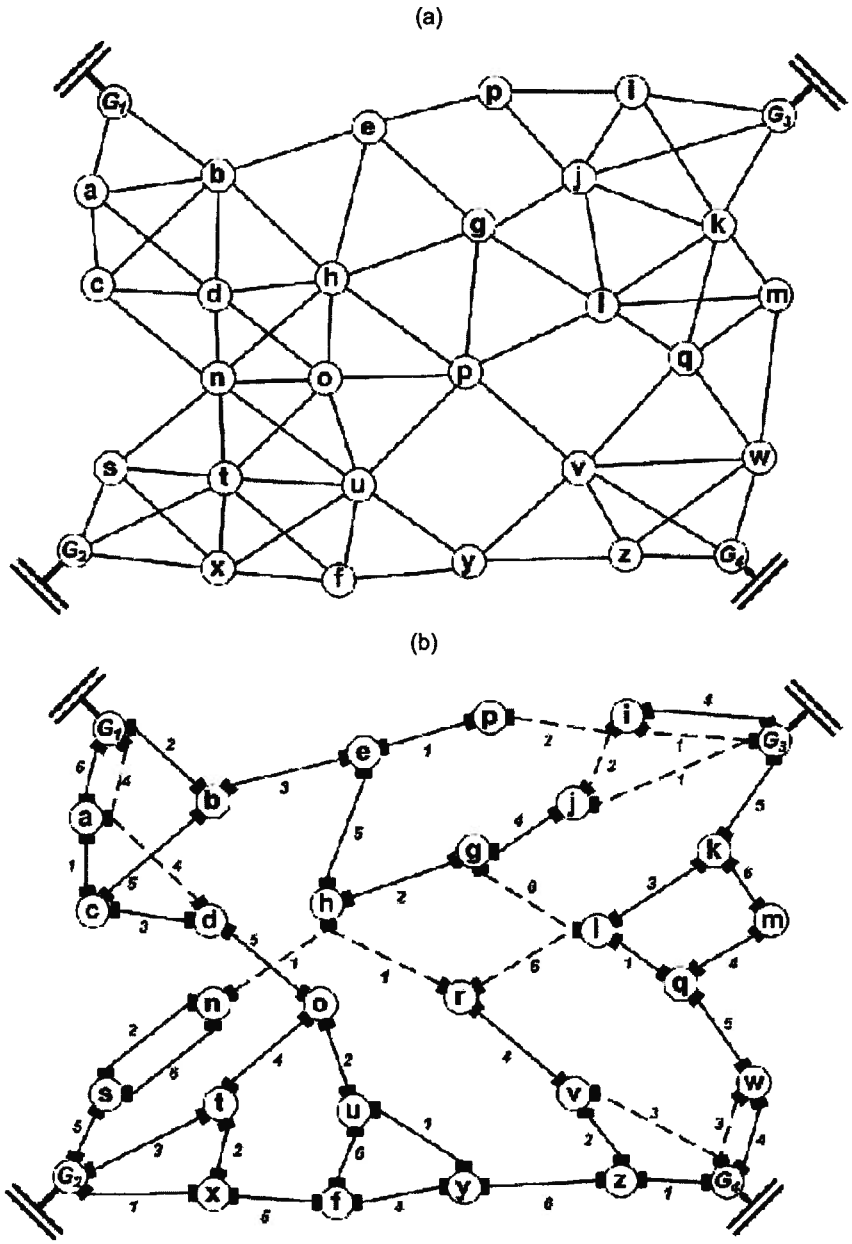


Figure 2.3: A random topology with 30 routers. Each router is equipped with 3 NICs. (a) Physical topology, (b) Logical topology, interface assignment, and channel allocation. Solid lines are wireless links that use only exclusive (not shared) NICs. Dashed lines are the links that share an interface with some other links. The number on each link indicates the channel number.

2.3.3 Impact of the Network Traffic

In this section, we first investigate the performance with UDP traffic. The number of active flows varies from 6 to 30. Fig. 2.4 shows the results of the packet delivery ratio and the average end-to-end delay. In this figure, each point is the average of measurements for all 10 simulated topologies. When the number of UDP flows increases, the network becomes congested. Since UDP does not have any congestion control mechanism, there is a reduction of the packet delivery ratio and an increase of the end-to-end delay. When there are 30 UDP flows, Fig. 2.4(a) shows that the packet delivery ratio obtained from *TiMesh* is 7% and 18% higher than *CLICA* and *Hyacinth*, respectively. Fig. 2.4(b) shows that the average end-to-end delay obtained from *TiMesh* is 28% and 52% lower than *CLICA* and *Hyacinth*, respectively.

Fig. 2.5 shows the results of the aggregated throughput and the average round-trip time when there are different number of TCP flows established in the network. In this figure, each point is the average of measurements for all 10 simulated topologies. If only a few flows are established (e.g., less than 6 flows), the *TiMesh*, *CLICA*, and *Hyacinth* architectures achieve almost the same performance. By increasing the number of flows, the network becomes congested and the round-trip time increases significantly. When there are 30 flows, Fig. 2.5(a) shows that the aggregated throughput obtained from *TiMesh* is 11% and 32% higher than *CLICA* and *Hyacinth*, respectively. Fig. 2.5(b) shows that the average round-trip time obtained from *TiMesh* is 29% and 52% lower than *CLICA* and *Hyacinth*, respectively.

The better performance of *TiMesh* can be explained based on the features of the three architectures. Unlike *Hyacinth* that concentrates the traffic on long routing paths with highly loaded links (especially the links connected to the gateways), *TiMesh* distributes and balances

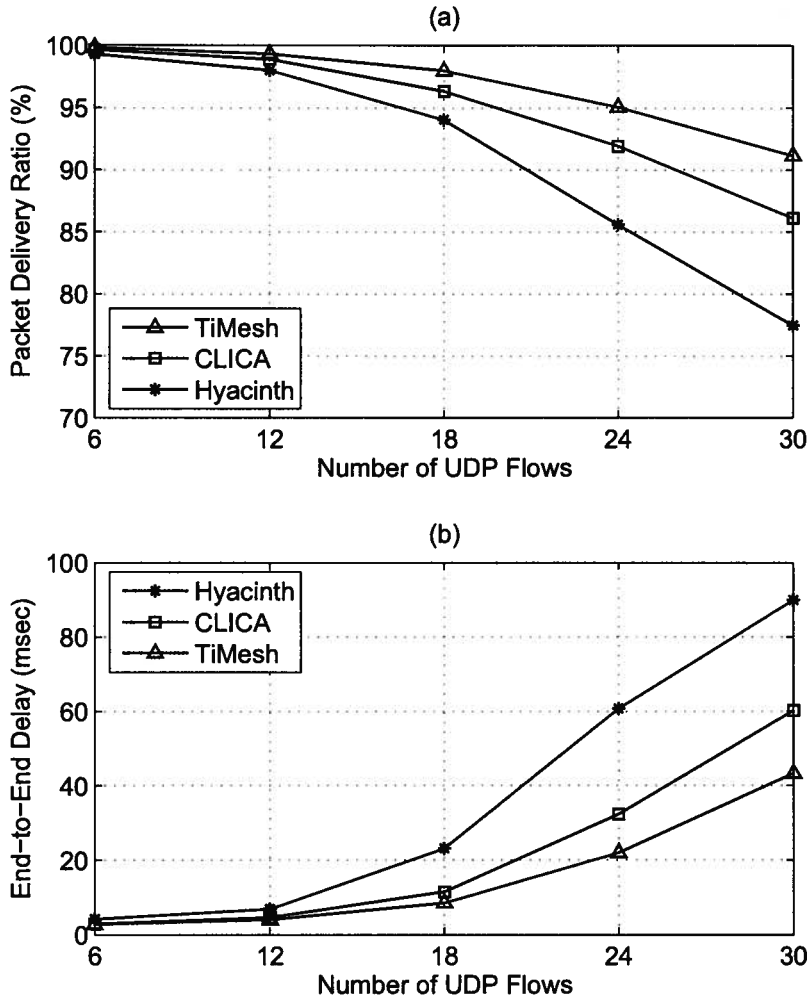


Figure 2.4: Comparison between *TiMesh*, *Hyacinth* [61], and *CLICA* [62] MC-WMN architectures in the presence of fixed-rate UDP traffic, (a) Packet delivery ratio, (b) End-to-end delay.

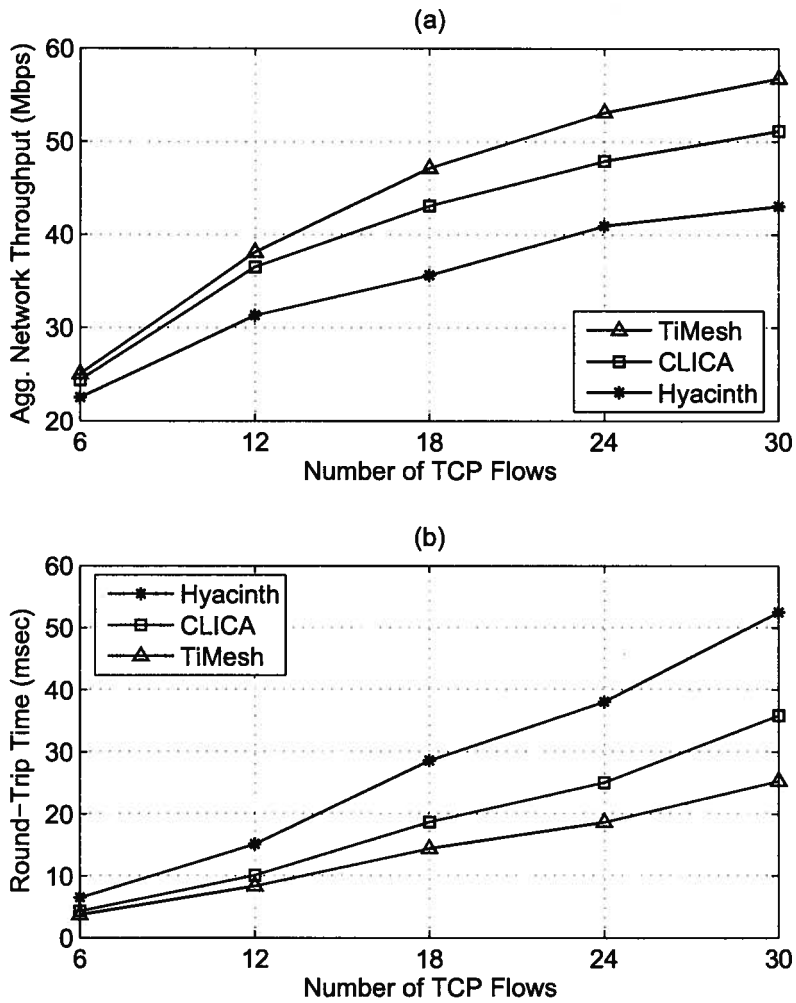


Figure 2.5: Comparison between *TiMesh*, *Hyacinth* [61], and *CLICA* [62] MC-WMN architectures in the presence of TCP traffic, (a) Aggregated throughput, (b) Average round-trip time.

the traffic among different links. It also assigns shorter routing paths. *TiMesh* has two distinct advantages when it is compared to *CLICA*. The logical topology created by *TiMesh* depends on the expected traffic demand. *TiMesh* also allows having multiple links between the routers. This further increases the effective data transmission rate between two neighboring routers.

2.3.4 Impact of the Number of NICs and Channels

In this section, we compare the performance by varying the number of interfaces and the available channels. Thirty TCP traffic flows are generated in the network. Results for the aggregate throughput and average round-trip time are shown in Table 2.1. We can see that both *TiMesh* and *CLICA* improve the performance significantly when the number of channels is increased from 6 to 9. The performance gain is less for *Hyacinth* as it has fewer logical links in its topology and cannot efficiently use the available resources. When the number of NICs increases from 3 to 4, the aggregate network throughput increases by 36%, 29%, and 37% for *TiMesh*, *CLICA*, and *Hyacinth*, respectively. The average round-trip time also decreases by 92%, 17%, and 48%, respectively. The observed high performance gain for *TiMesh* is due to the fact that it uses the extra available NICs to assign multiple links between the routers.

2.3.5 Fairness

Recall from Section 2.2.5 that *TiMesh* can achieve *max-min* fairness among logical links. To quantitatively measure the fairness that is attained among different flows, we let Ψ_{PDR} and Ψ_{EED} denote Jain's fairness indices [89] for packet delivery ratio and end-to-end delay, respectively:

$$\Psi_{\text{PDR}} = \frac{\left(\sum_{s,d\in\mathcal{N},\gamma^{sd}\neq 0} \text{PDR}(s,d)\right)^2}{W\sum_{s,d\in\mathcal{N},\gamma^{sd}\neq 0} \text{PDR}(s,d)^2}, \tag{2.15}$$

Table 2.1: Impact of varying the number of NICs and channels.

Architecture	Throughput (Mbps)			Round-Trip Time (msec)		
	$I = 3$, $C = 6$	$I = 4$, $C = 9$	$I = 4$, $C = 12$	$I = 3$, $C = 6$	$I = 4$, $C = 9$	$I = 4$, $C = 12$
TiMesh	57.9	68.3	92.0	25.5	21.8	11.6
CLICA	51.7	61.5	79.1	36.8	32.7	24.4
Hyacinth	43.1	48.2	66.4	52.7	49.9	41.0

Table 2.2: Achieved fairness among different flows.

Architecture	Fairness Index	
	Packet Delivery Ratio (Ψ_{PDR})	End-to-End Delay (Ψ_{EED})
TiMesh	0.914	0.903
CLICA	0.867	0.836
Hyacinth	0.712	0.678

where $\text{PDR}(s, d)$ denotes the packet delivery ratio for the flow from source s to destination d. The fairness index Ψ_{EED} can be expressed similarly. The measured Ψ_{PDR} and Ψ_{EED} for *TiMesh*, *CLICA*, and *Hyacinth* architectures are shown in Table 2.2. We can see that *TiMesh* offers better fairness among the flows. The lower fairness indices in *CLICA* and *Hyacinth* are the result of several highly congested bottlenecks. Those flows that traverse the bottleneck links experience higher delays and more packet loss compared to the rest of the flows. This is also the case for *Hyacinth* where the links connected to the gateways are the bottlenecks. The proposed topology control, routing, interface assignment, and channel allocation algorithms in *TiMesh* manage to avoid different links experiencing different congestion levels. Thus, different flows that are traversing different links achieve similar performance.

2.4 Summary

In this chapter, we proposed the *TiMesh* MC-WMN architecture by formulating the logical topology design, interface assignment, channel allocation, and routing as a joint linear mixed-integer optimization problem. Our model formulation takes into account the number of available NICs in routers, the number of available orthogonal frequency channels, expected traffic load between different source and destination pairs, and the effective capacity of the logical links. The proposed scheme balances the load among logical links and provides higher effective capacity for the bottleneck link(s). We conducted extensive *ns-2* simulation experiments to evaluate our algorithm and compared it with *Hyacinth* and *CLICA* MC-WMN architectures. Simulation results show that our proposed *TiMesh* architecture provides a higher aggregated network throughput and a lower end-to-end delay for both TCP and UDP traffic. The available NICs and channels are better utilized. The *TiMesh* also offers better fairness among different flows.

In this chapter, we mainly focused on centralized and static channel and interface assignment problem by formulating a linear mixed-integer problem in (2.14). In the next chapter, we will propose a dynamic and distributed channel assignment strategy. We will also study the possibility of formulating the channel and interface assignment problem as a convex optimization problem in Chapter 7.

2.5 Analytical Proofs

2.5.1 Proof of Lemma 1

Assume that node m is assigned to communicate with K neighboring nodes over channel i. Thus, constraint (2.4a) can be written as $0 \leq y_m^i \leq K$. If $K = 0$, then constraints (2.4a) and (2.4b)

become $0 \leq y_m^i \leq 0$ and $0 \leq y_m^i \leq 1$, respectively. This implies that $y_m^i = 0$. On the other hand, if $K > 0$ (i.e., if $K \geq 1$), then constraints (2.4a) and (2.4b) become $0 \leq y_m^i \leq K$ and $1 \leq y_m^i \leq 1$, respectively. This implies that $y_m^i = 1$. ■

2.5.2 Proof of Lemma 2

If there exists a logical link between nodes m and n over frequency channel i (i.e., $x_{mn}^i = 1$), then $\Lambda\, c^0(1 - x_{mn}^i) = 0$ and constraint (2.12) simply becomes $\delta_{\min} \leq (\Lambda\, c_{mn}^i - \lambda_{mn}^i)$. On the other hand, if there is no logical link between nodes m and n over frequency channel i (i.e., $x_{mn}^i = 0$), then $\Lambda\, c^0(1 - x_{mn}^i) = \Lambda\, c^0$. From eqs. (2.5) and (2.9) we also have $(\Lambda\, c_{mn}^i - \lambda_{mn}^i) = 0$. Thus, constraint (2.12) becomes $\delta_{\min} \leq \Lambda c^0$. Note that Λc^0 is an upper bound for variable $\delta_{\min}$. ■

Chapter 3

Cross-layer Fair Bandwidth Sharing for MC-WMNs

Most of the previously proposed channel assignment algorithms for MC-WMNs mainly focus on network *efficiency* (i.e., increasing the network throughput) while the issue of *fairness* remains less studied. Although several test-bed and simulation studies have shown that various channel and interface assignment algorithms can provide a higher throughput in MC-WMNs compared to their single-channel counterparts [61, 62, 66, 67, 69, 70, 71, 72, 74, 75, 76, 77, 79, 90, 91, 92, 93], it is not clear whether the same statement is true for the case of fairness. Note that, an efficient but unfair channel allocation may cause some flows to starve. In this chapter, we formulate a cross-layer bandwidth sharing problem in MC-WMNs as a NUM problem [19, 44] (see also Section 1.2.2). We then use the α-fair utility functions [50] to model a wide range of well-known fairness allocations. The contributions of our work are as follows.

- We mathematically model the channel and interface assignment problems by introducing *link* and *node* channel assignment *binary* vectors. Using these vectors, we also model the feasible region for the link-layer flow rates.

- We present a formulation for cross-layer fair bandwidth sharing problem as a non-linear

mixed-integer NUM. It takes into account the number of NICs at each router, the number of channels, and the interference constraints.

- We solve the NUM problem via both *exact* and *approximate* design schemes. The exact design results in an optimal static algorithm while the approximate design results in a near-optimal long-term basis dynamic and distributed algorithm.
- Our proposed designs take into account both network efficiency and fairness. In particular, some of the well-known fairness criteria, such as proportional fairness, harmonic-mean fairness, and max-min fairness, can be modeled using a tunable design parameter.

The rest of this chapter is organized as follows. The problem formulation is described in Section 3.1. The first design scheme (using exact binary linearization) is presented in Section 3.2. The second design scheme (using approximate dual decomposition) is described in Section 3.3. The performance of our algorithms is assessed through numerical examples in Section 3.4. A summary of the chapter is given in Section 3.5.

3.1 Problem Formulation

In this section, we describe the mathematical model to formulate a cross-layer fair bandwidth sharing problem in MC-WMNs. The terms wireless mesh routers and nodes will be used interchangeably. Consider an MC-WMN and let $\mathcal{N} = \{1, 2, \ldots, N\}$ denote the set of stationary nodes. Each node $m \in \mathcal{N}$ is equipped with I_m NICs. Different nodes can be equipped with different number of NICs. There are C orthogonal frequency channels available. We assume that the logical topology of the network has been pre-determined. Let $\mathcal{L}$ denote the set of all unidirectional logical links. The cardinality of set $\mathcal{L}$ is denoted by L. Notice that the logical topology

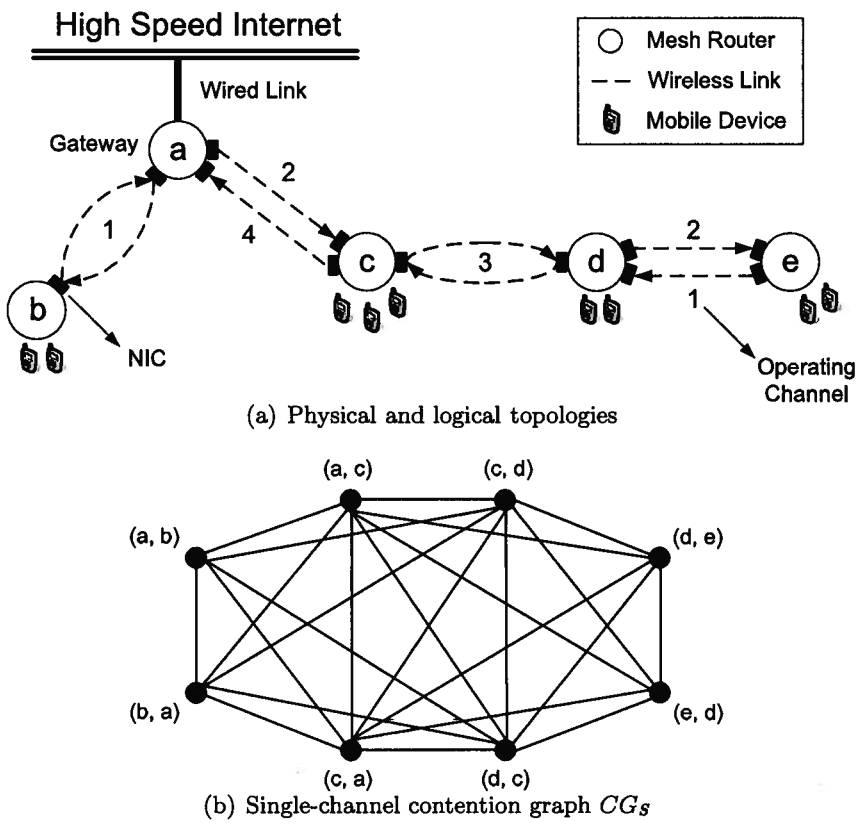


Figure 3.1: A sample MC-WMN with five routers, eight unidirectional logical links, and three frequency channels.

graph $G(\mathcal{N}, \mathcal{L})$ is always a subgraph of the physical topology graph $G(\mathcal{N}, \mathcal{E})$ where $\mathcal{E}$ is defined in Chapter 2. The logical link from node m to node n is denoted by $(m, n) \in \mathcal{L}$. We assume the connectivity to be symmetric. That is, link $(m, n) \in \mathcal{L}$ if and only if $(n, m) \in \mathcal{L}$.

3.1.1 Channel Assignment Model

For any two nodes $m, n \in \mathcal{N}$ such that there exists a logical link $(m, n) \in \mathcal{L}$, we define a $C \times 1$ *link* channel assignment vector $\boldsymbol{x}_{mn}$. The ith entry of $\boldsymbol{x}_{mn}$ is denoted by x^i_{mn}. If i^{th} frequency channel is assigned to unidirectional logical link (m, n), then $x^i_{mn} = 1$; otherwise, $x^i_{mn} = 0$. As

an example, for the MC-WMN in Fig.3.1(a) with $C = 4$, we have[1]:

$$\begin{aligned} \boldsymbol{x}_{ab} &= [\ 1 \quad 0 \quad 0 \quad 0\ ]^T, \\ \boldsymbol{x}_{ba} &= [\ 1 \quad 0 \quad 0 \quad 0\ ]^T, \\ \boldsymbol{x}_{ac} &= [\ 0 \quad 1 \quad 0 \quad 0\ ]^T, \\ \boldsymbol{x}_{ca} &= [\ 0 \quad 0 \quad 0 \quad 1\ ]^T. \end{aligned} \tag{3.1}$$

Since all logical links need to be assigned to a frequency channel, it is required that

$$\mathbf{1}^T \boldsymbol{x}_{mn} = 1, \quad \forall\, m, n \in \mathcal{N}, \quad (m, n) \in \mathcal{L}, \tag{3.2}$$

where $\mathbf{1}$ denotes a $C \times 1$ vector with all entries equal to 1. Notice that for any $m, n \in \mathcal{N}$, if $\boldsymbol{x}_{mn}^T \boldsymbol{x}_{nm} = 1$, then both logical links (m, n) and (n, m) are assigned to the same frequency channel (e.g., links (a, b) and (b, a) in Fig.3.1(a)). On the other hand, if $\boldsymbol{x}_{mn}^T \boldsymbol{x}_{nm} = 0$, then logical links (m, n) and (n, m) are assigned to two different channels (e.g., links (a, c) and (c, a) in Fig. 3.1(a)). In fact, for any pair of unidirectional logical links $(m, n), (p, q) \in \mathcal{L}$, we have:

$$\boldsymbol{x}_{mn}^T\, \boldsymbol{x}_{pq} = \begin{cases} 1, & \text{if } (m, n) \text{ and } (p, q) \text{ use the same channel,} \\ 0, & \text{otherwise.} \end{cases} \tag{3.3}$$

For any node $m \in \mathcal{N}$, we also define a $C \times 1$ *node* channel assignment vector $\boldsymbol{y}_m$. The ith entry of $\boldsymbol{y}_m$ is denoted by y_m^i. If i^{th} frequency channel is assigned to one of the NICs of node m,

[1]The link channel assignment model in this chapter is exactly the same as that in Chapter 2. However, here we consider the channel assignment variables in vector form.

then $y_m^i = 1$; otherwise, $y_m^i = 0$. Consider Fig. 3.1(a) as an example, we have[2]:

$$\begin{aligned} \boldsymbol{y}_a &= [\ 1 \quad 1 \quad 0 \quad 1\ ]^T, \\ \boldsymbol{y}_b &= [\ 1 \quad 0 \quad 0 \quad 0\ ]^T, \\ \boldsymbol{y}_c &= [\ 0 \quad 1 \quad 1 \quad 1\ ]^T. \end{aligned} \tag{3.4}$$

By definition, $\mathbf{1}^T \boldsymbol{y}_m$ indicates the total number of channels that are being used by node m to establish *outgoing* and *incoming* logical links with its neighboring nodes. Since each NIC operates on a distinct frequency channel, $\mathbf{1}^T \boldsymbol{y}_m$ cannot be larger than the total number of available NICs on node m. That is,

$$\mathbf{1}^T \boldsymbol{y}_m \leq I_m, \qquad \forall\, m \in \mathcal{N}. \tag{3.5}$$

The link and node channel assignment vectors are related. For each node $m \in \mathcal{N}$, we have $y_m^i = 1$ *if and only if* there exists $n \in \mathcal{N}$ such that either $x_{mn}^i = 1$ or $x_{nm}^i = 1$; otherwise, $y_m^i = 0$. The following Lemma, proved in Section 3.6.1, mathematically models the desired correspondence between link and node channel assignment vectors:

Lemma 3 *For each $m \in \mathcal{N}$, and any $i \in \{1, \ldots, C\}$,*

$$0 \leq y_m^i \leq \textstyle\sum_{n \in \mathcal{N}, (m,n) \in \mathcal{L}} x_{mn}^i + \sum_{n \in \mathcal{N}, (n,m) \in \mathcal{L}} x_{nm}^i, \tag{3.6}$$

$$x_{mn}^i \leq y_m^i \leq 1, \qquad \forall\, n \in \mathcal{N},\ (m,n) \in \mathcal{L}, \tag{3.7}$$

$$x_{nm}^i \leq y_m^i \leq 1, \qquad \forall\, n \in \mathcal{N},\ (n,m) \in \mathcal{L}. \tag{3.8}$$

[2]The node channel assignment variables here are slightly different from those in Chapter 2. Compare Lemma 3 with Lemma 1.

The link and node channel assignment vectors together provide all the required information to assign channels. They also implicity show how the *interfaces* should be assigned. For the example of Fig. (3.1), given $\boldsymbol{y}_a = [\,1\ 1\ 0\ 1\,]^T$, we assign channel 1 to the first NIC, channel 2 to the second NIC, and channel 4 to the third NIC of node a. Since $\boldsymbol{x}_{ab} = \boldsymbol{x}_{ba} = [\,1\ 0\ 0\ 0\,]^T$, $\boldsymbol{x}_{ac} = [\,0\ 1\ 0\ 0\,]^T$, and $\boldsymbol{x}_{ca} = [\,0\ 0\ 0\ 1\,]^T$, node a uses its first NIC to establish both links (a, b) and (b, a), its second NIC to establish (a, c), and its third NIC to establish (c, a).

We stack up all link channel assignment vectors and denote the obtained vector by $\boldsymbol{x}$. Similarly, we stack up all node channel assignment vectors and denote the obtained vector by $\boldsymbol{y}$. A *channel assignment strategy*, denoted by $\langle \boldsymbol{x}, \boldsymbol{y} \rangle$, is defined as determining vector $\boldsymbol{x}_{mn}$ for all links $(m, n) \in \mathcal{L}$, and vector $\boldsymbol{y}_m$ for all nodes $m \in \mathcal{N}$. Given an MC-WMN logical topology, a channel assignment strategy $\langle \boldsymbol{x}, \boldsymbol{y} \rangle$ is *feasible* if conditions (3.2) and (3.5)-(3.8) hold. The set of all feasible channel assignment strategies is denoted by Ψ.

3.1.2 Interference Model

In an MC-WMN, two logical links $(m, n), (p, q) \in \mathcal{L}$ are defined to *mutually interfere* with each other whenever both of the following conditions hold:

1. The logical links operate over the same frequency channel (i.e., $\boldsymbol{x}_{mn}^T\, \boldsymbol{x}_{pq} = 1$), and
2. The sender/receiver of one logical link is within the interference range of the sender/receiver of the other logical link.

To model the interference, we construct a *link-layer flow contention graph* (or simply contention graph [53]). In a contention graph, vertices correspond to the logical links. There is an edge between two vertices if the corresponding logical links mutually interfere with each other

and cannot be active simultaneously. The contention graph depends on the assigned channels. Given $\langle \boldsymbol{x}, \boldsymbol{y} \rangle$, the corresponding contention graph is denoted by $CG_{\langle \boldsymbol{x},\boldsymbol{y} \rangle}$. As a special case, consider a feasible channel assignment strategy that assigns all links to the first channel (i.e., $\boldsymbol{x}_{mn} = \boldsymbol{y}_m = [\,1 \;\; 0 \; \cdots \; 0\,]^T$ for all $m, n \in \mathcal{N}$ such that $(m, n) \in \mathcal{L}$). The corresponding contention graph is a *single-channel contention graph* and is denoted by $CG_{\boldsymbol{S}}$. The single-channel contention graph for the MC-WMN in Fig. 3.1(a) is shown in Fig. 3.1(b).

Although the vertices in $CG_{\langle \boldsymbol{x},\boldsymbol{y} \rangle}$ and $CG_{\boldsymbol{S}}$ are the same, for a general channel assignment strategy $\langle \boldsymbol{x}, \boldsymbol{y} \rangle$, $CG_{\langle \boldsymbol{x},\boldsymbol{y} \rangle}$ may have fewer edges than $CG_{\boldsymbol{S}}$. Thus,

$$CG_{\langle \boldsymbol{x},\boldsymbol{y} \rangle} \subseteq CG_{\boldsymbol{S}}, \qquad \forall \, \langle \boldsymbol{x}, \boldsymbol{y} \rangle \in \Psi. \tag{3.9}$$

Given $CG_{\langle \boldsymbol{x},\boldsymbol{y} \rangle}$, we can identify all of its maximal cliques[3]. The links which correspond to the vertices of a maximal clique cannot be active simultaneously [53, 64, 94]. Let $\mathbb{Q}_{\langle \boldsymbol{x},\boldsymbol{y} \rangle}$ denote the set of all maximal cliques in $CG_{\langle \boldsymbol{x},\boldsymbol{y} \rangle}$. The number of maximal cliques is denoted by $|\mathbb{Q}_{\langle \boldsymbol{x},\boldsymbol{y} \rangle}|$. For notation simplicity, we enumerate the maximal cliques. The ith maximal clique of $CG_{\langle \boldsymbol{x},\boldsymbol{y} \rangle}$ is denoted by $Q^i_{\langle \boldsymbol{x},\boldsymbol{y} \rangle}$. The set of vertices that form $Q^i_{\langle \boldsymbol{x},\boldsymbol{y} \rangle}$ is denoted by $\mathcal{V}^i_{\langle \boldsymbol{x},\boldsymbol{y} \rangle} \subseteq \mathcal{L}$.

Let $f_{mn} > 0$ denote the *normalized* link-layer flow rate on logical link $(m, n) \in \mathcal{L}$ (i.e., the proportion of time that link (m, n) is active). For notation simplicity, we stack up all link-layer flow rates and denote the obtained vector by $\boldsymbol{f}$. Since flows within the same maximal clique cannot transmit simultaneously, we have the following clique capacity constraint [53, 64, 94]:

$$\sum_{p,q:\, (p,q) \in \mathcal{V}^i_{\langle \boldsymbol{x},\boldsymbol{y} \rangle}} f_{pq} \;\leq\; 1, \qquad \forall \, i : Q^i_{\langle \boldsymbol{x},\boldsymbol{y} \rangle} \in \mathbb{Q}_{\langle \boldsymbol{x},\boldsymbol{y} \rangle}. \tag{3.10}$$

[3] A clique of a graph is a complete subgraph of the graph. Each clique is either a maximal clique or a subgraph of a maximal clique.

Recall that the contention graph $CG_{\langle \boldsymbol{x},\boldsymbol{y}\rangle}$ depends on selected channel assignment strategy. Any changes in $\langle \boldsymbol{x},\boldsymbol{y}\rangle$ will cause changes in $CG_{\langle \boldsymbol{x},\boldsymbol{y}\rangle}$ and the set of its maximal cliques $\mathbb{Q}_{\langle \boldsymbol{x},\boldsymbol{y}\rangle}$. This will result in changes in structure and number of inequalities in (3.10). Therefore, the current form of clique capacity constraint in (3.10) cannot be used to formulate an optimization-based channel assignment problem where $\boldsymbol{x}$ and $\boldsymbol{y}$ are optimization variables. The following theorem can overcome this problem.

Theorem 9 *Given $\langle \boldsymbol{x},\boldsymbol{y}\rangle \in \Psi$, the feasible region formed by constraint (3.10) is equivalent to the feasible region formed by the following constraint,*

$$\sum_{p,q:\ (p,q)\in\mathcal{V}_{\mathbf{S}}^{i}} \boldsymbol{x}_{mn}^{T}\boldsymbol{x}_{pq}f_{pq} \leq 1, \qquad \forall\, i : Q_{\mathbf{S}}^{i} \in \mathbb{Q}_{\mathbf{S}},\ \forall\, m,n \in \mathcal{N}, (m,n) \in \mathcal{V}_{\mathbf{S}}^{i} \tag{3.11}$$

where $\mathbb{Q}_{\mathbf{S}}$, $Q_{\mathbf{S}}^{i}$, and $\mathcal{V}_{\mathbf{S}}^{i}$ denote the set of maximal cliques, the ith maximal clique, and the set of vertices in the ith maximal clique of the single-channel contention graph $CG_{\mathbf{S}}$, respectively.

The proof of the above theorem is given in Section 3.6.1. Note that the number of constraints in (3.10) and (3.11) are not the same. Depending on $\langle \boldsymbol{x},\boldsymbol{y}\rangle$, the number of inequalities in (3.10) can *vary* from $|\mathbb{Q}_S|$ to $\frac{L}{2}$. However, the number of inequalities in (3.11) is *fixed* and is equal to $\sum_{i=1}^{|\mathbb{Q}_S|} |\mathcal{V}_S^i|$. In addition, all the inequalities in (3.10) are maximal clique constraints; while there may be some inequalities in (3.11) that are just clique (but not maximal clique) constraints.

As an example, consider CG_S in Fig. 3.1(b). Two maximal cliques are recognized:

$$\mathcal{V}_S^1 = \{(a,b),(b,a),(a,c),(c,a),(c,d),(d,c)\} \tag{3.12}$$

$$\mathcal{V}_S^2 = \{(a,c),(c,a),(c,d),(d,c),(d,e),(e,d)\} \tag{3.13}$$

They form $|V_S^1| + |V_S^2| = 12$ inequalities in (3.11). If we assign the frequency channels as shown in Fig. 3.1(a), then we have: $f_{ac} + f_{de} \leq 1$, $f_{ca} \leq 1$, $f_{cd} + f_{dc} \leq 1$, and $f_{ed} \leq 1$.

3.1.3 Cross-Layer Fair Bandwidth Sharing Problem

The model in (3.2)-(3.11) can be used in various cross-layer designs. In this chapter, we extend the fair bandwidth sharing framework in [53] to obtain two cross-layer fair bandwidth sharing algorithms for MC-WMNs. Given an MC-WMN logical topology with N nodes and L links, C orthogonal channels, I_m NICs per each router $m \in \mathcal{N}$, CG_S and the set of its maximal cliques $\mathbb{Q}_S$, our objective is to choose the normalized link-layer flow rates, and assign channels and interfaces, so as to solve the following NUM problem:

$$\begin{aligned} \underset{\mathbf{f}\succ\mathbf{0},\ \langle \boldsymbol{x},\boldsymbol{y}\rangle\in\Psi}{\textbf{maximize}} \quad & \sum_{m,n:\ (m,n)\in\mathcal{L}} u_{mn}(\kappa f_{mn}) \\ \textbf{subject to} \quad & \sum_{p,q:\ (p,q)\in\mathcal{V}_S^i} \boldsymbol{x}_{mn}^T \boldsymbol{x}_{pq} f_{pq} \leq 1, \qquad \forall\, i : Q_S^i \in \mathbb{Q}_S,\ m,n \in \mathcal{N},\ (m,n) \in \mathcal{V}_S^i \end{aligned} \tag{3.14}$$

where κ denotes the *nominal* link-layer data rate in bits per second, and u_{mn} is a continuously differentiable, increasing, and strictly concave utility function. The utility functions are assumed to be α-fair [50]. Recall from Section 1.2.2 that, if $\alpha = 1$, then proportional fairness among link-layer flows is obtained; $\alpha = 2$ corresponds to harmonic mean fairness; and $\alpha \to \infty$ corresponds to max-min fairness.

3.2 Design I: Exact Binary Linearization

Problem (3.14) is a non-linear mixed-integer problem and is not easy to solve. Note that:

1. It has real variables $\boldsymbol{f}$ and binary variables $\boldsymbol{x}$ and $\boldsymbol{y}$.

2. It has mixed binary-real *cubic* constraints.

After relaxing the binary constraints, problem (3.14) is still non-convex. Thus, even the relaxed problem cannot be easily solved. In this section, we present some binary linearization techniques to obtain the global optimal solution of problem (3.14) in a static and centralized manner. Let $\mathcal{E}_S$ denote the set of all edges in CG_S. We denote $e_{pq}^{mn} \in \mathcal{E}_S$ if there is an edge between vertices (m, n) and (p, q). The cubic constraint in (3.14) can be *linearized* as follows:

Step 1: For each pair of logical links $(m, n), (p, q) \in \mathcal{L}$ such that $e_{pq}^{mn} \in \mathcal{E}_S$, we define a $C \times 1$ *auxiliary* link channel assignment vector $\boldsymbol{v}_{pq}^{mn}$ as follows:

$$\boldsymbol{v}_{pq}^{mn} = \boldsymbol{x}_{mn} \circ \boldsymbol{x}_{pq}, \tag{3.15}$$

where $\circ$ denotes the Hadamard product[4]. From (3.15) we have, $\boldsymbol{x}_{mn}^T \, \boldsymbol{x}_{pq} = \mathbf{1}^T \, \boldsymbol{v}_{pq}^{mn}$. Notice that $\boldsymbol{x}_{mn}^T \, \boldsymbol{x}_{pq}$ is quadratic while $\mathbf{1}^T \, \boldsymbol{v}_{pq}^{mn}$ is linear. Since $\boldsymbol{x}_{mn}$, $\boldsymbol{x}_{pq}$, and $\boldsymbol{v}_{pq}^{mn}$ are $C \times 1$ binary, (3.15) is equivalent to the following *linear* constraints (see Theorem 3):

$$\begin{aligned} \boldsymbol{x}_{mn} + \boldsymbol{x}_{pq} - \boldsymbol{v}_{pq}^{mn} &\preceq \mathbf{1}, \\ -\boldsymbol{x}_{mn} - \boldsymbol{x}_{pq} + 2\boldsymbol{v}_{pq}^{mn} &\preceq \mathbf{0}. \end{aligned} \tag{3.16}$$

For notation simplicity, we stack up all vectors $\boldsymbol{v}_{pq}^{mn}$ as $\boldsymbol{v}$. A *linearized* channel assignment strategy, denoted by $\langle \boldsymbol{x}, \boldsymbol{y}, \boldsymbol{v} \rangle$, is defined as determining $\boldsymbol{x}_{mn}$ for all links $(m, n) \in \mathcal{L}$, $\boldsymbol{v}_{pq}^{mn}$ for all links $(m, n), (p, q) \in \mathcal{L}$ such that $e_{pq}^{mn} \in \mathcal{E}_S$, and $\boldsymbol{y}_m$ for all nodes $m \in \mathcal{N}$. A linearized strategy

[4]The Hadamard product of two $C \times 1$ vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ is a $C \times 1$ vector whose i^{th} entry is the product of the ith entry of $\boldsymbol{a}$ and the ith entry of $\boldsymbol{b}$.

$\langle \boldsymbol{x}, \boldsymbol{y}, \boldsymbol{v} \rangle$ is *feasible* if $\langle \boldsymbol{x}, \boldsymbol{y} \rangle \in \Psi$ and condition (3.16) holds. The set of all feasible linearized channel assignment strategies is denoted by Φ.

Step 2: For each pair of logical links $(m,n),(p,q) \in \mathcal{L}$ such that $e_{pq}^{mn} \in CG_S$, we define an *auxiliary* real scalar variable z_{pq}^{mn} as follows:

$$z_{pq}^{mn} = \boldsymbol{x}_{mn}^T \, \boldsymbol{x}_{pq} \, f_{pq} = (\mathbf{1}^T \, \boldsymbol{v}_{pq}^{mn}) f_{pq}. \tag{3.17}$$

Since $\mathbf{1}^T \boldsymbol{v}_{pq}^{mn}$ is a binary scalar and the normalized link-layer flow f_{pq} is upper bounded by one, equation (3.17) is equivalent to the following linear constraints (see Theorem 4):

$$0 \leq z_{pq}^{mn} \leq f_{pq}, \tag{3.18}$$

$$f_{pq} - 1 + \mathbf{1}^T \, \boldsymbol{v}_{pq}^{mn} \leq z_{pq}^{mn} \leq \mathbf{1}^T \, \boldsymbol{v}_{pq}^{mn}. \tag{3.19}$$

We stack up all scalars $\boldsymbol{z}_{pq}^{mn}$ and denote the obtained vector by $\boldsymbol{z}$. Combining steps 1 and 2, problem (3.14) is *equivalent* (cf. [16, pp. 130]) to the following problem:

$$\begin{array}{lll}
\textbf{maximize} & \displaystyle\sum_{m,n:\,(m,n)\in\mathcal{L}} u_{mn}(\kappa f_{mn}) & \\
\mathbf{f} \succ \mathbf{0},\ \boldsymbol{z} \succeq \mathbf{0}, & & \\
\langle \boldsymbol{x}, \boldsymbol{y}, \boldsymbol{v} \rangle \in \Phi & & \\
\textbf{subject to} & \displaystyle\sum_{p,q:\,(p,q)\in\mathcal{V}_S^i} z_{pq}^{mn} \leq 1, & \forall\, i : Q_S^i \in \mathbb{Q}_S,\ \forall\, m,n \in \mathcal{N},\ (m,n) \in \mathcal{V}_S^i, \\
& z_{pq}^{mn} \leq f_{pq}, & \forall\, m,n,p,q : e_{pq}^{mn} \in \mathcal{E}_S, \\
& f_{pq} - 1 + \mathbf{1}^T \, \boldsymbol{v}_{pq}^{mn} \leq z_{pq}^{mn}, & \forall\, m,n,p,q : e_{pq}^{mn} \in \mathcal{E}_S, \\
& z_{pq}^{mn} \leq \mathbf{1}^T \, \boldsymbol{v}_{pq}^{mn} & \forall\, m,n,p,q : e_{pq}^{mn} \in \mathcal{E}_S.
\end{array} \tag{3.20}$$

By relaxing the binary constraints on $\boldsymbol{x}$, $\boldsymbol{y}$, and $\boldsymbol{v}$, problem (3.20) becomes a convex optimization problem. There exist several efficient algorithms to solve convex problems [16]. By solving the relaxed problem, we can obtain the upper and lower bounds that are required in branch and bound algorithm [15, pp. 577-580]. By using branch and bound, we can find the global optimal solution of the mixed-integer problem in (3.20). Since problems (3.14) and (3.20) are equivalent, the global optimal solution of the mixed-integer problem in (3.14) is also readily found.

3.3 Design II: Approximate Dual Decomposition

The exact binary linearization scheme in Section 3.4 helps us to find the optimal solution of problem (3.14) in a static and centralized manner. In this section, we propose an alternative but approximate design which is more practical.

Consider the *dual* problem of the *primal* problem (3.14):

$$\underset{\boldsymbol{\rho}\succeq\mathbf{0}}{\textbf{minimize}}\ D(\boldsymbol{\rho}) \tag{3.21}$$

with partial dual function

$$D(\boldsymbol{\rho}) = \underset{\boldsymbol{f}\succ\mathbf{0},\langle\boldsymbol{x},\boldsymbol{y}\rangle\in\Psi}{\textbf{maximize}} \left[\sum_{m,n:\ (m,n)\in\mathcal{L}} u_{mn}(\kappa f_{mn}) + \sum_{i=1}^{|\mathbb{Q}_S|} \sum_{m,n:\ (m,n)\in\mathcal{V}_S^i} \rho_{mn}^i \left(1 - \sum_{p,q:\ (p,q)\in V_S^i} \boldsymbol{x}_{mn}^T \boldsymbol{x}_{pq} f_{pq} \right) \right], \tag{3.22}$$

where we relaxed the clique capacity constraint in (3.14). The Lagrange multiplier for the clique capacity constraint associated with clique $Q_S^i \in \mathbb{Q}_S$ and vertex $(m,n) \in \mathcal{V}_S^i$ is denoted by ρ_{mn}^i. For notation simplicity, we stacked up all Lagrange multipliers and denoted the obtained vector by $\boldsymbol{\rho}$. Our proposed joint algorithms are shown in Algorithm 2 and Algorithm 3, where

Algorithm 2 *Executed by each node $m \in \mathcal{N}$*

1: **For** $\forall n \in \mathcal{N}, (m,n) \in \mathcal{L}$ **do**
2: $\quad f_{mn} := \left[\frac{1}{\kappa} {u'_{mn}}^{-1}\left(\sum_{i:(m,n)\in\mathcal{V}_S^i} \sum_{p,q:(p,q)\in V_S^i} \rho_{pq}^i \boldsymbol{x}_{pq}^T \boldsymbol{x}_{mn}\right)\right]_0^1$
3: **End**
4: **For** $\forall n \in \mathcal{N}, (m,n) \in \mathcal{L}$ and $\forall i : (m,n) \in V_S^i$ **do**
5: $\quad \rho_{mn}^i := \left[\rho_{mn}^i - \xi\left(1 - \sum_{p,q:(p,q)\in V_S^i} \boldsymbol{x}_{pq}^T \boldsymbol{x}_{mn} f_{pq}\right)\right]_0^\infty$
6: **End**
7: Inform the updated values to all nodes $p \in \mathcal{N}$ such that $\exists n, q \in \mathcal{N},\ e_{pq}^{mn} \in \mathcal{E}_S$.

$[x]_a^b = \max(\min(x,b),a)$. We make the following assumptions:

1. The normalized link-layer flow rates and the Lagrangian multipliers are updated distributively and asynchronously every T_{MAC} time units using Algorithm 2.

2. The channels are updated every T_{CI} time units using Algorithm 3.

3. The time interval $T_{MAC} \ll T_{CI}$.

Consider the time interval between two consecutive channel updates (i.e., the period of length T_{CI} time units right after any channel assignment performed by Algorithm 3). During this period, Algorithm 2 is just a fair MAC [53] over *fixed* channels and *fixed* interfaces. Given $\boldsymbol{x}$ and $\boldsymbol{\rho}$ as constants, line 2 of Algorithm 2 selects f_{mn} to maximize the dual objective function in (3.22). Line 5 of Algorithm 2 also updates Lagrange multiplier ρ_{mn}^i using a sub-gradient method [15], where parameter ξ is a constant stepsize. We can interpret the Lagrange multipliers as *clique contention prices* to regulate between the supply and the demand. From line 5 in Algorithm 2, if the demand $\sum_{(p,q)\in\mathcal{V}_S^i} \boldsymbol{x}_{mn}^T \boldsymbol{x}_{pq} f_{pq}$ exceeds the supply (that is 1), the price ρ_{mn}^i will increase. The prices are then used to adjust the flow rates in the next iteration. If T_{CI} is large enough, stepsize ξ is small enough, and the *asynchronism measure* is bounded[5], then the convergence of

[5]The asynchronism measure of distributed Algorithm 2 is bounded if there exists a positive finite B such that 1) each node $m \in \mathcal{N}$ executes Algorithm 2 at least once during any interval of length B time units, and 2) the information is used by each node for executing Algorithm 2 is outdated by at most B time units [18, pp. 481].

Algorithm 3 *Executed by a pre-authorized gateway*

1: Gather the information on $\boldsymbol{f}$ and $\boldsymbol{\rho}$ from all nodes $m \in \mathcal{N}$.
2: **For** $\forall m, n, p, q : (m,n), (p,q) \in \mathcal{L}$ **do**
3: **If** $e_{pq}^{mn} \in \mathcal{E}_S$ **then**
4: $\omega_{pq}^{mn} := \frac{1}{2}\left(f_{mn} + f_{pq}\right)\left(\sum_{i:(m,n),(p,q)\in\mathcal{V}_S^i} \rho_{pq}^i\right)$
5: **else**
6: $\omega_{pq}^{mn} := 0$
7: **End if**
8: **End**
9: $d := \sum_{m,n,p,q:(m,n),(p,q)\in\mathcal{L}} \omega_{pq}^{mn}(\boldsymbol{x}_{mn}^T \boldsymbol{x}_{pq})$
10: $\langle \tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}, \tilde{\boldsymbol{v}} \rangle := \underset{\langle \tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}, \tilde{\boldsymbol{v}} \rangle \in \Phi}{\operatorname{argmin}} \sum_{m,n,p,q:(m,n),(p,q)\in\mathcal{L}} \omega_{pq}^{mn}(\mathbf{1}^T \tilde{\boldsymbol{v}}_{pq}^{mn})$
11: $\tilde{d} := \sum_{m,n,p,q:(m,n),(p,q)\in\mathcal{L}} \omega_{pq}^{mn}(\tilde{\boldsymbol{x}}_{mn}^T \tilde{\boldsymbol{x}}_{pq})$
12: **With** probability $\left[\delta\left(d/\tilde{d} - 1\right)\right]_0^1$ **do**
13: $\langle \boldsymbol{x}, \boldsymbol{y} \rangle := \langle \tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}} \rangle$
14: Inform $\langle \boldsymbol{x}, \boldsymbol{y} \rangle$ to all nodes $m \in \mathcal{N}$.
15: **End**
16: $\delta := \delta/2$.

Algorithm 2 is guaranteed [18, pp 527-535]. That is, before the new channels are being assigned by Algorithm 3 in its next iteration, Algorithm 2 will reach its fixed point. The fair MAC in Algorithm 2 can be implemented by modifying the contention window size adjustment mechanism within the IEEE 802.11 distributed coordination function as in [57].

Now consider the channel assignment scheme in Algorithm 3. We first gather all information on flow rates and Lagrangian multipliers every T_{CI} time units in a pre-authorized node (e.g., one of the gateways). Then, we select the linearized channel assignment strategy to minimize $\sum_{m,n,p,q:(m,n),(p,q)\in\mathcal{L}} \omega_{pq}^{mn}(\mathbf{1}^T \boldsymbol{v}_{pq}^{mn}) = \sum_{m,n,p,q:(m,n),(p,q)\in\mathcal{L}} \omega_{pq}^{mn}(\boldsymbol{x}_{mn}^T \boldsymbol{x}_{pq})$, where ω_{pq}^{mn} is as in lines 3-7. Recall that the Lagrange multipliers can be interpreted as clique contention prices and $\boldsymbol{x}_{mn}^T \boldsymbol{x}_{pq}$ indicates whether links (m,n) and (p,q) mutually interfere with each other. Thus, we can interpret ω_{pq}^{mn} as the *interference cost* of having the logical links (m,n) and (p,q) operate over the same frequency channel. By definition, the interference cost is high if the interfering

links are highly loaded and belong to highly contended maximal cliques. In the linear binary optimization problem in line 10, we minimize the aggregate interference cost across the network. The optimal solution of aggregate interference cost minimization problem (i.e., $\langle \tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}, \tilde{\boldsymbol{v}} \rangle$ in line 10) is taken into account with probability $\delta(d/\tilde{d} - 1)$ bounded between 0 and 1. Notice that d (in line 9) is always greater than or equal to $\tilde{d}$ (in line 11). We initially set $\delta = 1$. Since parameter δ is decreasing (see line 16), the probability of switching to new channel assignment strategies will gradually decrease through iterations. That is, Algorithm 3 becomes less willing to make changes in the assigned channels as time goes by. This will guarantee the convergence. To further explain how Algorithm 3 works, we present the following two Lemmas.

Lemma 4 *Let $Q_{\mathbf{S}}^i$ and $Q_{\mathbf{S}}^j$ be two arbitrary maximal cliques in single-channel contention graph $CG_{\mathbf{S}}$. For any links $(m,n),(p,q) \in \mathcal{L}$, we have: $(m,n) \in \mathcal{V}_{\mathbf{S}}^i$ and $(p,q) \in \mathcal{V}_{\mathbf{S}}^j$. Given the assigned channels $\langle \boldsymbol{x}, \boldsymbol{y} \rangle$, if*

$$\mathcal{V}_{\mathbf{S}}^i \setminus \left\{(l,k) : \boldsymbol{x}_{mn}^T \boldsymbol{x}_{lk} = 0\right\} \subset \mathcal{V}_{\mathbf{S}}^j \setminus \left\{(l,k) : \boldsymbol{x}_{pq}^T \boldsymbol{x}_{lk} = 0\right\}, \tag{3.23}$$

then

$$\sum_{l,k:(l,k)\in\mathcal{V}_{\mathbf{S}}^i} \boldsymbol{x}_{mn}^T \boldsymbol{x}_{lk}\, f_{lk} < \sum_{l,k:(l,k)\in\mathcal{V}_{\mathbf{S}}^j} \boldsymbol{x}_{pq}^T \boldsymbol{x}_{lk}\, f_{lk}, \tag{3.24}$$

and, if (3.24) holds for all $t > t_0$ for some t_0, then

$$\lim_{t\to\infty} \rho_{mn}^i(t) = 0. \tag{3.25}$$

The proof of Lemma 4 is given in Section 3.6.3. From (3.25), if T_{CI} is large enough, then the contention prices converge to zero for those cliques that are *not* maximal cliques of the

single-channel contention graph $CG_{\langle \boldsymbol{x},\boldsymbol{y}\rangle}$.

Lemma 5 *For arbitrary links $(m,n),(p,q) \in \mathcal{L}$ such that $e_{pq}^{mn} \in \mathcal{E}_{\mathbf{S}}$, if $\boldsymbol{x}_{mn}^T\, \boldsymbol{x}_{pq} = 1$ and $\rho_{mn}^i(0) = \rho_{pq}^i(0)$, then*

$$\rho_{mn}^i(t) = \rho_{pq}^i(t), \quad \forall\, i : (m,n),(p,q) \in \mathcal{V}_{\mathbf{S}}^i,\ \forall\, t \geq 0. \tag{3.26}$$

The proof of Lemma 5 is given in Section 3.6.4. For arbitrary logical links $(m,n),(p,q) \in \mathcal{L}$ such that $e_{pq}^{mn} \in \mathcal{E}_S$, we have:

$$\begin{aligned} \omega_{pq}^{mn} &= \frac{1}{2} \sum_{i:(m,n),(p,q)\in\mathcal{V}_S^i} \left(\rho_{pq}^i f_{mn} + \rho_{pq}^i f_{pq}\right) \\ &= \frac{1}{2} \sum_{i:(m,n),(p,q)\in\mathcal{V}_S^i} \left(\rho_{mn}^i f_{pq} + \rho_{pq}^i f_{mn}\right), \end{aligned} \tag{3.27}$$

where the first equality comes from line 3 of Algorithm 2 and the second equality results from Lemma 5. From (3.15), (3.27), and the fact that $\boldsymbol{x}_{mn}^T \boldsymbol{x}_{pq} = \boldsymbol{x}_{pq}^T \boldsymbol{x}_{mn}$, we have

$$\sum_{m,n,p,q:(m,n),(p,q)\in\mathcal{L}} \omega_{pq}^{mn} \left(\mathbf{1}^T \boldsymbol{v}_{pq}^{mn}\right) = \sum_{i=1}^{|\mathbb{Q}_S|} \sum_{m,n:\ (m,n)\in\mathcal{V}_S^i} \rho_{mn}^i \left(\sum_{p,q:\ (p,q)\in\mathcal{V}_S^i} \boldsymbol{x}_{mn}^T \boldsymbol{x}_{pq} f_{pq} \right). \tag{3.28}$$

This implies that solving the interference cost minimization problem in line 10 of Algorithm 3 is the same as selecting a feasible channel assignment strategy which maximizes the dual objective function in (3.22). In summary, both Algorithms 2 and 3 try to solve the dual problem of the primal NUM problem in (3.14). Algorithm 2 selects optimal $\boldsymbol{f}$ and $\boldsymbol{\rho}$ while $\boldsymbol{x}$ and $\boldsymbol{y}$ are assumed to be fixed. On the other hand, Algorithm 3 selects optimal $\boldsymbol{x}$ and $\boldsymbol{y}$ while assuming $\boldsymbol{f}$ and $\boldsymbol{\rho}$ are fixed. The optimality of joint Algorithms 2 and 3 is not guaranteed. We will investigate the sub-optimality of the solutions and their effects on network performance in Section 3.4.

Algorithm 4 *To replaced line 10 of Algorithm 3*

1: $K :=$ number of iterations
2: $\hat{\boldsymbol{x}}_{mn} := [\ 1\ 0\ \cdots\ 0\ ]^T,\ \ \forall\, m,n \in \mathcal{N},\ (m,n) \in \mathcal{L}$
3: $\hat{\boldsymbol{y}}_{m} := [\ 1\ 0\ \cdots\ 0\ ]^T,\ \ \forall\, m \in \mathcal{N}$
4: $\hat{\boldsymbol{v}}_{pq}^{mn} := [\ 1\ 0\ \cdots\ 0\ ]^T,\ \ \forall\, m,n,p,q \in \mathcal{N},\ e_{pq}^{mn} \in \mathcal{E}_S$
5: **For** $k := 1$ **to** K **do**
6: Randomly choose nodes $v, w \in \mathcal{N}$ such that $(v,w) \in \mathcal{L}$.
7: Using branch-and-bound [33], solve

$$\begin{aligned} \underset{\langle \tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}, \tilde{\boldsymbol{z}} \rangle \in \Psi}{\textbf{minimize}} \quad & \omega_{pq}^{mn}(\mathbf{1}^T \tilde{\boldsymbol{v}}_{pq}^{mn}) \\ \textbf{subject to} \quad & \tilde{\boldsymbol{x}}_{mn} = \hat{\boldsymbol{x}}_{mn},\ \ \forall\, m,n \in \mathcal{N}\backslash\{v,w\},\ (m,n) \in \mathcal{L} \\ & \tilde{\boldsymbol{y}}_{m} = \hat{\boldsymbol{y}}_{m},\ \ \forall\, m \in \mathcal{N}\backslash\{v,w\} \\ & \tilde{\boldsymbol{v}}_{pq}^{mn} = \hat{\boldsymbol{v}}_{pq}^{mn},\ \ \forall\, m,n,p,q \in \mathcal{N}\backslash\{v,w\},\ e_{pq}^{mn} \in \mathcal{E}_S \end{aligned}$$

8: $\langle \hat{\boldsymbol{x}}, \hat{\boldsymbol{y}}, \hat{\boldsymbol{v}} \rangle := \langle \tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}, \tilde{\boldsymbol{v}} \rangle$
9: **End**

3.3.1 A Simple Heuristic Algorithm to Solve Aggregate Interference Cost Minimization Problem

In line 10 of Algorithm 3, we need to solve a linear binary problem to minimize the interference cost across the network. There are effective commercial software packages (such as CPLEX [31] or MOSEK [32]) that can solve linear binary problems. However, the process can be time consuming for large scale MC-WMNs. An alternative is to use some simple and efficient metaheuristic methods to find suboptimal solutions [87]. Here we use the iterated local search [88]. Our algorithm is shown in Algorithm 4. We first assign all links to the first channel (lines 2-4). Then, at each iteration, we randomly choose a small neighborhood in the network (line 6) and *locally* solve the mixed-integer interference cost minimization problem for that neighborhood assuming the channels are fixed for the rest of the network (line 7). The iterations continue until a termination condition is met (e.g., we finish K iterations). Using heuristic Algorithm 4, Design II can easily be applied to large-scale MC-WMNs.

3.4 Simulation Results

In this section, we evaluate the performance of our proposed cross-layer designs. In the model, the size of the network field is 500 m × 500 m. Ten different random scenarios are considered. In each scenario, the WMN consists of twenty wireless mesh routers that are arbitrarily located in the field. Unless stated otherwise, the routers are equipped with four NICs (i.e., $I = 4$) and there are five orthogonal frequency channels available (i.e., $C = 5$). The communication and interference ranges are 100 m and 150 m, respectively. For each scenario, there is a logical link between each pair of nodes if they are within the communication range of each other. One of the network scenarios (scenario number 1) that we used in our analysis is shown in Fig. 3.4.

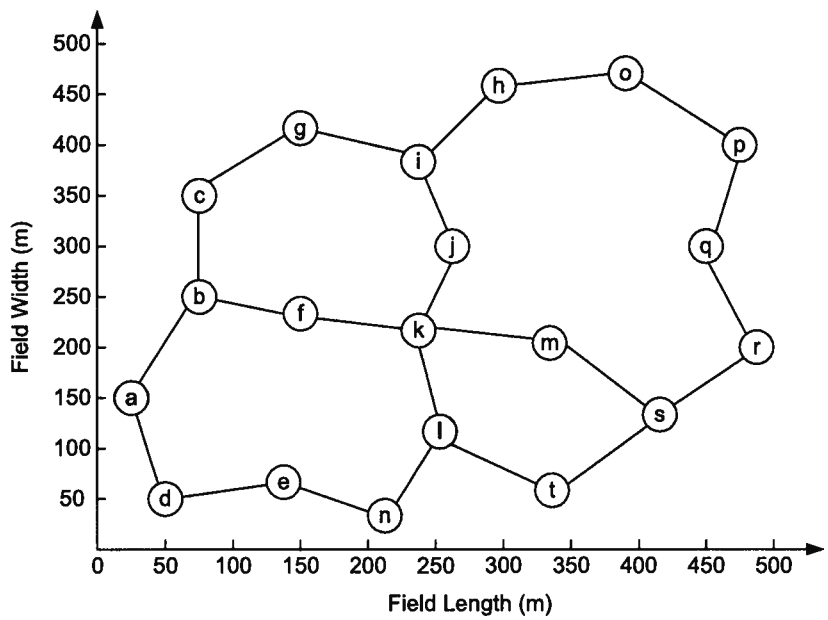


Figure 3.2: Scenario number 1: A wireless mesh network with 20 nodes (i.e., wireless mesh routers), 46 unidirectional logical links.

The utility functions are selected as in (1.17). Unless stated otherwise, we set $\alpha = 1$. Recall that the logarithmic utility functions lead to proportional fairness among the link-layer flow rates.

For the second design, we have: $T_{MAC} = 1$, $T_{CI} = 1500$, $\xi = 0.01$, $\kappa = 11$ Mbps, and $B = 5$. Note that depending on the selected value for the stepsize ξ and the value of the asynchronism measure B, the channel update interval T_{CI} should be large enough to let the fair MAC (i.e., Algorithm 2) reach its steady state. At time $t = 0$, we set $\delta = 10$. Later, we reduce δ by half every T_{CI} time units (as shown in line 16 of Algorithm 3). We use Algorithm 4 with $K = 25$ to solve the aggregate network interference cost minimization problem (see Section 3.3). All NICs initially (i.e., at $t = 0$) are assigned to a single channel.

Fig. 3.3 shows the evolution of the network utility for scenario number 1. We see that, after only four channel/interface update intervals, the utility reaches 99.46% of its optimal value. Later on, there is only one more slight channel/interface adjustment (at $t = 9000$) before the system reaches a steady state. We notice that right after each channel/interface assignment update, the utility experiences an overshoot. In fact, for the duration of overshooting, the assigned rates are *not* feasible as they were set accoridng to the previously assigned channels/interfaces. Thus, they result in higher utility until they settle down in their optimal feasible values.

The achieved network utilities for the ten different random scenarios are shown in Fig. 3.4. We see that the proposed MC-WMN deployments significantly increase the network utility in all scenarios. On average, the second (i.e., approximate) design scheme is able to find near optimal solutions with 99.6 % optimality. Recall that the second design scheme is simple to implement and its signalling overhead is not significant.

To evaluate the network performance, two metrics are considered: 1) *network throughput*, and 2) *fairness index*. The network throughput is the aggregate actual link-layer flow rate across all logical links in bits per second. That is, $\sum_{m,n:(m,n)\in\mathcal{L}} \kappa f_{mn}$. The fairness index is a dimensionless

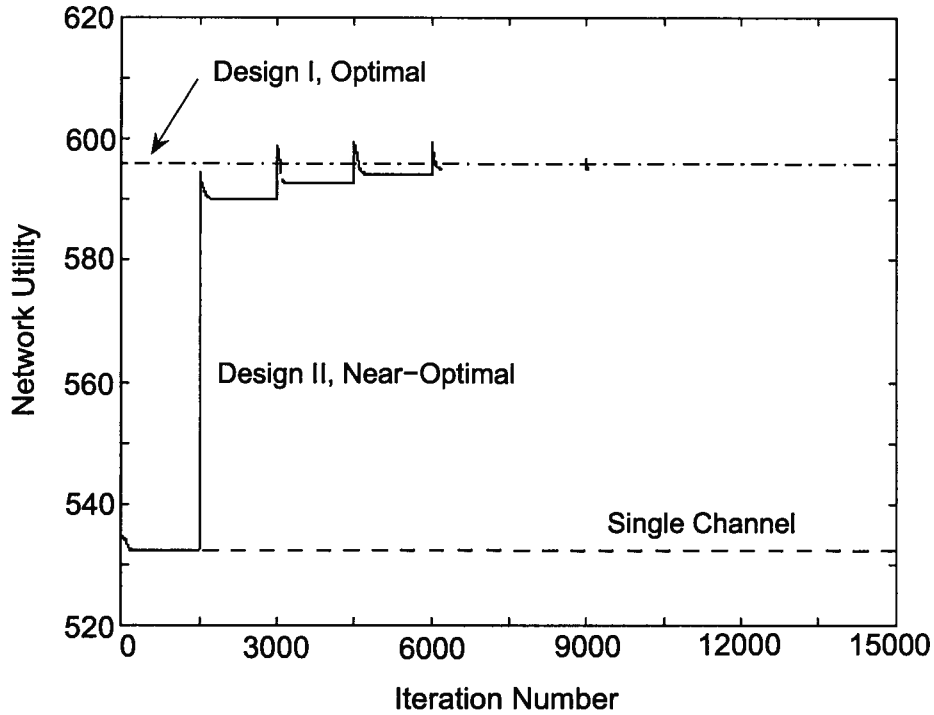


Figure 3.3: Evolution of network utility for scenario number 1.

metric between zero and one. It is defined as [89]:

$$\left(\sum_{m,n:(m,n)\in\mathcal{L}} f_{mn}\right)^2 / \left(L \sum_{m,n:(m,n)\in\mathcal{L}} (f_{mn})^2\right). \tag{3.29}$$

The higher the fairness index, the more fair the rate allocation is.

Fig. 3.5 shows the throughput and fairness index when the number of NICs varies between 2 and 4 and the number of orthogonal channels varies from 1 to 5. Each point is the average of the measurements for all ten scenarios. We can see that when each router is equipped with 3 NICs and there are 5 channels available, the throughput and fairness index increase by 242% and 3.4%, respectively, compared to the single-channel case. If each router is equipped with four NICs, then the throughput and fairness index further increase by 5% and 0.4%, respectively.

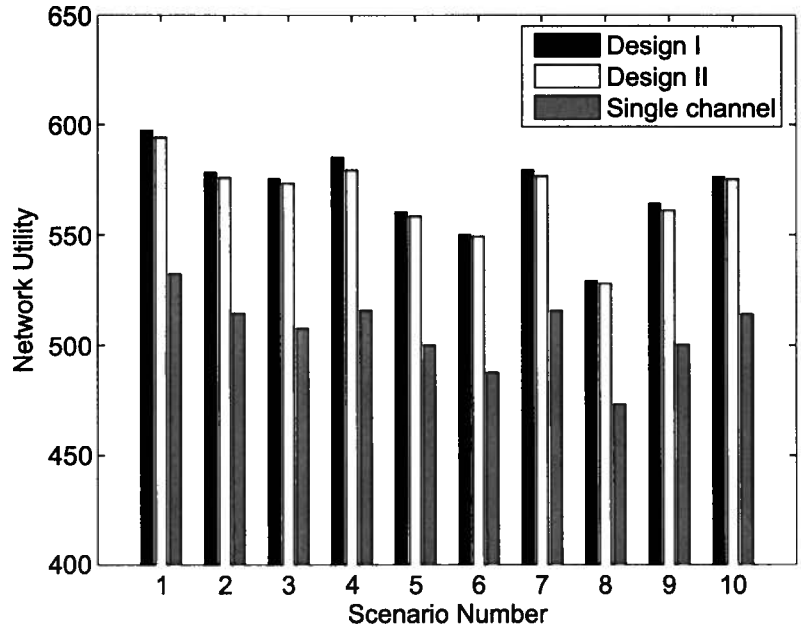
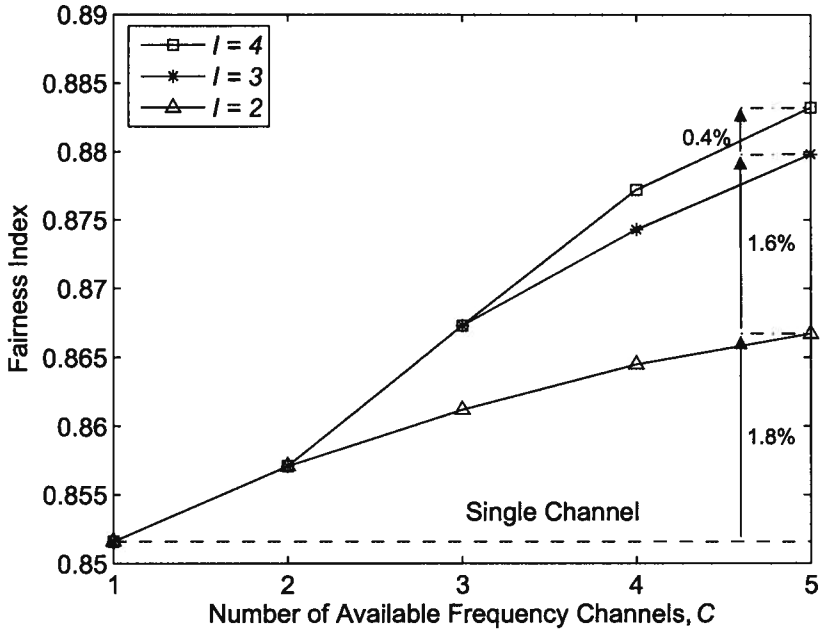


Figure 3.4: Network utility for ten different random scenarios. Each router is equipped with 4 NICs and there are 5 orthogonal channels available. On average, the second (i.e., approximate) design is able to find near optimal solutions with 99.6 % optimality. The average utility improvement compared to single-channel case is 12.5 %.

Results from Fig. 3.5 show that our proposed designs can lead to MC-WMN deployments which are not only more efficient but also more fair compared to their single-channel counterparts. Fig. 3.6 clarifies this issue in more details. In this figure, the average network throughput and the average fairness index across all ten topologies are shown when the number of channels varies from 1 to 12. Each mesh router is equipped with enough NICs so that the only resource limitation is the number of channels. We can see that the fairness index increases smoothly as the number of channels increases. To examine whether there is a similar trend for every channel and interface assignment algorithm, we consider the *load-aware* algorithm [66] which is a centralized long-term dynamic channel and interface assignment scheme. By monitoring the amount of traffic being transmitted over each frequency channel, the load-aware algorithm assigns



(a)

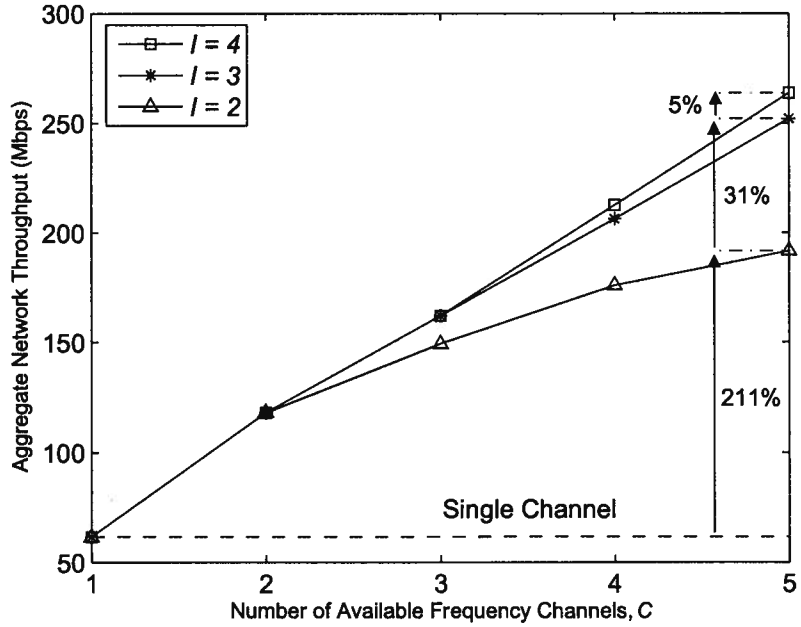


Figure 3.5: Impact of available network resources (i.e., available channels and NICs): (a) Aggregate network throughput, (b) Fairness index.

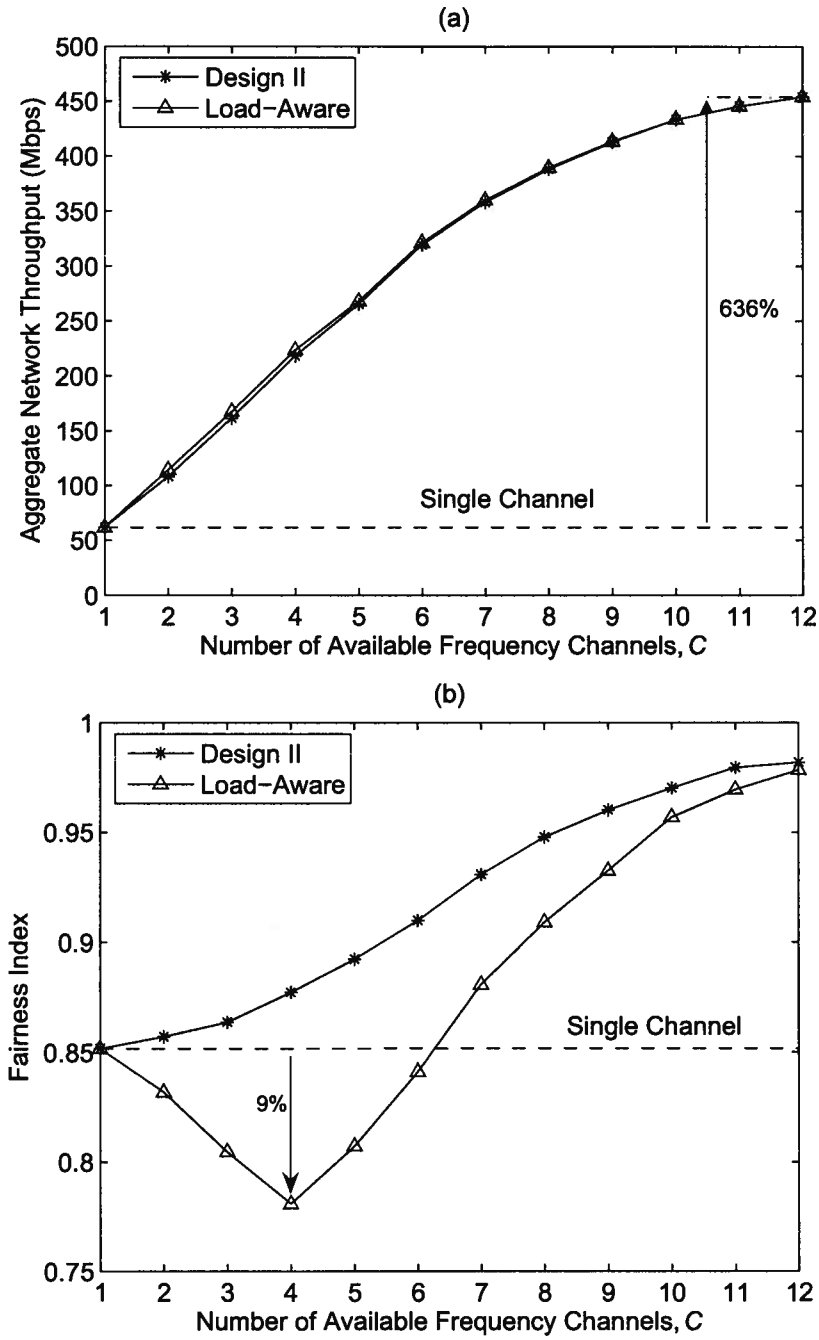


Figure 3.6: Effects of varying the number of available frequency channels with enough NICs: (a) Aggregate network throughput, (b) Fairness index.

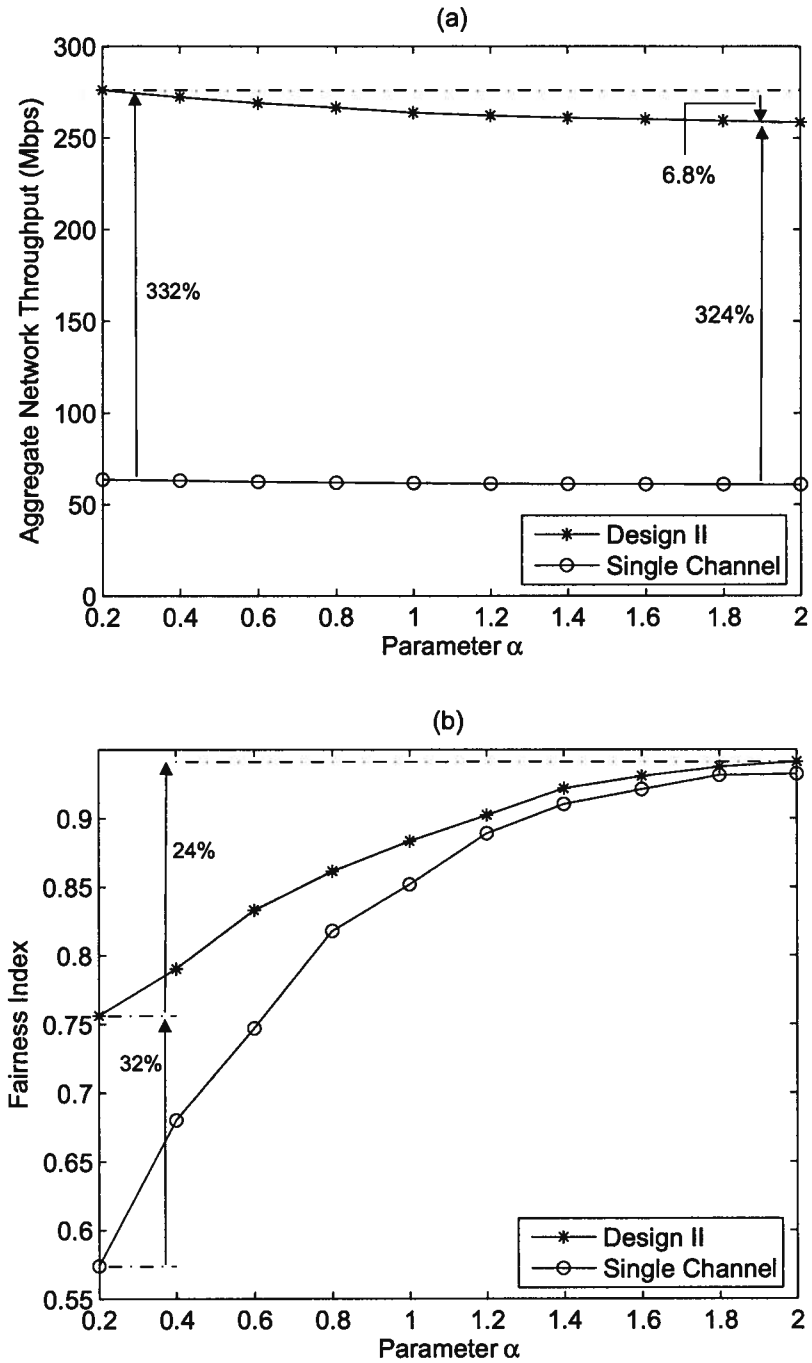


Figure 3.7: Effects of varying the fairness parameter: (a) Aggregate network throughput, (b) Fairness index.

the channel with minimum usage within the neighborhood of each link. We implemented the load-aware algorithm jointly with fair MAC [53]. To make the comparison consistent, channels and interfaces are updated every 1500 intervals as in our second design scheme. We can see that the average throughput is almost the same for both load-aware and our proposed schemes (see Fig. 3.6(a)); however, our proposed design is more fair (see Fig. 3.6(b)). In some cases (i.e., for $C = 2, \ldots, 6$), the MC-WMN is even less fair than the single-channel WMN when the load-aware algorithm is being used. Note that by increasing the number of channels, (e.g., $C \geq 7$), achieving fairness becomes trivial due to the availability of sufficient resources.

As stated in Section 1.2.2, different fairness criteria can be taken into account by tuning fairness parameter α. Fig. 3.7 shows the throughput and fairness index when $C = 5$, $I = 4$, and utility parameter α varies from 0.2 to 2. We can see that by increasing α, the system becomes more fair but less efficient. As an example, we can achieve 24% higher fairness index by setting $\alpha = 2$ (instead of $\alpha = 0.2$), at the expense of reducing the network throughput by 6.8%. From the results in Fig. 3.7, we can also conclude that as α decreases, the performance gain on fairness index becomes higher compared to the single-channel case.

3.5 Summary

In this chapter, we presented a formulation for cross-layer fair bandwidth sharing in MC-WMNs. We first modeled the channel and interface assignment problems by introducing binary channel assignment and binary interface assignment vectors. We then obtained the feasible region of the link-layer flow rates as a function of the channel and interface assignment vectors. A cross-layer fair bandwidth sharing problem was then formulated as a non-linear mixed-integer network utility maximization problem. An optimal design, based on exact binary linearization techniques, was

proposed which leads to a global maximum. A near-optimal design, based on approximate dual decomposition techniques, was also proposed which is more practical for implementation. Our proposed designs take into account both network efficiency and fairness. Some of the well-known fairness criteria, such as proportional fairness, harmonic-mean fairness, and max-min fairness, can also be modeled using a tunable design parameter.

3.6 Analytical Proofs

3.6.1 Proof of Lemma 3

Assume that node $m \in \mathcal{N}$ is assigned to establish K_{in}^i incoming and K_{out}^i outgoing logical links with its neighboring nodes over frequency channel i. Thus, constraint (3.6) can be re-written as $0 \leq y_m^i \leq (K_{in}^i + K_{out}^i)$. If $K_{in}^i = K_{out}^i = 0$, then (3.6) becomes $0 \leq y_m^i \leq 0$ and constraints (3.7) and (3.8) become $0 \leq y_m^i \leq 1$. This implies that $y_m^i = 0$. If $K_{in}^i \neq 0$ or $K_{out}^i \neq 0$, then constraint (3.6) or either one of (3.7) and (3.8) becomes $0 \leq y_m^i \leq K_{in}^i + K_{out}^i$ and $1 \leq y_m^i \leq 1$. ■

3.6.2 Proof of Theorem 9

Since (3.10) includes all maximal clique capacity constraints for the single-channel contention graph $CG_{\langle \boldsymbol{x},\boldsymbol{y} \rangle}$ and each inequality in (3.11) is a clique (not necessarily a maximal clique) capacity constraint for $CG_{\langle \boldsymbol{x},\boldsymbol{y} \rangle}$, then the feasible region formed by (3.10) is a subset of or equal to the feasible set formed by (3.11). We only need to prove that the reverse is also true. That is, the feasible region formed by (3.11) is a subset of or is equal to the feasible region formed by (3.10).

From (3.9) we have:

$$\forall\, Q^i_{\langle x,y\rangle} \in \mathbb{Q}_{\langle x,y\rangle} \;\Rightarrow\; \exists\, Q^j_S \in \mathbb{Q}_S :\; Q^i_{\langle x,y\rangle} \subseteq Q^j_S. \tag{3.30}$$

We refer to set Q^j_S as the *parent* of set $Q^i_{\langle x,y\rangle}$. In general, there may be more than one parent for set $Q^i_{\langle x,y\rangle}$. Consider an arbitrary maximal clique $Q^i_{\langle x,y\rangle}$ and one of its parents Q^j_S. Let (m,n) be a logical link in $Q^i_{\langle x,y\rangle}$. That is, $(m,n) \in \mathcal{V}^i_{\langle x,y\rangle}$. We can show (by contradiction) that,

$$\begin{aligned} \boldsymbol{x}^T_{mn}\, \boldsymbol{x}_{pq} &= 1, \qquad \forall\, p,q : (p,q) \in \mathcal{V}^i_{\langle x,y\rangle} \\ \boldsymbol{x}^T_{mn}\, \boldsymbol{x}_{pq} &= 0, \qquad \forall\, p,q : (p,q) \in \mathcal{V}^j_S \backslash \mathcal{V}^i_{\langle x,y\rangle} \end{aligned} \tag{3.31}$$

Thus, we have:

$$\begin{aligned} \sum_{p,q:\,(p,q)\in\mathcal{V}^i_{\langle x,y\rangle}} f_{pq} &= \sum_{p,q:\,(p,q)\in\mathcal{V}^i_{\langle x,y\rangle}} 1 \times f_{pq} + \sum_{p,q:\,(p,q)\in\mathcal{V}^j_S\backslash\mathcal{V}^i_{\langle x,y\rangle}} 0 \times f_{pq} \\ &= \sum_{p,q:\,(p,q)\in\mathcal{V}^i_{\langle x,y\rangle}} \boldsymbol{x}^T_{mn}\boldsymbol{x}_{pq} f_{pq} + \sum_{p,q:\,(p,q)\in\mathcal{V}^j_S\backslash\mathcal{V}^i_{\langle x,y\rangle}} \boldsymbol{x}^T_{mn}\boldsymbol{x}_{pq} f_{pq} \\ &= \sum_{p,q:\,(p,q)\in\mathcal{V}^j_S} \boldsymbol{x}^T_{mn}\boldsymbol{x}_{pq} f_{pq}, \end{aligned} \tag{3.32}$$

where the second equality follows from (3.31). Eq. (3.32) implies that for every inequality in (3.10), there is an equivalent inequality in (3.11). Therefore, the feasible region formed by (3.11) is a subset of or is equal to the one formed by (3.10). ■

3.6.3 Proof of Lemma 4

Inequality (3.24) is obtained from (3.23) as follows,

$$\begin{aligned}\sum_{l,k:(l,k)\in\mathcal{V}_S^i} \boldsymbol{x}_{mn}^T\boldsymbol{x}_{lk}\, f_{lk} &= \sum_{l,k:(l,k)\in\mathcal{V}_S^i\setminus\left\{(l,k):\boldsymbol{x}_{mn}^T\boldsymbol{x}_{lk}=0\right\}} f_{lk} \\ &< \sum_{l,k:(l,k)\in\mathcal{V}_S^j\setminus\left\{(l,k):\boldsymbol{x}_{pq}^T\,\boldsymbol{x}_{lk}=0\right\}} f_{lk} \\ &= \sum_{l,k:(l,k)\in\mathcal{V}_S^j} \boldsymbol{x}_{pq}^T\boldsymbol{x}_{lk}f_{lk}.\end{aligned} \tag{3.33}$$

From (3.11) and (3.33), we have:

$$\sum_{l,k:(l,k)\in\mathcal{V}_S^i} \boldsymbol{x}_{mn}^T\,\boldsymbol{x}_{lk}\,f_{lk} \;<\; 1. \tag{3.34}$$

Eq. (3.25) results from replacing (3.34) in the update equation of Lagrange multipliers in line 5 of Algorithm 2, assuming that $T_{\text{MAC}} \ll T_{CI}$. ■

3.6.4 Proof of Lemma 5

Consider a maximal clique $Q_S^i \in \mathbb{Q}_S$ so that $(m,n),(p,q) \in \mathcal{V}_S^i$. Since $\boldsymbol{x}_{mn}^T\,\boldsymbol{x}_{pq} = 1$,

$$\boldsymbol{x}_{mn}^T\,\boldsymbol{x}_{lk} = \boldsymbol{x}_{pq}^T\,\boldsymbol{x}_{lk}, \quad \forall\, l,k : (l,k) \in \mathcal{V}_S^i \tag{3.35}$$

Thus,

$$\sum_{l,k:(l,k)\in\mathcal{V}_S^i} \boldsymbol{x}_{mn}^T\,\boldsymbol{x}_{lk} = \sum_{l,k:(l,k)\in\mathcal{V}_S^i} \boldsymbol{x}_{pq}^T\,\boldsymbol{x}_{lk}. \tag{3.36}$$

Since $\rho_{mn}^i(0) = \rho_{pq}^i(0)$, the equality (3.26) follows from (3.36) and line 5 of Algorithm 2. ■

Chapter 4

Partially Overlapped Channel Assignment for MC-WMNs

As we showed in Chapters 2 and 3, the aggregate capacity of wireless mesh networks can be increased by the use of multiple frequency channels and multiple network interface cards in each router. However, so far we limited our studies to the case where only the orthogonal (i.e., non-overlapped) frequency channels are being used. Recent results have shown that the performance can further be increased when both non-overlapped and partially overlapped channels are used.

In this chapter, we systematically model a joint channel assignment, interface assignment, and scheduling design problem. The contributions of our work are as follows:

- Our proposed model takes into account various parameters including the number of available frequency channels, the number of available NICs in each wireless router, transmission power, path loss information, signal to interference plus noise ratio, expected traffic load, and frequency response of each channel filter.

- Since the model is formulated as a *linear* mixed-integer program with a few integer variables, the computation complexity is low and it is feasible for implementation.

- We propose the channel overlapping matrix and mutual interference matrices to model the

non-overlapped and partially overlapped channels.

- Simulation results show that the aggregate capacity increases by 90% when all partially overlapped channels within the IEEE 802.11b frequency band are used.

The rest of the chapter is organized as follows. The problem formulation is described in Section 4.1. Our proposed joint design is presented in Section 4.2. Performance evaluations are given in Section 4.3. The chapter is concluded in Section 4.4.

4.1 System Model

The system model in this chapter is very similar to the one in Chapter 3. Consider an MC-WMN and assume that $\mathcal{N} = \{1, 2, \ldots, N\}$ denotes the set of stationary wireless mesh routers. Each router is equipped with I NICs. There are C frequency channels available. We assume that the network's logical topology has been pre-determined. Let $\mathcal{L}$ denote the set of all unidirectional logical links. The cardinality of set $\mathcal{L}$ is denoted by L. The logical link from router a to router b is denoted by $(a, b) \in \mathcal{L}$. We assume the connectivity to be symmetric. That is, link $(a, b) \in \mathcal{L}$ if and only if $(b, a) \in \mathcal{L}$.

4.1.1 Channel and Interface Assignment Model

For any two routers $a, b \in \mathcal{N}$ such that there exists a logical link $(a, b) \in \mathcal{L}$, we define a $C \times 1$ *channel assignment vector* $\boldsymbol{x}_{ab}$. If router a communicates with router b over the ith frequency channel, then the ith element in $\boldsymbol{x}_{ab}$ is equal to 1; otherwise, it is equal to zero.

Same as Chapters 2 and 3, we have:

$$\boldsymbol{x}_{ab} = \boldsymbol{x}_{ba}, \quad \forall\, a, b \in \mathcal{N}, \quad (a, b) \in \mathcal{L} \tag{4.1}$$

and

$$\mathbf{1}^T\, \boldsymbol{x}_{ab} = 1, \quad \forall\, a, b \in \mathcal{N}, \quad (a, b) \in \mathcal{L} \tag{4.2}$$

where $\mathbf{1}$ denotes a $C \times 1$ vector with all entries equal to 1. The term $\mathbf{1}^T\, \boldsymbol{x}_{ab}$ is equal to 1 if router a assigns one of the available frequency channels to communicate with router b.

Similar to Chapter 3, for any two routers $a, b \in N$ such that $(a, b) \in L$, we define an $I \times 1$ *interface assignment vector* $\boldsymbol{y}_{ab}$. If the ith network interface in router a is used to communicate with router b, then the ith element in $\boldsymbol{y}_{ab}$ is equal to 1; otherwise, it is equal to zero. We have:

$$\mathbf{1}^T\, \boldsymbol{y}_{ab} = 1, \qquad \forall\, a, b \in \mathcal{N}, \quad (a, b) \in \mathcal{L} \tag{4.3}$$

where $\mathbf{1}$ denotes an $I \times 1$ vector with all entries equal to 1. The term $\mathbf{1}^T\, \boldsymbol{y}_{ab}$ is equal to 1 if router a assigns one of its NICs to communicate with router b. Notice that,

$$\boldsymbol{x}_{ab}^T\, \boldsymbol{x}_{ac} = \boldsymbol{y}_{ab}^T\, \boldsymbol{y}_{ac}, \qquad \forall\, a, b, c \in \mathcal{N}, \quad (a, b), (a, c) \in \mathcal{L}. \tag{4.4}$$

4.1.2 Channel Overlapping Matrix

Assume that m and n are two of the available channels within the 802.11 frequency band (i.e., $m, n \in \{1, \cdots, C\}$). Let $F_m(\omega)$ and $F_n(\omega)$ denote the power spectral density (PSD) functions of the band-pass filters for channels m and n, respectively. The PSD functions can be obtained from

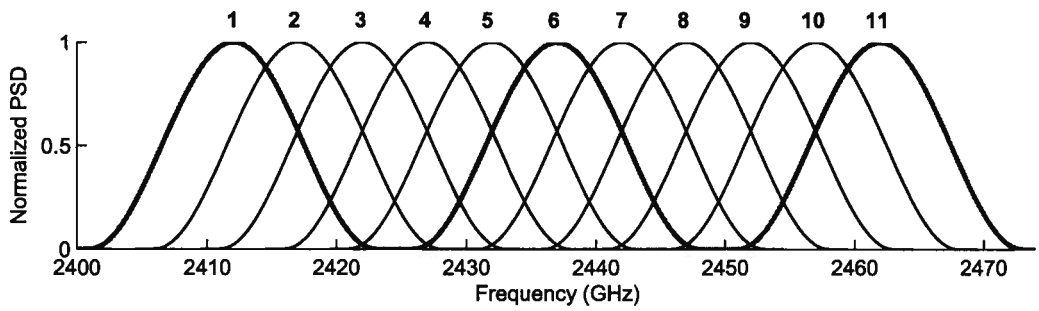


Figure 4.1: Available eleven partially overlapped channels in 802.11b frequency band. The number on each curve indicates the corresponding channel number. Channels 1, 6, and 11 are non-overlapped (orthogonal).

the channels' frequency responses. Without loss of generality, we assume the use of *raised cosine filters* [95]. Fig. 4.1 shows the frequency responses of the channel filters in the IEEE 802.11b frequency band. To model the overlapping among channels, we define a symmetric $C \times C$ *channel overlapping matrix* $\boldsymbol{W}$. The entry in the mth row and the nth column of $\boldsymbol{W}$ is denoted by scalar w_{mn} and is defined as:

$$w_{mn} = \frac{\int_{-\infty}^{\infty} F_m(\omega)\, F_n(\omega)\, d\,\omega}{\int_{-\infty}^{\infty} F_m^2(\omega)\, d\,\omega}. \tag{4.5}$$

Now assume that channels m and n are assigned to arbitrary links (a, b) and (c, d), respectively. Let p_a denote the transmission power of router a. Also let g_{ad} denote the *path loss* from router a to router d. The interference power from link (a, b) on link (c, d) can be modeled as:

$$\boldsymbol{x}_{ab}^T\, \boldsymbol{W}\, \boldsymbol{x}_{cd}\, g_{ad}\, p_a = w_{mn}\, g_{ad}\, p_a. \tag{4.6}$$

4.1.3 Mutual Interference Model

We first consider an MC-WMN where only non-overlapped channels are being used, two links $(a, b), (c, d) \in \mathcal{L}$ are defined to be *mutually interfered* with each other whenever they are assigned

to the same channel (i.e., $\boldsymbol{x}_{ab}^T \, \boldsymbol{x}_{cd} = 1$) and the sender or receiver of one link is within the *interference range* of the sender or receiver of the other link. The interference range is then defined as the region where a given receiver cannot decode the signal correctly if there is another transmission within that range. Given the modulation scheme, the interference range depends on the minimum required *signal to interference plus noise ratio* $SINR_{min}$.

Now consider an MC-WMN where both non-overlapped and partially-overlapped channels are being used. Two neighboring links $(a, b), (c, d) \in \mathcal{L}$ are assigned to channels m and n, respectively. If the interference power of the transmission on link (a, b) causes the signal to interference plus noise ratio on link (c, d) to be below $SINR_{min}$, then the transmitter of link (a, b) is within the interference range of the receiver of link (c, d):

$$\frac{g_{cd} \, p_c}{w_{mn} \, g_{ad} \, p_a + \eta_d} < SINR_{min}, \tag{4.7}$$

where η_d denotes the thermal noise power at the receiver router d. Without loss of generality, we model the path loss g_{ad} using the Friis free space model [95]:

$$g_{ad} = \frac{\alpha}{(r_{ad})^{\kappa}}, \tag{4.8}$$

where r_{ad} is the distance between routers a and d, κ is the path loss exponent, and α is a constant which depends on the transmitter and receiver antenna gains and signal wavelength. By substituting (4.8) into (4.7), link (a, b) interferes with link (c, d) if

$$r_{ad} < \sqrt[\kappa]{\left(\frac{\alpha \, p_a}{g_{cd} \, p_c / SINR_{min} - \eta_d}\right) w_{mn}}. \tag{4.9}$$

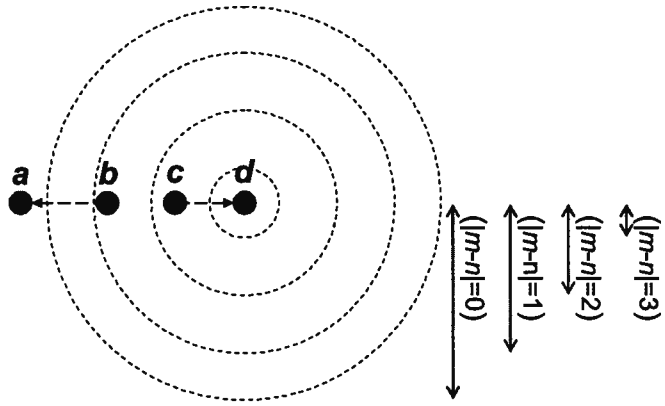


Figure 4.2: Different interference ranges depending on the frequency channel separation $|m-n|$. Logical links (a, b) and (c, d) use channels m and n, respectively.

The importance of (4.9) is that we now have different interference ranges depending on the assigned channels to the neighboring links. The less the frequency overlapped, the shorter the interference range is. Given that the bandwidth and roll-off factor are the same in all raised cosine channel filters, the interference range only depends on the *frequency channel separation* ($|m - n|$). This fact is illustrated in Fig. 4.2, where m is the channel assigned to link (a, b) and n is the channel assigned to link (c, d). The outermost circle indicates the interference range of receiver router d when $|m - n| = 0$ (i.e., the same channel is being assigned to links (a, b) and (c, d). The next circle shows the interference range when $|m - n| = 1$. The innermost circle corresponds to the interference range when $|m - n| = 3$. When $|m - n| > 3$, there is no overlap between frequency channels m and n for IEEE 802.11b (see Fig. 4.1). Thus, the corresponding interference ranges are equal to zero. Note that in this example, transmission on link (a, b) interferes with transmission on link (c, d) only when either $|m - n| = 0$ or $|m - n| = 1$.

For any two links $(a, b), (c, d) \in \mathcal{L}$, we define a symmetric $C \times C$ *mutual interference matrix* M_{cd}^{ab}. If we have either

$$r_{ad} < \sqrt[\kappa]{\left(\frac{\alpha\, p_a}{g_{cd}\, p_c / SINR_{min} - \eta_d}\right) w_{mn}},$$

or

$$r_{cb} < \sqrt[\kappa]{\left(\frac{\alpha\, p_c}{g_{ab}\, p_a / SINR_{min} - \eta_b}\right) w_{nm}},$$

then the entry in m^{th} row and n^{th} column of $\boldsymbol{M}_{cd}^{ab}$ is equal to 1; otherwise, it is equal to 0. If the transmission powers are fixed, the mutual interference matrices are constant. For the scenario in Fig. 4.2, the corresponding mutual interference matrices are *tridiagonal* with all diagonal, subdiagonal, and superdiagonal entries equal to 1:

$$\boldsymbol{M}_{cd}^{ab} = \boldsymbol{M}_{ab}^{cd} = \begin{bmatrix} 1 & 1 & 0 & 0 & \cdots & 0 \\ 1 & 1 & 1 & 0 & \cdots & 0 \\ 0 & 1 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 1 & 0 \\ 0 & \cdots & 0 & 1 & 1 & 1 \\ 0 & \cdots & 0 & 0 & 1 & 1 \end{bmatrix}_{11\times 11} \tag{4.10}$$

Note that if logical links (a, b) and (c, d) are far enough from each other, then all entries of $\boldsymbol{M}_{cd}^{ab}$ become zero. According to the definitions of channel assignment vectors and mutual interference matrices, we have:

$$\boldsymbol{x}_{ab}^T\, \boldsymbol{M}_{cd}^{ab}\, \boldsymbol{x}_{cd} = \begin{cases} 1, & \text{if links } (a,b) \text{ and } (c,d) \text{ are mutually interfered,} \\ 0, & \text{otherwise.} \end{cases} \tag{4.11}$$

4.1.4 Link-Layer Flow Rates

Let $0 \leq f_{ab} \leq 1$ denote the normalized link-layer flow rate on logical link $(a, b) \in \mathcal{L}$ (i.e., the proportion of time that link (b, a) is active). If two logical links $(a, b), (c, d) \in \mathcal{L}$ are mutually interfered (i.e., $\boldsymbol{x}_{ab}^T \boldsymbol{M}_{cd}^{ab} \boldsymbol{x}_{cd} = 1$), they cannot be active simultaneously. We can extend the interference constraint in [67] to have:

$$f_{ab} + \sum_{c,d:(c,d)\in\mathcal{L}} \boldsymbol{x}_{ab}^T \boldsymbol{M}_{cd}^{ab} \boldsymbol{x}_{cd} f_{cd} \leq 1. \tag{4.12}$$

Let c^0 denote the *nominal data transmission rate* (e.g., 11 Mbps data rate in IEEE 802.11b). The *effective link-layer data rate* on link (a, b) is equal to $f_{ab} c^0$.

4.2 Joint Channel Assignment, Interface Assignment, and Scheduling Problem

The mathematical formulation introduced in (4.1)-(4.12) models the channel assignment, interface assignment, mutual interference among the links, and link-layer flow rates when all non-overlapped and partially overlapped channels within the IEEE 802.11 frequency band are being used. The model can be employed to develop different cross-layer design schemes for MC-WMNs. In this chapter, we focus on a joint channel assignment, interface assignment, and scheduling design problem. We assume that the network logical topology and routing paths have been pre-determined.

4.2.1 Nonlinear Problem

Given the expected end-to-end traffic rates and routing paths, we can determine the expected aggregated traffic load on each logical link $(a, b) \in \mathcal{L}$. It is denoted by γ_{ab}. The *link utilization* on logical link (a, b) is defined as the total traffic load γ_{ab} divided by the effective link-layer data rate $f_{ab}\, c^0$. Based on the results from queuing theory, when the link utilization is close to 1, the queueing delay tends to be very large [80]. On the other hand, a small value of the link utilization tends to provide a small queueing delay. It also implies that the network is less prone to congestion. Thus, our objective function is to minimize the maximum (i.e., bottleneck) link utilization in an attempt to manage the network capacity according to the expected traffic load. Given the parameters $I, C, L, \gamma_{ab}, c^0$, and $\boldsymbol{M}_{cd}^{ab}$ for all $a, b, c, d \in \mathcal{N}$ such that $(a, b), (c, d) \in \mathcal{L}$,

$$
\begin{aligned}
\textbf{minimize} \quad & \underset{\boldsymbol{x},\boldsymbol{y},\boldsymbol{f}}{} \quad \underset{a,b:(a,b)\in\mathcal{L}}{\textbf{maximum}} \; \frac{\gamma_{ab}}{f_{ab}\, c^0} \\
\textbf{subject to} \quad & \boldsymbol{x}_{ab} = \boldsymbol{x}_{ba}, \\
& \mathbf{1}^T\, \boldsymbol{x}_{ab} = 1, \\
& \mathbf{1}^T\, \boldsymbol{y}_{ab} = 1, \\
& \boldsymbol{x}_{ab}^T\, \boldsymbol{x}_{ac} = \boldsymbol{y}_{ab}^T\, \boldsymbol{y}_{ac}, \\
& f_{ab} + \sum_{c,d:(c,d)\in\mathcal{L}} \boldsymbol{x}_{ab}^T\, \boldsymbol{M}_{cd}^{ab}\, \boldsymbol{x}_{cd}\, f_{cd} \leq 1, \\
\textbf{where} \quad & \boldsymbol{x}_{ab} \in \{0,1\}^C, \;\; \boldsymbol{y}_{ab} \in \{0,1\}^I, \;\; f_{ab} > 0, \;\; \forall\, a, b : (a,b) \in \mathcal{L}.
\end{aligned}
\tag{4.13}
$$

Intuitively, if a particular link is heavily loaded, its effective link-layer data rate should be increased by reducing the interfering links in its neighborhood. Note that problem (4.13) is a *non-linear* mixed-integer program and is not easy to solve. The terms $\boldsymbol{x}_{ab}^T\, \boldsymbol{x}_{ac}$ and $\boldsymbol{y}_{ab}^T\, \boldsymbol{y}_{ac}$ are

quadratic and the term $\boldsymbol{x}_{ab}^T \, \boldsymbol{M}_{cd}^{ab} \, \boldsymbol{x}_{cd} \, f_{cd}$ is cubical.

4.2.2 Equivalent Linear Problem

Following the binary linearization techniques (see Section 1.2.1), the non-linear problem (4.13) can be converted to its *equivalent* linear problem using these steps:

Step 1. For each pair of logical links $(a,b),(c,d) \in \mathcal{L}$, we define a $C \times 1$ *auxiliary vector* $\boldsymbol{u}_{cd}^{ab}$ to be as follows:

$$\boldsymbol{u}_{cd}^{ab} = \boldsymbol{x}_{ab} \circ \boldsymbol{x}_{cd} \tag{4.14}$$

where $\circ$ denotes the Hadamard product [96]. From (4.14) we have, $\boldsymbol{x}_{ab}^T \, \boldsymbol{x}_{cd} = \mathbf{1}^T \, \boldsymbol{u}_{cd}^{ab}$. Since $\boldsymbol{x}_{ab}$ and $\boldsymbol{x}_{cd}$ are binary vectors, (4.14) is equivalent to the following linear constraint [39]:

$$\begin{aligned} \boldsymbol{x}_{ab} + \boldsymbol{x}_{cd} - \boldsymbol{u}_{cd}^{ab} &\preceq \mathbf{1}, \\ \mathbf{0} \preceq \boldsymbol{u}_{cd}^{ab} &\preceq \boldsymbol{x}_{ab}, \\ \mathbf{0} \preceq \boldsymbol{u}_{cd}^{ab} &\preceq \boldsymbol{x}_{cd}. \end{aligned} \tag{4.15}$$

Step 2. For each pair of logical links $(a,b),(a,c) \in \mathcal{L}$, we define an $I \times 1$ *auxiliary vector* $\boldsymbol{v}_{ac}^{ab}$ to be as follows:

$$\boldsymbol{v}_{ac}^{ac} = \boldsymbol{y}_{ab} \circ \boldsymbol{y}_{ac}. \tag{4.16}$$

From (4.16), $\boldsymbol{y}_{ab}^T\, \boldsymbol{y}_{ac} = \mathbf{1}^T\, \boldsymbol{v}_{ac}^{ab}$. Eq. (4.16) is equivalent to the following constraint [39]:

$$\begin{aligned} \boldsymbol{y}_{ab} + \boldsymbol{y}_{ac} - \boldsymbol{v}_{ac}^{ab} &\preceq \mathbf{1}, \\ \mathbf{0} \preceq \boldsymbol{v}_{ac}^{ab} &\preceq \boldsymbol{y}_{ab}, \\ \mathbf{0} \preceq \boldsymbol{v}_{ac}^{ab} &\preceq \boldsymbol{y}_{ac}. \end{aligned} \tag{4.17}$$

Step 3. For each pair of logical links $(a,b),(c,d) \in L$, we define a $C \times C$ *auxiliary matrix* $\boldsymbol{Q}_{cd}^{ab}$ to be as follows:

$$\boldsymbol{Q}_{cd}^{ab} = \boldsymbol{x}_{ab}\, \boldsymbol{x}_{cd}^T\, f_{cd}. \tag{4.18}$$

The entry in the mth row and the nth column of matrix $\boldsymbol{Q}_{cd}^{ab}$ is equal to the product of the mth entry in vector $\boldsymbol{x}_{ab}$, the nth entry in vector $\boldsymbol{x}_{cd}$, and f_{cd}. Eq. (4.18) is equivalent to the following linear constraint for all $m,n \in \{1,\ldots,C\}$ (see Theorem 5):

$$\begin{aligned} \mathbf{1}_m^T\, \boldsymbol{Q}_{cd}^{ab}\, \mathbf{1}_n &\preceq \mathbf{1}_m^T \boldsymbol{x}_{ab}, \\ \mathbf{1}_m^T\, \boldsymbol{Q}_{cd}^{ab}\, \mathbf{1}_n &\preceq \mathbf{1}_n^T \boldsymbol{x}_{cd}, \\ \mathbf{1}_m^T \boldsymbol{x}_{ab} + \mathbf{1}_n^T \boldsymbol{x}_{cd} - 2 + f_{cd} &\preceq \mathbf{1}_m^T\, \boldsymbol{Q}_{cd}^{ab}\, \mathbf{1}_n, \\ \mathbf{1}_m^T\, \boldsymbol{Q}_{cd}^{ab}\, \mathbf{1}_n &\preceq 2 - \mathbf{1}_m^T \boldsymbol{x}_{ab} - \mathbf{1}_n^T \boldsymbol{x}_{cd} + f_{cd}, \end{aligned} \tag{4.19}$$

where $\mathbf{1}_m$ denotes the standard basis vector (i.e., a $C \times 1$ constant vector with all entries equal to zero, except the mth entry which is equal to 1). The terms $\mathbf{1}_m^T\, \boldsymbol{x}_{ab}$, $\mathbf{1}_n^T\, \boldsymbol{x}_{cd}$, and $\mathbf{1}_m^T\, \boldsymbol{Q}_{cd}^{ab}\, \mathbf{1}_n$ simply denote the mth entry of vector $\boldsymbol{x}_{ab}$, the nth entry of vector $\boldsymbol{x}_{cd}$, and the entry in the mth

row and n^{th} column of matrix $\boldsymbol{Q}_{cd}^{ab}$, respectively. From (4.18), we have:

$$\boldsymbol{x}_{ab}^T \, \boldsymbol{M}_{cd}^{ab} \, \boldsymbol{x}_{cd} f_{cd} = \mathbf{1}^T \left(\boldsymbol{Q}_{cd}^{ab} \circ \boldsymbol{M}_{cd}^{ab} \right) \mathbf{1}. \tag{4.20}$$

Note that $\boldsymbol{M}_{cd}^{ab}$ is a constant matrix, thus eq. (4.20) is a linear equality in terms of $\boldsymbol{Q}_{cd}^{ab}$. In addition, we have, $\mathrm{tr}(\boldsymbol{Q}_{cd}^{ab}) = \left(\mathbf{1}^T \boldsymbol{u}_{cd}^{ab}\right) f_{cd}$.

Step 4. We first note that:

$$\underset{\boldsymbol{x},\boldsymbol{y},f}{\textbf{minimize}} \; \underset{a,b:(a,b)\in\mathcal{L}}{\textbf{maximum}} \; \frac{\gamma_{ab}}{f_{ab}\, c^0} = \underset{\boldsymbol{x},\boldsymbol{y},f}{\textbf{maximize}} \; \underset{a,b:(a,b)\in\mathcal{L}}{\textbf{minimum}} \; \frac{f_{ab}\, c^0}{\gamma_{ab}} \tag{4.21}$$

Here equality indicates *equivalence* of the two optimization problems. By defining $\delta = \min(f_{ab}\, c^0/\gamma_{ab})$ for all $(a,b) \in \mathcal{L}$, solving the right hand side in (4.21) is equivalent to maximizing δ subject to the constraint $\delta \leq \frac{f_{ab}\, c^0}{\gamma_{ab}}$ for all links $(a,b) \in \mathcal{L}$.

Problem (4.13) can now be replaced by its *equivalent* linear mixed-integer program. Given

$I, C, L, \gamma_{ab}, c^0$, and $\boldsymbol{M}_{cd}^{ab}$ for all $a, b, c, d \in \mathcal{N}$ such that $(a,b),(c,d) \in \mathcal{L}$,

$$
\begin{aligned}
&\underset{\boldsymbol{x},\boldsymbol{y},\boldsymbol{u},\boldsymbol{v},\boldsymbol{Q},\delta,f}{\textbf{maximize}}\ \delta \\
&\textbf{subject to}\quad \boldsymbol{x}_{ab} = \boldsymbol{x}_{ba}, \\
&\qquad \mathbf{1}^T\, \boldsymbol{x}_{ab} = 1, \\
&\qquad \mathbf{1}^T\, \boldsymbol{y}_{ab} = 1, \\
&\qquad \mathbf{1}^T\, \boldsymbol{u}_{ac}^{ab} = \mathbf{1}^T\, \boldsymbol{v}_{ac}^{ab}, \\
&\qquad f_{ab} + \textstyle\sum_{c,d:(c,d)\in\mathcal{L}} \mathbf{1}^T\big(\boldsymbol{Q}_{cd}^{ab} \circ \boldsymbol{M}_{cd}^{ab}\big)\, \mathbf{1} \le 1, \\
&\qquad \boldsymbol{x}_{ab} + \boldsymbol{x}_{cd} - \boldsymbol{u}_{cd}^{ab} \preceq \mathbf{1}, \\
&\qquad \boldsymbol{u}_{cd}^{ab} \preceq \boldsymbol{x}_{ab}, \\
&\qquad \boldsymbol{u}_{cd}^{ab} \preceq \boldsymbol{x}_{cd}, \\
&\qquad \boldsymbol{y}_{ab} + \boldsymbol{y}_{ac} - \boldsymbol{v}_{ac}^{ab} \preceq \mathbf{1}, \\
&\qquad \boldsymbol{v}_{ac}^{ab} \preceq \boldsymbol{y}_{ab}, \\
&\qquad \boldsymbol{v}_{ac}^{ab} \preceq \boldsymbol{y}_{ac}, \\
&\qquad \mathbf{1}_m^T\, \boldsymbol{Q}_{cd}^{ab}\, \mathbf{1}_n \le \mathbf{1}_m^T\, \boldsymbol{x}_{ab}, && \forall\, m,n=1,\ldots,C \\
&\qquad \mathbf{1}_m^T\, \boldsymbol{Q}_{cd}^{ab}\, \mathbf{1}_n \le \mathbf{1}_n^T\, \boldsymbol{x}_{cd}, && \forall\, m,n=1,\ldots,C \\
&\qquad \mathbf{1}_m^T \boldsymbol{x}_{ab} + \mathbf{1}_n^T \boldsymbol{x}_{cd} - 2 + f_{cd} \le \mathbf{1}_m^T \boldsymbol{Q}_{cd}^{ab} \mathbf{1}_n, && \forall\, m,n=1,\ldots,C \\
&\qquad \mathbf{1}_m^T \boldsymbol{Q}_{cd}^{ab} \mathbf{1}_n \le 2 - \mathbf{1}_m^T \boldsymbol{x}_{ab} - \mathbf{1}_n^T \boldsymbol{x}_{cd} + f_{cd}, && \forall\, m,n=1,\ldots,C \\
&\qquad \delta \le f_{ab}\left(c^0/\gamma_{ab}\right), \\
&\textbf{where}\quad \boldsymbol{x}_{ab} \in \{0,1\}^C,\ \boldsymbol{y}_{ab} \in \{0,1\}^I,\ \delta, f_{ab} > 0,\ \boldsymbol{u}_{cd}^{ab} \succeq \mathbf{0},\ \boldsymbol{v}_{cd}^{ab} \succeq \mathbf{0}, \boldsymbol{Q}_{cd}^{ab} \succeq \mathbf{0}, \\
&\qquad \forall\, a,b,c,d \in \mathcal{N},\ (a,b),(c,d) \in \mathcal{L}.
\end{aligned}
\tag{4.22}
$$

Note that the optimal solutions obtained from (4.13) and (4.22) are the same. There are effective commercial software (such as CPLEX [31]) to solve linear mixed-integer programs. Most of them use branch-and-bound algorithm [33]. The linear mixed-integer problem (4.22) has $L(C+I)$ binary variables and $L^2(C+I+C^2)+L+1$ real variables, respectively. In general, problem (4.22) can be solved in practice for only *small-scale* and *medium-scale* MC-WMNs. Using some heuristic algorithms similar to Algorithm 4 in Section 3.3.1, we can also *approximately* solve problem (4.22) for large-scale MC-WMNs.

4.3 Performance Evaluation

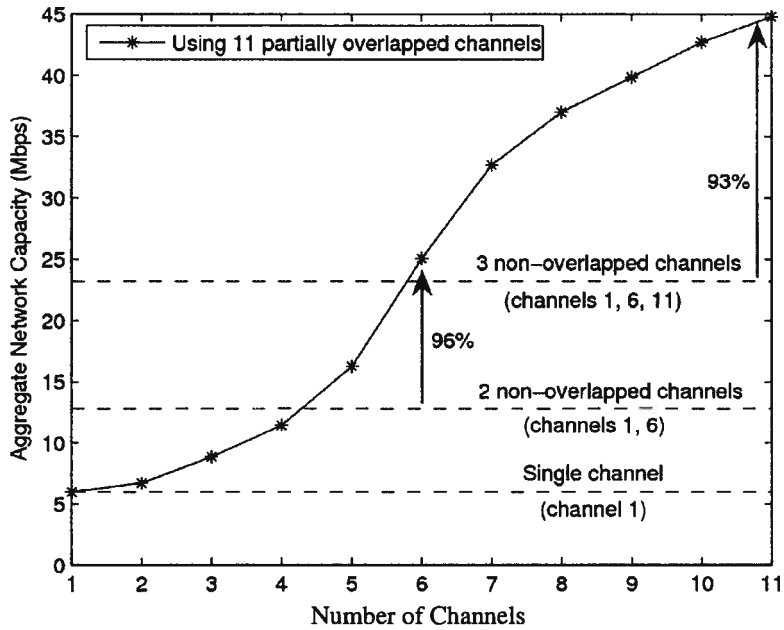
In this section, we evaluate the performance gain when not only the non-overlapped channels, but also the partially overlapped channels are being used. In the simulation model, the size of the network field is 1 km $\times$ 1 km. Five different random scenarios are simulated. In each scenario, the WMN consists of 15 wireless mesh routers that are randomly located in the field. The routers are equipped with 3 NICs. (i.e., $I = 3$). The IEEE 802.11b with 11 Mbps nominal data rate (i.e., $c^0 = 11 \times 10^6$) is being simulated. Thus, up to 11 partially overlapped frequency channels are available (see Fig. 4.1). Three of them (i.e., channels 1, 6, 11) are non-overlapped. The $SINR_{min}$ is set to be 13 dB. The value of κ is equal to 2. The transmission power is the same for all routers. For each scenario, 30 source and destination pairs are randomly selected to generate UDP traffic. We obtain the global optimal solution for linear mixed-integer program in (4.22) by using branch-and-bound algorithm [33]. To evaluate the performance, two metrics are considered: 1) aggregate network capacity, and 2) bottleneck link utilization (i.e., $1/\delta$).

The simulation results are shown in Fig. 4.3. In this figure, each point is the average of

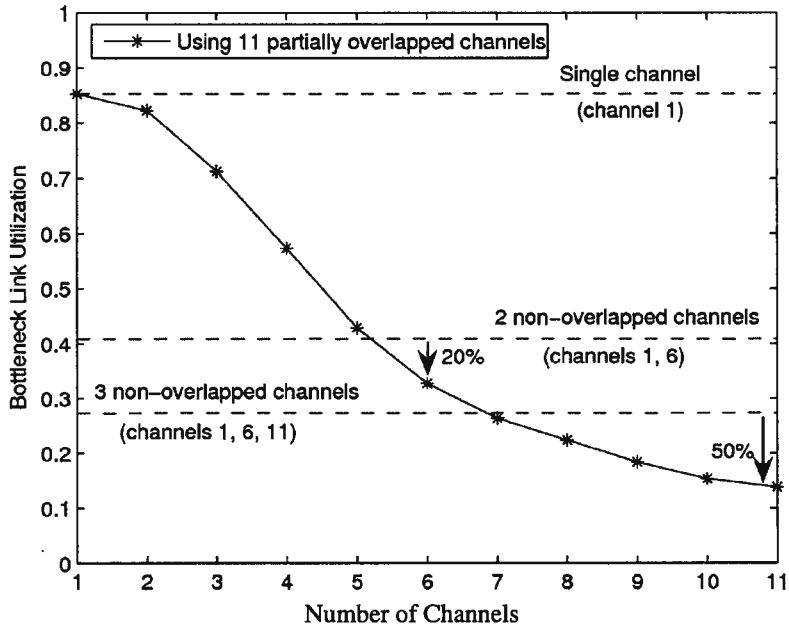
measurements for all five simulated scenarios. The dashed lines correspond to the measured performance metrics when a single channel (i.e., channel 1), two non-overlapped channels (i.e., channels 1, 6), and three non-overlapped channels (i.e., channels 1, 6, 11) are being used, respectively. It is observed that, by using the partially overlapped channels 1, 2, 3, 4, 5, and 6 instead of using only non-overlapped channels 1 and 6, the aggregate network capacity increases by 96% and the bottleneck link utilization decreases by 20%. On the other hand, by using all partially overlapped channels 1, 2, $\cdots$, 10, and 11 instead of using only non-overlapped channels 1, 6, and 11, the aggregate network capacity increases by 93% and the bottleneck link utilization decreases by 50%. Note that the performance improvements are achieved without using extra resources (frequency spectrum). Thus, the spectrum is utilized more efficiently when partially overlapped channels are being used.

4.4 Summary

In this chapter, we proposed a joint channel assignment, interface assignment, and scheduling algorithm for MC-WMNs when all non-overlapped and partially-overlapped channels are being used. The joint problem is formulated as a linear mixed-integer program with a few integer variables. The computational complexity is low and is feasible for implementation in practical networks. Simulation results show that there is a significant performance improvement in terms of a higher aggregate network capacity and a lower bottleneck link utilization when all the partially overlapped channels within the IEEE 802.11b frequency band are being used.



(a) Aggregate network capacity



(b) Bottleneck link utilization

Figure 4.3: Performance comparison within the IEEE 802.11b frequency band.

Chapter 5

Utility Optimal Random Access: Reduced Complexity, Fast Convergence, and Robust Performance

Different from the cross-layer designs in the previous chapters, in this chapter, we narrowly focus on the MAC problem in wireless ad hoc networks. In general, there are two major types of wireless MAC protocols: scheduling-based (e.g., in cellular systems) and contention-based (e.g., in wireless local area networks). In this chapter, we focus on the study of contention-based MAC, where wireless nodes *randomly* and *distributively* access the shared communication channel with certain transmission (persistent) probabilities.

The contention-based protocols are scalable and inherently flexible, but they typically have poor performance due to insufficient feedback. For example, in IEEE 802.11 distributed coordination function (DCF) [9], a node updates its transmission probability based on the *binary* feedback of its data transmission: success (no collision) or failure. This leads to low throughput

and unfair resource allocation. Such mechanism also cannot achieve a stable equilibrium [97].

In this chapter, we design distributed contention-based random MAC algorithms through the general framework of NUM [44]. Several related algorithms that are also proposed based on the same NUM framework include [47, 51, 56, 98, 99]. They have various performance bottlenecks due to one or more of the following: (1) extensive message passing among nodes to achieve semi-distributed implementation, (2) synchronous updates of contention probabilities that require homogeneous computational capabilities and software implementations among wireless nodes, (3) small update stepsizes to guarantee convergence with typically slow speed, and (4) supporting only a limited range of utility functions due to non-convexity.

Our proposed algorithms overcome the performance bottlenecks of previous proposed NUM-based random access algorithms in all four aspects. First, they only require limited message passing (i.e., signalling) among nodes. Based on the messages from other nodes, each node updates its persistent probabilities by solving a *local* and *myopic* optimization problem in an attempt to maximize the *total* network utility. Compared to the NUM-based random access algorithm in [98], our algorithms can reduce the total signalling overhead by more than a factor of 10. Second, our algorithms allow fully asynchronous updates of messages and contention probabilities. They can tolerate arbitrary large and finite asynchronism and message delays and are also robust to message losses. For example, even when the packet loss rate of the underlying communication channel is down to 0.5 (i.e., on average, half of the messages are lost), our algorithms can still achieve the optimal performance within a short time. Third, in our algorithms, nodes update their contention probabilities through best response updates, thus no small stepsizes are needed in the update. This enables our algorithms to achieve a much faster convergence compared with the previously proposed subgradient-based update methods (e.g., in [47, 51, 56, 98]). Finally,

our algorithms have provable convergence property under a wider range of utility functions, even if the NUM problem cannot be transformed into a convex optimization problem. The analysis techniques we use here are quite general and can be used to tackle other non-convex optimization problems in communications and networking.

Besides the NUM-based approach, another related thread of research focuses on the analysis of random MAC algorithms using game theory (e.g., [100, 101, 102, 103]). The focus is on noncooperative interaction among nodes, while here, we focus on *global* network optimization.

The rest of this chapter is organized as follows. The system model is described in Section 5.1. Our proposed algorithms are presented in Section 5.2. The convergence, optimality, and robustness of our proposed algorithms are analytically proved in Section 5.3. Simulation results are shown in Section 5.4. Conclusions are discussed in Section 5.5.

5.1 System Model

Consider a wireless ad-hoc network. Let $\mathcal{N}=\{1,\ldots,N\}$ denote the set of nodes and $\mathcal{L}=\{1,\ldots,L\}$ denote the set of unidirectional wireless links. For each node $n \in \mathcal{N}$, we denote the set of its *outgoing* links by $\mathcal{L}_n \subset \mathcal{L}$, with size $L_n = |\mathcal{L}_n|$. Each node n has L_n separate queues, each queue holds the packets for one of its outgoing links of node n (see Fig. 5.1). Time is divided into equal-length slots. At each time slot, node n may choose to transmit on one of its outgoing links $i \in \mathcal{L}_n$ with a *persistent probability* p_i. The probabilities need to satisfy $\sum_{i\in\mathcal{L}_n} p_i \leq P_n^{\max} < 1$, where $P_n^{\max}$ denotes the given maximum *total* persistent probability. That is, node n may remain silent in some slots. For the network in Fig. 5.1, node a has $L_a = 2$ outgoing links, where $\mathcal{L}_a = \{1, 2\}$. In this node, those packets which are destined to node b are enqueued in queue 1. Similarly, the

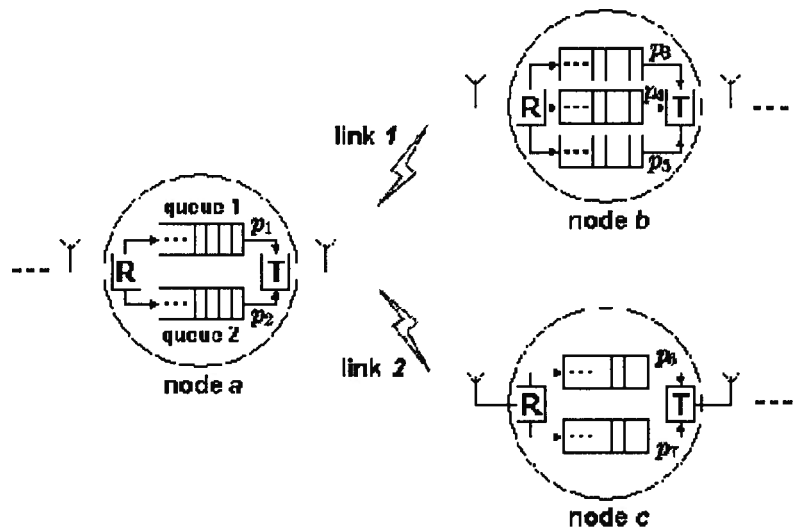


Figure 5.1: A sample wireless ad-hoc network. We have, $\mathcal{L}_a = \{1,2\}$, $\mathcal{L}_b = \{3,4,5\}$, and $\mathcal{L}_c = \{6,7\}$. In node a, those packets which are assigned to be sent to node b (over link 1) are enqueued in queue 1. Similarly, those packets that are assigned to be sent to node c (over link 2) are enqueued in queue 2. R and T boxes represent receiver and transmitter units, respectively.

packets which are destined to node c are enqueued in queue 2. At each time slot, a packet from queue 1 is sent over wireless link 1 with probability p_1, and a packet from queue 2 is sent over wireless link 2 with probability p_2. Notice that links 1 and 2 will *not* be active at the same time.

For each node $n \in \mathcal{N}$, if the *receiver* node of link $i \in \mathcal{L}_n$ is within the *interference range* of another node $s \in \mathcal{N}\backslash\{n\}$, then any transmission by node s (i.e., transmission on any link $j \in \mathcal{L}_s$) interferes with transmissions of link i. Those nodes which interfere with transmissions of link i are denoted by set $\mathcal{N}_i$. For each node $n \in \mathcal{N}$, let r_i denote the average data rate for link $i \in \mathcal{L}_n$, which is a function of the persistent probability vector $\boldsymbol{p} = (p_i, \forall i \in \mathcal{L})$ of all links. We have [80]:

$$r_i(\boldsymbol{p}) = \gamma_i p_i \prod_{s \in \mathcal{N}_i} (1 - \sum_{j \in \mathcal{L}_s} p_j), \tag{5.1}$$

where γ_i denotes the fixed peak data rate for link i (i.e., the rate achieved by link i if no node

in set $\mathcal{N}_i$ is active). To ensure that no link encounter starvation, for each node n and any link $i \in \mathcal{L}_n$, we require $p_i \geq P_n^{\min} > 0$ and $L_n P_n^{\min} \leq P_n^{\max}$. Here $P_n^{\min}$ denotes the given minimum persistent probability. Thus,

$$r_i(\boldsymbol{p}) \geq \gamma^{\min} P^{\min} (1 - P^{\max})^{N-1} > 0, \qquad \forall\, i \in \mathcal{L}, \tag{5.2}$$

where

$$P^{\min} = \min_{n \in \mathcal{N}} P_n^{\min} \tag{5.3}$$

$$P^{\max} = \max_{n \in \mathcal{N}} P_n^{\max} \tag{5.4}$$

$$\gamma^{\min} = \min_{i \in \mathcal{L}} \gamma_i \tag{5.5}$$

$$\gamma^{\max} = \max_{i \in \mathcal{L}} \gamma_i. \tag{5.6}$$

Each link $i \in \mathcal{L}$ has a *utility* which is an increasing and concave function of r_i and indicates link i's degree of satisfaction on its average data rate. The utility of link i is denoted by $u(r_i(\boldsymbol{p}))$, which is also a function of the persistent probabilities $\boldsymbol{p}$ of all links. We are interested in finding the value of $\boldsymbol{p}$ that solves the following NUM problem [53]:

$$\max_{\boldsymbol{p} \in \mathcal{P}} \sum_{i \in \mathcal{L}} u(r_i(\boldsymbol{p})), \tag{5.7}$$

where the feasible persistent probability region is

$$\mathcal{P} = \left\{ \boldsymbol{p} : p_i \geq P_n^{\min},\ \sum_{j \in \mathcal{L}_n} p_j \leq P_n^{\max},\ \forall\, n \in \mathcal{N}, i \in \mathcal{L}_n \right\}$$

and the utility function is α-fair (see Section 1.2.2). That is, [50]:

$$u(r_i) = \begin{cases} (1-\alpha)^{-1}\, r_i^{1-\alpha}, & \text{if } \alpha \in (0,1) \cup (1,\infty), \\ \log r_i, & \text{if } \alpha = 1, \end{cases} \qquad \forall\, i \in \mathcal{L}. \tag{5.8}$$

Notice that problem (5.7) has the form of the NUM problem in (1.18) where the optimization variables are the persistent probabilities. Also recall from Section 1.2.2 that using (5.8), a wide range of efficient and fair allocations can be modeled. In particular, problem (5.7) reduces to throughput maximization with $\alpha \to 0$, to proportional fair allocation with $\alpha = 1$, to harmonic mean fair allocation with $\alpha = 2$, and to max-min fairness with $\alpha \to \infty$.

Although the objective function in problem (5.7) is concave in link rates $\boldsymbol{r} = (r_i, \forall\, i \in \mathcal{L})$, it is not concave in persistent probabilities $\boldsymbol{p}$ due to the product form of the average rate in (5.1). Thus, finding the optimal solution for this *non-convex* and *constrained* optimization problem is difficult even in a centralized fashion. In this chapter, we propose two algorithms which are able to find the optimal solution of problem (5.7) in a distributed fashion under easily verifiable sufficient technical conditions. In comparison with the existing algorithms, our proposed algorithms do not require any synchronization, converge much faster, are more robust to message delay and message loss, and support a wider range of α values in the utility function.

5.2 Algorithms

In this section, we propose two distributed algorithms to solve problem (5.7), one for *single-cell* topologies in Section 5.2.1 and another one for *general* topologies in Section 5.2.2. In both algorithms, each node n performs a *myopic* and local optimization, i.e., optimizing the *total*

network utility by choosing the persistent probabilities of its own outgoing links, assuming others do not change theirs. Despite the complexity of the problem, we show that the solution of this local optimization problem can be obtained in closed form, facilitated by limited message passing among nodes and a simple local sorting procedure. Various properties of the algorithms, including convergence, optimality, and robustness, will be proved in Section 5.3.

5.2.1 Single-Cell Topology

We begin by considering a *single-cell* topology, where all links interfere with each other. That is, for each $n \in \mathcal{N}$ and any $i \in \mathcal{L}_n$, the interference node set $\mathcal{N}_i = \mathcal{N} \backslash \{n\}$. This models some important practical wireless networks including *wireless personal area networks* where multiple wireless devices interact with each other over short distances in a piconet, as well as indoor *wireless local area networks* where several wireless devices communicate with an access point and each other (e.g., in a large conference room). We extend our work to the general topology case in Section 5.2.2.

Node n's Local Optimization Problem

We begin by considering a *single-cell* topology, where all links interfere with each other. That is, for each $n \in \mathcal{N}$ and any $i \in \mathcal{L}_n$, the interference node set $\mathcal{N}_i = \mathcal{N} \backslash \{n\}$. This models some important practical wireless networks including *wireless personal area networks* where multiple wireless devices interact with each other over short distances, as well as indoor *wireless local area networks* where several wireless devices communicate with an access point and each other.

Node n's Local Optimization Problem

For each node n, let $\boldsymbol{p}_n = (p_i, \forall i \in \mathcal{L}_n)$ denote the persistent probabilities of its outgoing links. Also let $\boldsymbol{p}_{-n} = (p_j, \forall j \in \mathcal{L} \backslash \mathcal{L}_n)$ denote the persistent probabilities of all links *other than* the outgoing links of node n. Consider the following *local* and *myopic* optimization problem:

$$\max_{\boldsymbol{p}_n \in \mathcal{P}_n} \sum_{i \in \mathcal{L}} u(r_i(\boldsymbol{p}_n, \boldsymbol{p}_{-n})), \tag{LOCAL-NUM}$$

where the feasible persistent probability region for node n is

$$\mathcal{P}_n = \left\{ \boldsymbol{p}_n : \sum_{i \in \mathcal{L}_n} p_i \leq P_n^{\max} \leq 1, p_i \geq P_n^{\min} > 0, \forall\, i \in \mathcal{L}_n \right\}, \tag{5.9}$$

By solving problem (LOCAL-NUM), node n can select $\boldsymbol{p}_n$ such that the *total* network utility is maximized *assuming* that $\boldsymbol{p}_{-n}$ is fixed (i.e., none of the other nodes change their persistent probabilities). It is clear that nodes are *not* selfish in this case, and they *cooperate* with each other. This is necessary for achieving the optimal network performance in a distributed fashion.

Although problem (LOCAL-NUM) is difficult to solve, we can convert it to an equivalent instructive representation. Its objective function in the single-cell case can be written as:

$$\sum_{i \in \mathcal{L}} u(r_i(\boldsymbol{p}_n, \boldsymbol{p}_{-n})) = (1-\alpha)^{-1} \left(\prod_{c \in \mathcal{N} \backslash \{n\}} (1 - \sum_{l \in \mathcal{L}_c} p_l)\right)^{1-\alpha} \Big[\sum_{i \in \mathcal{L}_n} (\gamma_i p_i)^{1-\alpha} + \\ (1 - \sum_{i \in \mathcal{L}_n} p_i)^{1-\alpha} \sum_{s \in \mathcal{N} \backslash \{n\}} \sum_{j \in \mathcal{L}_s} (\gamma_j p_j)^{1-\alpha} / \left(1 - \sum_{l \in \mathcal{L}_s} p_l\right)^{1-\alpha} \Big].$$

Since the multiplicative term $\left(\prod_{c \in \mathcal{N} \backslash \{n\}} (1 - \sum_{l \in \mathcal{L}_c} p_l)\right)^{1-\alpha}$ does not depend on the vector variable $\boldsymbol{p}_n$, problem (LOCAL-NUM) can be *equivalently* written as:

$$\max_{\boldsymbol{p}_n \in \mathcal{P}_n} \sum_{i \in \mathcal{L}_n} u(\gamma_i p_i) + v_n(\boldsymbol{p}_{-n})\, u\left(1 - \sum_{i \in \mathcal{L}_n} p_i\right), \tag{5.10}$$

where

$$v_n(\boldsymbol{p}_{-n}) = \sum_{s \in \mathcal{N} \backslash \{n\}} \left(1 - \sum_{l \in \mathcal{L}_s} p_l\right)^{\alpha-1} \left(\sum_{j \in \mathcal{L}_s} (\gamma_j p_j)^{1-\alpha}\right). \tag{5.11}$$

Since $\sum_{i\in\mathcal{L}_n} u(\gamma_i p_i)$ and $u(1-\sum_{i\in\mathcal{L}_n} p_i)$ are strictly concave functions with respect to $\boldsymbol{p}_n$, and $v_n(\boldsymbol{p}_{-n})$ is independent of $\boldsymbol{p}_n$, problem (5.10) is strictly concave in local variable vector $\boldsymbol{p}_n$. In other words, there exists a unique optimal solution of problem (5.10) and thus problem (LOCAL-NUM).

Closed-Form Solution of Problem (LOCAL-NUM)

Next, we show how to obtain a *closed-form* optimal solution for problem (5.10). Consider a node $n \in \mathcal{N}$ and the set of its outgoing links $\mathcal{L}_n$. We define a permutation, $i_1, \cdots, i_{L_n}$, of the link indices in set $\mathcal{L}_n$ such that for any j and l that satisfy $1 \le j \le l \le L_n$, we have $\sqrt[\alpha]{\gamma_{i_j}^{\alpha-1}} \le \sqrt[\alpha]{\gamma_{i_l}^{\alpha-1}}$. Thus, in the case of $\alpha \ge 1$, we have $\gamma_{i_1} \le \cdots \le \gamma_{i_{L_n}}$, and in the case of $\alpha \in (0,1)$, we have $\gamma_{i_1} \ge \cdots \ge \gamma_{i_{L_n}}$. For example, let $\mathcal{L}_n = \{4, 7, 12\}$, $\gamma_4 = 18$ Mbps, $\gamma_7 = 24$ Mbps, and $\gamma_{12} = 6$ Mbps. If $\alpha \ge 1$, then we have $i_1 = 12$, $i_2 = 4$, and $i_3 = 7$. On the other hand, if $\alpha \in (0,1)$, then we have $i_1 = 7$, $i_2 = 4$, and $i_3 = 12$.

Let σ denote the *smallest* number in the set $\{0, \ldots, L_n - 1\}$ such that

$$1/P_n^{\min} - L_n + \sigma \le \sum_{l=1}^{\sigma} \sqrt[\alpha]{\left(\gamma_{i_{\sigma+1}}/\gamma_{i_l}\right)^{\alpha-1}} + \sqrt[\alpha]{\gamma_{i_{\sigma+1}}^{\alpha-1}\, v_n(\boldsymbol{p}_{-n})}. \tag{5.12}$$

We can show that (see (5.43) in Section 5.6.1), if condition (5.12) holds for σ, then it also holds for $\sigma + 1$. We define the set $\mathcal{B}_n = \{i_{\sigma+1}, \ldots, i_{L_n}\}$, with its size $B_n = |\mathcal{B}_n| = L_n - \sigma$. Notice that if condition (5.12) does not hold for any $\sigma \in \{0, \ldots, L_n - 1\}$, then we set $\mathcal{B}_n = \{\}$ and $B_n = 0$.

Similarly, let ς denote the *smallest* number in the set $\{0, \ldots, L_n - 1\}$ such that

$$P_n^{\max}/P_n^{\min} - L_n + \varsigma \le \sum_{l=1}^{\varsigma} \sqrt[\alpha]{\left(\gamma_{i_{\varsigma+1}}/\gamma_{i_l}\right)^{\alpha-1}}. \tag{5.13}$$

Again, we can show that (see (5.44) in Section 5.6.1), if condition (5.13) holds for ς, then it also holds for $\varsigma + 1$. We define $\mathcal{C}_n = \{i_{\varsigma+1}, \ldots, i_{L_n}\}$, and its size $C_n = |\mathcal{C}_n| = L_n - \varsigma$. If condition (5.13) does not hold for any $\varsigma \in \{0, \ldots, L_n - 1\}$, then we set $\mathcal{C}_n = \{\}$ with $C_n = 0$.

We now define $\mathcal{A}_n = \mathcal{B}_n \cup \mathcal{C}_n$ with its size $A_n = |\mathcal{A}_n| = L_n - \kappa$ where $\kappa = \min\{\sigma, \varsigma\}$. In fact, $\mathcal{A}_n = \{i_{\kappa+1}, \ldots, i_{L_n}\}$. Depending on the value of $v_n(\boldsymbol{p}_{-n})$, either $\mathcal{A}_n = \mathcal{B}_n$ or $\mathcal{A}_n = \mathcal{C}_n$ (see Section 5.6.1). Using $\mathcal{A}_n$, the closed-form solution of problem (5.10) can be obtained as follows.

Theorem 10 *For each node $n \in \mathcal{N}$, the unique optimal solution of problem (5.10) is $\boldsymbol{p}_n^*(\boldsymbol{p}_{-n}) = \boldsymbol{f}_n(\boldsymbol{p}_{-n}) = (f_i(\boldsymbol{p}_{-n}),\ \forall\, i \in \mathcal{L}_n)$, where for each link $i \in \mathcal{L}_n$, the mapping $f_i(\boldsymbol{p}_{-n})$ is defined as*

$$f_i(\boldsymbol{p}_{-n}) = \begin{cases} P_n^{\min}, & \text{if } i \in \mathcal{A}_n, \\ \left[\frac{1}{\sqrt[\alpha]{\gamma_i^{\alpha-1}}}\left(\frac{1-A_n P_n^{\min}}{w_n + \sqrt[\alpha]{v_n(\boldsymbol{p}_{-n})}}\right)\right]_{P_n^{\min}}^{(P_n^{\max} - A_n P_n^{\min})/(\sqrt[\alpha]{\gamma_i^{\alpha-1}} w_n)}, & \text{if } i \in \mathcal{L}_n \backslash \mathcal{A}_n, \end{cases} \tag{5.14}$$

where $[x]_b^a = \max[\min[x, a], b]$ and we have:

$$w_n = \textstyle\sum_{j \in \mathcal{L}_n \backslash \mathcal{A}_n} \sqrt[\alpha]{(1/\gamma_j)^{\alpha-1}}. \tag{5.15}$$

The proof of Theorem 10 is available in Section 5.6.1. The key is to show that $\boldsymbol{f}_n(\boldsymbol{p}_{-n})$ satisfies the *necessary* and *sufficient* Karush-Kuhn-Tucker (KKT) optimality conditions [16, pp. 244]. Since problems (LOCAL-NUM) and (5.10) are equivalent, $\boldsymbol{p}_n^*(\boldsymbol{p}_{-n})$ is also the unique optimal solution of problem (LOCAL-NUM). We notice that, regardless of the selected system parameters and the value of v_n, for each $i \in \mathcal{L}_n \backslash \mathcal{A}_n$, the upper bound $(P_n^{\max} - A_n P_n^{\min})/(\sqrt[\alpha]{\gamma_i^{\alpha-1}} w_n)$ in (5.14) is always greater than or equal to the lower bound $P_n^{\min}$ (see (5.63) and (5.75) in Section 5.6.1).

It is clear that to compute $f_i(\boldsymbol{p}_{-n})$ in (5.14), the only information node n needs from other nodes is $v_n(\boldsymbol{p}_{-n})$. If each node s announces a message m_s where

$$m_s = \left(1 - \textstyle\sum_{j \in \mathcal{L}_s} p_j\right)^{\alpha-1} \left(\textstyle\sum_{j \in \mathcal{L}_s} (\gamma_j p_j)^{1-\alpha}\right), \quad \forall s \in \mathcal{N} \backslash \{n\}, \tag{5.16}$$

node n can compute $v_n(\boldsymbol{p}_{-n}) = \sum_{s \in \mathcal{N} \backslash \{n\}} m_s$. This motivates us to propose our first algorithm.

Algorithm 5 *Executed by each node $n \in \mathcal{N}$ in a single-cell topology.*

1: Allocate memory for messages $\boldsymbol{m} = (m_1, \cdots, m_N)$.
2: Randomly choose $p_i \geq P_n^{\min} > 0$ for each link $i \in \mathcal{L}_n$ such that $\sum_{i \in \mathcal{L}_n} p_i \leq P_n^{\max} \leq 1$.
3: Randomly choose $m_s > 0$ for all $s \in \mathcal{N}$.
4: **repeat**
5: Transmit on outgoing link $i \in \mathcal{L}_n$ with probability p_i.
6: **if** $t \in T_{n,p}$ **then**
7: Set $\mathcal{A}_n = getA(n, v_n(\boldsymbol{p}_{-n}), L_n, \gamma_1, \ldots, \gamma_{L_n})$.
8: Update $p_i = P_n^{\min}$ for all $i \in \mathcal{A}_n$.
9: Update $p_i = \left[\frac{1}{\sqrt[\alpha]{\gamma_i^{\alpha-1}}} \frac{1 - A_n P_n^{\min}}{\left(w_n + \sqrt[\alpha]{v_n(\boldsymbol{p}_{-n})}\right)} \right]_{P_n^{\min}}^{(P^{\max} - A_n P_n^{\min})/(\sqrt[\alpha]{\gamma_i^{\alpha-1}} w_n)}$ for all $i \in \mathcal{L}_n \backslash \mathcal{A}_n$.
10: **if** $t \in T_{n,qm}$ **then**
11: Update $m_n = \left(1 - \sum_{i \in \mathcal{L}_n} p_i\right)^{\alpha-1} \left(\sum_{i \in \mathcal{L}_n} (1/\gamma_i)^{\alpha-1} (1/p_i)^{\alpha-1}\right)$.
12: Broadcast m_n.
13: **if** a message is received **then** Update $\boldsymbol{m}$.
14: **until** node n decides to leave the network.

A Distributed MAC Algorithm

Our distributed random MAC algorithm is given in Algorithm 5. In this algorithm, each node $n \in \mathcal{N}$, regardless of how many outgoing links it has, announces only a *single* message m_n. All nodes choose the persistent probabilities of their outgoing links based on the received messages from other nodes. The persistent probabilities and messages are *asynchronously* updated. Let $T_{n,p}$ and $T_{n,m}$ be two unbounded sets of time slots at which node n updates $\boldsymbol{p}_n$ and m_n, respectively. We assume that the asynchronism of the updates is bounded; i.e., there exists a finite H (called *asynchronism measure* [18]) such that:

$$\forall t_1 \in T_{n,p}\ ,\ \exists t_2 \in T_{n,p} \quad \text{such that} \quad t_2 - t_1 \leq H, \tag{5.17}$$

$$\forall t_3 \in T_{n,m}\ , \exists t_4 \in T_{n,m} \quad \text{such that} \quad (t_4 - t_3) + D \leq H, \tag{5.18}$$

where D denotes an upper bound on *communication delay* (e.g., queueing or propagation delay).

1: **function** $getA(n, v_n(\boldsymbol{p}_{-n}), L_n, \gamma_1, \ldots, \gamma_{L_n})$
2: Set $\mathcal{B}_n = \{\}$ and $\mathcal{C}_n = \{\}$.
3: **if** $\alpha \geq 1$ **then** Set $i_1, \ldots, i_{L_m}$ such that $\gamma_{i_1} \leq \gamma_{i_2} \leq \ldots \leq \gamma_{i_{L_n}}$.
4: **else** Set $i_1, \ldots, i_{L_m}$ such that $\gamma_{i_1} \geq \gamma_{i_2} \geq \ldots \geq \gamma_{i_{L_n}}$.
5: **for** $\sigma = 0, \ldots, L_n - 1$ **do**
6: **if** $1/P_n^{\min} - L_n + \sigma \leq \sum_{l=1}^{\sigma} \sqrt[\alpha]{\left(\gamma_{i_{\sigma+1}}/\gamma_{i_l}\right)^{\alpha-1}} + \sqrt[\alpha]{\gamma_{i_{\sigma+1}}{}^{\alpha-1} v_n(\boldsymbol{p}_{-n})}$ **then**
7: Set $\mathcal{B}_n = \{i_{\sigma+1}, \ldots, i_{L_n}\}$.
8: Goto Line 10.
9: **end for**
10: **for** $\varsigma = 0, \ldots, L_n - 1$ **do**
11: **if** $P_n^{\max}/P_n^{\min} - L_n + \varsigma \leq \sum_{l=1}^{\varsigma} \sqrt[\alpha]{\left(\gamma_{i_{\varsigma+1}}/\gamma_{i_l}\right)^{\alpha-1}}$ **then**
12: Set $\mathcal{C}_n = \{i_{\varsigma+1}, \ldots, i_{L_n}\}$.
13: Goto Line 15.
14: **end for**
15: Set $\mathcal{A}_n = \mathcal{B}_n \cup \mathcal{C}_n$.
16: **return** $\mathcal{A}_n$.

From (5.17), each node updates the persistent probabilities of its outgoing links at least once during any time interval of length H time slots. From (5.18), the information used by each node is outdated by at most H time slots. We notice that H can be arbitrarily large as long as it is bounded. The exact value of H is *not* important and needs *not* be known by all nodes.

Compared with the distributed MAC algorithms proposed in the literature, Algorithm 5 has several distinct features: (i) less explicit message passing is needed (e.g., in the subgradient algorithm proposed in [98], each node needs to announce *two* messages), (ii) asynchronous updates with arbitrarily finite delay, which minimizes the coordination overhead and allows maximum heterogeneity among wireless nodes, and (iii) does not use any stepsizes, which avoids the slow convergence problem due to small stepsizes in the commonly used subgradient methods.

We note that if Algorithm 5 has a fixed point, then every node n achieves the optimal solution of problem (LOCAL-NUM) and each node will not change its persistent probability vector. In Section 5.3.1, we show that this fixed point indeed corresponds to the unique global optimal

solution of the non-convex problem (5.7) under proper sufficient conditions, and Algorithm 5 globally converges to such a fixed point with fast and robust performance.

5.2.2 General Topology

Now let us consider the general case, where each node is within the interference range of an *arbitrary subset* of the other nodes. For each node $n \in \mathcal{N}$ and any of its outgoing links $i \in \mathcal{L}_n$, the set of nodes that interfere with link i is an arbitrary subset of all nodes, i.e., $\mathcal{N}_i \subseteq \mathcal{N}\backslash\{n\}$. In this case, the objective function of problem (LOCAL-NUM) can be written as:

$$\begin{aligned}\sum_{i\in\mathcal{L}} u(r_i(\boldsymbol{p}_n,\boldsymbol{p}_{-n})) = (1-\alpha)^{-1}\Big[&\sum_{i\in\mathcal{L}_n} \left((\gamma_i \prod_{c\in\mathcal{N}_i}(1-\sum_{l\in\mathcal{L}_c} p_l))p_i\right)^{1-\alpha} + (1-\sum_{i\in\mathcal{L}_n} p_i)^{1-\alpha} \\ &\sum_{s\in\mathcal{N}\backslash\{n\}} \sum_{j\in\mathcal{L}_s:n\in\mathcal{N}_j} (\gamma_j p_j \prod_{c\in\mathcal{N}_j\backslash\{n\}}(1-\sum_{l\in\mathcal{L}_c} p_l))^{1-\alpha} + \\ &\sum_{s\in\mathcal{N}\backslash\{n\}} \sum_{j\in\mathcal{L}_s:n\notin\mathcal{N}_j} (\gamma_j p_j \prod_{c\in\mathcal{N}_j}(1-\sum_{l\in\mathcal{L}_c} p_l))^{1-\alpha}\Big].\end{aligned}$$

Since the last term in the bracket does not depend on vector variable $\boldsymbol{p}_n$, problem (LOCAL-NUM) can be *equivalently* written as:

$$\max_{\boldsymbol{p}_n\in\mathcal{P}_n} \sum_{i\in\mathcal{L}_n} u\left(\gamma_i'(\boldsymbol{p}_{-n})p_i\right) + v_n'(\boldsymbol{p}_{-n})\, u\left(1-\sum_{i\in\mathcal{L}_n} p_i\right), \tag{5.19}$$

where

$$\gamma_i'(\boldsymbol{p}_{-n}) = \gamma_i \prod_{s\in\mathcal{N}_i}\left(1-\sum_{l\in\mathcal{L}_s} p_l\right), \quad \forall\, i\in\mathcal{L}_n, \tag{5.20}$$

$$v_n'(\boldsymbol{p}_{-n}) = \sum_{s\in\mathcal{N}\backslash\{n\}} \sum_{j\in\mathcal{L}_s:n\in\mathcal{N}_j} (\gamma_j p_j \prod_{c\in\mathcal{N}_j\backslash\{n\}}(1-\sum_{l\in\mathcal{L}_c} p_l))^{1-\alpha}. \tag{5.21}$$

Notice that $\gamma_i'(\boldsymbol{p}_{-n})$ does not represent the peak data rate of link i. We can show that problem (5.19) is strictly concave in $\boldsymbol{p}_n$ and has a unique optimal solution.

The closed-form solution of problem (5.19) can be obtained similarly as that of problem (5.10)

in the single-cell case. For the outgoing link set $\mathcal{L}_n$ of node n, we can define a permutation of link indices in this set, $i_1, \cdots, i_{L_n}$, such that for any j and l that satisfy $1 \le j \le l \le L_n$, we have $\sqrt[\alpha]{\gamma'_{i_j}(\boldsymbol{p}_{-n})^{\alpha-1}} \le \sqrt[\alpha]{\gamma'_{i_l}(\boldsymbol{p}_{-n})^{\alpha-1}}$. In the case of $\alpha \ge 1$, we have $\gamma'_{i_1} \le \cdots \le \gamma'_{i_{L_n}}$. In the case of $\alpha \in (0,1)$, we have $\gamma'_{i_1} \ge \cdots \ge \gamma'_{i_{L_n}}$. Let σ' denote the *smallest* value in $\{0, \ldots, L_n - 1\}$ such that

$$1/P_n^{\min} - L_n + \sigma' \le \sum_{l=1}^{\sigma'} \sqrt[\alpha]{\left(\gamma'_{i_{\sigma'+1}}(\boldsymbol{p}_{-n})/\gamma'_{i_l}(\boldsymbol{p}_{-n})\right)^{\alpha-1}} + \sqrt[\alpha]{\gamma'^{\alpha-1}_{i_{\sigma'+1}}(\boldsymbol{p}_{-n})\, v'_n(\boldsymbol{p}_{-n})}. \tag{5.22}$$

Similarly, let ς'denote the *smallest* value in $\{0, \ldots, L_n - 1\}$ such that

$$P_n^{\max}/P_n^{\min} - L_n + \varsigma' \le \sum_{l=1}^{\varsigma'} \sqrt[\alpha]{\left(\gamma'_{i_{\varsigma'+1}}(\boldsymbol{p}_{-n})/\gamma'_{i_l}(\boldsymbol{p}_{-n})\right)^{\alpha-1}}. \tag{5.23}$$

We define $\mathcal{B}'_n = \{i_{\sigma'+1}, \ldots, i_{L_n}\}$, with its size $B'_n = |\mathcal{B}'_n| = L_n - \sigma'$. If condition (5.22) does not hold for any $\sigma' \in \{0, \ldots, L_n - 1\}$, then we set $\mathcal{B}'_n = \{\}$ with $B'_n = 0$. Similarly, we define $\mathcal{C}'_n = \{i_{\varsigma'+1}, \ldots, i_{L_n}\}$, with its size $C'_n = |\mathcal{C}'_n| = L_n - \varsigma'$. If condition (5.23) does not hold for any $\varsigma' \in \{0, \ldots, L_n - 1\}$, then we set $\mathcal{C}'_n = \{\}$ with $C'_n = 0$. Given $\mathcal{B}'_n$ and $\mathcal{C}'_n$, we define $\mathcal{A}'_n = \mathcal{B}'_n \cup \mathcal{C}'_n$ with its size $A'_n = |\mathcal{A}'_n| = L_n - \kappa'$ where $\kappa' = \min\{\sigma', \varsigma'\}$.

Theorem 11 *For each node $n \in \mathcal{N}$, the unique optimal solution of problem (5.19) is $\boldsymbol{p}_n^*(\boldsymbol{p}_{-n}) = \boldsymbol{f}'_n(\boldsymbol{p}_{-n}) = (f'_i(\boldsymbol{p}_{-n}),\ \forall\, i \in \mathcal{L}_n)$, where for each link $i \in \mathcal{L}_n$, the mapping $f_i(\boldsymbol{p}_{-n})$ is defined as*

$$f'_i(\boldsymbol{p}_{-n}) = \begin{cases} P_n^{\min}, & \text{if } i \in \mathcal{A}'_n, \\ \left[\dfrac{1}{\sqrt[\alpha]{\gamma'_i(\boldsymbol{p}_{-n})^{\alpha-1}}}\left(\dfrac{1 - A'_n P_n^{\min}}{w'_n(\boldsymbol{p}_{-n}) + \sqrt[\alpha]{v'_n(\boldsymbol{p}_{-n})}}\right)\right]_{P_n^{\min}}^{\frac{P_n^{\max} - A'_n P_n^{\min}}{\sqrt[\alpha]{\gamma'_i(\boldsymbol{p}_{-n})^{\alpha-1}}\, w'_n(\boldsymbol{p}_{-n})}}, & \text{otherwise,} \end{cases} \tag{5.24}$$

with $w'_n(\boldsymbol{p}_{-n}) = \sum_{j \in \mathcal{L}_n \setminus \mathcal{A}'_n} \sqrt[\alpha]{\left(1/\gamma'_j(\boldsymbol{p}_{-n})\right)^{\alpha-1}}$.

The proof of Theorem 11 is similar to that of Theorem 10. It is clear that (5.24) provides the optimal solution for problem (LOCAL-NUM) in the general topology case, which includes the

Algorithm 6 *Executed by each node $n \in \mathcal{N}$ in a general topology.*

Allocate memory for $\boldsymbol{m} = (m_{1,n}, \ldots, m_{N,n})$ and $\boldsymbol{q} = (q_1, \ldots, q_N)$.
Randomly choose $p_i \geq P_n^{\min} > 0$ for all $i \in \mathcal{L}_n$ such that $\sum_{i \in \mathcal{L}_n} p_i \leq P_n^{\max} \leq 1$.
Randomly choose $m_{s,n} > 0$ and $q_s \in (0,1)$ for all $s \in \mathcal{N}$.
repeat
  Transmit on outgoing link $i \in \mathcal{L}_n$ with probability p_i.
  if $t \in T_{n,p}$ **then**
    Set $\mathcal{A}'_n = getA(n, v'_n(\boldsymbol{p}_{-n}), L_n, \gamma'_1, \ldots, \gamma'_{L_n})$.
    Set $p_i = P_n^{\min}$ for all $i \in \mathcal{A}'_n$.
    Set $p_i = \left[\frac{1}{\sqrt[\alpha]{\gamma_i'^{\alpha-1}(\boldsymbol{p}_{-n})}} \frac{1 - A'_n P_n^{\min}}{\left(w'_n(\boldsymbol{p}_{-n}) + \sqrt[\alpha]{v'_n(\boldsymbol{p}_{-n})}\right)} \right]_{P_n^{\min}}^{\frac{P^{\max} - A'_n P_n^{\min}}{\sqrt[\alpha]{\gamma_i'^{\alpha-1}(\boldsymbol{p}_{-n}) w'_n(\boldsymbol{p}_{-n})}}}$ for all $i \in \mathcal{L}_n \backslash \mathcal{A}'_n$.
  if $t \in T_{n,m}$ **then**
    Update $q_n = 1 - \sum_{i \in \mathcal{L}_n} p_i$.
    Update $m_{n,s} = \sum_{i \in \mathcal{L}_n : s \in \mathcal{N}_i} 1/(\gamma_i p_i \prod_{c \in \mathcal{N}_i \backslash \{s\}} q_c)^{\alpha-1}$ for any $s \neq n$.
    Inform $m_{n,s}$ to all $s \in \cup_{i \in \mathcal{L}_n} \mathcal{N}_i$.
    Inform q_n to all $s \in \mathcal{N} \backslash \{n\}$ such that $\exists\, j \in \mathcal{L}_s$ to have $n \in \mathcal{N}_j$.
  if a message is received **then** Update $\boldsymbol{m}$ and $\boldsymbol{q}$.
until node n decides to leave the network.

single-cell case as a special case. We can define node s' *messages* as

$$q_s = 1 - \sum_{j \in \mathcal{L}_s} p_j, \qquad \text{and} \qquad m_{s,n} = \sum_{j \in \mathcal{L}_s : n \in \mathcal{N}_j} 1/(\gamma_j p_j \prod_{c \in \mathcal{N}_j \backslash \{n\}} q_c)^{\alpha-1}. \tag{5.25}$$

Then $v'_n(\boldsymbol{p}_{-n}) = \sum_{s \in \mathcal{N} \backslash \{n\}} m_{s,n}$ and $\gamma'_i(\boldsymbol{p}_{-n}) = \gamma_i \prod_{s \in \mathcal{N}_i} q_s$ for all $i \in \mathcal{L}_n$. Message q_s simply denotes the probability that node s remains silent at a time slot. Also note that for each node $n \neq s$, if there does not exist any $j \in \mathcal{L}_s$ such that $n \in \mathcal{N}_j$, then $m_{s,n} = 0$ (i.e., node n does not cause interference to any outgoing link of node s).

Our second proposed algorithm works for any general topology and is shown in Algorithm 6. In this algorithm, each node $n \in \mathcal{N}$ informs $m_{n,s}$ to all nodes s whose transmissions interfere with transmissions of at least one of the outgoing links of node n. It also informs q_n to all nodes s whose outgoing transmissions is interfered by transmissions from node n. All nodes then choose the persistent probabilities of their outgoing links based on the received messages from

other nodes. In Algorithm 6, $T_{n,p}$ and $T_{n,qm}$ are two unbounded sets of time slots at which node n updates $\boldsymbol{p}_n$ and announces q_n, and $m_{n,s}$ for all $s \neq n$, respectively. The assumptions on asynchronism measure are the same as those in Algorithm 5. We will show in Section 5.3.2 that for any *general* topology, the fixed point of Algorithm 6 also corresponds to the global optimal solution of the *non-convex* problem (5.7) under proper conditions.

In comparison with the prior algorithms in the literature (e.g., [98]), Algorithms 5 and 6 are more *robust*, converge *faster*, and require *less signalling.* We further discuss the properties of our proposed algorithms in Sections 5.3 and 5.4.

5.3 Convergence, Optimality, and Robustness

In this section, we prove convergence, optimality and robustness of Algorithms 5 and 6 by using recent developments in the theory of parallel and distributed computation [18] (also see Section 1.2.3) and the theory of non-linear optimization [15, 16] (also see Section 1.2.1).

5.3.1 Single-Cell Topology

Here we study Algorithm 5 which was proposed to solve problem (LOCAL-NUM) in a single-cell topology. We first show that if Algorithm 5 has a unique fixed point, then it will globally converge to that fixed point. After that, we provide the conditions under which the uniqueness of the fixed point of Algorithm 5 is guaranteed. We also show that such unique fixed point corresponds to the unique global optimal solution of non-convex problem (5.7). We define $\boldsymbol{f}(\boldsymbol{p}) = (\boldsymbol{f}_n(\boldsymbol{p}), \forall n \in \mathcal{N})$, where $\boldsymbol{f}_n(\boldsymbol{p})$ is as in Theorem 10 for each node n. Recall that a fixed point of mapping $\boldsymbol{f}(\boldsymbol{p})$ is also a fixed point of Algorithm 5. We can show that:

Theorem 12 *Mapping $\boldsymbol{f}(\boldsymbol{p})$ is monotone* increasing *(see Section 1.2.3) if $\alpha \geq 1$, and is monotone* decreasing *if $\alpha \leq 1$.*

The proof of Theorem 12 is given in Section 5.6.2. This enables us to show the following:

Theorem 13 *Assume that $\boldsymbol{f}(\boldsymbol{p})$ has a unique fixed point $\boldsymbol{p}^\star$. Starting from any initial point $\boldsymbol{p} \in \mathcal{P}$, Algorithm 5 globally converges to $\boldsymbol{p}^\star$.*

The proof of Theorem 13 is given in Section 5.6.3. The key idea is to show that the monotone mapping $\boldsymbol{f}(\boldsymbol{p})$ satisfies *synchronous convergence* and *box* conditions; thus, the *asynchronous convergence theorem* [18, pp. 431] is applicable (see Theorem 8). Theorem 13 is general and applies to any choice of system parameters. It only requires that mapping $\boldsymbol{f}(\cdot)$ has a unique fixed point. Next, we will show that not only Algorithm 5 has a unique fixed point under mild technical conditions, the fixed point is indeed the global optimal solution of problem (5.7).

Let $\mathcal{F}$ denote the set of fixed points of Algorithm 5. For each $\boldsymbol{p}^\star \in \mathcal{F}$ and any link $i \in \mathcal{L}_n$, we have $p_i^\star = f_i(\boldsymbol{p}_{-n}^\star)$. We also let $\mathcal{S}$ denote the set of *stationary points* [15, pp. 194] of problem (5.7). Note that all local (and global) optimal solutions of problem (5.7) belong to set $\mathcal{S}$.

Theorem 14 $\mathcal{F} = \mathcal{S}$.

The proof of Theorem 14 is given in Section 5.6.4. From Theorems 13 and 14,

Corollary 1 *If either $\mathcal{S}$ or $\mathcal{F}$ is a* singleton *set (i.e., it has one element), then Algorithm 5 globally and asynchronously converges to the unique global optimal solution of problem (5.7).*

In [98], it has been shown that the set of stationary points $\mathcal{S}$ is a singleton set for all $\alpha \geq 1$. They used logarithmic mapping and transformed problem (5.7) to an equivalent convex problem and showed that it has a unique stationary point. However, this transformation does not work if

$\alpha \in (0,1)$. That is the reason the algorithm proposed in [98] does not support the α-fair utility functions with $\alpha \in (0,1)$. Here we are able to provide sufficient conditions under which the non-convex problem (5.7) has a unique optimal solution with $\alpha \in (0,1)$.

Theorem 15 *Consider the case where the fairness index parameter $\alpha \in (0,1)$. Set $\mathcal{F}$ is a singleton if the following holds:*

$$\left(\frac{1-\alpha}{\alpha}\,\Psi\,\Phi(V^{\min},V^{\max})\right)^2\left(\frac{\gamma^{\max}}{\gamma^{\min}}\Gamma\right)^{1-\alpha}\left(\Omega-\frac{1}{L/L^{\min}-1}\right)<1, \tag{5.26}$$

where

$$L^{\min}=\min_{n\in\mathcal{N}} L_n, \tag{5.27}$$

$$L^{\max}=\max_{n\in\mathcal{N}} L_n, \tag{5.28}$$

$$\Phi(V^{\min},V^{\max})=\begin{cases}\frac{(V^{\max})^{1/\alpha}}{(1+(V^{\max})^{1/\alpha})^2}, & \text{if } V^{\max}\le 1,\\ \frac{(V^{\min})^{1/\alpha}}{(1+(V^{\min})^{1/\alpha})^2}, & \text{if } V^{\min}\ge 1,\\ 0.25, & \text{otherwise,}\end{cases} \tag{5.29}$$

$$\Gamma=\left(P^{\max}(1-P^{\min})\right)/\left(P^{\min}(1-P^{\max})\right), \tag{5.30}$$

$$\Psi=L^{\max}/(1-P^{\max})+1/P^{\min}, \tag{5.31}$$

$$\Omega = \sum_{n \in \mathcal{N}} 1/(L/L_n - 1), \tag{5.32}$$

$$V^{\min} = (N-1)\left(\gamma^{\max}(1/P^{\min}-1)/\gamma^{\min}\right)^{\alpha-1}, \tag{5.33}$$

$$V^{\max} = (N-1)\left(\gamma^{\min}(1/P^{\min}-1)/\gamma^{\max}\right)^{\alpha-1}. \tag{5.34}$$

The proof of Theorem 15 is given in Section 5.6.5. The key is to show that if (5.26) holds, then mapping $\boldsymbol{f}$ is not only a monotone mapping, but also an l_2-norm contraction mapping (see Section 1.2.3 for the definition of contraction mapping).

The sufficient condition in Theorem 15 can further be simplified in some cases. For example, if all nodes have the same number of outgoing links, then $L^{\min} = L_n$ for each node n, and $L/L_n = L/L^{\min} = N$. Thus, $\Omega - 1/\left(L/L^{\min} - 1\right) = N/(N-1) - 1/(N-1) = 1$ and (5.26) becomes

$$\left(\frac{1-\alpha}{\alpha}\,\Psi\,\Phi(V^{\min}, V^{\max})\right)^2 \left(\frac{\gamma^{\max}}{\gamma^{\min}}\Gamma\right)^{1-\alpha} < 1. \tag{5.35}$$

In general, all the terms in (5.26) and (5.35), except Φ, are independent of the number of nodes N. The value of Φ can be arbitrarily close to 0 if N is large enough. Thus,

Corollary 2 *For any $\alpha \in (0,1)$ and any choice of other system parameters, there exists a positive integer $\hat{N}$ such that Algorithm 5 has a unique fixed point if the number of wireless nodes $N > \hat{N}$.*

Theorem 15 provides practical bounds on system parameters that guarantee the uniqueness of the fixed point. For example, consider the IEEE 802.11a standard where $\gamma^{\min} = 6$ Mbps and

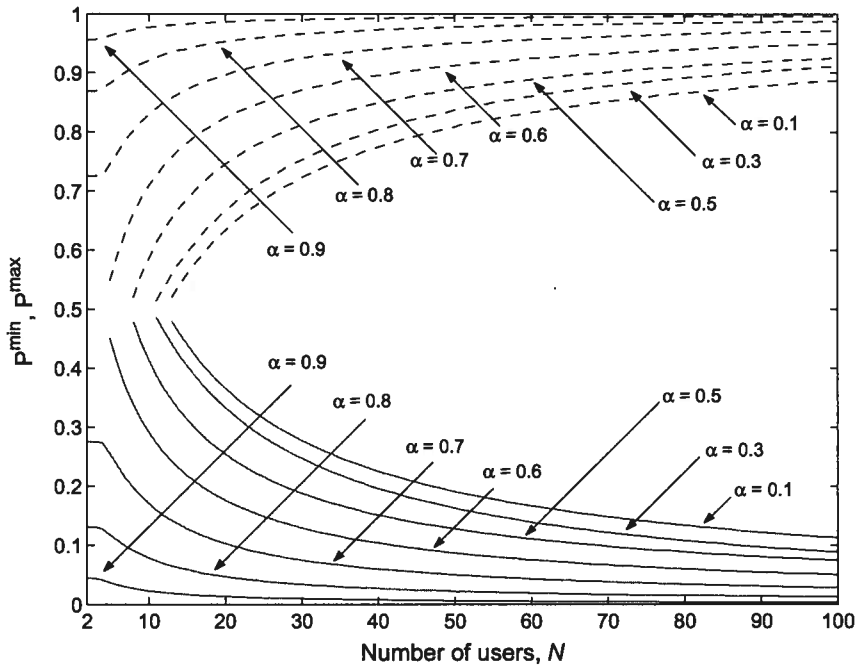


Figure 5.2: Sufficient conditions on the upper bounds of $P^{\max}$ and lower bounds of $P^{\min}$ for $\alpha \in [0.1, 0.9]$ and $N \in [2, 100]$ when Algorithm 5 is being used and each node has one outgoing link. Solid lines represent lower bounds on $P^{\min}$ and dashed lines represent upper bounds on $P^{\max}$.

$\gamma^{\max} = 54$ Mbps[6]. In Fig. 5.2, we plot the sufficient conditions on the upper bounds of $P^{\max}$ and lower bounds of $P^{\min}$ for utility parameter $\alpha \in [0.1, 0.9]$ and number of nodes $N \in [2, 100]$, where each node has one outgoing link. As we can see, the difference between the lower and upper bounds increases as α or N increases, indicating the convergence condition is less restrictive. In many cases, convergence of Algorithm 5 can be obtained even when the sufficient condition (5.26) is not satisfied. For example, it is easy to numerically verify that for $N = 2$, problem (5.7) with $\alpha \in (0.5, 1)$ has a unique global optimal solution with any choice of system parameters.

Theorems 13 to 15 together show that Algorithm 5 globally and asynchronously converges to the unique global optimal solution of the problem (5.7) when either $\alpha \in (0, 1)$ (under condition

[6]IEEE 802.11a supports 6, 9, 12, 18, 24, 36, 48, and 54 Mbps data rates [9].

(5.26)) or $\alpha \geq 1$ (with any system parameters). In particular, Algorithm 5 works properly under delayed or even occasionally lost messages. To have a better understanding on how the system behaves with message loss, consider M consecutive messages announced by an arbitrary node n. The first $M-1$ messages are lost (e.g., because of collision) while the last message is properly received by all other nodes $s \in \mathcal{N}\backslash\{n\}$. In this case, all derived results will go through with a redefined asynchronism measure. Let $\hat{H} = MH$. Since H and M are bounded, $\hat{H}$ is also bounded. Considering $\hat{H}$ as the new asynchronism measure, Theorems 13 to 15 can still be applied. Thus, convergence and optimality of Algorithm 5 are still guaranteed. Interestingly, this robust behavior is accompanied with fast convergence speed as shown in Section 5.4.

5.3.2 General Topology

Consider vector mapping $\boldsymbol{f}'(\boldsymbol{p}) = (\boldsymbol{f}'_n(\boldsymbol{p}), \forall\, i \in \mathcal{N})$, where $\boldsymbol{f}'_n(\boldsymbol{p})$ is defined in Theorem 11. We denote the set of fixed points of mapping $\boldsymbol{f}'(\boldsymbol{p})$ by $\mathcal{F}'$, which is the set of fixed points of Algorithm 6. We also denote the set of stationary points of problem (5.7) by $\mathcal{S}'$ in this case.

Theorem 16 $\mathcal{F}' = \mathcal{S}'$.

The proof of Theorem 16 is similar to that of Theorem 14. If $\alpha \geq 1$, then from [98, Lemma 1] we know that stationary point set $\mathcal{S}'$ is a singleton set. Together with Theorem 16, we have:

Corollary 3 *If $\alpha \geq 1$ and Algorithm 6 converges to some fixed point, then the fixed point is the unique global optimal solution of problem (5.7).*

If $\alpha \in (0, 1)$, we can use the same idea of Theorem 15 and obtain sufficient conditions to assure that the stationary point set $\mathcal{S}'$ is a singleton set. We first notice that since not all links interfere with each other, for each node $n \in \mathcal{N}$ and any link $i \in \mathcal{L}_n$, function f'_i may only depend on a small

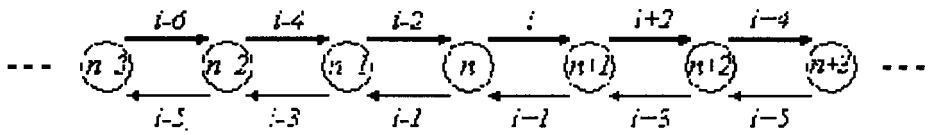


Figure 5.3: A chain topology with one hop interference. Arrows indicate the direction of the corresponding unidirectional wireless links. Function f_i' only depends on $(p_{i-5}, \ldots, p_{i-1}, p_{i+2}, \ldots, p_{i+4})$. Notice that γ_i', γ_{i-1}', $m_{n-2,n}$, $m_{n-1,n}$, $m_{n+1,n}$, and $m_{n+2,n}$ depend on $(p_{i+1}, \ldots, p_{i+4})$, $(p_{i-5}, \ldots, p_{i-2})$, $(p_{i-4}, p_{i-3}, p_{i-2})$, $(p_{i-2}, p_{i+1}, p_{i+2})$, $(p_{i-3}, p_{i-2}, p_{i+1})$, and $(p_{i+1}, p_{i+2}, p_{i+3})$, respectively.

subset of entries in vector $\boldsymbol{p}_{-n}$. For example, consider the *chain topology* in Fig. 5.3, where the interferences are within *one hop*. For each node i, $\mathcal{N}_i = \{n+1, n+2\}$. In this figure, f_i' only depends on $(p_{i-5}, \ldots, p_{i-1}, p_{i+2}, \ldots, p_{i+4})$. Notice that γ_i' depends on $(p_{i+1}, \ldots, p_{i+4})$ and γ_{i-1}' depends on $(p_{i-5}, \ldots, p_{i-2})$. Thus w_n' depends on $(p_{i-5}, \ldots, p_{i-1}, p_{i+2}, \ldots, p_{i+4})$. In addition, $m_{n-2,n}$, $m_{n-1,n}$, $m_{n+1,n}$, and $m_{n+2,n}$ depend on $(p_{i-4}, p_{i-3}, p_{i-2})$, $(p_{i-2}, p_{i+1}, p_{i+2})$, $(p_{i-3}, p_{i-2}, p_{i+1})$, and $(p_{i+1}, p_{i+2}, p_{i+3})$, respectively. Thus, v_n' also depends on $(p_{i-5}, \ldots, p_{i-1}, p_{i+2}, \ldots, p_{i+4})$. We define set $\mathcal{X}_i = \{i-5, \ldots, i-1, i+2, \ldots, i+4\}$ as the *dependency set* for link i. Similarly, we can define $\mathcal{X}_i$ for *all* $i \in \mathcal{L}$ in any *arbitrary* topology. That is, for any $i, j \in \mathcal{L}$, we have $j \in \mathcal{X}_i$ if and only if p_j appears in the formulation of f_i'. Let $X_i = |\mathcal{X}_i|$ denote the size of set $\mathcal{X}_i$. We define $X^{\max} = \max_{i \in \mathcal{L}} X_i$. As an example, for the chain topology in Fig. 5.3, $X^{\max} = 8$.

Theorem 17 *For* any *general topology, the fixed point set $\mathcal{F}'$ is a singleton if $\alpha \in (0,1)$ and*

$$\frac{1-\alpha}{\alpha} X^{\max} \Lambda \, \Phi(Z^{\min}, Z^{\max}) < 1, \tag{5.36}$$

where Φ is as in (5.29) and we have:

$$\Lambda = 1/P^{\min} + 1/(1 - P^{\max}), \tag{5.37}$$

$$Z^{\min} = \left(\frac{\gamma'^{\min}}{\gamma'^{\max}}\right)^{1-\alpha}(L^{\min}-1)^{\alpha}, \tag{5.38}$$

$$Z^{\max} = \left(\frac{\gamma'^{\max}}{\gamma'^{\min}}\right)^{1-\alpha}\left((L^{\min}-1)+\sqrt[\alpha]{(L-L^{\min})P^{\max}}\right)^{\alpha} \tag{5.39}$$

$$\gamma'^{\min} = \gamma^{\min}(1-P^{\max})^{N-1}, \tag{5.40}$$

$$\gamma'^{\max} = \gamma^{\max}. \tag{5.41}$$

Theorem 17 guarantees that Algorithm 6 has a unique fixed point which is indeed the global optimal solution of problem (5.7). Notice that $Z^{\min}$ and $Z^{\max}$ are lower and upper bounds on $\gamma'^{\alpha-1}(w_n(\boldsymbol{p}_{-n})+\sqrt[\alpha]{v_n(\boldsymbol{p}_{-n})})^{\alpha}$ for any $n\in\mathcal{N}$ and any $i\in\mathcal{L}_n$. The proof of Theorem 17 is similar to that of Theorem 15. The key is to show that condition (5.36) is *sufficient* to make $\boldsymbol{f}'$ an l_2-norm contraction mapping. Theorem 17 is general and applies to any topology. Given the particular topology of interest, we can further refine (5.36), e.g., as in (5.26) for single-cell topologies. We note that condition (5.36) is only sufficient as in the single-cell case. For example,we can numerically verify that for many practical topologies (e.g., chain topologies), problem (5.7) with $\alpha\in(0.5,1)$ has a unique local (thus global) optimal solution for *any* choice of system parameters.

Unlike mapping $\boldsymbol{f}$ in the single cell case, mapping $\boldsymbol{f}'$ here may not always be a monotone mapping. Thus, the convergence results in the single cell case (i.e., Theorem 13) do not apply in the general topology case. On the other hand, we can also find sufficient conditions under which the convergence to the unique fixed point is guaranteed.

Theorem 18 *For any general topology, Algorithm 6 globally and asynchronously converges to the unique global optimal solution of problem (5.7) if*

$$\frac{|1-\alpha|}{\alpha}\sqrt{N}\,X^{\max}\Lambda\,\Phi(Z^{\min},Z^{\max})<1, \tag{5.42}$$

where Λ is as in (5.37) and $Z^{\min}$ and $Z^{\max}$ are defined in (5.38) and (5.39), respectively.

The proof of Theorem 18 is given in Section 5.6.6. The idea is to use the relationship between l_2 and l_∞ norms to obtain a sufficient condition under which $\boldsymbol{f}'(\boldsymbol{p})$ is a weighted maximum norm contraction mapping with *unit* weights (see Section 1.2.3). Thus, Algorithm 6 asynchronously converges to its unique fixed point. From Theorem 16, the convergence is indeed towards the unique global optimal solution of problem (5.7). Notice that condition (5.42) is a sufficient (but not necessary) condition for asynchronous convergence. Simulation results in Section 5.4 verify that Algorithm 6 converges under a wide range of system parameters. We also notice that if (5.42) holds for some $\alpha \in (0,1)$, then (5.36) also holds for the same α. Thus, Theorem 18 also implies Theorem 17.

The exact value of asynchronism measure H is *not* important for any of the proofs. Following the same argument in Section 5.3.1, Algorithm 6 works properly under delayed or occasionally lost messages. In Section 5.4, we assess the optimality, robustness, and convergence speed of Algorithm 6 for several randomly selected topologies and under different channel conditions.

5.4 Simulation Results

In this section, we assess the optimality, convergence and robust performance of our proposed algorithms. In particular, we show the advantages of our algorithms compared with the previously proposed subgradient-based algorithm [98] and IEEE 802.11 DCF.

5.4.1 Convergence and Optimality

We first consider a single-cell topology with $N=3$ nodes and $L=6$ links. In this network, each node has two outgoing links, one to each of the other two nodes. For all nodes $n \in \mathcal{N}$, we set the minimum persistent probability $P_n^{\min}=0.01$ and the maximum total probability $P_n^{\max}=0.99$. We also set asynchronism measure $H=10$. The peak rates of 6 links are $\gamma_1=6$ Mbps, $\gamma_2=36$ Mbps, $\gamma_3=9$ Mbps, $\gamma_4=12$ Mbps, $\gamma_5=18$ Mbps, and $\gamma_6=54$ Mbps. Communication delays among nodes are up to 9 slots and the packet error rate is 0.1 (i.e., on average, 10% of the messages are lost). Fig.5.4(a) shows the trajectories of adjusted persistent probabilities and their optimal values when $\alpha = 2$ (which is *greater* than 1). In this case, the global optimal persistent probabilities $\boldsymbol{p}^\star=[0.26, 0.11, 0.21, 0.18, 0.16, 0.09]^T$. We see that Algorithm 5 converges to $\boldsymbol{p}^\star$ within less than 300 slots, even with communication delay and message loss. Similar results when $\alpha = 0.6$ (which is *less* than 1) are shown in Fig.5.4(b). In this case, $\boldsymbol{p}^\star=[0.06, 0.21, 0.07, 0.09, 0.18, 0.38]^T$. Again, we see that Algorithm 5 asynchronously converges to $\boldsymbol{p}^\star$ within 320 time slots.

Next, consider a chain topology with $N = 3$ nodes and $L = 4$ links (see Fig. 5.3). Note that this belongs to the general topology case, where each node only interferes with a subset of other links. For all nodes $n \in \mathcal{N}$, we set $P_n^{\min} = 0.01$ and $P_n^{\max} = 0.99$. We also set, $\gamma_1 = 6$ Mbps, $\gamma_2 = 36$ Mbps, $\gamma_3 = 9$ Mbps, and $\gamma_4 = 12$ Mbps. Communication delays, asynchronism measure, and packet loss rate are the same as the previous experiment. Simulation results when $\alpha = 2$ and $\alpha = 0.6$ are shown in Fig. 5.5(a) and Fig. 5.5(b), respectively. We see that the optimal persistent probabilities are achieved very fast and Algorithm 6 can tolerate both delay and message loss.

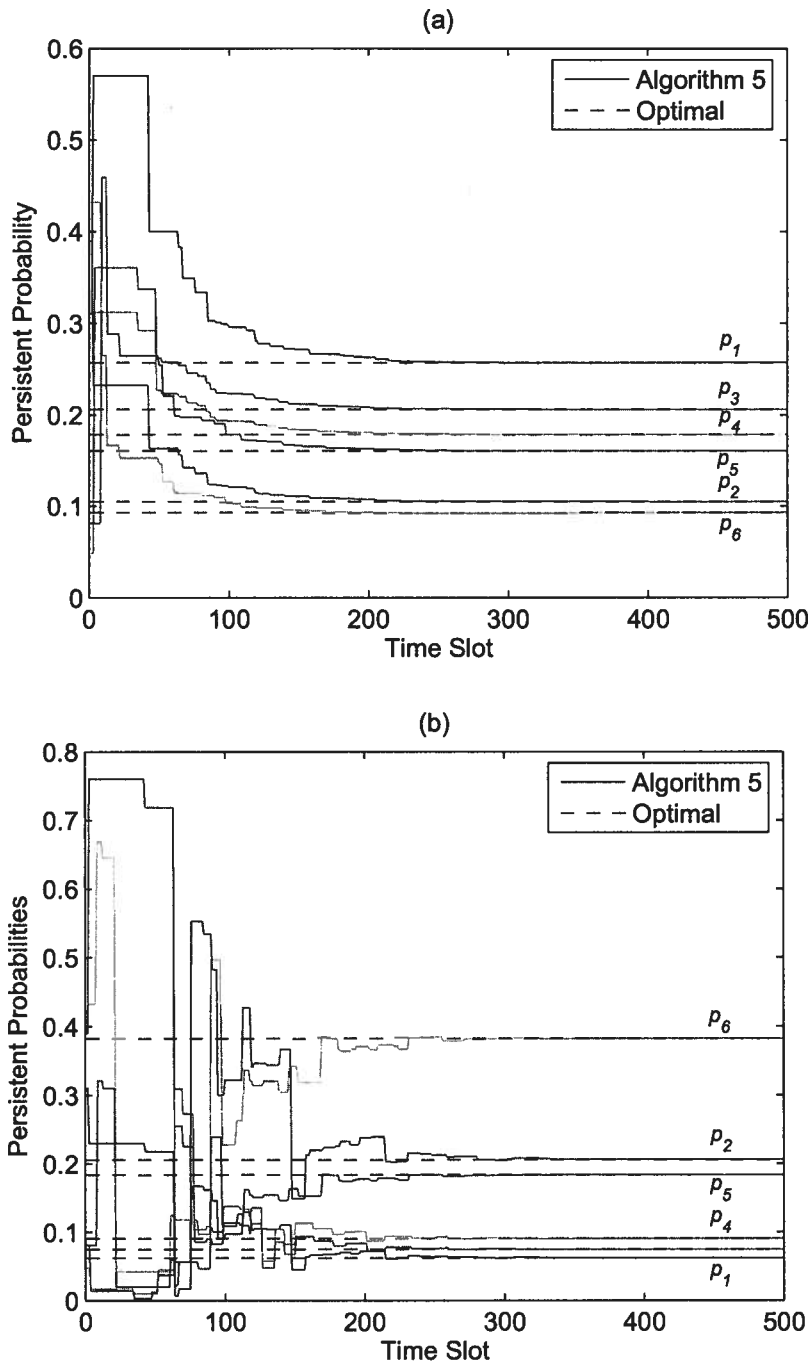


Figure 5.4: Simulation results for a single-cell topology with three nodes and six links. Algorithm 5 is being used. (a) $\alpha = 2$, (b) $\alpha = 0.6$.

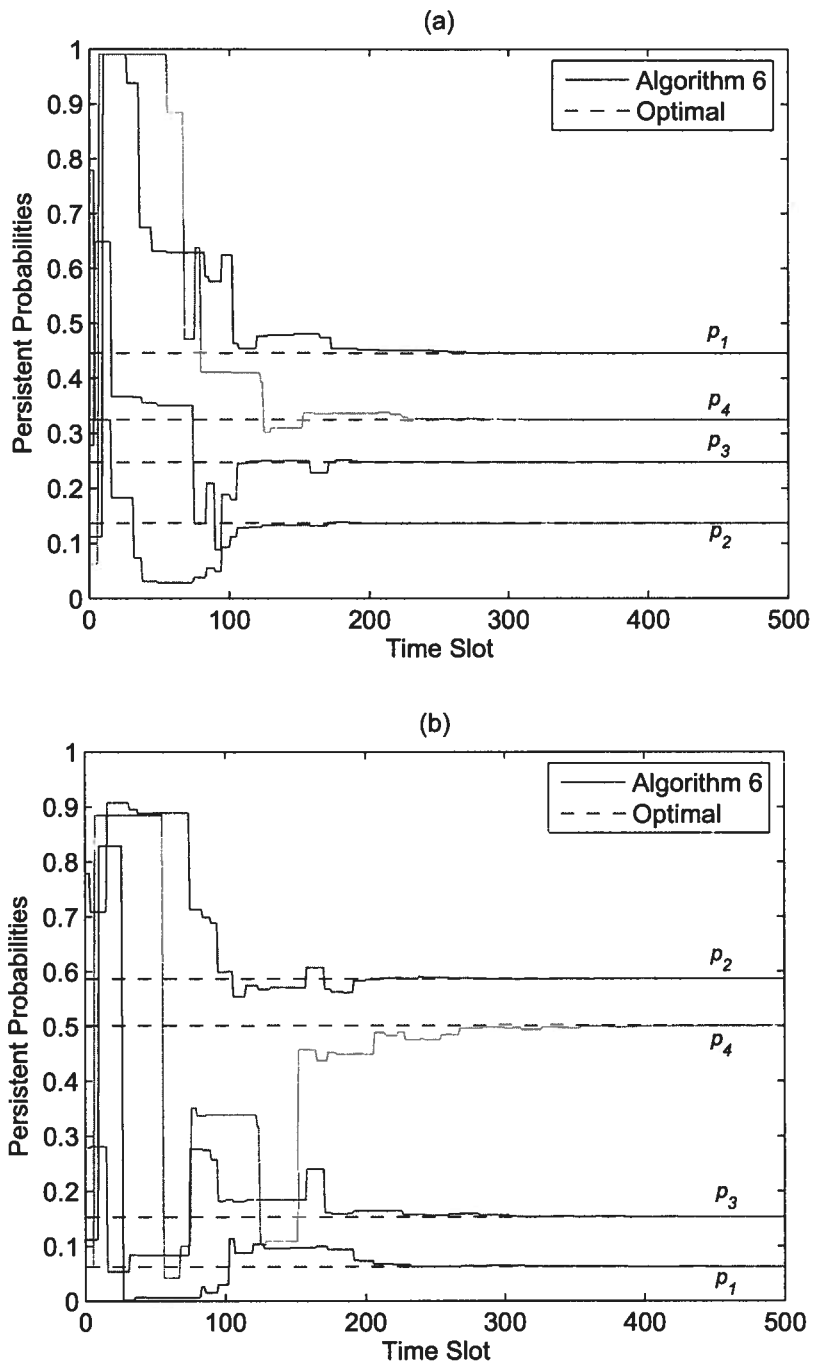


Figure 5.5: Simulation results for a chain topology with three nodes and four links. Algorithm 6 is being used. (a) Utility parameter $\alpha = 2$, (b) Utility parameter $\alpha = 0.6$.

5.4.2 Signalling Overhead

High signalling overhead is a critical problem for algorithms which require cooperation among nodes in a wireless ad-hoc network. In this section, we compare the signalling overhead in our proposed algorithms with the subgradient-based algorithm [98]. In the simulation, wireless nodes are *randomly* located in 100 m $\times$ 100 m and 1 km $\times$ 1 km fields for single-cell and general topologies, respectively. The communication and interference ranges are 150 m and 300 m, respectively. Each node has an outgoing link to any of its neighboring nodes within its communication range. The peak transmission rates (i.e., γ_i for all $i \in \mathcal{L}$) are selected *randomly* between 6 Mbps to 54 Mbps. Utility parameter α is set to 2 which models harmonic mean fair allocation. We assume that each message value requires two bytes. Thus, the signalling overhead for each algorithm is defined as the total required message exchange (in KBytes) that the algorithm needs before it reaches the corresponding optimal solution of problem (5.7). Results for single-cell and general topologies when number of nodes N varies from 5 to 30 are shown in Fig. 5.6(a) and Fig. 5.6(b), respectively. We see that increasing the number of nodes increases the signalling overhead. However, Algorithm 5 and 6 manage to reach the optimal solutions via much less signalling. Compared to the subgradient-based algorithm and when $N = 30$, Algorithm 5 and Algorithm 6 reduce the signalling overhead by 1120% (from 55.2 KByte to 4.5 KByte) and 810% (from 111.3 KByte to 10.8 KByte), respectively. Notice that one reason for the superiority of our algorithms is their faster convergence. In addition, Algorithm 5 reduces the message amount by half, which also contributes to reducing the signalling overhead.

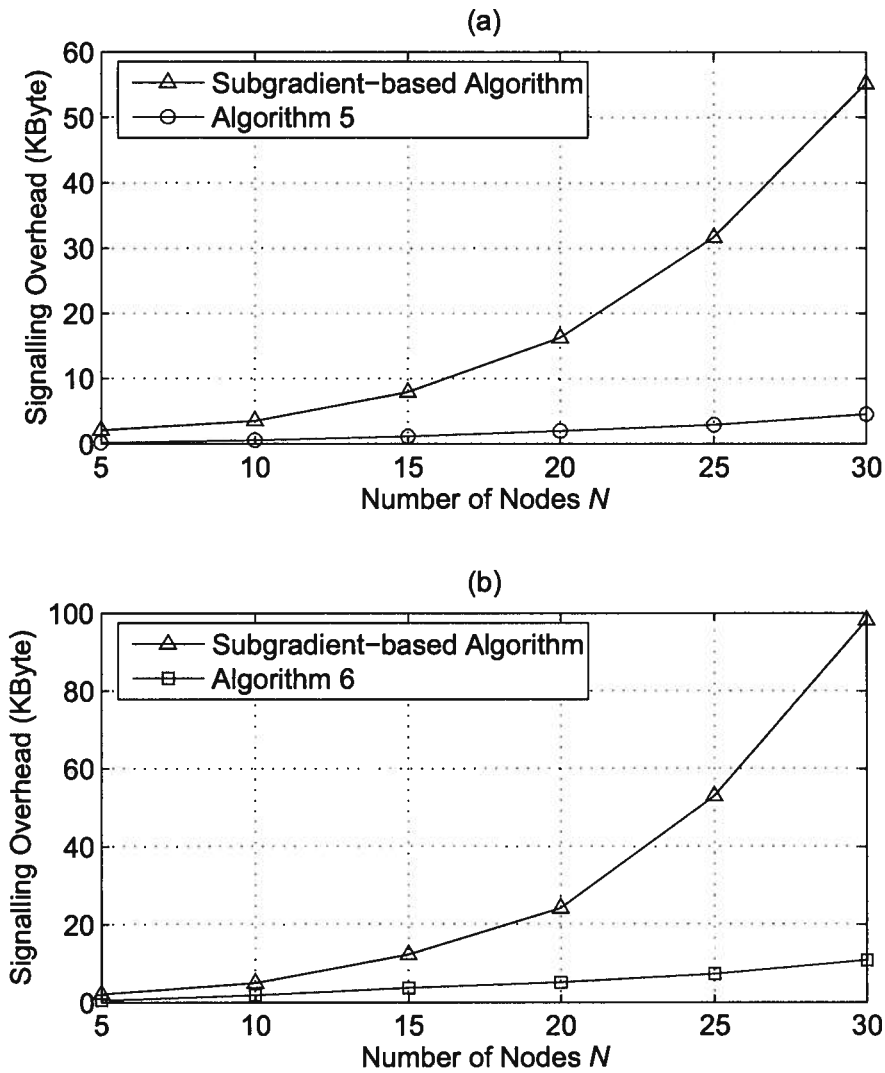


Figure 5.6: Comparison between our proposed algorithms and the subgradient-based algorithm [98] in term of signalling overhead when the number of wireless nodes varies from 5 to 30. Each point represents the average results from simulating 10 random topologies. (a) Results when the simulated topologies are single-cell and Algorithm 5 is being used, (b) Results when the simulated topologies are general and Algorithm 6 is being used.

5.4.3 Robustness

Since the underlying communication channels are not ideal, transmitted messages by MAC protocols may be delayed or even be lost. In this section, we show that our proposed algorithms are robust with respect to both message delay and loss. The simulation model is the same as that in Section 5.4.2. We only consider general topologies with $N = 30$ nodes. Results for single-cell topologies are similar. The simulation time is set to $50,000$ time slots. For each algorithm, we measure the optimality of the achieved network utility at the end of each simulation run.

First, we assume that the communication delay varies from 10 to 50 time slots. Results are shown in Fig. 5.7(a). We see that by increasing delay up to 50 time slots, the subgradient-based algorithm leads to 8.4% optimality loss while Algorithm 6 can always find the exact optimal solutions. Notice that although Algorithm 6 is robust to communication delay, higher delays can cause more iterations for the algorithm to converge. In our simulation model, Algorithm 6 converges to the optimal persistent probabilities on average after 421, 1581, 3641, 6472, and 9923 time slots when communication delay is 10, 20, 30, 40, and 50 time slots, respectively. Convergence time versus the communication delay is illustrated in Fig. 5.8.

Next, we consider the effect of message delay when the packet error rate varies from 0.1 (i.e. 10% of the messages are lost) to 0.5 (i.e., 50% of the messages are lost). Results are shown in Fig. 5.7(b). We see that Algorithm 6 is robust to message loss. On average, it converges to the optimal persistent probabilities after 312, 473, 531, 629, and 727 time slots when packet error rate is 0.1, 0.2, 0.3, 0.4, and 0.5, respectively. Notice that both Algorithm 6 and the subgradient-based algorithm are less sensitive to message loss compared to long delays.

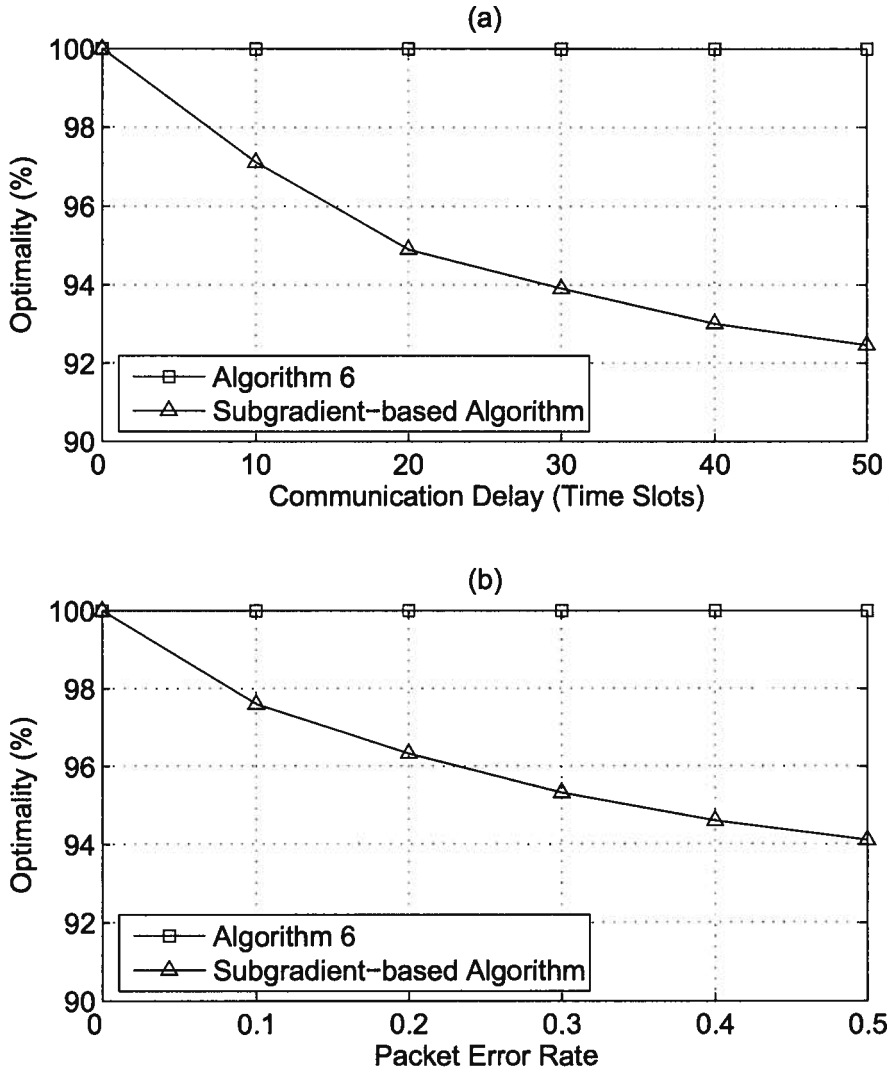


Figure 5.7: Comparison between Algorithm 6 and the subgradient-based algorithm [98] in term of robustness with respect to communication delay and message loss. Each point represents the average results from simulating 10 random general topologies, each including 30 nodes. (a) Optimality in percentage when maximum communication delay varies from 10 to 50 time slots, (b) Optimality in percentage when packet error rate varies from 0.1 to 0.5.

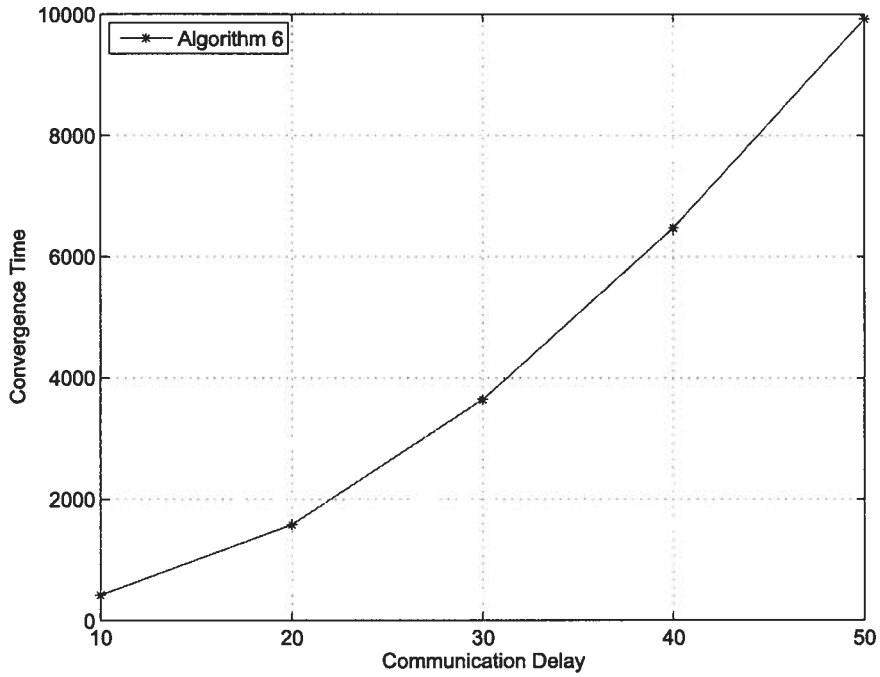


Figure 5.8: Convergence time in presence of various communication delays.

5.4.4 Comparison with IEEE 802.11 DCF

Since our proposed algorithms achieve the global optimal solution of problem (5.7), they establish a performance upper-bound for all MAC algorithms that are designed to solve the same problem. On the other hand, they can resolve some of the existing problems in the current 802.11 DCF, e.g., its well-known *short-term fairness* problem due to binary exponential backoff. Next, we compare Algorithm 6 with DCF in terms of both system throughput and Jain's fairness index [89] when $N = 30$. The short-term fairness is obtained using a sliding window of size 200 slots. Results when utility parameter α varies from 0.5 to 5 are shown in Fig. 5.9. Each point represents the average results from simulating 10 random general topologies. We see that, parameter α acts as a *knob* in Algorithm 6 to control the tradeoff between efficiency and fairness. By increasing α

we can make the system more fair but less efficient (and vice versa). If $\alpha = 0.5$, then the system throughput is 54.9% higher than DCF (see Fig. 5.9(a)). Besides, for any choice of $\alpha \in [0.5, 5]$, the fairness is much better than DCF (see Fig. 5.9(b)).

5.5 Summary

In this chapter, we designed two distributed contention-based MAC algorithms to solve a NUM problem at link-layer in wireless ad hoc networks. Both algorithms globally converge to the unique global optimal solution of the NUM problem under mild technical conditions on the system parameters. In particular, it is true for α-fair utility functions with $\alpha \in (0, 1)$, in which case the NUM problem cannot be transformed into a convex optimization problem using logarithmic change of variables as in the previous literature, and thus, it is already challenging to prove the uniqueness of the optimal solution before designing any algorithm.

Besides supporting a wider range of utility functions, our algorithms have several other advantages over previous algorithms, including less message passing, fully asynchronous updates, robustness to message delays and losses, and fast convergence. Simulation results show that our algorithms converge faster than the recently proposed algorithm in [98] with significantly less signaling overhead. They are also robust to message delays and losses due to channel errors. Moreover, they achieve better efficiency-fairness trade-off compared to the IEEE 802.11 DCF.

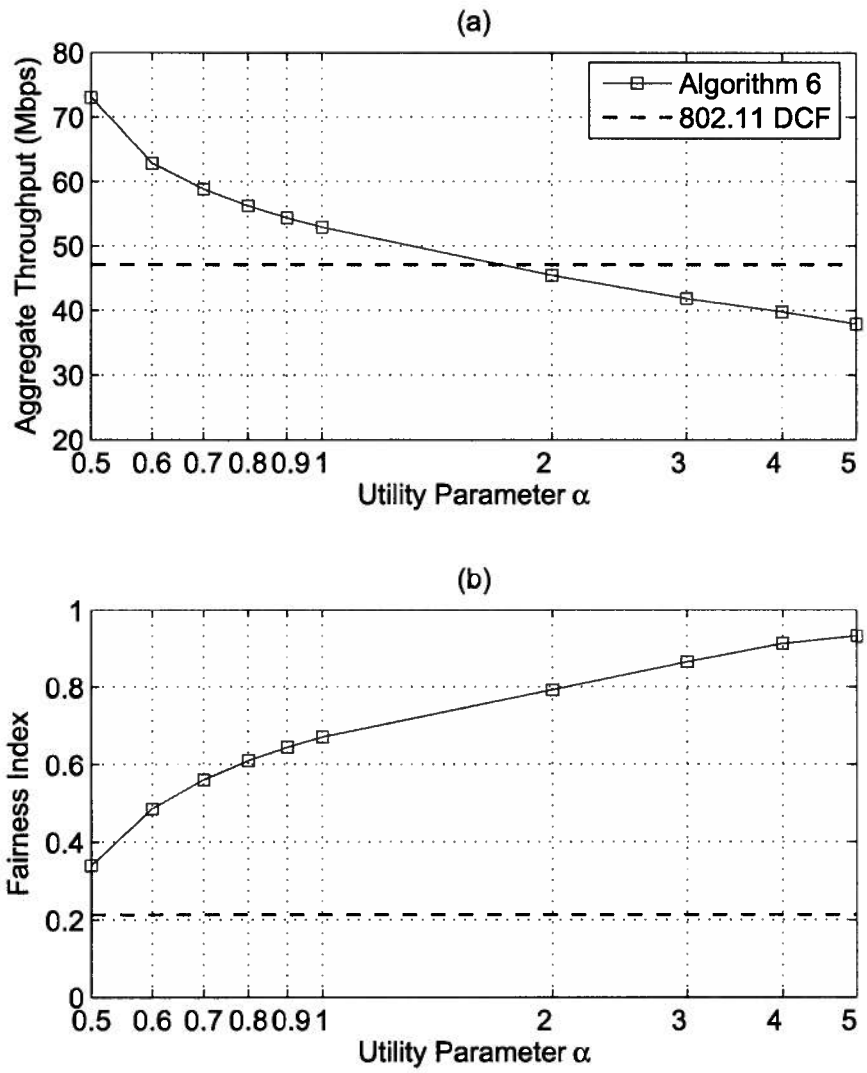


Figure 5.9: Comparison between DCF and Algorithm 6 in terms of system throughput and fairness index when utility parameter α varies from 0.5 to 5. Each point represents the average results from simulating 10 random general topologies, each including 30 nodes. (a) Aggregate network throughput, (b) Jain's fairness index.

5.6 Analytical Proofs

5.6.1 Proof of Theorem 10

Lemma 6 *For each node $n \in \mathcal{N}$, we have:*

$$1/P_n^{\min} - B_n \leq \sqrt[\alpha]{\gamma_i^{\alpha-1}}\left(w_n + \sqrt[\alpha]{v_n(\boldsymbol{p}_{-n})}\right), \qquad \forall\, i \in \mathcal{B}_n, \tag{5.43}$$

$$P_n^{\max}/P_n^{\min} - C_n \leq \sqrt[\alpha]{\gamma_i^{\alpha-1}}w_n, \qquad \forall\, i \in \mathcal{C}_n, \tag{5.44}$$

$$1/P_n^{\min} - B_n > \sqrt[\alpha]{\gamma_i^{\alpha-1}}\left(w_n + \sqrt[\alpha]{v_n(\boldsymbol{p}_{-n})}\right), \qquad \forall\, i \in \mathcal{L}_n\backslash\mathcal{B}_n, \tag{5.45}$$

$$P_n^{\max}/P_n^{\min} - C_n > \sqrt[\alpha]{\gamma_i^{\alpha-1}}w_n, \qquad \forall\, i \in \mathcal{L}_n\backslash\mathcal{C}_n. \tag{5.46}$$

Proof: From (5.12) and knowing that $\mathcal{L}_n\backslash\mathcal{A}_n = \{i_1, \ldots, i_\kappa\}$, for each link $i \in \mathcal{B}_n$ we have:

$$\begin{aligned} 1/P_n^{\min} - B_n &\leq \textstyle\sum_{j\in\mathcal{L}_n\backslash\mathcal{A}_n} \sqrt[\alpha]{\left(\gamma_{i_{\kappa+1}}/\gamma_j\right)^{\alpha-1}} + \sqrt[\alpha]{\gamma_{i_{\kappa+1}}^{\alpha-1}v_n(\boldsymbol{p}_{-n})} \\ &= \left(\sqrt[\alpha]{\gamma_{i_{\kappa+1}}^{\alpha-1}}/\sqrt[\alpha]{\gamma_i^{\alpha-1}}\right)\left(\textstyle\sum_{j\in\mathcal{L}_n\backslash\mathcal{A}_n} \sqrt[\alpha]{(\gamma_i/\gamma_j)^{\alpha-1}} + \sqrt[\alpha]{\gamma_i^{\alpha-1}v_n(\boldsymbol{p}_{-n})}\right) \\ &\leq \sqrt[\alpha]{\gamma_i^{\alpha-1}}\left(w_n + \sqrt[\alpha]{v_n(\boldsymbol{p}_{-n})}\right), \end{aligned} \tag{5.47}$$

where the last inequality comes from the fact that $\sqrt[\alpha]{\gamma_{i_{\kappa+1}}^{\alpha-1}} \leq \sqrt[\alpha]{\gamma_{i_{\sigma+1}}^{\alpha-1}} \leq \sqrt[\alpha]{\gamma_i^{\alpha-1}}$ for all $i \in \mathcal{B}_n$. Recall that $\kappa = \min\{\sigma, \varsigma\}$ and $i_{\kappa+1} \leq i_{\sigma+1}$. Proof of (5.44) is similar. We prove (5.45) by *contradiction.* Assume that there exists $i_\varepsilon \in \mathcal{L}_n\backslash\mathcal{A}_n$ such that (5.45) does *not* hold. It is clear that $\varepsilon \leq \sigma$. We

have:

$$1/P_n^{\min} - L_n + \sigma \leq \sqrt[\alpha]{\gamma_{i_\varepsilon}} \left(\sum_{l=1}^{\sigma} \sqrt[\alpha]{(1/\gamma_l)^{\alpha-1}} + \sqrt[\alpha]{v_n(\boldsymbol{p}_{-n})} \right) \tag{5.48}$$

$$\frac{1}{P_n^{\min}} - L_n + (\varepsilon - 1) + \delta - (\varepsilon - 1) \leq \left(\sum_{l=1}^{\varepsilon-1} \sqrt[\alpha]{(\tfrac{\gamma_{i_\varepsilon}}{\gamma_l})^{\alpha-1}} + \sqrt[\alpha]{\gamma_{i_\varepsilon} v_n(\boldsymbol{p}_{-n})} \right) + \sum_{l=\varepsilon}^{\sigma} \sqrt[\alpha]{(\tfrac{\gamma_{i_\varepsilon}}{\gamma_l})^{\alpha-1}} \tag{5.49}$$

$$1/P_n^{\min} - L_n + \varepsilon - 1 \leq \sum_{l=1}^{\varepsilon-1} \sqrt[\alpha]{(\gamma_{i_\varepsilon}/\gamma_l)^{\alpha-1}} + \sqrt[\alpha]{\gamma_{i_\varepsilon} v_n(\boldsymbol{p}_{-n})}, \tag{5.50}$$

where the last inequality results from the fact that $\frac{\gamma_{i_\varepsilon}}{\gamma_l} \leq 1$ for all $l = \varepsilon, \ldots, \delta$ which implies $\sum_{l=\varepsilon}^{\sigma} \sqrt[\alpha]{(\gamma_{i_\varepsilon}/\gamma_l)^{\alpha-1}} \leq \sigma - \varepsilon + 1$. Comparing with (5.12), inequality (5.50) implies that $i_\varepsilon \in \mathcal{A}_n$ which contradicts the assumption that $i_\varepsilon \in \mathcal{L}_n \backslash \mathcal{A}_n$. Proof of (5.46) is similar.

Since the objective function of problem (LOCAL-NUM) is strictly concave in $\boldsymbol{p}_n$, its solution is *unique* [16, pp. 137] and satisfies the following *necessary* and *sufficient* KKT conditions[7] [16, pp. 244]:

$$p_i \geq P_n^{\min}, \qquad \forall\, i \in \mathcal{L}_n, \tag{5.51}$$

$$\sum_{i \in \mathcal{L}_n} p_i \leq P_n^{\max}, \tag{5.52}$$

$$\frac{1}{\gamma_i^{\alpha-1} p_i^{\alpha}} - \frac{v_n(\boldsymbol{p}_{-n})}{\left(1 - \sum_{i \in \mathcal{L}_n} p_i\right)^{\alpha}} = \lambda_n - \delta_i, \qquad \forall\, i \in \mathcal{L}_n, \tag{5.53}$$

$$\lambda_n \left(\sum_{i \in \mathcal{L}_n} p_i - P_n^{\max} \right) = 0, \tag{5.54}$$

$$\delta_i \left(P_n^{\min} - p_i \right) = 0, \qquad \forall\, i \in \mathcal{L}_n, \tag{5.55}$$

$$\lambda_n \geq 0, \quad \delta_i \geq 0, \qquad \forall\, i \in \mathcal{L}_n, \tag{5.56}$$

where λ_n denotes the Lagrange multiplier corresponding to constraint $\sum_{i \in \mathcal{L}_n} p_i \leq P_n^{\max}$ and δ_i denotes the Lagrange multiplier corresponding to constraint $p_i \geq P_n^{\min}$ for each $i \in \mathcal{L}_n$. Now we

[7] It is clear that Slater's condition [16, pp. 226] always holds as long as $L_n P_n^{\min} < P_n^{\max}$ for all nodes $n \in \mathcal{N}$.

need to show that (5.14) leads to (5.51) to (5.56).

Condition (5.51) directly results from (5.14). We also have:

$$\begin{aligned}\textstyle\sum_{i\in\mathcal{L}_n} f_i(\boldsymbol{p}_{-n}) &\leq A_n P_n^{\min} + \textstyle\sum_{i\in\mathcal{L}_n\setminus\mathcal{A}_n} \left(P_n^{\max} - A_n P_n^{\min}\right)/(\sqrt[\alpha]{\gamma^{\alpha-1}}w_n) \\ &= A_n P_n^{\min} + \frac{P_n^{\max} - A_n P_n^{\min}}{w_n}\left(\textstyle\sum_{i\in\mathcal{L}_n\setminus\mathcal{A}_n} \sqrt[\alpha]{(1/\gamma_i)^{\alpha-1}}\right) \overset{\text{by (5.15)}}{=} P_n^{\max}.\end{aligned} \tag{5.57}$$

Thus, condition (5.52) also holds. Two cases are possible:

Case I: $\sum_{i\in\mathcal{L}_n} f_i(\boldsymbol{p}_{-n}) < P_n^{\max}$, which only happens if $\sum_{i\in\mathcal{L}_n\setminus\mathcal{A}_n} f_i(\boldsymbol{p}_{-n}) < P_n^{\max} - A_n P_n^{\min}$. Thus, there exists $l \in \mathcal{L}_n\setminus\mathcal{A}_n$ such that

$$\begin{aligned}(1/\sqrt[\alpha]{\gamma_l^{\alpha-1}})\left(1 - A_n P_n^{\min}\right)/\left(w_n + \sqrt[\alpha]{v_n(\boldsymbol{p}_{-n})}\right) &< \left(P_n^{\max} - A_n P_n^{\min}\right)/(\sqrt[\alpha]{\gamma_l^{\alpha-1}}w_n), \\ \left(1 - A_n P_n^{\min}\right)/\left(w_n + \sqrt[\alpha]{v_n(\boldsymbol{p}_{-n})}\right) &< \left(P_n^{\max} - A_n P_n^{\min}\right)/w_n.\end{aligned} \tag{5.58}$$

Multiplying both sides of this inequality by $\sqrt[\alpha]{(1/\gamma_i)^{\alpha-1}}$, for *any* link $i \in \mathcal{L}_n\setminus\mathcal{A}_n$, we have

$$(1/\sqrt[\alpha]{\gamma_i^{\alpha-1}})\left(1 - A_n P_n^{\min}\right)/\left(w_n + \sqrt[\alpha]{v_n(\boldsymbol{p}_{-n})}\right) < \left(P_n^{\max} - A_n P_n^{\min}\right)/(\sqrt[\alpha]{\gamma_i^{\alpha-1}}w_n).$$

Hence,

$$f_i(\boldsymbol{p}_{-n}) = \max\left(P_n^{\min}, (1/\sqrt[\alpha]{\gamma_i^{\alpha-1}})\left(1 - A_n P_n^{\min}\right)/\left(w_n + \sqrt[\alpha]{v_n(\boldsymbol{p}_{-n})}\right)\right), \forall i \in \mathcal{L}_n\setminus\mathcal{A}_n. \tag{5.59}$$

In this case, we have $\sigma \leq \varsigma$ and $\kappa = \min\{\sigma, \varsigma\} = \sigma$. In other words, $\mathcal{A}_n = \mathcal{B}_n$ and $A_n = B_n$. We

prove this by *contradiction.* Assume that $\sigma > \varsigma$ and $\kappa = \varsigma$. We have:

$$\frac{1-(L_n-\varsigma)P_n^{\min}}{\sum_{l=1}^{\varsigma}\sqrt[\alpha]{(1/\gamma_{i_l})^{\alpha-1}}+\sqrt[\alpha]{v_n(\boldsymbol{p}_{-n})}} \overset{\text{by (5.58)}}{<} \frac{P_n^{\max}-(L_n-\varsigma)P_n^{\min}}{\sum_{l=1}^{\varsigma}\sqrt[\alpha]{(1/\gamma_{i_l})^{\alpha-1}}} \overset{\text{by (5.13)}}{\leq} \sqrt[\alpha]{\gamma_{i_{\varsigma+1}}^{\alpha-1}}P_n^{\min}.$$

Thus,

$$1/P_n^{\min}-L_n+\varsigma < \sqrt[\alpha]{\gamma_{i_{\varsigma+1}}^{\alpha-1}}\left(\sum_{l=1}^{\sigma}\sqrt[\alpha]{(1/\gamma_{i_l})^{\alpha-1}}+\sqrt[\alpha]{v_n(\boldsymbol{p}_{-n})}\right). \tag{5.60}$$

Comparing (5.60) and (5.12), we have $i_{\varsigma+1} \in \mathcal{B}_n$. This implies that $\sigma \leq \varsigma$ which contradicts our counter assumption $\sigma > \varsigma$. Thus, $\kappa = \sigma$ and we have:

$$\mathcal{A}_n = \mathcal{B}_n, \quad A_n = B_n. \tag{5.61}$$

We can verify that conditions (5.53)-(5.55) hold for $\boldsymbol{p}_n = \boldsymbol{f}_n(\boldsymbol{p}_{-n})$ if we set $\lambda_n = 0$, and

$$\delta_i = \begin{cases} \frac{v_n(\boldsymbol{p}_{-n})}{\left(1-A_nP_n^{\min}-\sum_{j\in\mathcal{L}_n\setminus\mathcal{A}_n}f_j(\boldsymbol{p}_{-n})\right)^{\alpha}} - \left(\frac{1}{\sqrt[\alpha]{\gamma_i^{\alpha-1}}P_n^{\min}}\right)^{\alpha}, & \text{if } i \in \mathcal{A}_n, \\ 0, & \text{if } i \in \mathcal{L}_n\setminus\mathcal{A}_n. \end{cases} \tag{5.62}$$

Next, we need to show that (5.56) holds. For each $i \in \mathcal{A}_n$, we have:

$$\begin{aligned}
\frac{v_n(\boldsymbol{p}_{-n})}{\left(1 - A_n P_n^{\min} - \sum_{j\in\mathcal{L}_n\setminus\mathcal{A}_n} f_j(\boldsymbol{p}_{-n})\right)^\alpha} &\overset{\text{by (5.59)}}{\geq} \frac{v_n(\boldsymbol{p}_{-n})}{(1-A_n P_n^{\min})\left(1-\sum_{j\in\mathcal{L}_n\setminus\mathcal{A}_n} \frac{\sqrt[\alpha]{1/\gamma_j^{\alpha-1}}}{w_n+\sqrt[\alpha]{v_n(\boldsymbol{p}_{-n})}}\right)^\alpha} \\
&\overset{\text{by (5.15)}}{=} \left(\frac{\sqrt[\alpha]{\gamma_i^{\alpha-1}}\left(w_n + \sqrt[\alpha]{v_n(\boldsymbol{p}_{-n})}\right)}{\sqrt[\alpha]{\gamma_i^{\alpha-1}}\,(1 - A_n P_n^{\min})}\right)^\alpha \\
&\underset{\text{and (5.61)}}{\overset{\text{by (5.43)}}{\geq}} \left(\frac{1/P_n^{\min} - A_n}{\sqrt[\alpha]{\gamma_i^{\alpha-1}}\,(1 - A_n P_n^{\min})}\right)^\alpha \\
&= \left(\frac{1}{\sqrt[\alpha]{\gamma_i^{\alpha-1}}\,P_n^{\min}}\right)^\alpha,
\end{aligned}$$

Thus, $\delta_i \geq 0$ for all $i \in \mathcal{L}_n$ and (5.56) holds. Finally, for each $i \in \mathcal{L}_n\setminus\mathcal{A}_n$ we have:

$$\frac{P_n^{\max} - A_n P_n^{\min}}{\sqrt[\alpha]{\gamma_i^{\alpha-1}}\,w_n} \overset{\text{by (5.58)}}{>} \frac{1 - A_n P_n^{\min}}{\sqrt[\alpha]{\gamma_i^{\alpha-1}}\left(w_n + \sqrt[\alpha]{v_n(\boldsymbol{p}_{-n})}\right)} \underset{\text{by (5.61)}}{\overset{\text{and (5.45)}}{\geq}} P^{\min}, \tag{5.63}$$

which guarantees that the upper bound in (5.14) is indeed greater than the lower bound.

Case II: $\sum_{i\in\mathcal{L}_n} f_i(\boldsymbol{p}_{-n}) = P_n^{\max}$, which happens only if

$$\sum_{i\in\mathcal{L}_n\setminus\mathcal{A}_n} f_i(\boldsymbol{p}_{-n}) = P_n^{\max} - A_n P_n^{\min}. \tag{5.64}$$

We already know that

$$\sum_{i\in\mathcal{L}_n\setminus\mathcal{A}_n} (P_n^{\max} - A_n P_n^{\min})/(\sqrt[\alpha]{\gamma_i^{\alpha-1}}\,w_n) \overset{\text{by (5.15)}}{=} P_n^{\max} - A_n P_n^{\min}. \tag{5.65}$$

From (5.64) and (5.65) and since $f_i(\boldsymbol{p}_{-n}) \leq (P_n^{\max} - A_n P_n^{\min})/(\sqrt[\alpha]{\gamma_i^{\alpha-1}} w_n)$ for all $i \in \mathcal{L}_n \backslash \mathcal{A}_n$,

$$f_i(\boldsymbol{p}_{-n}) = (P_n^{\max} - A_n P_n^{\min})/(\sqrt[\alpha]{\gamma_i^{\alpha-1}} w_n), \quad \forall i \in \mathcal{L}_n \backslash \mathcal{A}_n. \tag{5.66}$$

From (5.14) and (5.66) we have

$$\left(1 - A_n P_n^{\min}\right)/(w_n + \sqrt[\alpha]{v_n(\boldsymbol{p}_{-n})}) \geq \left(P_n^{\max} - A_n P_n^{\min}\right)/w_n. \tag{5.67}$$

In this case, we have $\sigma \geq \varsigma$ and $\kappa = \min\{\sigma, \varsigma\} = \varsigma$. In other words, $\mathcal{A}_n = \mathcal{C}_n$, and $A_n = C_n$. We prove this by *contradiction.* Assume that $\sigma < \varsigma$ and $\kappa = \sigma$. We have:

$$\frac{P_n^{\max} - (L_n - \sigma) P_n^{\min}}{\sum_{l=1}^{\sigma} \sqrt[\alpha]{(1/\gamma_{i_l})^{\alpha-1}}} \overset{\text{by (5.67)}}{\leq} \frac{1 - (L_n - \sigma) P_n^{\min}}{\sum_{l=1}^{\sigma} \sqrt[\alpha]{(1/\gamma_{i_l})^{\alpha-1}} + \sqrt[\alpha]{v_n(\boldsymbol{p}_{-n})}} \overset{\text{by (5.12)}}{\leq} \sqrt[\alpha]{\gamma_{i_{\sigma+1}}^{\alpha-1}} P_n^{\min}. \tag{5.68}$$

Thus,

$$P_n^{\max}/P_n^{\min} - L_n + \sigma \leq \sqrt[\alpha]{\gamma_{i_{\sigma+1}}^{\alpha-1}} \left(\sum_{l=1}^{\sigma} \sqrt[\alpha]{(1/\gamma_{i_l})^{\alpha-1}} \right). \tag{5.69}$$

Comparing (5.69) and (5.13), we have $i_{\sigma+1} \in \mathcal{C}_n$. This implies that $\sigma \geq \varsigma$ which contradicts our counter assumption $\sigma < \varsigma$. Thus, $\kappa = \varsigma$ and we have:

$$\mathcal{A}_n = \mathcal{C}_n, \qquad A_n = C_n. \tag{5.70}$$

We can verify that conditions (5.53)-(5.55) hold for each $\boldsymbol{p}_n = \boldsymbol{f}_n(\boldsymbol{p}_{-n})$ if we set

$$\lambda_n = \left(\frac{w_n}{P_n^{\max} - A_n P_n^{\min}} \right)^{\alpha} - \left(\frac{\sqrt[\alpha]{v_n(\boldsymbol{p}_{-n})}}{1 - P_n^{\max}} \right)^{\alpha}, \tag{5.71}$$

and

$$\delta_i = \begin{cases} \left(\frac{w_n}{P_n^{\max} - A_n P_n^{\min}}\right)^\alpha - \left(\frac{1}{\sqrt[\alpha]{\gamma_i^{\alpha-1}} P_n^{\min}}\right)^\alpha, & \text{if } i \in \mathcal{A}_n, \\ 0, & \text{if } i \in \mathcal{L}_n \backslash \mathcal{A}_n. \end{cases} \tag{5.72}$$

Next, we need to show that condition (5.56) holds. From (5.67) and by reordering,

$$\begin{aligned} w_n(1 - A_n P_n^{\min}) &\geq (w_n + \sqrt[\alpha]{v_n(\boldsymbol{p}_{-n})})(P_n^{\max} - A_n P_n^{\min}) \\ w_n/(P_n^{\max} - A_n P_n^{\min}) &\geq \sqrt[\alpha]{v_n(\boldsymbol{p}_{-n})}/(1 - P^{\max}). \end{aligned} \tag{5.73}$$

On the other hand, from (5.44) and (5.70) and by reordering,

$$w_n/(P_n^{\max} - A_n P_n^{\min}) \geq 1/(\sqrt[\alpha]{\gamma_i^{\alpha-1}} P^{\min}). \tag{5.74}$$

Replacing (5.73) in (5.71), we have $\lambda_n \geq 0$. On the other hand, replacing (5.74) in (5.72), we have $\delta_i \geq 0$ for all $i \in \mathcal{L}_n$. Thus, condition (5.56) holds. Finally, for each $i \in \mathcal{L}_n \backslash \mathcal{A}_n$ we have:

$$(P_n^{\max} - A_n P_n^{\min})/(\sqrt[\alpha]{\gamma_i^{\alpha-1}} w_n) \underset{\text{by (5.70)}}{\overset{\text{and (5.46)}}{\geq}} P^{\min}, \tag{5.75}$$

which guarantees that the upper bound in (5.14) is no less than the lower bound. ■

5.6.2 Proof of Theorem 12

If $\alpha \geq 1$, then for any $\tilde{\boldsymbol{p}}, \hat{\boldsymbol{p}} \in \mathcal{P}$ such that $\tilde{\boldsymbol{p}} \preceq \hat{\boldsymbol{p}}$, we have: $\tilde{p}_j \leq \hat{p}_j$, for all $j \in \mathcal{L}$. Thus,

$$1 - \textstyle\sum_{j \in \mathcal{L}_s} \tilde{p}_j \geq 1 - \sum_{j \in \mathcal{L}_s} \hat{p}_j, \qquad \forall\, s \in \mathcal{N}, \tag{5.76}$$

$$1/\tilde{p}_j \geq 1/\hat{p}_j, \qquad \forall\, j \in \mathcal{L}. \tag{5.77}$$

From (5.76) and (5.77),

$$m_s(\tilde{\boldsymbol{p}}_{-n}) \geq m_s(\hat{\boldsymbol{p}}_{-n}), \qquad \forall\ s \in \mathcal{N}, \tag{5.78}$$

$$v_i(\tilde{\boldsymbol{p}}_{-n}) \geq v_i(\hat{\boldsymbol{p}}_{-n}), \qquad \forall\ i \in \mathcal{L}, \tag{5.79}$$

$$w_i + \sqrt[\alpha]{v_i(\tilde{\boldsymbol{p}}_{-n})} \geq w_i + \sqrt[\alpha]{v_i(\hat{\boldsymbol{p}}_{-n})} \quad \forall\ i \in \mathcal{L}, \tag{5.80}$$

$$f_i(\tilde{\boldsymbol{p}}_{-n}) \leq f_i(\hat{\boldsymbol{p}}_{-n}), \qquad \forall\ i \in \mathcal{L}. \tag{5.81}$$

Thus, vector function $\boldsymbol{f}(\boldsymbol{p})$ is a monotone increasing mapping. If $\alpha \leq 1$, then the sign of the inequalities in (5.78)-(5.81) is reversed and $\boldsymbol{f}(\boldsymbol{p})$ becomes monotone decreasing. ■

5.6.3 Proof of Theorem 13

Since H is bounded, the local memory of each node $n \in \mathcal{N}$ is updated infinitely often as $t \to \infty$. Thus, the *total asynchronism* assumption [18, pp. 430] holds. First consider the case with $\alpha \geq 1$. Let $\boldsymbol{p}^{\min}$ and $\boldsymbol{p}^{\max}$ denote two $N \times 1$ vectors with all entries equal to $P^{\min}$ and $P^{\max}$, respectively. Since $\boldsymbol{f}(\boldsymbol{p}^\star) = \boldsymbol{p}^\star, \boldsymbol{p}^{\min} \preceq \boldsymbol{p}^\star \preceq \boldsymbol{p}^{\max}$, and $\boldsymbol{f}(\cdot)$ is monotone increasing, we have:

$$\boldsymbol{p}^{\min} \preceq \boldsymbol{f}(\boldsymbol{p}^{\min}) \preceq \boldsymbol{p}^\star \preceq \boldsymbol{f}(\boldsymbol{p}^{\max}) \preceq \boldsymbol{p}^{\max}. \tag{5.82}$$

Note that, for each $n \in \mathcal{N}$ and $i \in \mathcal{L}_n$, $f_i(\boldsymbol{p}^{\min}) \geq P_n^{\min} \geq P^{\min}$ and $f_i(\boldsymbol{p}^{\max}) \leq P_n^{\max} \leq P^{\max}$. Let $\boldsymbol{f}^k(\boldsymbol{p})$ denote the composition of $\boldsymbol{f}$ with itself k times, where $\boldsymbol{f}^0(\boldsymbol{p}) = \boldsymbol{p}$. Starting from (5.82) and applying $\boldsymbol{f}$ for $k \geq 0$ times, we have:

$$\boldsymbol{f}^k(\boldsymbol{p}^{\min}) \preceq \boldsymbol{f}^{k+1}(\boldsymbol{p}^{\min}) \preceq \boldsymbol{p}^\star \preceq \boldsymbol{f}^{k+1}(\boldsymbol{p}^{\max}) \preceq \boldsymbol{f}^k(\boldsymbol{p}^{\max}). \tag{5.83}$$

We also define: $\mathcal{P}^k = \{\boldsymbol{p} : \boldsymbol{f}^k(\boldsymbol{p}^{\min}) \preceq \boldsymbol{p} \preceq \boldsymbol{f}^k(\boldsymbol{p}^{\max})\}$. If $\boldsymbol{p} \in \mathcal{P}^k$, then $\boldsymbol{f}(\boldsymbol{p}) \in \mathcal{P}^{k+1}$. From (5.83), $\mathcal{P}^{k+1} \subseteq \mathcal{P}^k$ for all $k \geq 0$. We can also show (by contradiction) that if $\mathcal{P}^k \neq \{\boldsymbol{p}^\star\}$, then $\mathcal{P}^{k+2} \neq \mathcal{P}^k$. That is, $\mathcal{P}^{k+2} \subset \mathcal{P}^k$. Thus, $\lim_{k\to\infty} \boldsymbol{f}^k(\boldsymbol{p}^{\min}) = \lim_{k\to\infty} \boldsymbol{f}^k(\boldsymbol{p}^{\max}) = \boldsymbol{p}^\star$. Besides, if $\boldsymbol{p}, \hat{\boldsymbol{p}} \in \mathcal{P}^k$, then $(p_1, \cdots, p_{i-1}, \hat{p}_i, p_{i+1}, \cdots, p_L) \in \mathcal{P}^k$ for any $k \geq 0$. Therefore, both *synchronous convergence* and *box* conditions [18, pp. 431] hold. From *asynchronous convergence theorem* [18, pp. 431], starting from any point in $\mathcal{P}^0$, Algorithm 5 will converge to $p^\star$. Since $\mathcal{P} \subseteq \mathcal{P}^0$, the proof for $\alpha \geq 1$ is complete. Now consider the case with $\alpha \leq 1$, from Theorem 12, the mapping $\boldsymbol{f}(\cdot)$ is monotone decreasing. We have:

$$\boldsymbol{p}^{\min} \preceq \boldsymbol{f}(\boldsymbol{p}^{\max}) \preceq \boldsymbol{p}^\star \preceq \boldsymbol{f}(\boldsymbol{p}^{\min}) \preceq \boldsymbol{p}^{\max}. \tag{5.84}$$

Comparing with (5.82), the orders of $\boldsymbol{f}(\boldsymbol{p}^{\min})$ and $\boldsymbol{f}(\boldsymbol{p}^{\max})$ is exchanged in (5.84). Applying $f(\cdot)$ once more, we have:

$$\boldsymbol{f}(\boldsymbol{p}^{\max}) \preceq \boldsymbol{f}^2(\boldsymbol{p}^{\min}) \preceq \boldsymbol{p}^\star \preceq \boldsymbol{f}^2(\boldsymbol{p}^{\max}) \preceq \boldsymbol{f}(\boldsymbol{p}^{\min}).$$

By mathematical induction, we can show that

$$\begin{aligned} &\boldsymbol{f}^k(\boldsymbol{p}^{\max}) \preceq \boldsymbol{f}^{k+1}(\boldsymbol{p}^{\min}) \preceq \boldsymbol{p}^\star \preceq \boldsymbol{f}^{k+1}(\boldsymbol{p}^{\max}) \preceq \boldsymbol{f}^k(\boldsymbol{p}^{\min}), && \text{if } k \text{ is } \textit{odd}, \\ &\boldsymbol{f}^k(\boldsymbol{p}^{\min}) \preceq \boldsymbol{f}^{k+1}(\boldsymbol{p}^{\max}) \preceq \boldsymbol{p}^\star \preceq \boldsymbol{f}^{k+1}(\boldsymbol{p}^{\min}) \preceq \boldsymbol{f}^k(\boldsymbol{p}^{\max}), && \text{if } k \text{ is } \textit{even}. \end{aligned}$$

For each (either odd or even) $k \geq 0$, we define:

$$\mathcal{P}^k = \left\{\boldsymbol{p} : \min\left[\boldsymbol{f}^k(\boldsymbol{p}^{\min}), \boldsymbol{f}^k(\boldsymbol{p}^{\max})\right] \preceq \boldsymbol{p} \preceq \max\left[\boldsymbol{f}^k(\boldsymbol{p}^{\min}), \boldsymbol{f}^k(\boldsymbol{p}^{\max})\right]\right\}.$$

The rest of the proof follows the case of $\alpha \geq 1$. ■

5.6.4 Proof of Theorem 14

For each $\hat{\boldsymbol{p}}^\star \in \mathcal{F}$ and each link $i \in \mathcal{N}$, Theorem 10 states that $\hat{p}_i^\star$ is the unique global optimal solution of convex problem (LOCAL-NUM), and satisfies the KKT conditions in (5.51)-(5.56). We denote the corresponding Lagrange multipliers by $\hat{\lambda}_n^\star$ and $\hat{\delta}_i^\star$ for all $i \in \mathcal{L}_n$. We also define $\hat{\boldsymbol{\lambda}}^\star = (\hat{\lambda}_n^\star,\ \forall\, n \in \mathcal{N})$ and $\hat{\boldsymbol{\delta}}^\star = (\hat{\delta}_i^\star,\ \forall\, i \in \mathcal{L})$. On the other hand, since any local optimum $\hat{\boldsymbol{p}} \in \mathcal{S}$ is a *regular* point [15, pp. 315] of non-convex problem (5.7), and so must satisfy the KKT *necessary* conditions [15, Proposition 3.3.1], for any $i \in \mathcal{N}$,

$$\tilde{p}_i^\star \geq P_n^{\min}, \qquad \forall\, n \in \mathcal{N},\ i \in \mathcal{L}_n, \tag{5.85}$$

$$\textstyle\sum_{i \in \mathcal{L}_n} \tilde{p}_i^\star \leq P_n^{\max}, \qquad \forall\, n \in \mathcal{N}, \tag{5.86}$$

$$\frac{1}{y_n(\tilde{\boldsymbol{p}}_{-n}^\star)} \left(\frac{1}{\gamma_i^{\alpha-1} p_i^\alpha} - \frac{v_n(\boldsymbol{p}_{-n}^\star)}{\left(1 - \sum_{i \in \mathcal{L}_n} p_i\right)^\alpha} \right) = \tilde{\lambda}_n^\star - \tilde{\delta}_i^\star, \qquad \forall\, n \in \mathcal{N}, i \in \mathcal{L}_n, \tag{5.87}$$

$$\tilde{\lambda}_n^\star \left(\textstyle\sum_{i \in \mathcal{L}_n} p_i - P_n^{\max} \right) = 0, \qquad \forall\, n \in \mathcal{N}, \tag{5.88}$$

$$\tilde{\delta}_i^\star \left(P_n^{\min} - p_i \right) = 0, \qquad \forall\, n \in \mathcal{N},\ i \in \mathcal{L}_n, \tag{5.89}$$

$$\tilde{\lambda}_n^\star \geq 0, \quad \tilde{\delta}_i^\star \geq 0, \qquad \forall\, i \in \mathcal{L}_n, \tag{5.90}$$

where

$$y_n(\boldsymbol{p}_{-n}) = \left(\textstyle\prod_{s \in \mathcal{N} \setminus \{n\}} (1 - \sum_{j \in \mathcal{L}_s} p_j)\right)^{\alpha-1} > 0. \tag{5.91}$$

We define, $\tilde{\boldsymbol{\lambda}}^\star = (\tilde{\lambda}_n^\star,\ \forall\, n \in \mathcal{N})$ and $\hat{\boldsymbol{\delta}}^\star = (\hat{\delta}_i^\star,\ \forall\, i \in \mathcal{L})$. If $\left(\hat{\boldsymbol{p}}^\star, \hat{\boldsymbol{\lambda}}^\star, \hat{\boldsymbol{\delta}}^\star\right)$ satisfies the KKT

conditions in (5.51)-(5.56) for all wireless nodes $n \in \mathcal{N}$, then the tuple:

$$\left(\hat{\boldsymbol{p}}^\star, (\hat{\lambda}_n^\star / y_n(\hat{\boldsymbol{p}}_{-n}^\star), \forall n \in \mathcal{N}), (\hat{\delta}_i^\star / y_n(\hat{\boldsymbol{p}}_{-n}^\star), \forall n \in \mathcal{N},\ i \in \mathcal{L}_n)\right) \tag{5.92}$$

satisfies (5.85)-(5.90). On the other hand, if tuple $\left(\tilde{\boldsymbol{p}}^\star, \tilde{\boldsymbol{\lambda}}^\star, \tilde{\boldsymbol{\delta}}^\star\right)$ satisfies the KKT conditions in (5.85)-(5.90), then the tuple

$$\left(\tilde{\boldsymbol{p}}^\star, (\tilde{\lambda}_n^\star y_n(\tilde{\boldsymbol{p}}_{-n}^\star), \forall n \in \mathcal{N}), (\tilde{\delta}_i^\star y_n(\tilde{\boldsymbol{p}}_{-n}^\star), \forall n \in \mathcal{N},\ i \in \mathcal{L}_n)\right) \tag{5.93}$$

satisfies conditions (5.51)-(5.56) for all $n \in \mathcal{N}$. The former implies that $\mathcal{F} \subseteq \mathcal{S}$ while the later implies $\mathcal{S} \subseteq \mathcal{F}$. Thus, set $\mathcal{F} = \mathcal{S}$. ■

5.6.5 Proof of Theorem 15

For any $\boldsymbol{p} \in \mathcal{P}$, the Jacobian $J(\boldsymbol{p})$ is defined as an $L \times L$ matrix whose entry in row i and column j is $\partial f_i / \partial p_j$. Consider node $n \in \mathcal{N}$ and each $i \in \mathcal{L}_n$, we have:

$$J_{i,j}(\boldsymbol{p}) = \begin{cases} 0, & \text{if } j \in \mathcal{L}_n, \\ \dfrac{\frac{\alpha-1}{\alpha}\frac{(1-A_n P_n^{\min})}{1-\sum_{k\in\mathcal{L}_s} p_k}\left[m_s + \left(\frac{1-\sum_{k\in\mathcal{L}_s} p_k}{p_j \gamma_j}\right)^{\alpha-1}\left(\frac{1-\sum_{k\in\mathcal{L}_s} p_k}{p_j}\right)\right]}{\sqrt[\alpha]{\gamma_i^{\alpha-1}} v_n(\boldsymbol{p}_{-n})^{1-1/\alpha}(w_n + v_n(\boldsymbol{p}_{-n})^{1/\alpha})^2}, & \text{if } k \in \mathcal{L}_s,\ s \neq n. \end{cases} \tag{5.94}$$

This is under the assumption that $f_i \in \left(P_n^{\min}, (P_n^{\max} - A_n P_n^{\min}) / \sqrt[\alpha]{\gamma_i^{\alpha-1}} w_n\right)$. If f_i is chosen to be one of the boundary points, then the corresponding entry is 0. Thus, (5.94) provides

upper-bounds (in absolute values) on the entries of the Jacobian. We can show that:

$$\begin{aligned}\|J\|_\infty &= \max_{i\in\mathcal{L}_n, n\in\mathcal{N}} \left\{ \sum_{s\in\mathcal{N}\setminus\{n\}} \frac{\frac{1-\alpha}{\alpha}(1-A_nP_n^{\min})\sum_{j\in\mathcal{L}_s}\left(\frac{1-\sum_{k\in\mathcal{L}_s}p_k}{p_j\gamma_j}\right)^{\alpha-1}\left(\frac{\mathcal{L}_s}{1-\sum_{k\in\mathcal{L}_s}p_k}+\frac{1}{p_j}\right)}{\sqrt[\alpha]{\gamma_i^{\alpha-1}}v_n(\boldsymbol{p}_{-n})^{1-1/\alpha}(w_n+v_n(\boldsymbol{p}_{-n})^{1/\alpha})^2} \right\} \\ &\le \frac{1-\alpha}{\alpha}\max_{i\in\mathcal{L}_n, n\in\mathcal{N}}\left\{\frac{(\gamma_i v_n(\boldsymbol{p}_{-n}))^{1/\alpha}}{(1+(\gamma_i v_n(\boldsymbol{p}_{-n}))^{1/\alpha})^2}\right\}\left(\frac{L^{\max}}{1-P^{\max}}+\frac{1}{P^{\min}}\right) \\ &\le \frac{1-\alpha}{\alpha}\Psi\,\Phi(V^{\min},V^{\max}).\end{aligned} \tag{5.95}$$

Note that $1-A_nP_n^{\min}\le 1$ and $\sqrt[\alpha]{\gamma_i^{\alpha-1}}w_n\ge 1$ for all $i\in\mathcal{L}_n\setminus\mathcal{A}_n$. Function $x^{1/\alpha}/(1+x^{1/\alpha})^2$ is always non-negative and less than 0.25. It has a unique maximum at $x=1$. The function is monotonically increasing for $0\le x<1$ and monotonically decreasing for $x>1$. Its value approaches zero as either $x\to 0$ or $x\to\infty$. We can similarly show that:

$$\|J(\boldsymbol{p})\|_1 \le \frac{1-\alpha}{\alpha}\,\Psi\,\Phi(V^{\min},V^{\max})\left(\frac{\gamma^{\max}}{\gamma^{\min}}\Gamma\right)^{1-\alpha}\left(\Omega-\frac{1}{L/L^{\min}-1}\right). \tag{5.96}$$

From (5.26), (5.95), and (5.96), we have [18, pp. 635]:

$$\|J(\boldsymbol{p})\|_2 \le \sqrt{\|J(\boldsymbol{p})\|_\infty\,\|J(\boldsymbol{p})\|_1} < 1. \tag{5.97}$$

Let $\tilde{\boldsymbol{p}},\hat{\boldsymbol{p}}\in\mathcal{P}$. From (5.97) and by Cauchy Schwarz inequality,

$$\|\boldsymbol{f}(\tilde{\boldsymbol{p}})-\boldsymbol{f}(\hat{\boldsymbol{p}})\|_2 \le \|J(\boldsymbol{p})\|_2\,\|\tilde{\boldsymbol{p}}-\hat{\boldsymbol{p}}\|_2 < \|\tilde{\boldsymbol{p}}-\hat{\boldsymbol{p}}\|_2,$$

where $\boldsymbol{p}$ is a convex combination of $\tilde{\boldsymbol{p}}$ and $\hat{\boldsymbol{p}}$. Thus, $\boldsymbol{f}$ is l_2-norm contraction mapping. ■

5.6.6 Proof of Theorem 18

Following the same argument as in Section 5.6.5, condition (5.36) implies that for each $\boldsymbol{p}\in\mathcal{P}$, we have: $\sqrt{N}\|J'(\boldsymbol{p})\|_2<1$ where $J'(\boldsymbol{p})$ denotes the Jacobian matrix of $\boldsymbol{f}'(\boldsymbol{p})$. From linear algebra, we also know that for each $N\times 1$ vector $\boldsymbol{a}$, we have $\|\boldsymbol{a}\|_\infty\le\|\boldsymbol{a}\|_2\le\sqrt{N}\|\boldsymbol{a}\|_\infty$. Let $\tilde{\boldsymbol{p}},\hat{\boldsymbol{p}}\in\mathcal{P}$.

Using the Cauchy-Schwarz inequality,

$$\|\boldsymbol{f}'(\tilde{\boldsymbol{p}}) - \boldsymbol{f}'(\hat{\boldsymbol{p}})\|_\infty \leq \|\boldsymbol{f}'(\tilde{\boldsymbol{p}}) - \boldsymbol{f}'(\hat{\boldsymbol{p}})\|_2 \leq \|J'(\boldsymbol{p})\|_2 \; \|\tilde{\boldsymbol{p}} - \hat{\boldsymbol{p}}\|_2 < \|\tilde{\boldsymbol{p}} - \hat{\boldsymbol{p}}\|_2/\sqrt{N} \leq \|\tilde{\boldsymbol{p}} - \hat{\boldsymbol{p}}\|_\infty.$$

Condition (5.36) guarantees that function $\boldsymbol{f}'(\cdot)$ is an l_∞ norm contraction. Thus, it asynchronously converges to its *unique* fixed point. From Theorem 16, this implies that Algorithm 6 converges to the unique global optimal solution of problem (5.7). ■

Chapter 6

Utility Optimal Random Access: Optimal Performance without Frequent Explicit Message Passing

In the existing contention-based MAC protocols, there is a tradeoff between system performance (e.g., throughput and fairness) and the amount of explicit message passing required among wireless users. One example is the IEEE 802.11 DCF, where users do not explicitly exchange any message related to their transmission probabilities and adapt their transmission probabilities only based on the binary implicit feedback from the network (e.g., collision or not). In this chapter, we use "messages" to denote control signals that are explicitly related to users' transmission probabilities. IEEE 802.11 DCF does not have any explicit message passing, although it has various other control signals (e.g., RTS/CTS/ACK). Lack of message exchange typically leads to low throughput and unfair resource allocation [97]. On the other hand, several MAC algorithms (e.g., [47, 51, 98]) have been designed based on the framework of NUM which lead to the optimal system performance without taking the signalling overhead into account. However, these algorithms require extensive frequent message passing among users. Considering the fact that any message transmission leads to additional contention in a random access network, this chapter

aims to address the following question: *Is it possible to design a MAC algorithm that can achieve the optimal performance without frequent explicit message passing?*

We provide a positive answer to the above question in some special but important cases, based on the NUM-based MAC algorithms we proposed in [104]. Compared with the previous algorithms (e.g., [47, 51, 98]), the algorithms in [104] support a wider range of utility functions, converge faster, and allow fully asynchronous operations among users. However, frequent explicit message passings are still needed in [104]. In this chapter, we show that in the simple case of a single-cell interference topology (e.g., as in wireless personal and local area networks), we can completely eliminate the need for frequent message passing. Users will be able to estimate the required information through local observation of the channel contention history. We prove the convergence of our algorithm under various channel conditions. If the channel is perfect and the estimations are asymptotically accurate, then the optimality of the algorithm is also guaranteed. The estimation techniques we use here are related to [100, 105]. However, our estimation model is more elaborate and captures more information (i.e., each user's transmission probability). Simulation results show that our algorithm is robust to changes in user populations and channel conditions. These encouraging results provide important insights and useful hints to design fully distributed utility optimal MAC algorithms without frequent explicit message passing for more general topologies.

The rest of this chapter is organized as follows. The system model is described in Section 6.1. Our algorithm is presented in Section 6.2. Convergence and optimality are proved in Section 6.3. Simulation results are shown in Section 6.4. A summary is given in Section 6.5.

6.1 System Model

Consider a single-hop wireless ad-hoc network with $\mathcal{N} = \{1, \ldots, N\}$ as the set of wireless links. Each link, together with its dedicated transmitter and receiver nodes, is called a *user*. A sample network with 3 users is shown in Fig. 6.1. We assume that each user's receiver node can hear every other user's transmissions. Thus, each user interferes with all other users. This models some important wireless networks including *wireless personal area networks* where wireless devices interact with each other (e.g., in an office) and indoor *wireless local area networks* where the nodes interact with each other and an access point (e.g., in a large conference room). Time is divided into equal-length slots. At each slot, user i transmits with probability $p_i \in \mathcal{P}_i = [P_i^{\min}, P_i^{\max}]$, with $0 < P_i^{\min} < P_i^{\max} < 1$. A transmission is successful only if it is the only transmission in the current slot. Let r_i denote the average data rate for user i. We have [80]:

$$r_i(\boldsymbol{p}) = \gamma_i p_i \prod_{j \in \mathcal{N} \setminus \{i\}} (1 - p_j), \qquad \forall\, i \in \mathcal{N}, \tag{6.1}$$

where $\boldsymbol{p} = (p_i, \forall i \in \mathcal{L})$ is the vector of all users' transmission probabilities and γ_i denotes the fixed peak data rate for user i. Each link $i \in \mathcal{L}$ maintains a *utility* which is an increasing and concave function of r_i and indicates link i's level of satisfaction on its average data rate. The utility of link i is denoted by $u_i(r_i(\boldsymbol{p}))$ which is also a function of $\boldsymbol{p}$. We are interested in finding the value of $\boldsymbol{p}$ that solves the following NUM problem [44, 53]:

$$\max_{\boldsymbol{p} \in \mathcal{P}} \sum_{i \in \mathcal{N}} u_i(r_i(\boldsymbol{p})), \tag{6.2}$$

where $\mathcal{P} = \{\boldsymbol{p} : p_i \in \mathcal{P}_i, \forall i \in \mathcal{N}\}$, and the utility functions are α-fair [50]. That is, $u_i(r_i(\boldsymbol{p})) = (1-\alpha)^{-1} r_i(\boldsymbol{p})^{1-\alpha}$ if $\alpha \in (0,1) \cup (1, \infty)$, and $u_i(r_i(\boldsymbol{p})) = \log r_i(\boldsymbol{p})$, if $\alpha = 1$. In [104], we have shown that the α-fair utility functions can model a wide range of efficient and fair allocations.

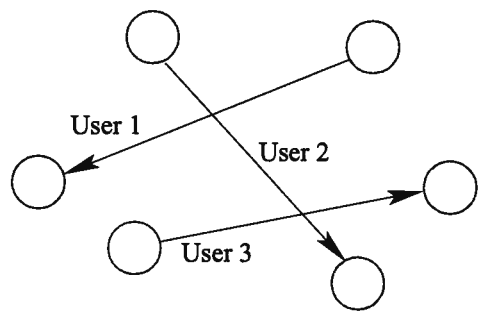


Figure 6.1: A single-hop wireless ad-hoc network with $N = 3$ users. Each user includes a wireless link and its dedicated transmitter and receiver nodes.

6.2 Algorithm with No Explicit Message Passing

Local Optimization

For each user i, consider the following *local* optimization problem:

$$\max_{p_i \in \mathcal{P}_i} \sum_{j \in \mathcal{N}} u_j(r_j(p_i, \boldsymbol{p}_{-i})), \tag{6.3}$$

where $\boldsymbol{p}_{-i} = (p_j,\ \forall j \in \mathcal{N} \backslash \{i\})$ denotes the transmission probabilities of all users *other than* user i. To solve problem (6.3), user i will choose p_i to maximize the *total* network utility, *assuming* that none of the other users change their transmission probabilities.

Theorem 19 *For each user $i \in \mathcal{N}$, the unique global optimal solution of problem (6.3) is $p_i^*(\boldsymbol{p}_i) = f_i(\boldsymbol{p}_i)$, where the mapping function $f_i(\boldsymbol{p}_i)$ is defined as*

$$f_i(\boldsymbol{p}_{-i}) = \left[1/\left(1 + \sqrt[\alpha]{v_i(\boldsymbol{p}_{-i})}\right)\right]_{P_i^{\min}}^{P_i^{\max}}. \tag{6.4}$$

Here $[x]_b^a = \max[\min[x, a], b]$ and $v_i(\boldsymbol{p}_{-i}) = {\gamma_i}^{\alpha-1} \sum_{j \in \mathcal{N} \backslash \{i\}} (1/\gamma_j)^{\alpha-1} (1/p_j - 1)^{\alpha-1}$.

The proof of Theorem 19 is similar to that of Theorem 13 and is omitted for brevity. It is clear that if user i wants to compute (6.4), the only information it needs from other users is $v_i(\boldsymbol{p}_{-i})$.

If each user i can *estimate* the value of

$$m_j = (1/\gamma_j)^{\alpha-1}\,(1/p_j - 1)^{\alpha-1}, \quad \forall\, j \in \mathcal{N}, \tag{6.5}$$

then it can compute $v_i(\boldsymbol{p}_{-i}) = {\gamma_i}^{\alpha-1} \sum_{j\in\mathcal{N}\setminus\{i\}} m_j$ and set $p_i = f_i(\boldsymbol{p}_{-i})$. Notice that for each $j \in \mathcal{N}$, m_j is bounded between $M^{\min}$ and $M^{\max}$. If $\alpha \geq 1$, then $M^{\min} = (1/\gamma^{\max})^{\alpha-1}(1/P^{\max} - 1)^{\alpha-1}$ and $M^{\max} = (1/\gamma^{\min})^{\alpha-1}(1/P^{\min} - 1)^{\alpha-1}$ where $P^{\min} = \min_{i\in\mathcal{N}} P_i^{\min}$, $P^{\max} = \max_{i\in\mathcal{N}} P_i^{\max}$, $\gamma^{\min} = \min_{i\in\mathcal{N}} \gamma_i$, and $\gamma^{\max} = \max_{i\in\mathcal{N}} \gamma_i$. If $\alpha < 1$, then $M^{\min} = (1/\gamma^{\min})^{\alpha-1}(1/P^{\min} - 1)^{\alpha-1}$ and $M^{\max} = (1/\gamma^{\max})^{\alpha-1}(1/P^{\max} - 1)^{\alpha-1}$. As shown in [104, Section IV-A], if each user i updates its transmission probability p_i accordingly to (6.4), then the whole system will converge to the optimal solution of problem (6.2). The key question is how to obtain the values of m_j for all $j \neq i$. Next, we answer this question through local observations of the shared channel.

Learning from Contention History

From (6.5), we see that only the values of γ_j and p_j are required to calculate the value of m_j. Notice that α is the same for all users. The value of the peak rate γ_j depends on the channel gain between the transmitter and receiver of user j; thus, it can only be measured by user i and then announced to the whole network once user i joins the network. The remaining task is to determine how to obtain the value of p_j.

From user i's viewpoint, any time slot falls into one of the following possible states: *idle* (no user transmits), *busy* (at least one other user transmits), *success* (user i transmits successfully), and *failure* (user i transmits but it fails). Let p_i^{idle}, p_i^{busy}, p_i^{succ}, and p_i^{fail} denote the probabilities of experiencing these four states, respectively. Also let $p_{i,j}^{\text{err}}$ denote the *packet error rate* of the

channel from the transmitter node of user j to the receiver node of user i. We have:

$$p_i^{\text{idle}} = \prod_{j\in\mathcal{N}}(1-p_j), \tag{6.6}$$

$$\begin{aligned} p_i^{\text{busy}} &= (1-p_i)(1-\prod_{j\in\mathcal{N}\setminus\{i\}}(1-p_j)) = (1-p_i) - \prod_{j\in\mathcal{N}}(1-p_j) \\ &= (1-p_i) - p_i^{\text{idle}}, \end{aligned} \tag{6.7}$$

$$\begin{aligned} p_i^{\text{succ}} &= p_i(\prod_{j\in\mathcal{N}\setminus\{i\}}(1-p_j))(1-p_{i,i}^{\text{err}}) = \tfrac{p_i}{1-p_i}(\prod_{j\in\mathcal{N}}(1-p_j))(1-p_{i,i}^{\text{err}}) \\ &= \tfrac{p_i}{1-p_i}p_i^{\text{idle}}(1-p_{i,i}^{\text{err}}), \end{aligned} \tag{6.8}$$

$$\begin{aligned} p_i^{\text{fail}} &= p_i((1-\prod_{j\in\mathcal{N}\setminus\{i\}}(1-p_j)) + p_{i,i}^{\text{err}} - (1-\prod_{j\in\mathcal{N}\setminus\{i\}}(1-p_j))p_{i,i}^{\text{err}}) \\ &= p_i - (\frac{p_i}{1-p_i}p_i^{\text{idle}})(1-p_{i,i}^{\text{err}}). \end{aligned} \tag{6.9}$$

From (6.6), the channel is idle if all users are silent. We note that for each user $j \in \mathcal{N}$, the probability of being silent is $(1-p_j)$. From (6.7), the channel is busy from user i's viewpoint if user i is silent (so that it can sense the channel) and at least one other user $j \neq i$ is transmitting packets. The former has probability $(1-p_i)$ while the latter has probability $(1-\prod_{j\in\mathcal{N}\setminus\{i\}}(1-p_j))$. Multiplying the two terms and after reordering, the probability in (6.7) is resulted. From (6.8), the transmission from the transmitter node of user i to the receiver node of user i is successful if the transmitter node of user i transmits the packet, no other user $j \neq i$ transmits any packet at the same time, and the transmitted packet is not corrupted. Notice that the latter happens with probability $(1-p_{i,i}^{\text{err}})$. Finally, from (6.9), user i observes a failure slot if it transmits a packet and the transmitted packet either collides with some other transmission(s), or gets corrupted, or both. We notice that p_i^{idle} and p_i^{busy} do not depend on the value of $p_{i,i}^{\text{err}}$, while p_i^{succ} and p_i^{fail} do. Since user i knows the local transmission probability p_i, it can estimate p_i^{idle} using either (6.6) or any of the expressions in (6.7)-(6.9). However, it is clear that user i is still unable to estimate the *individual* transmission probability p_j for any $j \neq i$ even if it can accurately estimate all the state

probabilities p_i^{idle}, p_i^{busy}, p_i^{succ}, and p_i^{fail}. In fact, finding the values of *individual* transmission probabilities requires gathering more *individual* information from other users as we explain next.

Recall that, at a busy slot seen by user $i \in \mathcal{N}$, at least one other user transmits. Since users can hear each other, user i may successfully *decode* the transmission of user $j \neq i$ with probability:

$$p_{i,j}^{\text{decd}} = p_j(\prod_{l\in\mathcal{N}\setminus\{j\}}(1-p_l))(1-p_{i,j}^{\text{err}}) = (p_j/(1-p_j))\left(\prod_{l\in\mathcal{N}}(1-p_l)\right)(1-p_{i,j}^{\text{err}}). \tag{6.10}$$

Let $n_{i,j}^{\text{decd}}$ denote the number of slots between any two consecutive successful decoding of transmissions of user j by user i. We have:

$$p_{i,j}^{\text{decd}} = 1/(1+\bar{n}_{i,j}^{\text{decd}}), \tag{6.11}$$

where $\bar{n}_{i,j}^{\text{decd}}$ is the mean value of $n_{i,j}^{\text{decd}}$ and can be *locally* estimated by user i through observation of the channel contention history. Notice that in practice, the transmitted signal by user j can be decoded by the network interface of user i's receiver node; however, as its destination MAC address is not the same as the one in user i, the packet is simply discarded. Now, user i needs to obtain the sender's MAC address from the packet header before discarding the packet.

Similarly, let n_i^{idle} denote the number of non-idle slots that user i observes between any two consecutive idle time slots. User i can estimate p_i^{idle} as follows [100]:

$$p_i^{\text{idle}} = 1/(1+\bar{n}_i^{\text{idle}}), \tag{6.12}$$

where $\bar{n}_i^{\text{idle}}$ is the mean value of n_i^{idle}. Substituting (6.6), (6.11), and (6.12) into (6.10), for each $j \in \mathcal{N}\setminus\{i\}$, we have:

$$1/p_j - 1 = \left((1+\bar{n}_{i,j}^{\text{decd}})/(1+\bar{n}_i^{\text{idle}})\right)(1-p_{i,j}^{\text{err}}). \tag{6.13}$$

Let T_{idle}^i and $T_{j,\text{idle}}^i$ denote the set of time slots at which user i observes an idle slot and

decodes the transmissions of user $j \neq i$, respectively. We estimate $\bar{n}_i^{\text{idle}}$ and $\bar{n}_{i,j}^{\text{decd}}$ iteratively as:

$$\bar{n}_i^{\text{idle}}(t+1) = (1-\rho_i(t))\bar{n}_i^{\text{idle}}(t)+\rho_i(t)n_i^{\text{idle}}(t)I\{t\in T_{\text{idle}}^i\}, \tag{6.14}$$

$$\bar{n}_{i,j}^{\text{decd}}(t+1) = (1-\varrho_{i,j}(t))\bar{n}_{i,j}^{\text{decd}}(t)+\varrho_{i,j}(t)n_{i,j}^{\text{decd}}(t)I\{t\in T_{j,\text{decd}}^i\}, \tag{6.15}$$

where $\bar{n}_i^{\text{idle}}(t)$, $n_i^{\text{idle}}(t)$, $\bar{n}_{i,j}^{\text{decd}}(t)$ and $n_{i,j}^{\text{decd}}(t)$ denote the estimation of $\bar{n}^{\text{idle}}$, the measurement of n_i^{idle}, the estimation of $\bar{n}_{i,j}^{\text{decd}}$, and the measurement of $n_{i,j}^{\text{decd}}$ at time slot t, respectively, and $I\{\cdot\}$ is an indicator function. Here ρ_i and $\varrho_{i,j}$ are *tapering* stepsizes. Based on the asynchronous stochastic approximation theory [106], we know that the estimation error decreases to zero when users do not change their transmission probabilities.

For each user i and any other user $j \neq i$, given γ_j, $\bar{n}_{i,j}^{\text{decd}}$ and $\bar{n}_i^{\text{idle}}$, we define:

$$m_j^i(t) = (1/\gamma_j)^{\alpha-1}\left((1+\bar{n}_{i,j}^{\text{decd}}(t))/(1+\bar{n}_i^{\text{idle}}(t))\right)^{\alpha-1}, \qquad \forall\, j \in \mathcal{N}\backslash\{i\}, \tag{6.16}$$

where $m_j^i(t)$ denotes the estimation of m_j made by user i at time slot t. In general,

$$m_j^i(t) = \beta_j^i(t)\, m_j(t), \tag{6.17}$$

where $\beta_j^i(t) > 0$ is the *estimation gain*, which can represent accurate estimation (i.e., $\beta_j^i(t) = 1$), over-estimation (i.e., $\beta_j^i(t) > 1$) or under-estimation (i.e., $\beta_j^i(t) < 1$). From (6.13), if the estimations on $\bar{n}_{i,j}^{\text{decd}}$ and $\bar{n}_i^{\text{idle}}$ are accurate and the channel is perfect (with zero packet error rate), then $\beta_j^i(t) = 1$ and we have $m_j^i(t) = m_j(t)$ for all $j \in \mathcal{N}\backslash\{i\}$. Notice that if the value of the existing packet error rate $p_{i,j}^{\text{err}}$ is known (e.g., via measurements at the physical layer), then we can redefine $m_j^i(t) = (1/\gamma_j)^{\alpha-1}\left((1+\bar{n}_{i,j}^{\text{decd}}(t))/(1+\bar{n}_i^{\text{idle}}(t))\right)^{\alpha-1}/(1-p_{i,j}^{\text{err}})^{\alpha-1}$ and obtain a more accurate estimation by canceling out the effect of channel imperfections. However, in this chapter, we consider the general case and assume that the packet error rates are not known.

For each user $i \in \mathcal{N}$ and for all $j \in \mathcal{N}\backslash\{i\}$, we set $T_{j,m}^i$ such that as time goes by, the minimum difference between any two consecutive time slots in the union of sets $\{T_{j,m}^i, \forall\, j \in$

$\mathcal{N}\backslash\{i\}\}$ increases. This implies that for each j, we update m_j less frequently to be able to collect more samples of n_i^{idle} and $n_{i,j}^{\text{decd}}$. Thus, the estimations of mean values $\bar{n}_i^{\text{idle}}$ and $\bar{n}_{i,j}^{\text{decd}}$ improve gradually and become asymptotically accurate. We also reset the tapering stepsizes ρ_i and $\varrho_{i,j}$ to 1 after each $t \in T_{j,m}^i$ so that the errors in previous estimations do not affect new estimations. Based on these assumptions, there exists a $\beta_j^i > 0$ such that $\lim_{t\to\infty} \beta_j^i(t) = \beta_j^i$. From (6.16) and (6.17),

$$\beta_j^i = 1/(1 - p_{i,j}^{\text{err}})^{\alpha-1}, \quad \forall i, j \in \mathcal{N},\ i \neq j.$$

If the channel is perfect, then $\beta_j^i = 1$ and all estimations are *asymptotically accurate.* For a lossy channel, if $\alpha < 1$, then $\beta_j^i < 1$ and m_j^i is *asymptotically under-estimated* for all $j \neq i$. On the other hand, if $\alpha > 1$, then $\beta_j^i > 1$ and m_j^i is *asymptotically over-estimated.*

Distributed Algorithm

Our proposed distributed MAC algorithm with no explicit message passing (except when each user joins or leaves the network) is shown in Algorithm 7. In this algorithm, each user $i \in \mathcal{N}$ continuously updates $\bar{n}_i^{\text{idle}}$ and $\bar{\boldsymbol{n}}_i^{\text{decd}} = (\bar{n}_{i,j}^{\text{decd}},\ \forall j \in \mathcal{N}\backslash\{i\})$ based on its local observations from the shared channel to estimate $\boldsymbol{m}_i = (m_j^i,\ \forall j \in \mathcal{N}\backslash\{i\})$. Then, it chooses p_i according to (6.4) with $v_i = \sum_{j\in\mathcal{N}\backslash\{i\}} m_j^i$. Sets $T_{i,p}$ and $T_{i,m}$ are two unbounded sets of time slots at which user i updates p_i and $\boldsymbol{m}_i$, respectively. Notice that the updates are *asynchronous* across users which includes synchronous updates as a special case.

Algorithm 7 *Executed by each user $i \in \mathcal{N}$.*

Allocate memory for p_i and $\boldsymbol{m}^i = (m_1, \cdots, m_N)$.
Allocate memory for $\bar{n}_i^{\text{decd}}$ and $\bar{\boldsymbol{n}}_i^{\text{decd}} = (\bar{n}_{i,1}^{\text{decd}}, \cdots, \bar{n}_{i,N}^{\text{decd}})$.
Randomly choose $p_i \in \left[P_i^{\min}, P_i^{\max}\right]$.
Randomly choose $m_j^i \in \left[M^{\min}, M^{\max}\right]$ for all $j \in \mathcal{N}$.
Choose $\bar{n}_i^{\text{idle}} = 1$ and $\bar{n}_{i,j}^{\text{decd}} = 1$ for all $j \in \mathcal{N}$.
Broadcast the fixed data rate γ_i to all other users.
repeat
  Transmit with probability p_i.
  Update $\bar{n}_i^{\text{idle}}$ and $\bar{\boldsymbol{n}}_i^{\text{decd}}$ according to Eqs. (6.14) and (6.15).
  if $t \in T_{i,p}$ **then**
    Update $p_i = \left[1/\left(1+\sqrt[\alpha]{\gamma_i{}^{\alpha-1}\sum_{j\in\mathcal{N}\setminus\{i\}} m_j^i}\,\right)\right]_{P_i^{\min}}^{P_i^{\max}}$.
  end if
  if $t \in T_{j,m}^i$ **then**
    Update m_j^i according to Eq. (6.16).
  end if
until the user decides to leave the network.
Broadcast termination message.

6.3 Convergence and Optimality

For each $i \in \mathcal{N}$, and at any time $t \in T_{i,p}$, Algorithm 7 updates

$$p_i(t+1) = f_i'(\boldsymbol{p}_{-i}, t) = \left[1/\left(1+\sqrt[\alpha]{v_i'(\boldsymbol{p}_{-i}, t)}\,\right)\right]_{P_i^{\min}}^{P_i^{\max}},$$

where $v_i'(\boldsymbol{p}_{-i}, t) = \sum_{j\in\mathcal{N}\setminus\{i\}} (\gamma_i/\gamma_j)^{\alpha-1} (1/p_j - 1)^{\alpha-1} \beta_j^i(t)$. For any $t \geq 0$, we define $f'(\boldsymbol{p}, t) = (f_i'(\boldsymbol{p}_{-i}, t), \forall i \in \mathcal{N})$. Notice that $f'(\boldsymbol{p}, t)$ is a time-varying vector mapping. Since $\beta_j^i(t)$ approaches β_j^i as $t \to \infty$ for all $i, j \in \mathcal{N}$, the sequence of mapping $\{f'(\boldsymbol{p}, t)\}$ converges to a unique mapping $f'(\boldsymbol{p}, \infty)$ as $t \to \infty$. That is, for any $\boldsymbol{p} \in \mathcal{P}$ and any $\epsilon' > 0$, there exists $t_{\epsilon'} \geq 0$ such that $\|f'(\boldsymbol{p}, t) - f'(\boldsymbol{p}, \infty)\| < \epsilon'$ for all $t \geq t_{\epsilon'}$.

Theorem 20 *Assume there exists $t_0' \geq 0$ such that for all $t \geq t_0'$ and any $\boldsymbol{p} \in \mathcal{P}$, we have:*

$$\frac{\beta^{\max}(t)}{\beta^{\min}(t)} \left(\frac{|1-\alpha|}{\alpha}\, \Psi\, \Phi(V'^{\min}, V'^{\max})\right)^2 \left(\frac{\gamma^{\max}}{\gamma^{\min}}\Gamma\right)^{|1-\alpha|} < 1, \tag{6.18}$$

where

$$\beta^{\min}(t) = \min_{i,j\in\mathcal{N}} \beta_j^i(t), \tag{6.19}$$

$$\beta^{\max}(t) = \max_{i,j\in\mathcal{N}} \beta_j^i(t), \tag{6.20}$$

$$\Psi = \max\left\{\frac{1}{P^{\min}(1-P^{\min})}, \frac{1}{P^{\max}(1-P^{\max})}\right\}, \tag{6.21}$$

$$\Gamma = \frac{P^{\max}(1-P^{\min})}{P^{\min}(1-P^{\max})}, \tag{6.22}$$

$$\Phi(V'^{\min}, V'^{\max}) = \begin{cases} \frac{(V'^{\max})^{1/\alpha}}{(1+(V'^{\max})^{1/\alpha})^2}, & \text{if } V'^{\max} \leq 1, \\ \frac{(V'^{\min})^{1/\alpha}}{(1+(V'^{\min})^{1/\alpha})^2}, & \text{if } V'^{\min} \geq 1, \\ 0.25, & \text{otherwise.} \end{cases} \tag{6.23}$$

Then, Algorithm 7 globally and asynchronously converges to the unique fixed point of $f'(\boldsymbol{p}, \infty)$. *Notice that* $V'^{\min}$ *and* $V'^{\max}$ *are the lower and upper bounds on* $v_i'(\boldsymbol{p}, t)$ *for each* $i \in \mathcal{N}$ *and at any time* t. *If* $\alpha \geq 1$, *then* $V'^{\min} = (N-1)M^{\min}(\gamma^{\min})^{\alpha-1}$ *and* $V'^{\max} = (N-1)M^{\max}(\gamma^{\max})^{\alpha-1}$. *If* $\alpha < 1$, *then* $V'^{\min} = (N-1)M^{\min}(\gamma^{\max})^{\alpha-1}$ *and* $V'^{\max} = (N-1)M^{\max}(\gamma^{\min})^{\alpha-1}$.

The proof of Theorem 20 is given in Section 6.6.1. Notice that, at any time $t \geq 0$,

$$(M^{\min}/M^{\max}) \leq \beta^{\min}(t) \leq \beta^{\max}(t) \leq (M^{\max}/M^{\min}). \tag{6.24}$$

We notice that all the terms in (6.18), except Φ, are bounded and independent of the number of users N. Thus, Φ can be arbitrarily close to 0 if N is large enough:

Corollary 4 *For any choice of system parameters, there exists an integer $\hat{N} > 0$, such that Algorithm 7 globally and asynchronously converges to the unique fixed point of mapping $f'(\boldsymbol{p}, \infty)$, if the number of users $N > \hat{N}$, i.e., there are enough users competing for the channel.*

Theorem 20 is general and does not depend on the exact values of the estimation errors as $t \to \infty$; however, the performance at the asymptotic fixed point still depends on the accuracy of the estimations. The following Theorem can be shown for perfect channel case.

Theorem 21 *If the channel is perfect such that $\lim_{t\to\infty} \beta^{\min}(t) = \lim_{t\to\infty} \beta^{\max}(t) = 1$, then the unique fixed point of Algorithm 7 is the unique global optimal solution of problem (6.2).*

The proof of Theorem (21) is similar to that of Theorem 15 and is omitted. Note that since $\lim_{t\to\infty} \beta_j^i(t) = 1$, $f'(\boldsymbol{p}, \infty) = f(\boldsymbol{p}) = (f_i(\boldsymbol{p}),\ \forall i \in \mathcal{N})$ where $f_i(\boldsymbol{p})$ is as in (6.4).

From Theorems 20 and 21, if the channel is perfect and (6.18) holds, Algorithm 7 asynchronously converges to the unique optimal solution of non-convex problem (6.2). If the channel is not perfect, although the algorithm still converges, optimality is not always guaranteed.

6.4 Simulation Results

To evaluate the performance of our proposed distributed algorithm, we develop a discrete-event simulator that implements Algorithms 7 and the IEEE 802.11 DCF access method.

We first consider a network with $N = 4$, $P^{\min} = 0.01$, and $P^{\max} = 0.99$. We set $\gamma_1 = 6$, $\gamma_2 = 18$, $\gamma_3 = 36$, and $\gamma_4 = 54$, all in Mbps. Utility parameter $\alpha = 0.5 < 1$. Notice that none of the previous NUM-based MAC algorithms (e.g, [47, 51, 98]) support α-fair utility functions with $\alpha \in (0, 1)$ because of non-convexity (see [104, Sections II and IV-A]). Each slot is 20 μs (as in

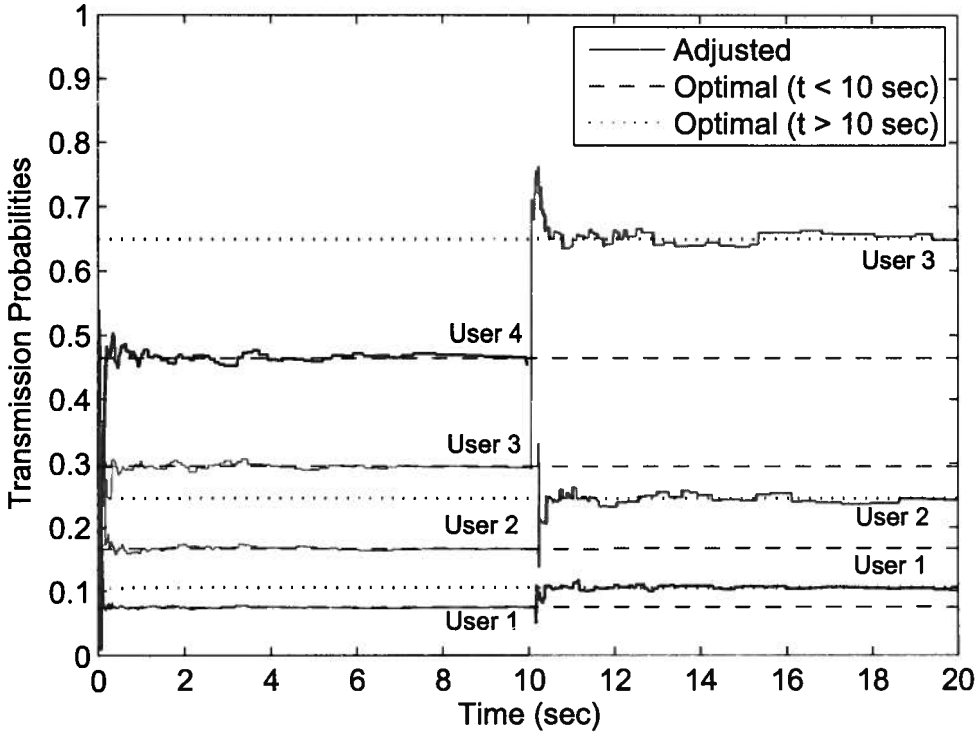


Figure 6.2: Simulation results for Algorithm 7 when $\alpha = 0.6$. The number of users and the features of the communication channel change after $t = 10s$. The optimal probabilities before $t = 10s$ (i.e., dashed lines) and after $t = 10s$ (i.e., dotted lines) are accurate and obtained using Algorithm 5.

802.11a) and the simulation time is $20s$. We assume that from time $t = 0$ to $t = 10s$, the channel is perfect and $N = 4$. Then, from $t = 10s$ to $t = 20s$, the channel is lossy and $N = 3$ (i.e., user 4 leaves the network). Packet error rates are randomly selected between 0 and 0.01 (i.e., the maximum allowed packet error rate in 802.11a) at $t = 10s$ and then become fixed until $t = 20s$. Results are shown in Fig. 6.2. We see that Algorithm 7 converges to a small neighborhood of the optimal values very fast. It is also robust to the change of user population and channel conditions. Similar results have also been obtained for $\alpha \geq 1$.

It is well-known that 802.11 DCF has a *short-term fairness* problem, due to binary exponential

backoff. Next, we compare 802.11 DCF with Algorithm 7 in terms of both system throughput and Jain's fairness index [89]. The short-term fairness is obtained using sliding windows with size of 200 slots. There are $N = 10$ users in the network and their fixed peak rates are randomly selected between 6 and 54 Mbps. Simulation time is $100s$. The results when α varies between 0.5 to 5 are shown in Fig. 6.3. We see that, parameter α acts as a *knob* to control the tradeoff between efficiency and fairness. By increasing α we can make the system more fair but less efficient (and vice versa). If $\alpha = 0.5$, then throughput is 29.7% higher than DCF (see Fig. 6.3(a)). Besides, for any choice of $\alpha \in [0.5, 5]$, the fairness is much better than DCF (Fig. 6.3(b)).

6.5 Summary

In this chapter, we designed a distributed contention-based MAC algorithm to solve a NUM without frequent explicit message passing among users. Our algorithm is fully asynchronous problem, enjoys fast convergence, and supports a wider range of utility functions compared to previously proposed NUM-based MAC algorithms. Simulation results show that our algorithm achieves a better efficiency-fairness trade-off compared with the IEEE 802.11 DCF. It is also robust to the changes of user population and channel conditions.

6.6 Analytical Proofs

6.6.1 Proof of Theorem 20

For any $\boldsymbol{p} \in \mathcal{P}$ and $t \geq t_0'$, the Jacobian $J(\boldsymbol{p}, t)$ is defined as an $N \times N$ matrix whose entry in row i and column j is $\partial f_i(\boldsymbol{p}, t)/\partial p_j$. We can show that,

$$\|J'(\boldsymbol{p},t)\|_\infty \leq (|1-\alpha|/\alpha)\,\Psi\,\Phi, \tag{6.25}$$

$$\|J'(\boldsymbol{p},t)\|_1 \leq (|1-\alpha|/\alpha)\,\left(\beta^{\max}(t)/\beta^{\min}(t)\right)\,\Psi\,\Phi\left((\gamma^{\max}/\gamma^{\min})\,\Gamma\right)^{1-\alpha}. \tag{6.26}$$

Let $\tilde{\boldsymbol{p}}, \hat{\boldsymbol{p}} \in \mathcal{P}$. From (6.18), (6.25), (6.26), and by Cauchy Schwarz inequality, we have:

$$\|f'(\tilde{\boldsymbol{p}},t) - f'(\hat{\boldsymbol{p}},t)\|_2 \leq \|J'(\boldsymbol{p},t)\|_2\,\|\tilde{\boldsymbol{p}}-\hat{\boldsymbol{p}}\|_2 \leq \sqrt{\|J'(\boldsymbol{p},t)\|_\infty \|J'(\boldsymbol{p},t)\|_1}\,\|\tilde{\boldsymbol{p}}-\hat{\boldsymbol{p}}\|_2 < \|\tilde{\boldsymbol{p}}-\hat{\boldsymbol{p}}\|_2,$$

where $\boldsymbol{p}$ is any convex combination of $\tilde{\boldsymbol{p}}$ and $\hat{\boldsymbol{p}}$. Thus, for any $t \geq t_0'$, vector function $f'(\boldsymbol{p},t)$ is a contraction mapping [18, pp. 181] and has a unique fixed point [18, pp. 183], denoted by $\boldsymbol{p}_t^*$. We also denote the unique fixed point of mapping $f'(\boldsymbol{p},\infty)$ by $\boldsymbol{p}_\infty^*$. Thus,

$$\left\|f'(\boldsymbol{p},t) - \boldsymbol{p}_t^*\right\|_2 \leq \eta_t\,\|\boldsymbol{p}-\boldsymbol{p}_t^*\|_2 \leq \eta\xi, \tag{6.27}$$

where $\eta_t = \|J'(\boldsymbol{p},t)\|$, $\eta = \max_{t>t_0'}\eta_t$, and $\xi = \|\boldsymbol{p}-\boldsymbol{p}_t^*\|_2$. Note that $\eta < 1$, and ξ is bounded. Since $f'(\boldsymbol{p},t)$ is continuous at $\boldsymbol{p}_t^*$ and $\lim_{t\to\infty} f'(\boldsymbol{p},t) = f'(\boldsymbol{p},\infty)$, we have $\lim_{t\to\infty}\boldsymbol{p}_t^* = \boldsymbol{p}_\infty^*$. In other words, $\forall\epsilon > 0$, $\exists t_0 \geq t_0'$, such that $\forall t \geq t_0$,

$$\|\boldsymbol{p}_t^* - \boldsymbol{p}_\infty^*\|_2 \leq \epsilon. \tag{6.28}$$

Together with (6.27), we have $\|f'(\boldsymbol{p},t) - \boldsymbol{p}_\infty^*\|_2 \leq \|f'(\boldsymbol{p},t) - \boldsymbol{p}_t^*\|_2 + \|\boldsymbol{p}_t^* - \boldsymbol{p}_\infty^*\|_2 \leq \eta\xi + \epsilon$. Similarly,

$$\begin{aligned}\left\|f'\left(f'(\boldsymbol{p},t),t+1\right) - \boldsymbol{p}_\infty^*\right\|_2 &\leq \left\|f'\left(f'(\boldsymbol{p},t),t+1\right) - \boldsymbol{p}_{t+1}^*\right\|_2 + \left\|\boldsymbol{p}_{t+1}^* - \boldsymbol{p}_\infty^*\right\|_2 \\ &\leq \eta\left(\left\|f'(\boldsymbol{p},t) - \boldsymbol{p}_\infty^*\right\|_2 + \left\|\boldsymbol{p}_{t+1}^* - \boldsymbol{p}_\infty^*\right\|_2\right) + \epsilon \leq \eta(\eta\xi + \epsilon + \epsilon) + \epsilon = \eta(\eta\xi + 2\epsilon) + \epsilon.\end{aligned} \tag{6.29}$$

For any $k \geq 0$, we recursively define $f'^k(\boldsymbol{p},t) = f'(f'^{k-1}(\boldsymbol{p},t), t+k-1)$ where $f'^0 = \boldsymbol{p}$. From (6.29), and by mathematical induction, we can show that for any $k \geq 0$,

$$\left\|f'^k(\boldsymbol{p},t) - \boldsymbol{p}_\infty^*\right\|_2 \leq \eta^k\xi + \frac{2\left(1-\eta^k\right)}{1-\eta}\epsilon - \epsilon < \eta^k\xi + \frac{1+\eta}{1-\eta}\epsilon.$$

For any $\varepsilon > 0$, there exist k_ε such that if $k \geq k_\varepsilon$, then $\eta^k \xi \leq \frac{\varepsilon}{2}$. By choosing $\epsilon = \left(\frac{1-\eta}{1+\eta}\right)\frac{\varepsilon}{2}$,

$$\left\| f'^k(\boldsymbol{p}, t) - \boldsymbol{p}^*_\infty \right\|_\infty \leq \left\| f'^k(\boldsymbol{p}, t) - \boldsymbol{p}^*_\infty \right\|_2 < \tfrac{\varepsilon}{2} + \tfrac{\varepsilon}{2} = \varepsilon, \tag{6.30}$$

For all $t \geq t_0$, define $\epsilon'_t = \max_{k\geq 0} \left\| \boldsymbol{p}^*_{k+t} - \boldsymbol{p}^*_\infty \right\|_\infty$, $\varepsilon'_t = \max_{k\geq 0, \boldsymbol{p}' \in \mathcal{P}} \| f'^{k+t-t_0}(\boldsymbol{p}', t_0) - \boldsymbol{p}^*_\infty \|_\infty$, and

$$\varepsilon_t = \begin{cases} \max\left[\varepsilon'_t, \frac{2(1+\eta)}{1-\eta}\,\epsilon'_t\right], & \text{if } t < t_0 + C, \\ \max\left[\varepsilon'_t, \frac{2(1+\eta)}{1-\eta}\,\epsilon'_t, \chi(C)\,\varepsilon_{t-C}\right], & \text{otherwise,} \end{cases} \tag{6.31}$$

where function $\chi(C) = \frac{1}{2}(\frac{3+\eta}{1+\eta}\eta^C + 1)$, integer constant $C = \lceil \log(\frac{1+\eta}{3+\eta}) / \log(\eta) \rceil + 1$, and $\lceil \cdot \rceil$ denotes the ceiling function. From (6.28) and (6.30), $\{\epsilon'_t\}$ and $\{\varepsilon_t\}$ are infinite decreasing sequences and converge to zero as $t \to \infty$. Construct a new *time sequence* $\{\bar{t}_l\}$ where $\bar{t}_l = t_0 + lC$ for all integer $l \geq 0$. Since $\chi(C) < 1$, sequence $\{\varepsilon_t\}$ is also decreasing and in particular, we have $\lim_{l\to\infty} \varepsilon_{\bar{t}_l} = 0$. For each $l \geq 0$, define $\mathcal{P}_{\bar{t}_l} = \left\{\boldsymbol{p} : \|\boldsymbol{p} - \boldsymbol{p}^*_\infty\|_\infty \leq \varepsilon_{\bar{t}_l}\right\}$. It is clear that $\boldsymbol{p}^*_\infty \in \mathcal{P}_{\bar{t}_l}$ and $\mathcal{P}_{\bar{t}_{l+1}} \subseteq \mathcal{P}_{\bar{t}_l}$ for all $l \geq 0$. Furthermore, $\mathcal{P}_{\bar{t}_{l+l'}} \subset \mathcal{P}_{\bar{t}_l}$ for some finite l'. For any $\boldsymbol{p} \in \mathcal{P}_{\bar{t}_l}$,

$$\left\| \boldsymbol{p} - \boldsymbol{p}^*_{\bar{t}_l} \right\|_\infty \leq \|\boldsymbol{p} - \boldsymbol{p}^*_\infty\|_\infty + \left\| \boldsymbol{p}^*_{\bar{t}_l} - \boldsymbol{p}^*_\infty \right\|_\infty \leq \varepsilon_{\bar{t}_l} + \epsilon'_t.$$

From (6.30), we know that $\left\| f'^k(\boldsymbol{p}, \bar{t}_l) - \boldsymbol{p}^*_\infty \right\|_\infty < \eta^k \left(\varepsilon_{\bar{t}_l} + \epsilon'_{\bar{t}_l}\right) + \frac{1+\eta}{1-\eta}\epsilon'_{\bar{t}_l}$. If $\varepsilon_{\bar{t}_l} = \varepsilon'_{\bar{t}_l}$, then $\epsilon'_{\bar{t}_l} \leq \frac{1-\eta}{1+\eta}\frac{\varepsilon_{\bar{t}_l}}{2}$. On the other hand, if $\varepsilon_{\bar{t}_l} = \frac{2(1+\eta)}{1-\eta}\epsilon'_{\bar{t}_l}$, or if $\varepsilon_{\bar{t}_l} = \chi(C)\varepsilon_{\bar{t}_l - C}$, then $\varepsilon'_{\bar{t}_l} \leq \varepsilon_{\bar{t}_l}$ and $\epsilon'_{\bar{t}_l} \leq \frac{1-\eta}{1+\eta}\frac{\varepsilon_{\bar{t}_l}}{2}$. Thus, for all three possibilities in (6.31), we have

$$\|f'^C(\boldsymbol{p}, \bar{t}_l) - \boldsymbol{p}^*_\infty\|_\infty < \eta^C \left(\varepsilon_{\bar{t}_l} + \frac{1-\eta}{1+\eta}\frac{\varepsilon_{\bar{t}_l}}{2}\right) + \frac{1+\eta}{1-\eta}\frac{1-\eta}{1+\eta}\frac{\varepsilon_{\bar{t}_l}}{2} = \chi(C)\,\varepsilon_{\bar{t}_l} \leq \varepsilon_{\bar{t}_{l+1}}.$$

Thus, $\forall\, \boldsymbol{p} \in \mathcal{P}_{\bar{t}_l}$, $f'^C(\boldsymbol{p}, \bar{t}_l) \in \mathcal{P}_{\bar{t}_{l+1}}$. Since *synchronous convergence* and *box* conditions hold, Algorithm 7 globally and asynchronously converges to $\boldsymbol{p}^*_\infty$ [18, pp. 431]. ■

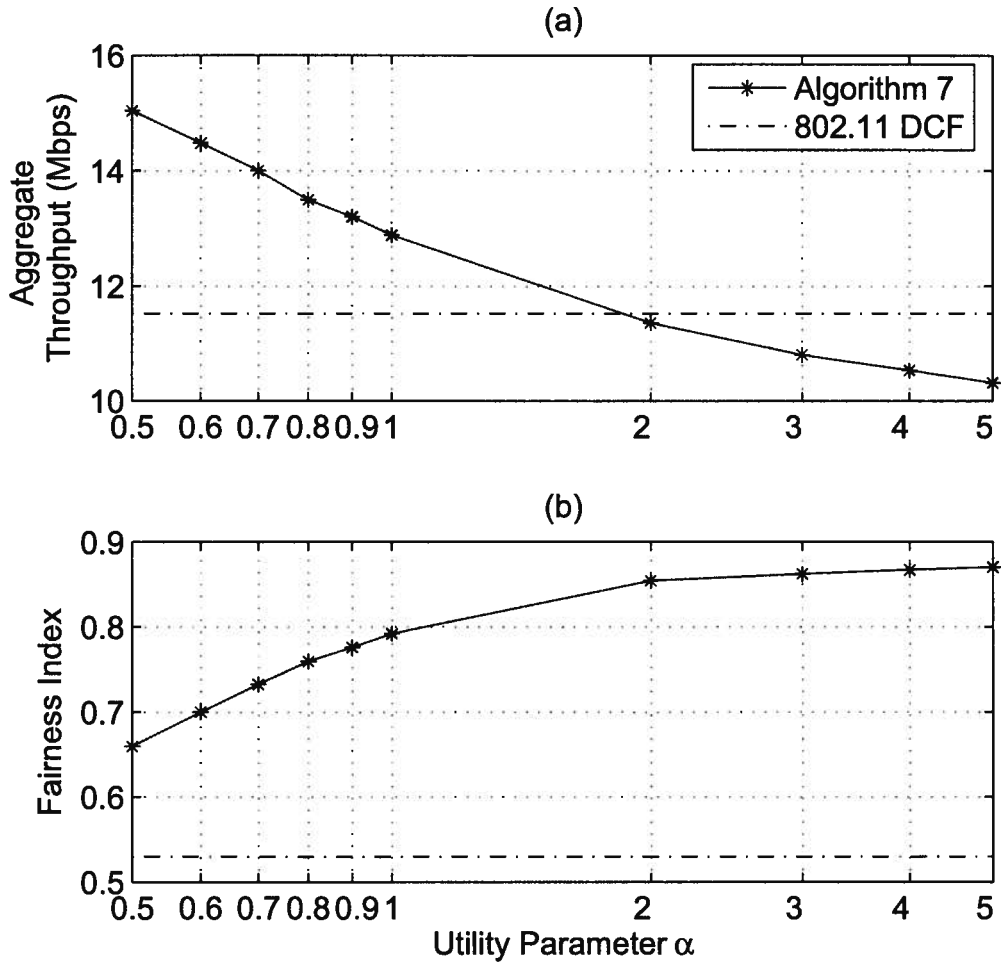


Figure 6.3: Comparison between Algorithm 7 and 802.11 DCF when the number of users $N = 10$. Notice that, neither Algorithm 7 nor 802.11 DCF use any frequent explicit message passing. However, Algorithm 7 results in significantly better throughput-fairness tradeoff.

Chapter 7

Distributed Optimal Multi-Interface Multi-Channel Random Access

Most of the recently proposed channel and interface assignment algorithms for MC-WMNs (e.g., see Chapters 2, 3, and 4) are based on formulating *combinatorial* channel assignment problems and discrete optimization. Each NIC is assumed to be assigned to exactly *one* fixed channel. Examples include *graph coloring problems* [78], *integer optimization problems* [61, 107], and *mixed-integer optimization problems* [70, 108]. It is known that the combinatorial problems are *NP*-hard [34]. That is, finding the optimal solutions may require examining *all* the possible combinations within the search space. Thus, the combinatorial channel assignment algorithms are usually computationally complicated. In addition, they display poor performance gain as the ratio between the number of channels and the number of NICs at each node increases [61, 108]. Finally, the distributed implementation of the combinatorial channel assignment algorithms is difficult because of several design challenges such as the ripple-effect problem [61].

In this chapter, we overcome the performance bottlenecks of the previous combinatorial channel and interface assignment algorithms in all the aforementioned aspects. We first derive the mathematical models for average link data rates in *single-channel reception* and *multi-channel re-*

ception[8] scenarios. We also consider the case when not only the orthogonal (i.e., non-overlapped) channels, but also the *partially-overlapped* frequency channels are being used. The NUM problem is then formulated for each scenario. Our models are extensions of the results on NUM-based single-interface single-channel random access in Chapter 5. Finally, we propose two fast, fully distributed, and easy to implement algorithms, called *distributed multi-interface multi-channel random access* (DMMRA), to solve the formulated NUM problem for each scenario. We prove that DMMRA always outperforms combinatorial channel assignment. We also analytically study the optimality and convergence properties of our proposed algorithms. Simulation results show that DMMRA algorithm with single-channel reception leads to 36% and 23% higher network utility and aggregate throughput compared to the *utility-optimal combinatorial interface assignment and channel allocation* algorithm in [107] (i.e., a modified version of the algorithm in Chaper 3. When multi-channel reception model is implemented, the utility and throughput are further increased by 57% and 71%, respectively. On the other hand, in both single-channel reception and multi-channel reception scenarios, using all available partially overlapped channels result in significant performance improvement compared to the case where only the non-overlapped channels are being used. Our proposed algorithms are also robust to communication delay and delayed information exchange.

The rest of this paper is organized as follows. The data rates for various scenarios are modeled in Section 7.1. The NUM problems are formulated in Section 7.2. The DMMRA algorithms are proposed in Section 7.3. Simulation results are presented in Section 7.4. A summary of the chapter is given in Section 7.5. Analytical proofs of all theorems are given in Section 7.6.

[8]In this paper, *multi-channel reception* refers to a hardware implementation scenario, where each NIC can correctly decode multiple simultaneously received packets if they are transmitted over orthogonal channels. This differs from the *receiver diversity* model in MIMO systems (cf. [109]). Multi-channel reception is studied in Section 7.1.2.

7.1 System Model

Consider a multi-interface multi-channel wireless ad-hoc network with $\mathcal{N} = \{1, \ldots, N\}$ as the set of wireless nodes and $\mathcal{L} = \{1, \ldots, L\}$ as the set of unidirectional wireless links. For each node $n \in \mathcal{N}$, we denote the set of *incoming* links by $\mathcal{L}_n^{\text{in}} \subset \mathcal{L}$, with size $L_n^{\text{in}} = |\mathcal{L}_n^{\text{in}}|$, and the set of outgoing links by $\mathcal{L}_n^{\text{out}} \subset \mathcal{L}$, with size $L_n^{\text{out}} = |\mathcal{L}_n^{\text{out}}|$. We also define $\mathcal{N}_n^{\text{in}} = \{m : (m, n) \in \mathcal{L}_n^{\text{in}}\}$ as the set of *in-neighbors* and $\mathcal{N}_n^{\text{out}} = \{m : (n, m) \in \mathcal{L}_n^{\text{out}}\}$ as the set of *out-neighbors* of node n, respectively. The set of available frequency channels is denoted by $\mathcal{C} = \{1, \ldots, C\}$. The set of NICs for each node $n \in \mathcal{N}$ is denoted by $\mathcal{I}_n$, with size $I_n = |\mathcal{I}_n|$. Each node $n \in \mathcal{N}$ has L_n^{out} separate queues, where each queue stores the packets for one of the outgoing links of node n (see Fig. 7.1). Time is divided into equal-length slots[9]. At each time slot, node n may choose to transmit packets to each of its out-neighbors $m \in \mathcal{N}_n^{\text{out}}$ using its NIC $i \in \mathcal{I}_n$ over channel $c \in \mathcal{C}$ with a *link persistent probability* $p_{nm}^{(i)(c)}$. For the network in Fig. 7.1, node n has $I_n = 2$ NICs and $L_n = 2$ outgoing links, where $\mathcal{I}_n = \{i, j\}$ and $\mathcal{N}_n^{\text{out}} = \{m, s\}$. We also have: $\mathcal{C} = \{1, 2, 3\}$. In node n, those packets which are destined to node m are enqueued in queue $[n, m]$. Similarly, the packets which are destined to node s are enqueued in queue $[n, s]$. At each time slot, a packet from queue $[n, m]$ is sent to node m (i.e., through link (n, m)) using NIC i over channels 1, 2, or 3, with probabilities $p_{nm}^{(i)(1)}$, $p_{nm}^{(i)(2)}$, and $p_{nm}^{(i)(3)}$, respectively. Each NIC may only transmit one packet at a time. Next, we obtain the average data rate model for each wireless link for various multi-interface multi-channel wireless networking scenarios.

[9]The length of the time slots should be large enough, compared to the *channel switching delay* at NICs, i.e., the time it takes for an NIC to switch from one channel to another. In the current commercial NICs, the channel switching delay is between 100 μs to 224 μs [9].

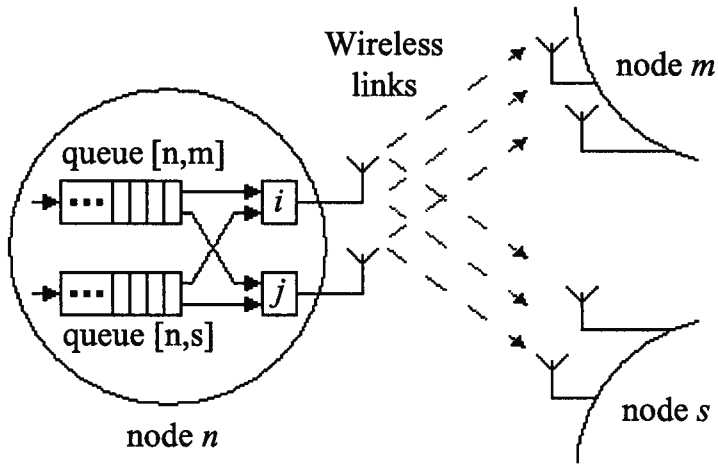


Figure 7.1: An ad-hoc network with $\mathcal{N} = \{n, m, s\}$ as the set of nodes. Each node has 2 NICs and there are $C = 3$ channels available, denoted by 3 colors. Nodes m and s are the out-neighbors of node n. We have: $\mathcal{I}_n = \{i, j\}$.

7.1.1 NICs with Single-Channel Reception

In this section, we consider the case where each NIC can decode the received packets over only *one* channel *at a time*. We assume that all the available channels are orthogonal[10]. For each node $n \in \mathcal{N}$, let $Q_n^{(i)(c)}$ denote the probability that node n *listens* to channel $c \in \mathcal{C}$ using its NIC $i \in \mathcal{I}_n$. To be able to listen to channel c, NIC i on node n needs to be in the *receive mode* (i.e., does not transmit any packet) and also *operates* over channel c. The key feature of the single-channel reception model is that if node n is in the receive mode, and operates over channel $d \neq c$, then it *cannot* decode the signals transmitted over channel c. Most of the existing commercial NICs implement such a single-channel reception model. In this case, for each node $n \in \mathcal{N}$, we have[11]:

$$\sum_{c \in \mathcal{C}} \left(P_n^{(i)(c)} + Q_n^{(i)(c)} \right) = 1, \quad \forall i \in \mathcal{I}_n, \tag{7.1}$$

[10]The case where some of the available channels are partially-overlapped is studied in Section 7.1.3.

[11]Here we assume that each NIC operates either in *transmit* or *receive* mode. If an NIC also operates in *idle* mode, then the equality in (7.1) is replaced with non-strict inequality "$\leq$" and NIC $i \in \mathcal{I}_n$ would be operating in *idle* mode with probability $1 - \sum_{c \in \mathcal{C}} (P_n^{(i)(c)} + Q_n^{(i)(c)})$.

where $P_n^{(i)(c)}$ denotes the probability that node n transmits some data from NIC $i \in \mathcal{I}_n$ over channel $c \in \mathcal{C}$ to *one* of its out-neighbors. We call $P_n^{(i)(c)}$ the *node persistent probability* for NIC i of node n over channel c. We have:

$$P_n^{(i)(c)} = \sum_{m \in \mathcal{N}_n^{\text{out}}} p_{nm}^{(i)(c)}. \tag{7.2}$$

For each link $(n, m) \in \mathcal{L}$, we first consider the case where there is *no interference* in the network (i.e., assuming that there are only two nodes). Let $\tilde{A}_m^{(c)}$ denote the action set for all cases where at least one NIC $j \in \mathcal{I}_m$ transmits packets over frequency channel c. The probability of this happening is:

$$\mathbb{P}\left(\tilde{A}_m^{(c)}\right) = 1 - \prod_{j \in \mathcal{I}_m} \left(1 - P_m^{(j)(c)}\right). \tag{7.3}$$

Let $\hat{A}_m^{(c)}$ denote the action set for all cases where no NIC on node m transmits packets over channel c, and no NIC listens to channel c either. The probability of this happening is:

$$\mathbb{P}\left(\hat{A}_m^{(c)}\right) = \prod_{j \in \mathcal{I}_m} \left(1 - P_m^{(j)(c)} - Q_m^{(j)(c)}\right). \tag{7.4}$$

Since the sets $\tilde{A}_m^{(c)}$ and $\hat{A}_m^{(c)}$ are two *disjoint* sets (i.e., $\tilde{A}_m^{(c)} \cap \hat{A}_m^{(c)}$ is an empty set), we have:

$$\mathbb{P}\left(\tilde{A}_m^{(c)} \cup \hat{A}_m^{(c)}\right) = \mathbb{P}\left(\tilde{A}_m^{(c)}\right) + \mathbb{P}\left(\hat{A}_m^{(c)}\right). \tag{7.5}$$

The transmission from NIC $i \in \mathcal{I}_n$ on sending node $n \in \mathcal{N}$ over channel $c \in \mathcal{C}$ can be received *correctly* by the receiving node $m \in \mathcal{N}_n^{\text{out}}$ only if *at least* one NIC $j \in \mathcal{I}_m$ is *listening* to channel c and *none* of the other NICs on node m are transmitting packets over frequency channel c. From (7.3)-(7.5), this happens with probability:

$$1 - \mathbb{P}\left(\tilde{A}_m^{(c)} \cup \hat{A}_m^{(c)}\right) = \prod_{j \in \mathcal{I}_m} \left(1 - P_m^{(j)(c)}\right) - \prod_{j \in \mathcal{I}_m} \left(1 - P_m^{(j)(c)} - Q_m^{(j)(c)}\right). \tag{7.6}$$

Next, we model the effect of interference in a network with $N > 2$ nodes. For each pair of nodes $s, m \in \mathcal{N}$, we define $\delta_{sm} = 1$ if node s is within the *interference range* of node m, and $\delta_{sm} = 0$

otherwise. Since the interference range is at least as large as the communication range, $\delta_{sm}=1$ if $s \in \mathcal{N}_m^{\text{in}}$. A transmission from NIC i on node n to node m via link $(n,m) \in \mathcal{L}$ over channel c *does not encounter collision* if there is no simultaneous transmission over channel c from any NIC $j \in \mathcal{I}_n \backslash \{i\}$ on node n, and any NIC $k \in \mathcal{I}_s$ on node s with $\delta_{sm}=1$. This happens with probability:

$$\left(\prod_{j\in\mathcal{I}_n\backslash\{i\}}\left(1-P_n^{(j)(c)}\right)\right)\left(\prod_{s\in\mathcal{N}\backslash\{n,m\}}\prod_{k\in\mathcal{I}_s}\left(1-\delta_{sm}P_s^{(k)(c)}\right)\right). \tag{7.7}$$

For each wireless link $(n,m) \in \mathcal{L}$, let r_{nm} denote the average data rate, which is a function of the following persistent and listening probability vectors:

$$\boldsymbol{p} = \left(p_{nm}^{(i)(c)},\ \forall n \in \mathcal{N},\ m \in \mathcal{N}_n^{\text{out}},\ i \in \mathcal{I}_n,\ c \in \mathcal{C}\right), \tag{7.8}$$

$$\boldsymbol{Q} = \left(Q_n^{(i)(c)},\ \forall n \in \mathcal{N},\ i \in \mathcal{I}_n,\ c \in \mathcal{C}\right). \tag{7.9}$$

From (7.2), (7.6) and (7.7), we have [80][12]:

$$\begin{aligned} r_{nm}(\boldsymbol{p},\boldsymbol{Q}) = & \sum_{i\in\mathcal{I}_n}\sum_{c\in\mathcal{C}}\gamma_{nm}^{(c)}p_{nm}^{(i)(c)}\left(\prod_{j\in\mathcal{I}_n\backslash\{i\}}\left(1-P_n^{(j)(c)}\right)\right) \\ & \left(\prod_{s\in\mathcal{N}\backslash\{n,m\}}\prod_{k\in\mathcal{I}_s}\left(1-\delta_{sm}P_s^{(k)(c)}\right)\right) \\ & \left(\prod_{j\in\mathcal{I}_m}\left(1-P_m^{(j)(c)}\right)-\prod_{j\in\mathcal{I}_m}\left(1-P_m^{(j)(c)}-Q_m^{(j)(c)}\right)\right), \end{aligned} \tag{7.10}$$

where $\gamma_{nm}^{(c)}$ denotes the fixed peak data rate[13] for link (n,m) over frequency channel c (i.e., the data rate achieved by link (n,m) over channel c if there is no other transmission in the network at the same time). The data rate model in (7.10) sums up all the average data rates that can be achieved by transmitting packets from each NIC $i \in \mathcal{I}_n$ and over each channel $c \in \mathcal{C}$.

[12]The *node* persistent probability vector $\boldsymbol{P}=(P_n^{(i)(c)},\ \forall n\in\mathcal{N},\ i\in\mathcal{I}_n,\ c\in\mathcal{C})$ can always be constructed from the *link* persistent probability vector $\boldsymbol{p}$ using (7.2). Thus, for each link $(n,m)\in\mathcal{L}$, we can denote the average data rate for the single-channel reception scenario as $r_{nm}(\boldsymbol{p},\boldsymbol{Q})$, rather than $r_{nm}(\boldsymbol{p},\boldsymbol{P},\boldsymbol{Q})$, to avoid redundancy.

[13]In general, $\gamma_{nm}^{(c)} \neq \gamma_{nm}^{(d)}$ for any $c \neq d$ as the channel properties are different at different frequency bands.

7.1.2 NICs with Multi-Channel Reception

Next, consider the case where each NIC can decode *multiple* simultaneously received packets as long as they are transmitted over different orthogonal channels. That is, each NIC does not listen to only one channel while it is in the receive mode. Instead, it listens to *all* frequency channels and applies the band-pass channel filters to *all* the received signals. The output of each filter is then processed separately (i.e., in parallel) to distinguish transmissions over different channels [95]. Figs. 7.2 (a) and (b) show the basic building blocks of the receiver device when the single-channel reception and multi-channel reception models are being implemented, respectively. We notice that, most of the existing commercial NICs do not yet implement multi-channel reception model. However, we will show in Section 7.4 that it can significantly improve the network performance. Thus, it is an attractive and promising candidate for future deployment.

As in Section 7.1.1, we first assume that there is no interference in the network. Let $\breve{A}_m^{(-c)}$ denote the action set where all NICs on node m transmit packets on some channels *other than* channel c, and no NIC is in the receive mode. We have:

$$\mathbb{P}\left(\breve{A}_m^{(-c)}\right) = \prod_{j\in\mathcal{I}_m} \sum_{d\in\mathcal{C}\setminus\{c\}} P_m^{(j)(d)}. \tag{7.11}$$

Since the sets $\tilde{A}_m^{(c)}$ (defined in Section 7.1.1) and $\breve{A}_m^{(-c)}$ are *disjoint* (i.e., $\tilde{A}_m^{(c)} \cap \breve{A}_m^{(-c)}$ is an empty set), we have:

$$\mathbb{P}\left(\tilde{A}_m^{(c)} \cup \breve{A}_m^{(-c)}\right) = \mathbb{P}\left(\tilde{A}_m^{(c)}\right) + \mathbb{P}\left(\breve{A}_m^{(-c)}\right). \tag{7.12}$$

In the multi-channel reception model, for any link $(n,m) \in \mathcal{L}$, the transmission from NIC $i \in \mathcal{I}_n$ on node $n \in \mathcal{N}$ over channel $c \in \mathcal{C}$ can be received correctly by node $m \in \mathcal{N}_n^{\text{out}}$ if *at least* one NIC $j \in \mathcal{I}_m$ is in the *receive mode* and *none* of the other NICs on node m are transmitting

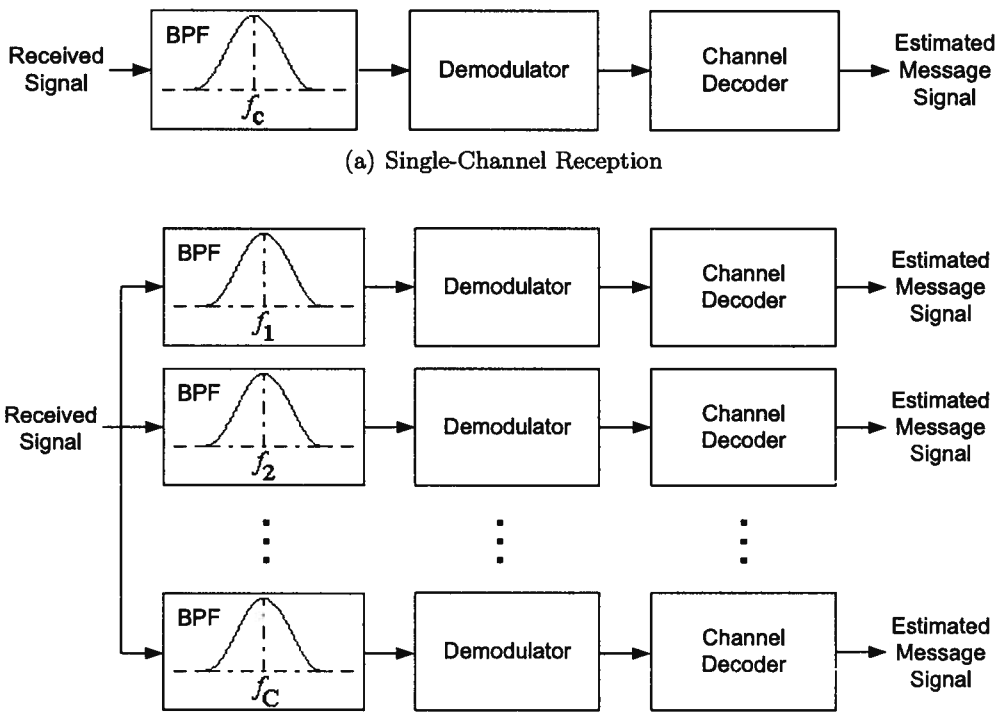


Figure 7.2: Building blocks of the receiver unit in single-channel reception and multi-channel reception models. BPF stands for band-pass filter. For each channel $c \in \mathcal{C}$, the central frequency of the channel filter is shown by f_c.

packets over channel c. From (7.3), (7.11), and (7.12), this happens with probability:

$$1 - \mathbb{P}\left(\tilde{A}_m^{(c)} \cup \breve{A}_m^{(-c)}\right) = \prod_{j \in \mathcal{I}_m} \left(1 - P_m^{(j)(c)}\right) - \prod_{j \in \mathcal{I}_m} \sum_{d \in \mathcal{C} \setminus \{c\}} P_m^{(j)(d)}. \tag{7.13}$$

When the interference is taken into account, from (7.2), (7.7), and (7.13), for each link $(n, m) \in \mathcal{L}$, we have:

$$\begin{aligned} r_{nm}(\boldsymbol{p}) = \; & \sum_{i \in \mathcal{I}_n} \sum_{c \in \mathcal{C}} \gamma_{nm}^{(c)} p_{nm}^{(i)(c)} \left(\prod_{j \in \mathcal{I}_n \setminus \{i\}} \left(1 - P_n^{(j)(c)}\right)\right) \\ & \left(\prod_{s \in \mathcal{N} \setminus \{n,m\}} \prod_{k \in \mathcal{I}_s} \left(1 - \delta_{sm} P_s^{(k)(c)}\right)\right) \\ & \left(\prod_{j \in \mathcal{I}_m} \left(1 - P_m^{(j)(c)}\right) - \prod_{j \in \mathcal{I}_m} \sum_{d \in \mathcal{C} \setminus \{c\}} P_m^{(j)(d)}\right). \end{aligned} \tag{7.14}$$

Note that since a node can listen to all channels when it is in the receive mode, the data rate model in (7.14) does *not* depend on the listening probability vector $\boldsymbol{Q}$.

7.1.3 Partially Overlapped Frequency Channels

In this section, we extend the data rate models in (7.10) and (7.14) to the general cases, where both orthogonal (i.e., non-overlapped) and partially overlapped channels are being used. In this regard, we borrow the concept of *multiple interference ranges* from our recent work in [110, Section II-D]. For each pair of nodes $s, m \in \mathcal{N}$, we define $\delta_{sm}^{(cd)} = 1$ if node s is within the interference range of node m, while node s is operating over channel c and node m is operating over channel d; otherwise, $\delta_{sm}^{(cd)} = 0$. In general, *the smaller the frequency spectrum overlapping between two channels c and d, the shorter the corresponding interference range* (cf. [110, Fig. 3]). In fact, as two wireless links use lower overlapped channels, the less is the interference power that they cause on each other's transmissions. Thus, the interfering transmissions need to be in shorter distance to corrupt each other's packets.

We first assume that NICs use single-channel reception model and there is no interference in the network. For each link $(n, m) \in \mathcal{L}$, let $\tilde{B}_m^{(c)}$ denote the action set for all cases where at least one NIC $j \in \mathcal{I}_m$ transmits packets on some channel $d \in \mathcal{C}$ such that $\delta_{mm}^{(cd)} = 1$. Also, let $\hat{B}_m^{(c)}$ denote the action set for all cases where no NIC on node m transmits packets on any channel $d \in \mathcal{C}$ such that $\delta_{mm}^{(cd)} = 1$, and no NIC listens to channel c either. We have:

$$\mathbb{P}\left(\tilde{B}_m^{(c)}\right) = 1 - \prod_{j \in \mathcal{I}_m} \left(1 - \sum_{d \in \mathcal{C}} \delta_{mm}^{(cd)} P_m^{(j)(d)}\right), \tag{7.15}$$

$$\mathbb{P}\left(\hat{B}_m^{(c)}\right) = \prod_{j \in \mathcal{I}_m} \left(1 - \sum_{d \in \mathcal{C}} \delta_{mm}^{(cd)} P_m^{(j)(d)} - Q_m^{(j)(c)}\right). \tag{7.16}$$

Since the sets $\tilde{B}_m^{(c)}$ and $\hat{B}_m^{(c)}$ are disjoint sets, we have:

$$1 - \mathbb{P}\left(\tilde{B}_m^{(c)} \cup \hat{B}_m^{(c)}\right) = \prod_{j \in \mathcal{I}_m} \left(1 - \sum_{d \in \mathcal{C}} \delta_{mm}^{(cd)} P_m^{(j)(d)}\right) - \prod_{j \in \mathcal{I}_m} \left(1 - \sum_{d \in \mathcal{C}} \delta_{mm}^{(cd)} P_m^{(j)(d)} - Q_m^{(j)(c)}\right). \tag{7.17}$$

To model the interference, we can modify the collision avoidance probability model in (7.7)

as:

$$\left(\prod_{j\in\mathcal{I}_n\setminus\{i\}}\left(1-\sum_{d\in\mathcal{C}}\delta_{nm}^{(cd)}P_n^{(j)(d)}\right)\right)\left(\prod_{s\in\mathcal{N}\setminus\{n,m\}}\prod_{k\in\mathcal{I}_s}\left(1-\sum_{d\in\mathcal{C}}\delta_{sm}^{(cd)}P_s^{(k)(d)}\right)\right). \tag{7.18}$$

From (7.17) and (7.18), when the NICs implement single-channel reception and all partially overlapped channels are available, the average data rate of link $(n,m)\in\mathcal{L}$ is:

$$\begin{aligned} r_{nm}(\boldsymbol{p},\boldsymbol{Q}) = {} & \sum_{i\in\mathcal{I}_n}\sum_{c\in\mathcal{C}}\gamma_{nm}^{(c)}p_{nm}^{(i)(c)}\left(\prod_{j\in\mathcal{I}_n\setminus\{i\}}\left(1-\sum_{d\in\mathcal{C}}\delta_{nm}^{(cd)}P_n^{(j)(d)}\right)\right) \\ & \left(\prod_{s\in\mathcal{N}\setminus\{n,m\}}\prod_{k\in\mathcal{I}_s}\left(1-\sum_{d\in\mathcal{C}}\delta_{sm}^{(cd)}P_s^{(k)(d)}\right)\right) \\ & \left(\prod_{j\in\mathcal{I}_m}\left(1-\sum_{d\in\mathcal{C}}\delta_{mm}^{(cd)}P_m^{(j)(d)}\right)-\prod_{j\in\mathcal{I}_m}\left(1-\sum_{d\in\mathcal{C}}\delta_{mm}^{(cd)}P_m^{(j)(d)}-Q_m^{(j)(c)}\right)\right). \end{aligned} \tag{7.19}$$

We now consider the multi-channel reception scenario. For any link $(n,m)\in\mathcal{L}$, let $\breve{B}_m^{(-c)}$ denote the set of actions for all cases where no NIC on node m transmits packets over channel c or any other channel $d\in\mathcal{C}\setminus\{c\}$ such that $\delta_{mm}^{(cd)}=1$, and no NIC is silent either. In other words, all NICs on node m transmit packets on some channels *other than* those channels that have (full or partial) overlapping with channel c. We have:

$$\mathbb{P}\left(\breve{B}_m^{(-c)}\right)=\prod_{j\in\mathcal{I}_m}\sum_{d\in\mathcal{C}}\left(1-\delta_{mm}^{(cd)}\right)P_m^{(j)(d)}. \tag{7.20}$$

Notice that for any node $m\in\mathcal{N}$, if frequency channels $c,d\in\mathcal{C}$ are either fully or partially overlapped, then $1-\delta_{mm}^{(cd)}=0$. From (7.15), (7.18), and (7.20), when the NICs use multi-channel reception and all non-overlapped as well as partially overlapped channels are being available, the average data rate of link $(n,m)\in\mathcal{L}$ is:

$$\begin{aligned} r_{nm}(\boldsymbol{p}) = {} & \sum_{i\in\mathcal{I}_n}\sum_{c\in\mathcal{C}}\gamma_{nm}^{(c)}p_{nm}^{(i)(c)}\left(\prod_{j\in\mathcal{I}_n\setminus\{i\}}\left(1-\sum_{d\in\mathcal{C}}\delta_{nm}^{(cd)}P_n^{(j)(d)}\right)\right) \\ & \left(\prod_{s\in\mathcal{N}\setminus\{n,m\}}\prod_{k\in\mathcal{I}_s}\left(1-\sum_{d\in\mathcal{C}}\delta_{sm}^{(cd)}P_s^{(k)(d)}\right)\right) \\ & \left(\prod_{j\in\mathcal{I}_m}\left(1-\sum_{d\in\mathcal{C}}\delta_{mm}^{(cd)}P_m^{(j)(d)}\right)-\prod_{j\in\mathcal{I}_m}\sum_{d\in\mathcal{C}}\left(1-\delta_{mm}^{(cd)}\right)P_m^{(j)(d)}\right). \end{aligned} \tag{7.21}$$

It can be verified that, if all the channels are orthogonal (i.e., $\delta_{sm}^{cd}=0$ for all $s,m\in\mathcal{N}$ and

any $c \neq d$), then the rates in (7.19) and (7.21) reduce to (7.10) and (7.14), respectively.

7.2 Network Utility Maximization

The formulation in (7.1)-(7.21) models the average link data rates for different multi-interface multi-channel random access scenarios. In this section, we formulate the random access problems in those scenarios according to the NUM framework (cf. [44]).

7.2.1 Problem Formulation

Within the NUM framework, the resource allocation problem can be formulated either at *link layer* [98, 104] or at *transport layer* [44]. Here, for the ease of exposition, we limit our study to the link-layer NUM[14]. In this regard, each link $(n, m) \in \mathcal{L}$ is assumed to maintain a *utility* $u(r_{nm})$, which is an increasing and concave function of its rate r_{nm} and indicates the degree of satisfaction of link (n, m) on its data rate. The utility of link (n, m) is also a function of all the persistent and listening probabilities $\boldsymbol{p}$ and $\boldsymbol{Q}$. Assuming that the *single*-channel reception model is used, we are interested in finding the optimal solution of the following NUM problem:

$$\underset{\langle \boldsymbol{p}, \boldsymbol{Q} \rangle \in \Phi}{\text{maximize}} \quad \sum_{n \in \mathcal{N}} \sum_{m \in \mathcal{N}_n^{\text{out}}} u(r_{nm}(\boldsymbol{p}, \boldsymbol{Q})), \tag{NUM-S}$$

where the data rates are as in (7.10) if only the orthogonal channels are used, and as in (7.19) if both orthogonal and partially overlapped channels are used. We also have:

$$\begin{aligned}\Phi = \{ \langle \boldsymbol{p}, \boldsymbol{Q} \rangle : & \; p_{nm}^{(i)(c)}, P_n^{(i)(c)}, Q_n^{(i)(c)} \in [0, 1], \\ & P_n^{(i)(c)} = \textstyle\sum_{m \in \mathcal{N}_n^{\text{out}}} p_{nm}^{(i)(c)}, \; \sum_{d \in \mathcal{C}} \left(P_n^{(i)(d)} + Q_n^{(i)(d)} \right) = 1, \\ & \forall n \in \mathcal{N}, \; m \in \mathcal{N}_n^{\text{out}}, \; i \in \mathcal{I}_n, \; c \in \mathcal{C} \}.\end{aligned}$$

[14]We can extend the model to a transport-layer NUM similar to the joint congestion control and medium access control design in [111].

On the other hand, if *multi*-channel reception is used, then the following NUM problem is being solved:

$$\underset{\boldsymbol{p}\in\Psi}{\text{maximize}} \quad \sum_{n\in\mathcal{N}}\sum_{m\in\mathcal{N}_n^{\text{out}}} u(r_{nm}(\boldsymbol{p})), \tag{NUM-M}$$

where the data rates are as in (7.14) if only the orthogonal channels are used, and as in (7.21) if both orthogonal and partially overlapped channels are used. We have:

$$\Psi = \{\, \boldsymbol{p} : p_{nm}^{(i)(c)}, P_n^{(i)(c)} \in [0,1],\ P_n^{(i)(c)} = \sum_{m\in\mathcal{N}_n^{\text{out}}} p_{nm}^{(i)(c)},$$

$$\sum_{d\in\mathcal{C}} P_n^{(i)(d)} \le 1,\ \forall n\in\mathcal{N},\ m\in\mathcal{N}_n^{\text{out}},\ i\in\mathcal{I}_n,\ c\in\mathcal{C}\,\}.$$

Notice that the sets Φ and Ψ are formed by linear constraints. Thus, Φ and Ψ are *convex* sets [16]. Various concave utility functions can also be considered to achieve different network design objectives. A popular class of utility functions are α-fair utilities [50], where for each link $(n,m)\in\mathcal{L}$, we have:

$$u(r_{nm}) = \begin{cases} (1-\alpha)^{-1} r_{nm}^{1-\alpha}, & \text{if } \alpha\in(0,1)\cup(1,\infty), \\ \log r_{nm}, & \text{if } \alpha = 1. \end{cases} \tag{7.22}$$

Using (7.22), a wide range of efficient and fair resource allocations among the link-layer flows can be modeled. In particular, optimization problems (NUM-S) and (NUM-M) reduce to throughput maximization with $\alpha\to 0$, to proportional fair allocation with $\alpha=1$, to harmonic mean fairness with $\alpha=2$, and to max-min fairness with $\alpha\to\infty$.

Unlike most of the previously proposed optimization-based channel assignment models, where the formulated optimization problems are combinatorial and *discrete-valued* (cf. [61, 72, 107, 108, 112, 113]), problems (NUM-S) and (NUM-M) are *continuous-valued* optimization problems. They make the analysis of our models substantially easier. Notice that, within the NUM framework at the link-layer, the combinatorial channel assignment problem can be formulated as the following

mixed-integer optimization problem [107]:

$$\begin{aligned}
&\underset{\langle \boldsymbol{p},\boldsymbol{Q}\rangle\in\Phi,\ \boldsymbol{x}\in\Upsilon}{\text{maximize}} \ \textstyle\sum_{n\in\mathcal{N}}\sum_{m\in\mathcal{N}_n^{\text{out}}} u(r_{nm}(\boldsymbol{p},\boldsymbol{Q})), \\
&\text{subject to } \textstyle\sum_{c\in\mathcal{C}} x_n^{(i)(c)}=1, && \forall n\in\mathcal{N},\, i\in\mathcal{I}_n, \\
&\qquad P_n^{(i)(c)} \le x_n^{(i)(c)}, && \forall n\in\mathcal{N},\, i\in\mathcal{I}_n,\, c\in\mathcal{C}, \\
&\qquad Q_n^{(i)(c)} \le x_n^{(i)(c)}, && \forall n\in\mathcal{N},\, i\in\mathcal{I}_n,\, c\in\mathcal{C},
\end{aligned} \tag{NUM-C}$$

where for each node $n \in \mathcal{N}$, any NIC $i \in \mathcal{I}_n$, and each channel $c \in \mathcal{C}$, the *integer* variable $x_n^{(i)(c)}$ is defined as:

$$x_n^{(i)(c)} = \begin{cases} 1, & \text{if NIC } i \text{ operates over channel } c, \\ 0, & \text{otherwise.} \end{cases} \tag{7.23}$$

We also have $\boldsymbol{x} = (x_n^{(i)(c)},\ \forall n\in\mathcal{N},\ i\in\mathcal{I}_n\ c\in\mathcal{C})$ and

$$\Upsilon = \left\{\boldsymbol{x} : x_n^{(i)(c)} \in \{0,1\},\ \forall n\in\mathcal{N},\ i\in\mathcal{I}_n\ c\in\mathcal{C}\right\}.$$

From the first constraint in (NUM-C), each NIC can be assigned to exactly one channel. In the context of combinatorial channel assignment, the selection of the *operating frequency channel* for each NIC is called *interface-to-channel binding* [61]. From the second and the third constraints, NIC i cannot transmit over or listen to channel $c \in \mathcal{C}$ if it is *not* operating on channel c. By solving the mixed-integer optimization problem (NUM-C), we can select not only the operating channel for each NIC, but also the persistent and listening probabilities corresponding to the operating channel of each NIC to achieve utility-optimal network performance within the combinatorial channel and interface assignment framework.

Theorem 22 *Let $U^\star_{SCR}$, $U^\star_{MCR}$, and $U^\star_{Comb}$ denote the optimal solution of problems (NUM-S), (NUM-M), and (NUM-C), respectively. We can show the following:*

(a) Random access with multi-channel reception outperforms random access with single-channel reception:

$$U^{\star}_{SCR} \leq U^{\star}_{MCR}. \tag{7.24}$$

(b) Random access with single-channel reception outperforms NUM-based combinatorial channel assignment:

$$U^{\star}_{Comb} \leq U^{\star}_{SCR}. \tag{7.25}$$

The proof of Theorem 22 is given in Section 7.7. For better understanding of the inequalities in (7.24) and (7.25), we consider two examples and compare the optimal utilities $U^{\star}_{Comb}$, $U^{\star}_{SCR}$, and $U^{\star}_{MCR}$.

7.2.2 Examples

First, consider a *unidirectional ring topology* with $N=3$ nodes, $L=3$ links, and $C=3$ channels. The utilities are logarithmic (i.e., $\alpha = 1$). Each node has one NIC. We have: $\mathcal{N} = \{n, m, s\}$, $\mathcal{L} = \{(n,m),(m,s),(s,n)\}$, and $\mathcal{C} = \{1,2,3\}$. For any $c \in \mathcal{C}$, $\gamma^{(c)}_{nm} = \gamma^{(c)}_{ns} = \gamma^{(c)}_{sn} = 11$ Mbps. In this scenario, combinatorial channel assignment strategies can only assign the *same* channel to all NICs in the network. Otherwise, at least two nodes *cannot* communicate with each other[15]. Thus, we have: $U^{\star}_{Comb} = 3\log(11 \times \frac{1}{3} \times (1-\frac{1}{3}) \times (1-\frac{1}{3})) = 1.465$, where each link is optimally active with probability $\frac{1}{3}$. On the other hand, $U^{\star}_{SCR} = 3\log(11 \times \frac{1}{2} \times (1-0) \times (1-\frac{1}{2})) = 3.035$, where $P^{(1)(1)}_{n} = Q^{(1)(3)}_{n} = \frac{1}{2}$, $P^{(1)(2)}_{n} = P^{(1)(3)}_{n} = Q^{(1)(1)}_{n} = Q^{(1)(2)}_{n} = 0$, $P^{(1)(2)}_{m} = Q^{(1)(1)}_{m} = \frac{1}{2}$, $P^{(1)(1)}_{m} = P^{(1)(3)}_{m} = Q^{(1)(2)}_{m} = Q^{(1)(3)}_{m} = 0$, $P^{(1)(3)}_{s} = Q^{(1)(2)}_{s} = \frac{1}{2}$, and $P^{(1)(1)}_{s} = P^{(1)(2)}_{s} = Q^{(1)(1)}_{s} =$

[15]This is one of the key limitations of the combinatorial channel assignment models. In fact, as shown in [61, 108], combinatorial channel assignment algorithms display poor performance gain as the ratio between the number of channels and the number of NICs at each node increases. In this example, there are multiple channels available, while each node has only one NIC.

$Q_s^{(1)(3)} = 0$. That is, on average, each wireless node transmits to its out-neighbor on one channel during half of the time slots and listens to its in-neighbor on a *different* channel during the other half of the time slots. As a result, each of the 3 wireless links in the network is active on a *distinct* channel and the available frequency spectrum is fully utilized. Performance improvement is in the factor of $\frac{3.035}{1.465} = 2.1$. In this example, we have:

$$U^{\star}_{Comb} < U^{\star}_{SCR} = U^{\star}_{MCR}. \tag{7.26}$$

Next, consider a *bidirectional ring topology*, where $\mathcal{N} = \{n,\ m, s\}$ and $\mathcal{L} = \{(n,m),(m,n)$ $,(m,s),(s,m),\ (s,n),(n,s)\}$. The rest of the parameters are the same as the previous example. Again, any combinatorial channel assignment assigns the same channel to all NICs. We have: $U^{\star}_{Comb} = 6\log(11 \times \frac{1}{6} \times (1 - \frac{1}{3}) \times (1 - \frac{1}{3})) = -1.229$, where each link is optimally active with probability $\frac{1}{6}$. Each node is also optimally active with probability $2\times\frac{1}{6}=\frac{1}{3}$ as it has two outgoing links. On the other hand, $U^{\star}_{SCR} = 6\log(11 \times 0.2113 \times (1 - 0.2113) \times 0.5774) = 0.341$, where each node listens to one distinct channel with probability 0.5774. Each node also transmits to its out-neighbors using two different channels, other than the channel that it listens to. Finally, $U^{\star}_{MCR} = 6\log(11 \times \frac{1}{4} \times (1-0) \times (1-\frac{1}{2})) = 1.9107$, where each node transmits to both of its out-neighbors using one of the three channels with probability $\frac{1}{4}$. Each NIC is silent with probability $\frac{1}{2}$. In this example, we have:

$$U^{\star}_{Comb} < U^{\star}_{SCR} < U^{\star}_{MCR}. \tag{7.27}$$

Theorem 22 and the above examples show that our proposed random access models can outperform combinatorial channel and interface assignment. We will investigate this issue further in Section 7.4.2.

7.3 DMMRA Algorithms

Although the objective functions in problems (NUM-S) and (NUM-M) are concave in link rates $\boldsymbol{r} = (r_{nm}, \forall (n,m) \in \mathcal{L})$, they are *not* concave in persistent and listening probabilities $\boldsymbol{p}$ and $\boldsymbol{Q}$ due to the *product forms* in (7.10), (7.14), (7.19), and (7.21). Thus, finding the optimal solutions of these *non-convex* optimization problems are not easy in general. In this section, we discuss some of the features of problems (NUM-S) and (NUM-M) which will help us to develop our distributed multi-interface multi-channel random access (DMMRA) algorithms.

7.3.1 Local NUM Problems

For each node $n \in \mathcal{N}$ and any NIC $i \in \mathcal{I}_n$, we define:

$$\boldsymbol{p}_n^{(i)} = \left(p_{nm}^{(i)(c)}, \forall m \in \mathcal{N}_n^{\text{out}}, c \in \mathcal{C}\right), \tag{7.28}$$

$$\boldsymbol{Q}_n^{(i)} = \left(Q_n^{(i)(c)}, \forall c \in \mathcal{C}\right), \tag{7.29}$$

to be the persistent and listening probabilities of NIC i, respectively. Consider the following *local* and *myopic* optimization problem in NIC $i \in \mathcal{I}_n$, when the single-channel reception model is being used:

$$\underset{\langle \boldsymbol{p}_n^{(i)}, \boldsymbol{Q}_n^{(i)} \rangle \in \Phi_n^{(i)}}{\text{maximize}} \quad \sum_{m \in \mathcal{N}} \sum_{s \in \mathcal{N}_m^{\text{out}}} u(r_{ms}(\boldsymbol{p}, \boldsymbol{Q})). \tag{LOCAL-NUM-S}$$

Here, the average data rates are as in (7.10) and we have:

$$\begin{aligned} \Phi_n^{(i)} = \{ & \langle \boldsymbol{p}_n^{(i)}, \boldsymbol{Q}_n^{(i)} \rangle : p_{nm}^{(i)(c)}, P_n^{(i)(c)}, Q_n^{(i)(c)} \in [0,1], \\ & P_n^{(i)(c)} = \textstyle\sum_{m \in \mathcal{N}_n^{\text{out}}} p_{nm}^{(i)(c)}, \\ & \textstyle\sum_{d \in \mathcal{C}} \left(P_n^{(i)(d)} + Q_n^{(i)(d)}\right) = 1, \ \forall m \in \mathcal{N}_n^{\text{out}}, \ c \in \mathcal{C} \}. \end{aligned} \tag{7.30}$$

Similarly, consider the following local and myopic problem when the multi-channel reception model is used:

$$\underset{\boldsymbol{p}_n^{(i)} \in \Psi_n^{(i)}}{\text{maximize}} \ \sum_{m \in \mathcal{N}} \sum_{s \in \mathcal{N}_m^{\text{out}}} u(r_{ms}(\boldsymbol{p})). \tag{LOCAL-NUM-M}$$

Here, the average data rates are as in (7.14) and we have:

$$\begin{aligned} \Psi_n^{(i)} = \{ \boldsymbol{p}^{(i)} : p_{nm}^{(i)(c)}, P_n^{(i)(c)} \in [0,1], \ \textstyle\sum_{d \in \mathcal{C}} P_n^{(i)(d)} \leq 1, \\ P_n^{(i)(c)} = \textstyle\sum_{m \in \mathcal{N}_n^{\text{out}}} p_{nm}^{(i)(c)}, \ \forall m \in \mathcal{N}_n^{\text{out}}, \ c \in \mathcal{C} \}. \end{aligned} \tag{7.31}$$

The objective functions in (LOCAL-NUM-S) and (LOCAL-NUM-M) are the same as the objective functions in (NUM-S) and (NUM-M), respectively. However, the optimization variables in (LOCAL-NUM-S) and (LOCAL-NUM-M) are local to NIC i in node n.

Consider the case where the single-channel reception model is being used. We define:

$$\boldsymbol{p}_n^{(-i)} = \left(\boldsymbol{p}_n^{(j)}, \forall j \in \mathcal{I}_n \backslash \{i\}, \ \boldsymbol{p}_m^{(k)}, \forall k \in \mathcal{I}_m, m \in \mathcal{N} \backslash \{n\} \right), \tag{7.32}$$

$$\boldsymbol{Q}_n^{(-i)} = \left(\boldsymbol{Q}_n^{(j)}, \forall j \in \mathcal{I}_n \backslash \{i\}, \ \boldsymbol{Q}_m^{(k)}, \forall k \in \mathcal{I}_m, m \in \mathcal{N} \backslash \{n\} \right). \tag{7.33}$$

The above are the persistent and listening probabilities corresponding to all NICs in the network *other than* NIC i in node n. By solving problem (LOCAL-NUM-S), we can select $\boldsymbol{p}_n^{(i)}$ and $\boldsymbol{Q}_n^{(i)}$ such that the total utility is maximized *assuming* that $\boldsymbol{p}_n^{(-i)}$ and $\boldsymbol{Q}_n^{(-i)}$are *fixed* (i.e., the persistent and the listening probabilities of the other NICs in the network do not change). Solving problem (LOCAL-NUM-M) leads to similar results if multi-channel reception model is being used. We can show the following key theorem:

Theorem 23 *Problems (LOCAL-NUM-S) and (LOCAL-NUM-M) are* convex *optimization problems.*

The proof of Theorem 23 is given in Section 7.8. From Theorem 23, we can use various *convex*

programming techniques (cf. [16]) to solve problems (LOCAL-NUM-S) and (LOCAL-NUM-M). The optimal solutions of problems (LOCAL-NUM-S) and (LOCAL-NUM-M) can also be obtained in *closed-form* for some *simple* scenarios:

Theorem 24 *For a two-node single-interface multi-channel network (i.e., when $N = 2$, $C > 1$, and $I_n = 1$ for all $n \in \mathcal{N}$) with α-fair utilities, the optimal solution of problem (LOCAL-NUM-M) can be obtained as:*

$$\boldsymbol{p}_n^{(i)^*} = \boldsymbol{M}\left(\boldsymbol{p}_n^{(-i)}\right)^{-1} \mathbf{1}, \qquad \forall n \in \mathcal{N},\ i \in \mathcal{I}_n, \tag{7.34}$$

where $\mathbf{1}$ is a unit $C \times 1$ vector and $\boldsymbol{M}(\boldsymbol{p}_n^{(-i)})$ is a $C \times C$ matrix with its entry in the cth row and d^{th} column is:

$$\boldsymbol{M}^{(cd)}\left(\boldsymbol{p}_n^{(-i)}\right) = 1 + \frac{\gamma_{nm}^{(d)}/\gamma_{nm}^{(c)}}{\sqrt[\alpha]{\left(\frac{1-\sum_{e\in C} p_{mn}^{(j)(e)}}{\sum_{e\in C} \gamma_{mn}^{(e)} p_{mn}^{(j)(e)}} \gamma_{nm}^{(c)}\right)^{\alpha-1}}}. \tag{7.35}$$

Here j denotes the (only) NIC of node m.

The proof of Theorem 24 is given in Section 7.9. Although the solution in (7.34) always exists, it may not always be unique. In general, when the network has a large number of nodes, for any arbitrary topology, we can use the *interior-point method* (IPM) [16, Chapter 11] to solve the convex problems in (LOCAL-NUM-S) and (LOCAL-NUM-M) via *local* iterations. IPM can find the optimal solution of convex optimization problems in *polynomial time* [16].

Finally, we notice that for each node $n \in \mathcal{N}$ and any of its NICs $i \in \mathcal{I}_n$, we can rewrite the objective function of problem (LOCAL-NUM-S) to be as follows:

$$\sum_{s\in\mathcal{N},\delta_{ns}=1} \sum_{m\in\mathcal{N}_s^{\text{in}}} u(r_{ms}(\boldsymbol{p},\boldsymbol{Q})) + \Gamma(\boldsymbol{p}_n^{(-i)}, \boldsymbol{Q}_n^{(-i)}), \tag{7.36}$$

where Γ only depends on $\boldsymbol{p}_n^{(-i)}$ and $\boldsymbol{Q}_n^{(-i)}$, but not $\boldsymbol{p}_n^{(i)}$ and $\boldsymbol{Q}^{(i)}$. Recall that $\delta_{ns} = 1$ if

node n is within the interference range of node s. By writing the objective function of problem (LOCAL-NUM-S) in the form of (7.36), we divide it into two parts. The first part (i.e., $\sum_{s\in\mathcal{N},\delta_{sn}=1}\sum_{m\in\mathcal{N}_s^{\text{in}}} u(r_{ms}(\boldsymbol{p},\boldsymbol{Q})))$ depends on the optimization variables $\boldsymbol{p}_n^{(i)}$ and $\boldsymbol{Q}_n^{(i)}$. The second part (i.e., $\Gamma(\boldsymbol{p}_n^{(-i)},\boldsymbol{Q}_n^{(-i)}))$ is a constant. Thus, the value of Γ needs *not* be known to solve problem (LOCAL-NUM-S). On the other hand, for any link (m,s), where $s\in\mathcal{N}$, $\delta_{ns}=1$, and $m\in\mathcal{N}_s^{\text{in}}$, the data rate r_{ms} is fully modeled if the persistent and listening probabilities of nodes within $2R^{\max}$ distance of node n are known. Here, $R^{\max}=\max_{m\in\mathcal{N}}R_m$ and for each node $m\in\mathcal{N}$, R_m is the interference range of node m. Thus, the convex optimization problem (LOCAL-NUM-S) can easily be solved *locally*, as long as all nodes within $2R^{\max}$ distance of node n can inform their persistent and listening probabilities to node n. The same statement is true for the convex optimization problem (LOCAL-NUM-M).

7.3.2 Algorithms

Our proposed DMMRA algorithms, when the *single*-channel reception and *multi*-channel reception models are being used, are shown in Algorithm 8 (DMMRA-S) and Algorithm 9 (DMMRA-M), respectively. For each node $n\in\mathcal{N}$ and any of its NICs $i\in\mathcal{I}_n$, let $\mathcal{T}_n^{(i)}$ be an unbounded set of time slots at which node n updates NIC i's persistent and listening probabilities. We assume that the updates are *asynchronous* across the network. That is, $\mathcal{T}_n^{(i)}\cap\mathcal{T}_n^{(j)}=\{\}$ for all $j\in\mathcal{I}_n\backslash\{i\}$ and $\mathcal{T}_n^{(i)}\cap\mathcal{T}_m^{(k)}=\{\}$ for all $m\in\mathcal{N}\backslash\{n\}$ and any $k\in\mathcal{I}_m$. In line 2 of Algorithm 8, node n randomly initiates all of its persistent and listening probabilities. Lines 4 to 14 are then executed repeatedly at every time slot until node n leaves the network or switches off. In lines 4 to 6, node n either transmits or receives packets according to its persistent and listening probabilities. On the other hand, lines 8 to 10 are executed only if there exists an NIC $i\in\mathcal{I}_n$ such

Algorithm 8 - DMMRA-S: Executed by each node $n \in \mathcal{N}$ when the NICs implement *single*-channel reception.

1: Allocate memory for $\boldsymbol{p}$ and $\boldsymbol{Q}$.
2: Randomly choose $\boldsymbol{p}$ and $\boldsymbol{Q}$ such that $\langle \boldsymbol{p}, \boldsymbol{Q} \rangle \in \Phi$.
3: **repeat**
4: **for** each NIC $i \in \mathcal{I}_n$ **do**
5: Either transmit to node $m \in \mathcal{N}_n^{\text{out}}$ on channel $c \in \mathcal{C}$ or listen to channel $c \in \mathcal{C}$ with probabilities $p_{nm}^{(i)(c)}$ and $Q_n^{(i)(c)}$, respectively.
6: **end for**
7: **if** $t \in \mathcal{T}_n^{(i)}$ for some $i \in \mathcal{I}_n$ **then**
8: Solve problem (LOCAL-NUM-S) using IPM [16].
9: Update $\boldsymbol{p}_n^{(i)}$ and $\boldsymbol{Q}_n^{(i)}$ according to the solution.
10: Inform $\boldsymbol{p}_n^{(i)}$ and $\boldsymbol{Q}_n^{(i)}$ to nodes in $2R^{\max}$ distance.
11: **end if**
12: **if** a message is received from another node **then**
13: Update $\boldsymbol{p}$ and $\boldsymbol{Q}$ accordingly.
14: **end if**
15: **until** node n leaves the network.

that $t \in \mathcal{T}_n^{(i)}$. That is, the *current* time slot is a time slot at which the persistent and listening probabilities of NIC i need to be updated by solving problem (LOCAL-NUM-S). Recall from Theorem 23 that problem (LOCAL-NUM-S) is convex. Thus, it can easily be solved using IPM. In line 10, node n announces its updated persistent and listening probabilities to all nodes within its $2R^{\max}$ distance, where $R^{\max}$ is defined in Section 7.3.1. This can be done using *limited-scope message flooding*[16] [12, pp. 408]. Upon reception of the new probability values from other nodes, in line 13, node n updates its local memory accordingly. Finally, for distributed implementation of Algorithm 8, we need to slightly modify the feasible sets in (7.30) such that for each $n \in \mathcal{N}$, any NIC $i \in \mathcal{I}_n$, and each $m \in \mathcal{N}_n^{\text{out}}$, we have: $p_{nm}^{(i)(c)}, Q_n^{(i)(c)} \in [\epsilon, 1-\epsilon]$, where $0 < \epsilon \ll \frac{1}{2}$ is a *small* design parameter (e.g., $\epsilon = 10^{-6}$). This requires all NICs to listen to all channels and transmit over all channels with arbitrarily small but *non-zero* probabilities. Similar assumptions are made to avoid node starvation in the single-channel random access algorithms in [98] and [104].

[16]We measure the signalling overhead for each of our proposed DMMRA algorithms in Section 7.4.4.

Algorithm 9 - DMMRA-M: Executed by each node $n \in \mathcal{N}$ when the NICs implement *multi-channel* reception.

```
Allocate memory for p.
Randomly choose p such that p ∈ Ψ.
repeat
    for each NIC i ∈ I_n do
        Transmit to node m ∈ N_n^out on channel c ∈ C with probability p_nm^(i)(c).
    end for
    if t ∈ T_n^(i) for some i ∈ I_n then
        Solve problem (LOCAL-NUM-M) using IPM.
        Update p_n^(i) according to the solution.
        Inform p_n^(i) to nodes within 2R^max distance.
    end if
    if a message is received from another node then
        Update p accordingly.
    end if
until node n leaves the network.
```

Algorithm 9 works similarly. The persistent probabilities are adjusted by solving the convex optimization problem in (LOCAL-NUM-M). Notice that, in general, Algorithm 9 is computationally less complicated compared to Algorithm 8 as problem (LOCAL-NUM-M) has fewer variables and also fewer constraints compared to (LOCAL-NUM-S). Both Algorithms 8 and 9 are fully distributed and allow each node to adjust its operation based on a few simple local tasks and some limited-scope message exchange with other nodes.

7.3.3 Optimality and Convergence

In this section, we analytically investigate the optimality and convergence properties of DMMRA algorithms. For each node $n \in \mathcal{N}$ and any NIC $i \in \mathcal{I}_n$, we define:

$$\begin{aligned} & f_{n,SCR}^{(i)}\left(\boldsymbol{p}_n^{(-i)}, \boldsymbol{Q}_n^{(-i)}\right) \\ & = \underset{\langle \boldsymbol{p}_n^{(i)}, \boldsymbol{Q}_n^{(i)} \rangle \in \Phi_n^{(i)}}{\arg\max} \sum_{m \in \mathcal{N}} \sum_{s \in \mathcal{N}_m^{\text{out}}} u(r_{ms}(\boldsymbol{p}, \boldsymbol{Q})), \end{aligned} \tag{7.37}$$

and

$$f_{n,MCR}^{(i)}\left(p_n^{(-i)}\right) = \arg\max_{p_n^{(i)} \in \Psi_n^{(i)}} \sum_{m \in \mathcal{N}} \sum_{s \in \mathcal{N}_m^{\text{out}}} u(r_{ms}(p)). \tag{7.38}$$

We also define $f_{SCR} = (f_{n,SCR}^{(i)}, \forall i \in \mathcal{I}_n, n \in \mathcal{N})$ and $f_{MCR} = (f_{n,MCR}^{(i)}, \forall n \in \mathcal{N}, i \in \mathcal{I}_n)$ as the *mapping functions* corresponding to Algorithms 8 and 9, respectively. Let $\mathcal{F}_{SCR}$ and $\mathcal{F}_{MCR}$ denote the set of *fixed points* (cf. [18, pp. 181]) of mappings f_{SCR} and f_{MCR}, respectively. That is, if $\langle p^*, Q^* \rangle \in \mathcal{F}_{SCR}$, then $f_{SCR}(p, Q) = \langle p^*, Q^* \rangle$. Similarly, if $p^* \in \mathcal{F}_{MCR}$, then $f_{MCR}(p) = p^*$. Also let $\mathcal{S}_{SCR}$ and $\mathcal{S}_{MCR}$ denote the set of all *stationary points* (cf. [15, pp. 194]) of optimization problems (NUM-S) and (NUM-M), respectively. Note that all local (and thus global) optimal solutions of problems (NUM-S) and (NUM-M) belong to the sets $\mathcal{S}_{SCR}$ and $\mathcal{S}_{MCR}$, respectively. We can show the following key results:

Theorem 25 *$\mathcal{F}_{SCR} = \mathcal{S}_{SCR}$ and $\mathcal{F}_{MCR} = \mathcal{S}_{MCR}$.*

The proof of Theorem 25 is given in Section 7.10. From Theorem 25, any fixed point of Algorithm 8 is indeed a stationary point of optimization problem (NUM-S) and vice versa. The same statement is true for Algorithm 9 and optimization problem (NUM-M).

Theorem 26 *(a) For any choice of system parameters, the fixed point of Algorithm 8 (Algorithm 9) always exists. (b) If the number of channels $C = 1$, then Algorithm 8 (Algorithm 9) has a unique fixed point. (c) If $C > 1$, then Algorithm 8 (Algorithm 9) has at least two (i.e., non-unique) fixed points.*

The proof of Theorem 26 is given in Section 7.11. From Theorem 25 and Theorem 26(b), if the number of channels $C = 1$, then the unique fixed point of Algorithm 8 (Algorithm 9) is the

unique global optimal solution (i.e., the only stationary point) of problem (NUM-S) (problem (NUM-M)). On the other hand, from Theorem 25 and Theorem 26(c), if $C > 1$, then any fixed point of Algorithm 8 (Algorithm 9) is at least a local maximum of problem (NUM-S) (problem (NUM-M)).

Next, we discuss convergence. Let $U_{SCR}(t)$ and $U_{MCR}(t)$ denote the aggregate network utilities at time slot t, while running Algorithms 8 and 9, respectively. We can show that:

Theorem 27 *For any choice of system parameters, (a) At each time slot t, the instantaneous aggregate network utilities $U_{SCR}(t)$ and $U_{MCR}(t)$ are upper bounded:*

$$U_{SCR}(t), U_{MCR}(t) \leq L\, u(C\, I^{\max}\, \gamma^{\max}), \quad \forall (n,m) \in \mathcal{L}, \tag{7.39}$$

where $I^{\max} = \max_{n \in \mathcal{N}} I_n$ and $\gamma^{\max} = \max_{(n,m) \in \mathcal{L}, c \in \mathcal{C}} \gamma_{nm}^{(c)}$. (b) The instantaneous network utilities $U_{SCR}(t)$ and $U_{MCR}(t)$ are non-decreasing. *That is, for any $T \geq 2$,*

$$U_{SCR}(1) \leq \quad U_{SCR}(2) \quad \leq \cdots \leq U_{SCR}(T), \tag{7.40}$$

$$U_{MCR}(1) \leq \quad U_{MCR}(2) \quad \leq \cdots \leq U_{MCR}(T). \tag{7.41}$$

(c) Starting from any arbitrary initial point $\langle \boldsymbol{p}, \boldsymbol{Q} \rangle$ and $\boldsymbol{p}$, Algorithms 8 and 9 asynchronously converge to one of their fixed points, respectively. That is, there exist U^*_{SCR} and U^*_{MCR} such that:

$$U^*_{SCR} = \lim_{t \to \infty} U_{SCR}(t), \tag{7.42}$$

and

$$U^*_{MCR} = \lim_{t \to \infty} U_{MCR}(t). \tag{7.43}$$

The proof of Theorem 27 is given in Section 7.12. Notice that, from Theorem 25, U^*_{SCR} and

U^*_{MCR} are the local maxima of optimization problems (NUM-S) and (NUM-M), respectively. In many cases, the achieved fixed points are not only locally optimal, but also *globally* optimal. For example, we can verify that for the sample topologies in Section 7.2.2, we have: $U^*_{SCR} = U^\star_{SCR}$ and $U^*_{MCR} = U^\star_{MCR}$. We further discuss optimality in Section 7.4.7.

7.4 Performance Evaluation

In this section, we evaluate the performance of our proposed DMMRA algorithms. We study the convergence, robustness, and optimality properties, evaluate the performance gain of assigning partially overlapped channels, and measure the signalling overhead for both Algorithms 8 and 9. We also compare DMMRA with *utility-optimal combinatorial interface assignment and channel allocation* (UO-CIACA) [107] and *multi-channel medium access control* (MMAC) [114] algorithms.

In our simulation model, we consider ten different random topologies. Unless otherwise is stated, each topology includes $N = 10$ nodes, randomly located in a 500 m $\times$ 500 m square field. Communication and interference ranges are 150 m and 250 m, respectively. The peak data rates (i.e., $\gamma^{(c)}_{nm}$ for all $(n, m) \in \mathcal{L}$ and $c \in \mathcal{C}$) are selected randomly between 6 and 54 Mbps, as in the IEEE 802.11a standard. Except in Section 7.4.3, where we study the impact of utility parameter α, the utility functions are assumed to be logarithmic. Each time slot is 1 sec. The simulation time is 1000 time slots.

7.4.1 Convergence

In this experiment, we set the number of channels $C = 6$ and the number of NICs $I_n = 2$ for all $n \in \mathcal{N}$. The trends of the network utilities for the first simulated topology (i.e., topology number 1) when the DMMRA algorithms are being used are shown in Fig. 7.3. We can see that both Algorithms 8 and 9 converge to their fixed-points very fast, i.e., within 152 and 146 time slots, respectively. We can also observe that the utility values are *non-decreasing* and *bounded*, which confirm the results in Theorem 27(b). At the steady state, Algorithm 8 results in 34% higher utility, compared to UO-CIACA. Using Algorithm 9, the utility is further increased by 40%. Thus, equations (7.40) and (7.41) hold as *strict* inequalities in this case. Similar results are observed for other topologies.

7.4.2 Comparison with Combinatorial Channel Assignment

Next, we compare DMMRA with UO-CIACA [107] in terms of both utility and throughput. Simulation setting is the same as in Section 7.4.1. Results are shown in Fig. 7.4. In this figure, each point is the average of the measurements for all ten simulated topologies. We can see that both utility and throughput increase as there are more channels available. Algorithm 8 results in 36% and 23% higher utility and throughput, compared to UO-CIACA, respectively (see Theorem 22(b)). Using Algorithm 9 leads to further 57% and 71% increase in utility and throughout, respectively (see Theorem 22(a)).

7.4.3 Impact of Utility Parameter α

Recall from Section 7.2.1 that by changing the utility parameter α, various design objectives can be modeled. In particular, α can act as a *knob* in Algorithms 8 and 9 to control the trade-off

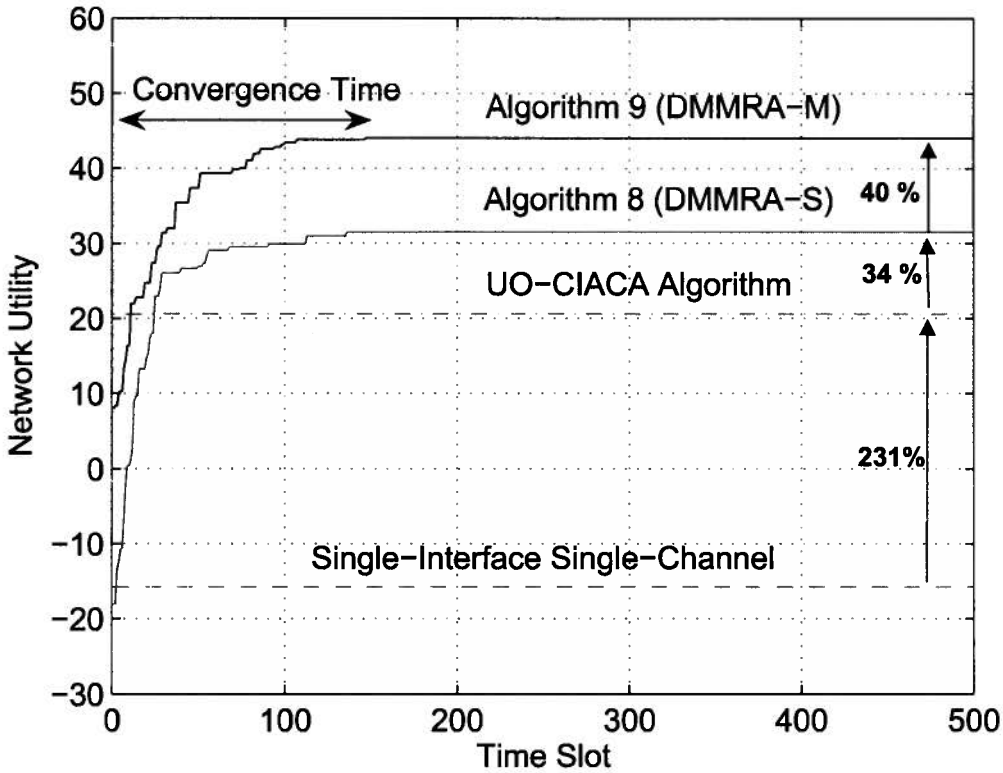


Figure 7.3: Trend of the network utility versus time slots using Algorithms 8 and 9 for the first simulated topology.

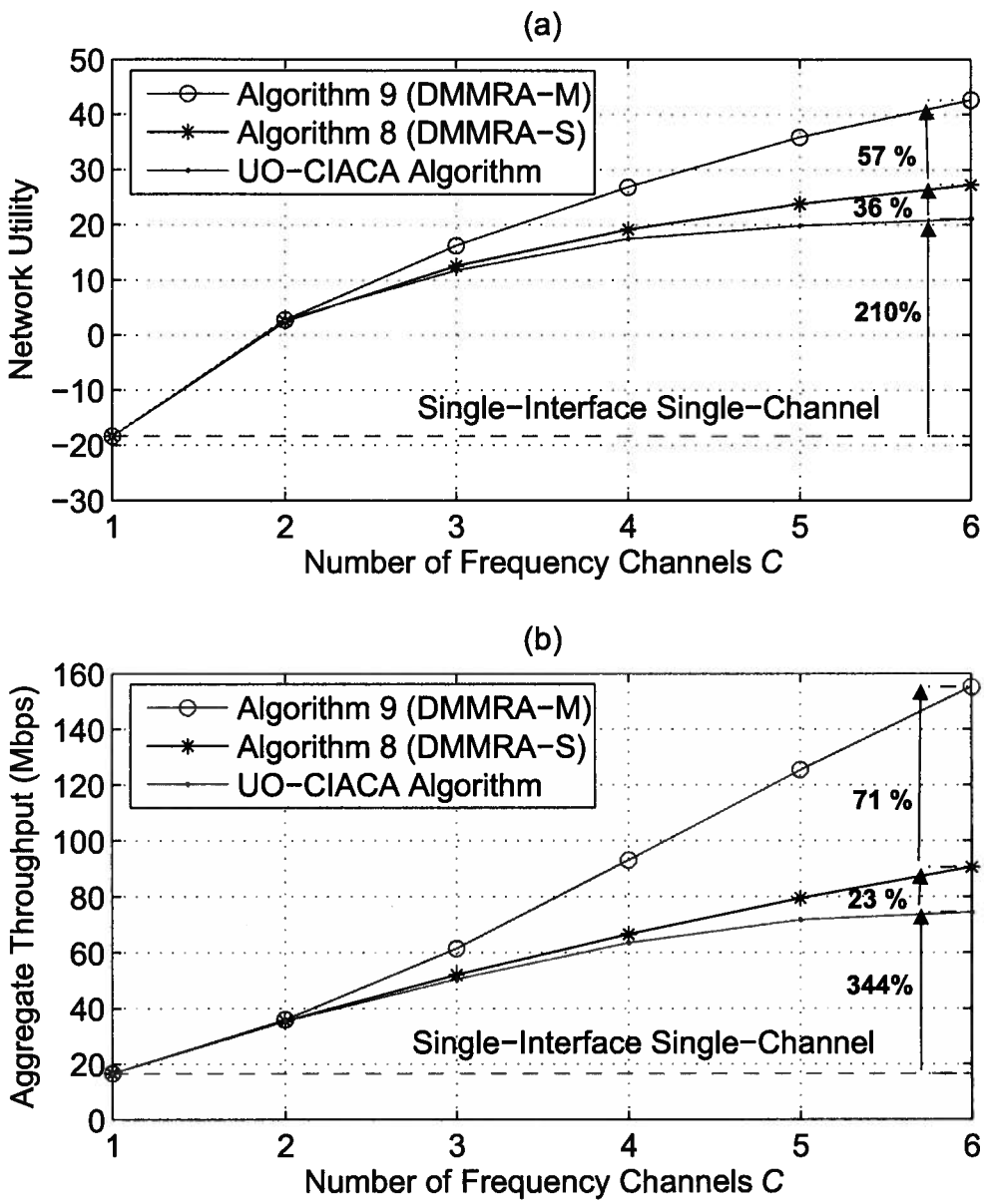


Figure 7.4: Comparison between Algorithms 8 and 9 with (UO-CIACA) algorithm [107] in terms of both utility and throughput. The number of available channels varies from 1 to 6. (a) Network utility, (b) Aggregate throughput.

between efficiency and fairness. In this section, we compare DMMRA-S algorithm with MMAC [114]. MMAC is the multi-channel extension of the IEEE 802.11 distributed coordination function. For each NIC, it assigns the channel which has the *least scheduled transmission* within the neighborhood. MMAC is designed for *single*-interface *multi*-channel networks. It also assumes that each NIC can listen to only one channel at a time. Thus, MMAC is most comparable with Algorithm 8, where $I_n = 1$ for all $n \in \mathcal{N}$. We assume that the number of channels $C = 6$. Running both Algorithm 8 and MMAC for all ten topologies, the network throughput and fairness index, when α varies between 0.5 and 5, are shown in Fig. 7.5 (a) and (b), respectively. The fairness index is calculated among the data rates of all links as in [89]. We can see that, by increasing α, we can make the system more fair, but less efficient (and vice versa). If $\alpha = 0.5$, then Algorithm 8 results in 74% higher throughput, compared to MMAC. If $\alpha = 5$, then Algorithm 8 results in 97% higher fairness index. Thus, Algorithm 8 can be set to achieve efficiency or fairness (or both). Notice that for $\alpha \in (0.5, 2)$, Algorithm 8 makes the system both more fair and also more efficient. Similar results are observed for Algorithm 9.

7.4.4 Signalling Overhead

Both Algorithms 8 and 9 require message exchange among the neighboring nodes. In this section, we measure the signalling overhead for each algorithm and compare it with the aggregate network throughput. We assume that each probability value occupies two bytes. Thus, for each node $n \in \mathcal{N}$, the message size is $2C(L_n^{\text{out}} + 2)$ bytes and $2C(L_n^{\text{out}} + 1)$ bytes for Algorithms 8 and 9, respectively. For each NIC $i \in \mathcal{I}_n$, we assume the use of *limited-scope message flooding* [12, pp. 408] to distribute $\langle \boldsymbol{p}_n^{(i)}, \boldsymbol{Q}_n^{(i)} \rangle$ (for single-channel reception scenario) and $\boldsymbol{p}_n^{(i)}$ (for multi-channel reception scenario) to all nodes within $2R^{\max}$ distance. The signalling overhead (in Kbps) and

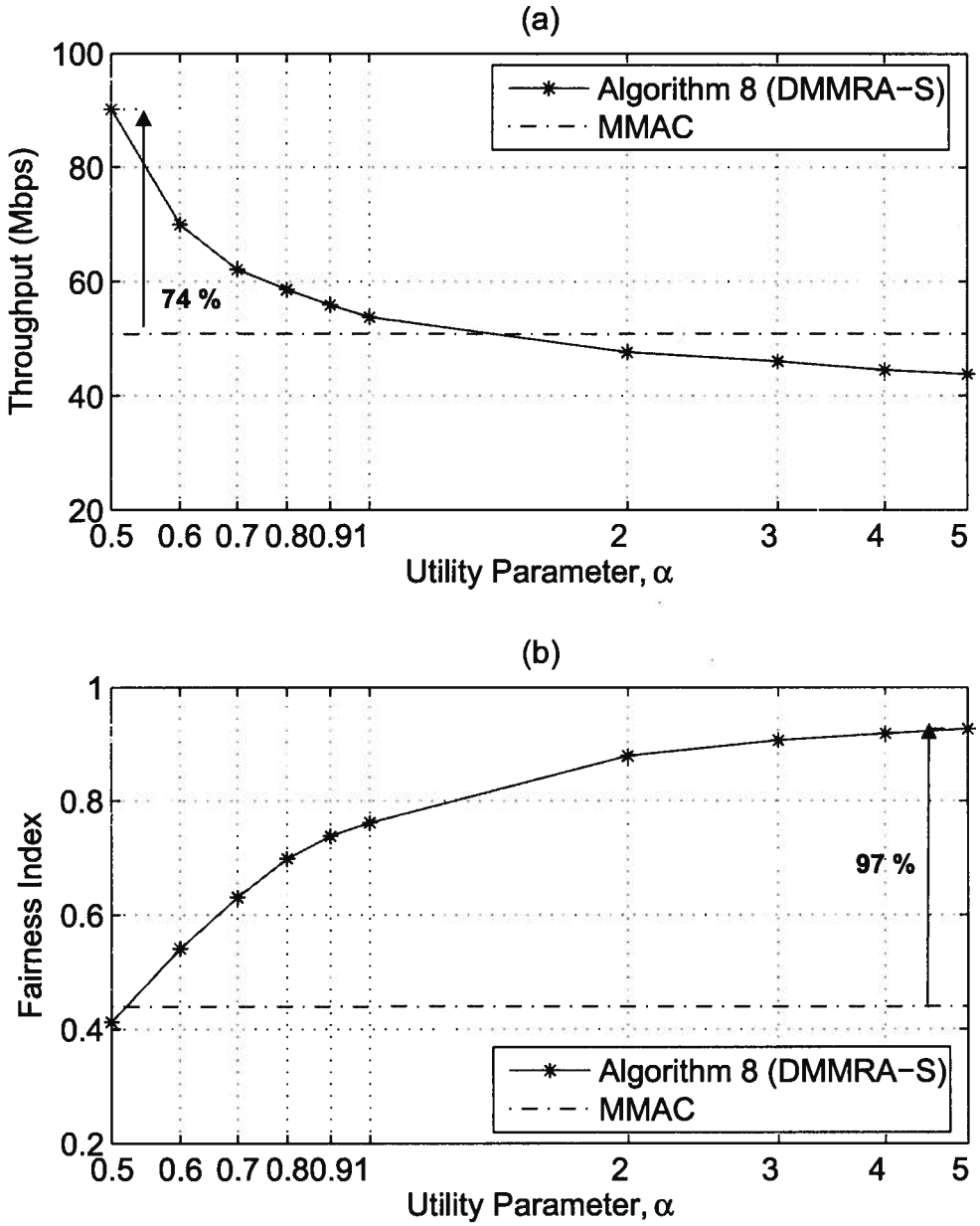


Figure 7.5: Comparison between Algorithm 8 and MMAC [114]. Each node has one NIC and there are 6 channels available. For Algorithm 8, the parameter α is varied from 0.5 to 5. (a) Throughput, (b) Fairness index.

throughput (in Mbps), when the number of nodes N varies from 10 to 50, are shown in Fig. 7.6 (a) and (b), respectively. The signalling overhead increases as the number of nodes increases. However, it is always negligible compared to the throughput (i.e., less than 0.002%). We also notice that Algorithm 9 always incurs lower signalling overhead as it has smaller messages and converges faster. When $N = 50$, Algorithm 9 results in 37% lower signalling overhead and 91% higher throughput, compared to Algorithm 8.

7.4.5 Partially Overlapped Channel Assignment

Given the data rate models in (7.19) and (7.21), Algorithms 8 and 9 can be used to assign not only the non-overlapped channels, but also the partially overlapped channels. This is particularly important when the number of orthogonal channels is limited; e.g., as in IEEE 802.11b standard, where only 3 out of 11 channels are non-overlapped. The throughput, when the IEEE 802.11b standard is used, is shown in Fig. 7.7. The channel filters are assumed to be *raised cosine* with *roll-off factors* equal to 1 (cf. [95]). The dashed lines correspond to the measured throughput when either single channel (i.e., channel 1), two non-overlapped channels (i.e., channels 1 and 6), or three non-overlapped channels (i.e., channels 1, 6, and 11) are used. We see that, using Algorithm 9, assigning the partially overlapped channels 1, ..., 6, instead of assigning only non-overlapped channels 1 and 6, results in 11% higher throughput. By assigning all partially overlapped channels 1, ..., 11, instead of assigning only the non-overlapped channels 1, 6, and 11, the throughput is increased by 13%. Similar results are observed for Algorithm 8. Here, the improvements are achieved without using extra resources. Thus, the available spectrum is utilized more efficiently.

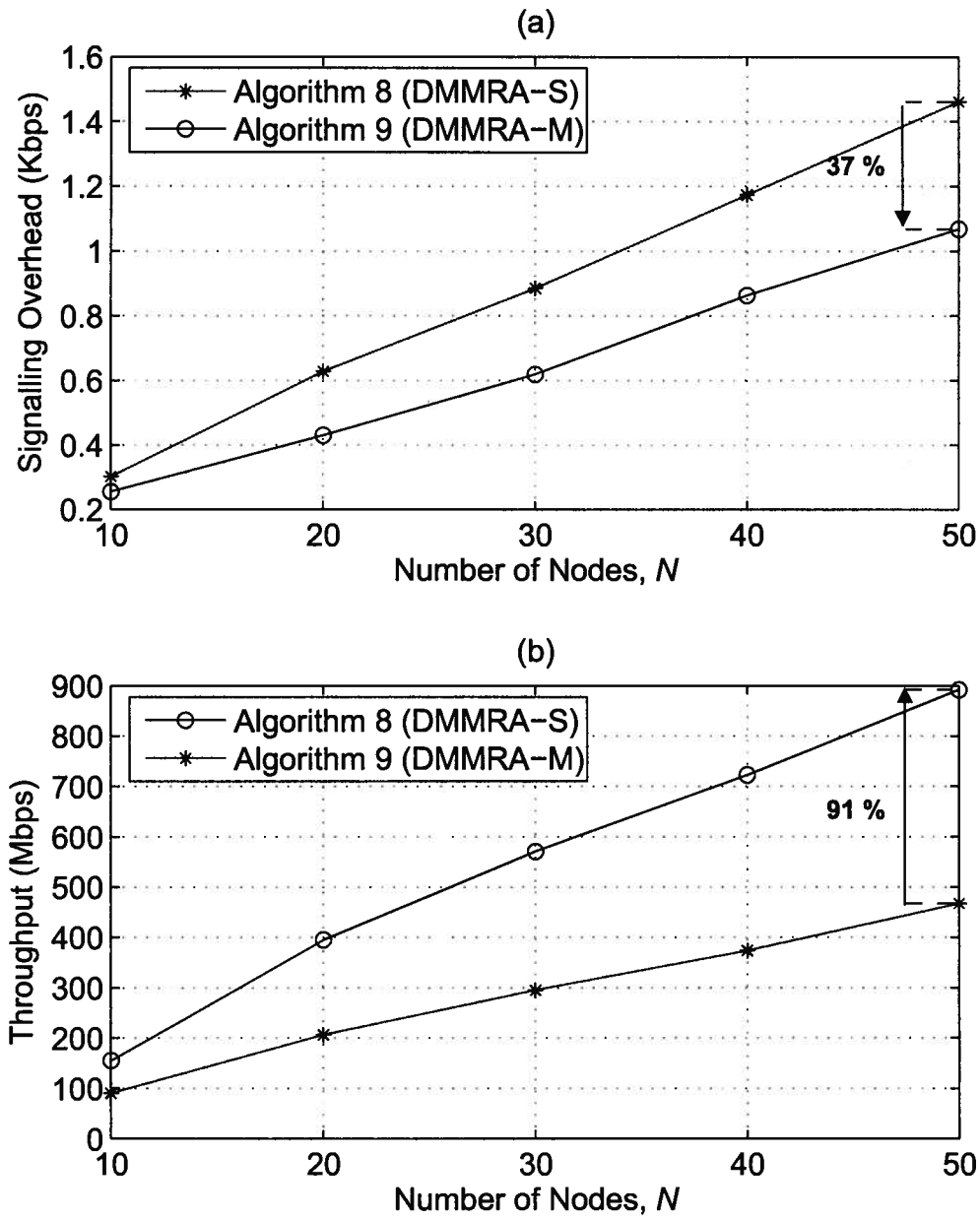


Figure 7.6: Signalling overhead and aggregate network throughput for Algorithms 8 and 9 when the number of nodes N varies from 10 to 50. Each node is equipped with 2 NICs and there are 6 orthogonal channels available.

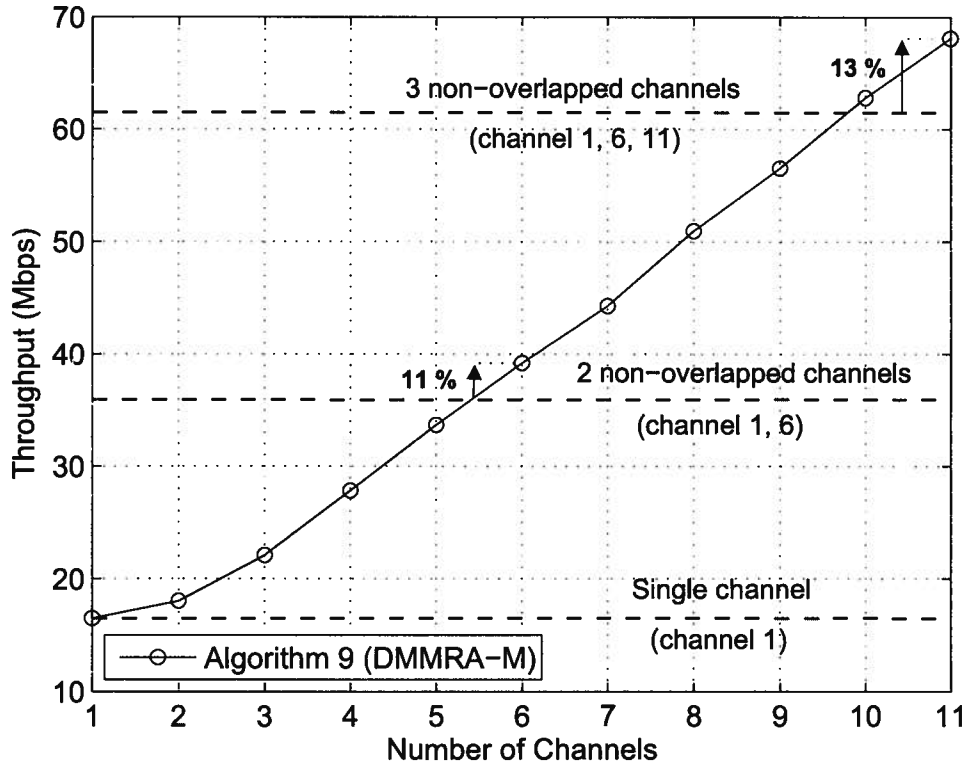


Figure 7.7: Performance improvement when both orthogonal and partially overlapped channels are being used. The number of available channels is varied from 1 to 11.

7.4.6 Impact of Delayed and Outdated Information

In some practical scenarios, the nodes may receive *outdated* information about the persistent and listening probabilities of other nodes. This can be due to *communication delay* (e.g., queueing or propagation delay) or *message loss.* The latter can occur due to *channel imperfections* (e.g., fading) or *packet collision.* In this section, we study the effect of outdated information exchange on the performance of DMMRA algorithms. In particular, we consider the case where the communication medium imposes random delay on the exchanged messages. The trend of the network utility for the first simulated topology, when Algorithm 8 is being used and the messages experience delay up to 10 time slots (i.e., 10 seconds), is shown in Fig. 7.8. We can see that Algorithm

8 still converges to its fixed-point, even though the information used by the nodes is outdated. However, communication delay may cause *utility fluctuation* (compare Fig. 7.8 and Fig. 7.3). That is, at some time instances, since the node, which executes DMMRA, may not have accurate information about the persistent and listing probabilities of its neighboring nodes, the value of the utility may be decreased. Nevertheless, Algorithm 8 still (asynchronously) converges to its fixed point, but with lower convergence speed. Similar results are obtained for Algorithm 9. From the results in Fig. 7.9, we can also see that as the delay increases from 0 to 10 time slots, the convergence time increases by 98% and 83% for Algorithms 8 and 9, respectively. We notice that Theorem 25 still holds. In fact, the fixed-points of DMMRA algorithms are independent of delay. Thus, the fixed points of Algorithms 8 and 9 are still guaranteed to be the stationary points of problems (NUM-S) and (NUM-M), respectively.

7.4.7 Optimality

From Theorem 25, every fixed point of DMMRA algorithms is a stationary point of the formulated NUM problems. That is, each fixed-point is at least a *local* maximum. However, the fixed-points may not always be *globally* maximum. In this section, we investigate the optimality of Algorithms 8 and 9. Results are shown in Fig. 7.10. In this figure, the *average* utility is compared with the *optimal* utility for each topology. To obtain the results, we calculate the utility at *all* fixed-points. This is done by partitioning the search space and running DMMRA with the *initial points* varying among the partitions. Specifically, for each topology, we obtain both *mean utility value* and *maximum utility value* among all fixed-points. The former indicates the *average* performance of our proposed DMMRA algorithms, while the latter indicates the *optimal* performance. From the results in Fig. 7.10, DMMRA achieves *near-optimal* solutions for all ten simulated topologies. On

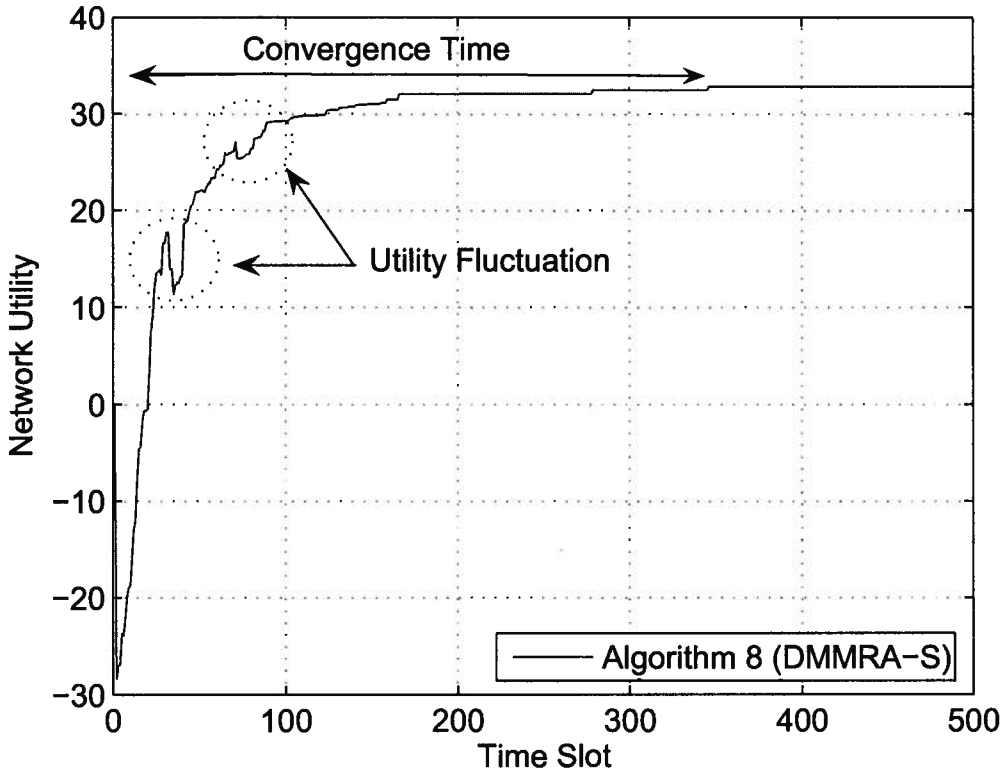


Figure 7.8: Trend of the network utility versus time slots using Algorithms 8 for the first simulated topology in the presence of communication delay.

average, Algorithms 8 and 9 result in 96.5% and 97.4% optimality, respectively. In fact, although the fixed-points are not always globally optimal, they lead to near-optimal performance.

7.5 Summary

In this chapter, we formulated a novel continuous multi-interface multi-channel random access model. Both single-channel reception and multi-channel reception scenarios were considered and the data rate models were obtained accordingly. We then formulated a multi-interface multi-channel NUM problem and proved that its optimal solution outperforms most of the previously proposed combinatorial channel assignment strategies. We also proposed a simple, fast, and

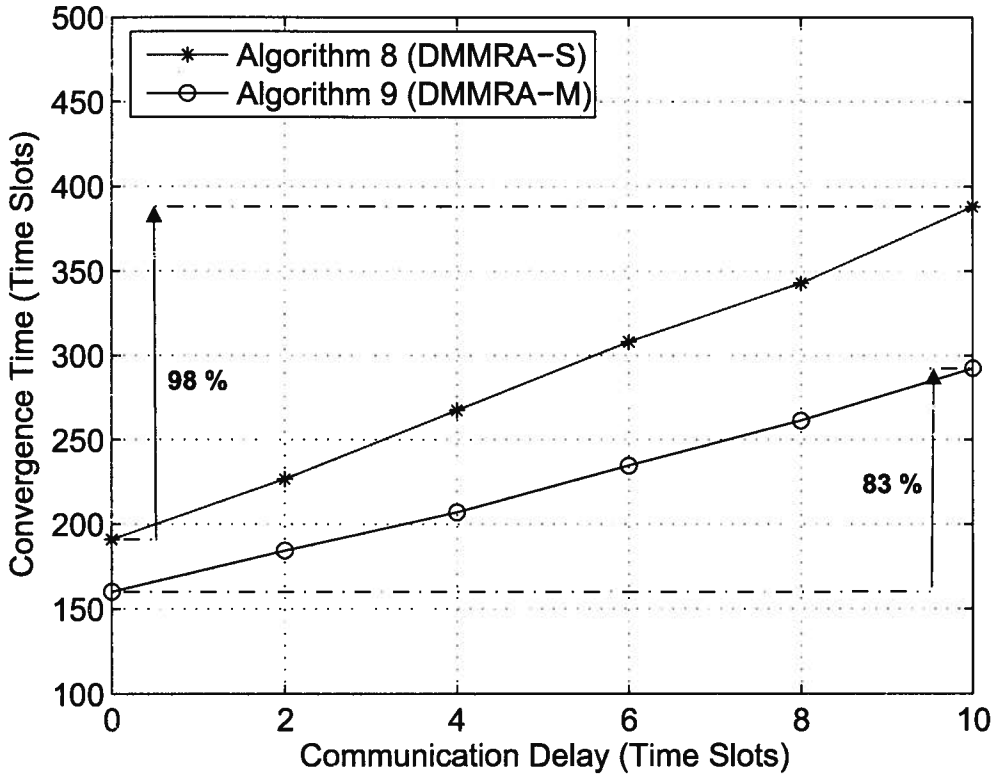


Figure 7.9: Impact of delay on convergence time.

distributed algorithm, called DMMRA, to solve the formulated NUM problem. DMMRA requires each node to only iteratively solve a *local*, *myopic*, and *convex* optimization problem. The convergence and optimality of the proposed algorithm were proved. Simulation results show that DMMRA with single-channel reception results in 36% and 23% higher network utility and throughput compared to combinatorial channel assignment. If multiple-channel reception is being used, then the utility and the throughput further increase by 57% and 71%, respectively.

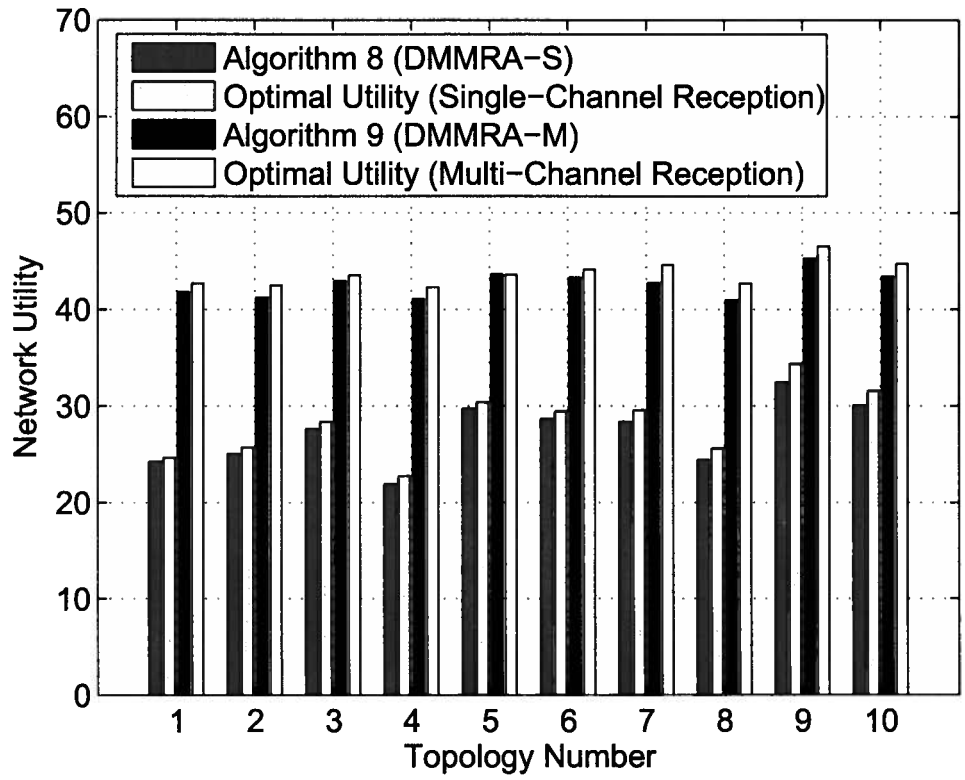


Figure 7.10: Comparison between achieved aggregate network utility and its optimal value for each proposed algorithm and each simulated topology.

7.6 Analytical Proofs

7.7 Proof of Theorem 22

Part (a): For each wireless node $m \in \mathcal{N}$, any of its NICs $j \in \mathcal{I}_m$, and each frequency channel $c \in \mathcal{C}$, we have:

$$1 - P_m^{(j)(c)} - Q_m^{(j)(c)} \overset{\text{by } (7.1)}{=} \sum_{d \in \mathcal{C} \setminus \{c\}} P_m^{(j)(d)} + \sum_{d \in \mathcal{C} \setminus \{c\}} Q_m^{(j)(d)}$$

$$\geq \sum_{d \in \mathcal{C} \setminus \{c\}} P_m^{(j)(d)}.$$

Replacing the above inequality in (7.4) and (7.11), we have:

$$\mathbb{P}\left(\hat{A}_m^{(c)}\right) \geq \mathbb{P}\left(\breve{A}_m^{(-c)}\right), \tag{7.44}$$

which also implies that $(1-\mathbb{P}(\tilde{A}_m^{(c)} \cup \hat{A}_m^{(c)})) \leq (1-\mathbb{P}(\tilde{A}_m^{(c)} \cup \check{A}_m^{(-c)}))$. Hence r_{nm} in (7.14) is always greater than or equal to the one in (7.10). Since the utility function $u(r_{nm})$ is an increasing function of r_{nm}, the inequality (7.24) is resulted when we sum up the utilities of all links.

Part (b): Let Λ denote the feasible set of problem (NUM-C). It is clear that, $\Lambda \subseteq \Phi$. Thus, any $\langle \bar{\boldsymbol{p}}, \bar{\boldsymbol{Q}} \rangle \in \Lambda$ is also a feasible solution of problem (NUM-S). Hence, *no* $\langle \bar{\boldsymbol{p}}, \bar{\boldsymbol{Q}} \rangle \in \Lambda$ can lead to an aggregate network utility which is greater than the *optimal* utility $U_{SCR}^{\star}$ over the set Φ. Therefore, the inequality in (7.25) always holds. ■

7.8 Proof of Theorem 23

For each node $n \in \mathcal{N}$ and any NIC $i \in \mathcal{I}_n$, the objective function of problem (LOCAL-NUM-S) can be written as:

$$\begin{aligned}
&\textstyle\sum_{m\in\mathcal{N}_n^{\text{out}}} u\left(\sum_{c\in\mathcal{C}}\left(\xi_{n,m}^{(i)(c)} p_{nm}^{(i)(c)} + \zeta_{n,m}^{(i)(c)}\left(1-P_n^{(i)(c)}\right)\right)\right)\\
&+\textstyle\sum_{m\in\mathcal{N}_n^{\text{in}}} u\left(\sum_{c\in\mathcal{C}}\left(\left(\theta_{n,m}^{(i)(c)}-\vartheta_{n,m}^{(i)(c)}\right)\left(1-P_n^{(i)(c)}\right)+\vartheta_{n,m}^{(i)(c)} Q_n^{(i)(c)}\right)\right)\\
&+\textstyle\sum_{m\in\mathcal{N}\setminus\{n\}} \sum_{s\in\mathcal{N}_m^{\text{out}}\setminus\{n\}} u\left(\sum_{c\in\mathcal{C}} \beta_{n,ms}^{(i)(c)}\left(1-\delta_{ns}P_n^{(i)(c)}\right)\right),
\end{aligned}$$

where for each node $m \in \mathcal{N}_n^{\text{out}}$ and any channel $c \in \mathcal{C}$,

$$\begin{aligned}
\xi_{n,m}^{(i)(c)} = {}& \gamma_{nm}^{(c)}\left(\textstyle\prod_{j\in\mathcal{I}_n\setminus\{i\}}\left(1-P_n^{(j)(c)}\right)\right)\\
&\left(\textstyle\prod_{s\in\mathcal{N}\setminus\{n,m\}}\prod_{k\in\mathcal{I}_s}\left(1-\delta_{sm}P_s^{(k)(c)}\right)\right)\\
&\left(\textstyle\prod_{l\in\mathcal{I}_m}\left(1-P_m^{(l)(c)}\right)-\prod_{l\in\mathcal{I}_m}\left(1-P_m^{(l)(c)}-Q_m^{(l)(c)}\right)\right),\\
\zeta_{n,m}^{(i)(c)} = {}& \textstyle\sum_{j\in\mathcal{I}_n\setminus\{i\}} \gamma_{nm}^{(c)} p_{nm}^{(j)(c)}\left(\prod_{k\in\mathcal{I}_n\setminus\{i,j\}}\left(1-P_n^{(k)(c)}\right)\right)\\
&\left(\textstyle\prod_{s\in\mathcal{N}\setminus\{n,m\}}\prod_{k\in\mathcal{I}_s}\left(1-\delta_{sm}P_s^{(k)(c)}\right)\right)\\
&\left(\textstyle\prod_{j\in\mathcal{I}_m}\left(1-P_m^{(j)(c)}\right)-\prod_{j\in\mathcal{I}_m}\left(1-P_m^{(j)(c)}-Q_m^{(j)(c)}\right)\right).
\end{aligned}$$

For any node $m \in \mathcal{N}_n^{\text{in}}$ and any channel $c \in \mathcal{C}$,

$$\theta_{n,m}^{(i)(c)} = \sum_{k\in\mathcal{I}_m} \gamma_{mn}^{(c)} p_{mn}^{(k)(c)} \left(\prod_{s\in\mathcal{N}\setminus\{m,n\}} \prod_{l\in\mathcal{I}_s} \left(1 - \delta_{sn} P_s^{(l)(c)}\right)\right)$$
$$\left(\prod_{v\in\mathcal{I}_m\setminus\{k\}} \left(1 - P_m^{(v)(c)}\right)\right) \left(\prod_{j\in\mathcal{I}_n\setminus\{i\}} \left(1 - P_n^{(j)(c)}\right)\right),$$
$$\vartheta_{n,m}^{(i)(c)} = \sum_{k\in\mathcal{I}_m} \gamma_{mn}^{(c)} p_{mn}^{(k)(c)} \left(\prod_{s\in\mathcal{N}\setminus\{m,n\}} \prod_{l\in\mathcal{I}_s} \left(1 - \delta_{sn} P_s^{(l)(c)}\right)\right)$$
$$\left(\prod_{v\in\mathcal{I}_m\setminus\{k\}} \left(1 - P_m^{(v)(c)}\right)\right)\left(\prod_{j\in\mathcal{I}_n\setminus\{i\}} \left(1 - P_n^{(j)(c)} - Q_n^{(j)(c)}\right)\right).$$

For any link $(m,n) \in \mathcal{L}\setminus(\mathcal{L}_n^{\text{in}} \cup \mathcal{L}_n^{\text{out}})$ and each $c \in \mathcal{C}$,

$$\beta_{n,ms}^{(i)(c)} = \sum_{k\in\mathcal{I}_m} \gamma_{ms}^{(c)} p_{ms}^{(i)(c)} \left(\prod_{j\in\mathcal{I}_m\setminus\{k\}} \left(1 - P_m^{(j)(c)}\right)\right)$$
$$\left(\prod_{v\in\mathcal{N}\setminus\{m,s,n\}} \prod_{l\in\mathcal{I}_v} \left(1 - \delta_{nv} P_v^{(l)(c)}\right)\right)$$
$$\left(\prod_{l\in\mathcal{I}_s} (1 - P_s^{(l)(c)}) - \prod_{l\in\mathcal{I}_s} \left(1 - P_s^{(l)(c)} - Q_s^{(l)(c)}\right)\right).$$

Notice that $\xi_{n,m}^{(i)(c)}$, $\zeta_{n,m}^{(i)(c)}$, $\theta_{n,m}^{(i)(c)}$, $\vartheta_{n,m}^{(i)(c)}$, and $\beta_{n,ms}^{(i)(c)}$ only depend on $\boldsymbol{p}_n^{(-i)}$ and $\boldsymbol{Q}_n^{(-i)}$. In fact, they can be treated as *constants* as far as problem (LOCAL-NUM-S) is concerned. Since the utility functions are concave, the objective function of problem (LOCAL-NUM-S) is a summation of *concave-affine* compositions over $\boldsymbol{p}_n^{(i)}$ and $\boldsymbol{Q}_n^{(i)}$. Therefore, it is concave [16, pp. 84]. Clearly, $\Phi_n^{(i)}$ is also a convex set. Together, these imply that problem (LOCAL-NUM-S) is a convex optimization problem. The proof for (LOCAL-NUM-M) is similar and is omitted. ■

7.9 Proof of Theorem 24

Let $\mathcal{N} = \{n, m\}$, $\mathcal{L}_n = \{i\}$, and $\mathcal{L}_m = \{j\}$. For wireless link $(n,m) \in \mathcal{L}$, we have:

$$r_{nm}(\boldsymbol{p}) = \sum_{c\in\mathcal{C}} \gamma_{nm}^{(c)} p_{nm}^{(i)(c)} \left(1 - \sum_{d\in\mathcal{C}} p_{mn}^{(j)(d)}\right), \tag{7.45}$$

The data rate $r_{mn}(\boldsymbol{p})$ can also be obtained similarly. We can re-write optimization problem (LOCAL-NUM-M) as:

$$\begin{aligned}\underset{\boldsymbol{p}_n^{(i)}\in\Psi_n^{(i)}}{\text{maximize}}\ & u\Big(\Big(1-\textstyle\sum_{e\in\mathcal{C}} p_{mn}^{(j)(e)}\Big)\Big(\textstyle\sum_{c\in\mathcal{C}} \gamma_{nm}^{(c)} p_{nm}^{(i)(c)}\Big)\Big) \\ & +u\Big(\Big(\textstyle\sum_{e\in\mathcal{C}}\gamma_{mn}^{(e)} p_{mn}^{(j)(e)}\Big)\Big(1-\textstyle\sum_{c\in\mathcal{C}} p_{nm}^{(i)(c)}\Big)\Big).\end{aligned} \tag{7.46}$$

From Theorem 23, problem (7.46) is a convex optimization problem. By solving the Karush-Kuhn-Tucker (KKT) *sufficient* and *necessary* optimality conditions (cf. [16, pp. 244]) and rearrangement the terms, we can show that for any fairness index α and for any channel $c \in \mathcal{C}$,

$$\left(\frac{\gamma_{nm}^{(c)}\left(1-\sum_{e\in\mathcal{C}} p_{mn}^{(j)(e)^*}\right)}{\left(\sum_{e\in\mathcal{C}} \gamma_{mn}^{(e)} p_{mn}^{(j)(e)^*}\right)}\right)^{1-\alpha} = \left(\frac{\sum_{d\in\mathcal{C}}(\gamma_{nm}^{(d)}/\gamma_{nm}^{(c)}) p_{nm}^{(i)(d)^*}}{1-\sum_{d\in\mathcal{C}} p_{nm}^{(i)(d)^*}}\right)^{\alpha}. \tag{7.47}$$

From (7.35) and (7.47), the optimal solution of convex optimization problem (7.46) can be obtained by solving the following *system of linear equations*:

$$\textstyle\sum_{d\in\mathcal{C}} \boldsymbol{M}^{(cd)}\left(\boldsymbol{p}_n^{(-i)}\right) p_{nm}^{(i)(d)^*} = 1, \qquad \forall c \in \mathcal{C}. \tag{7.48}$$

In vector representation, (7.48) is equivalent to:

$$\boldsymbol{M}\left(\boldsymbol{p}_n^{(-i)}\right) \boldsymbol{p}_n^{(i)^*} = \mathbf{1}. \tag{7.49}$$

The solution of the system of linear equations in (7.49) is obtained as in (7.34). In general, the closed-form solution can also be obtained using *substitution* method (cf. [115]). Notice that, any optimal solution of problem (7.46) should satisfy (7.49). Since problem (7.46) is always a *feasible* optimization problem (cf. [15, pp. 9]), the solution in (7.34) always exists, regardless of the choice of system parameters. However, the solution may not be unique. For example, if $\alpha = 1$, any persistent probability vector $\boldsymbol{p}_n^{(i)}$ is globally optimal as long as $\sum_{c\in\mathcal{C}} p_{nm}^{(i)(c)} = \frac{1}{2}$. ■

7.10 Proof of Theorem 25

Consider the case where the NICs implement single-channel reception. For any fixed point $\langle \boldsymbol{p}^*, \boldsymbol{Q}^* \rangle \in \mathcal{F}_{SCR}$, the tuple $\langle \boldsymbol{p}_n^{(i)^*}, \boldsymbol{Q}_n^{(i)^*} \rangle$ is an optimal solution of the convex optimization problem in (LOCAL-NUM-S) for any $i \in \mathcal{I}_n$ for all $n \in \mathcal{N}$; otherwise, there would exist some NIC $j \in \mathcal{I}_n$ for some node $n \in \mathcal{N}$ which deviates its persistent or listening probabilities from the fixed point to achieve a better solution for its local optimization problem (see line 10 in Algorithm 8). Thus, for any node $n \in \mathcal{N}$ and each of its NICs $i \in \mathcal{I}_n$, the tuple $\langle \boldsymbol{p}_n^{(i)^*}, \boldsymbol{Q}_n^{(i)^*} \rangle$ needs to satisfy the following KKT conditions [16, pp. 244]. For each channel $c \in \mathcal{C}$,

$$\frac{\xi_{n,m}^{(i)(c)} - \zeta_{n,m}^{(i)(c)}}{\left(\sum_{d\in\mathcal{C}}\left(\xi_{n,m}^{(i)(d)} p_{nm}^{(i)(d)^*} + \zeta_{n,m}^{(i)(d)}(1-P_n^{(i)(d)^*})\right)\right)^{\alpha}} - \sum_{v\in\mathcal{N}_n^{\text{out}}} \frac{\zeta_{n,v}^{(i)(c)}}{\left(\sum_{d\in\mathcal{C}}\left(\xi_{n,v}^{(i)(d)} p_{nv}^{(i)(d)^*} + \zeta_{n,v}^{(i)(d)}(1-P_n^{(i)(d)^*})\right)\right)^{\alpha}}$$

$$- \sum_{v\in\mathcal{N}_n^{\text{in}}} \frac{\theta_{n,v}^{(i)(c)} - \vartheta_{n,v}^{(i)(c)}}{\left(\sum_{d\in\mathcal{C}}\left((\theta_{n,v}^{(i)(d)} - \vartheta_{n,v}^{(i)(d)})(1-P_n^{(i)(d)^*}) + \vartheta_{n,v}^{(i)(d)} Q_n^{(i)(d)^*}\right)\right)^{\alpha}} \tag{7.50}$$

$$- \sum_{v\in\mathcal{N}\setminus\{n\}} \sum_{s\in\mathcal{N}_v^{\text{out}}\setminus\{n\}} \frac{\beta_{n,vs}^{(i)(c)}}{\left(\sum_{d\in\mathcal{C}} \beta_{n,vs}^{(i)(d)}(1-\delta_{ns}P_n^{(i)(d)^*})\right)^{\alpha}} = \lambda_n^{(i)} - \delta_{nm}^{(i)(c)}, \qquad \forall\, m \in \mathcal{N}_n^{\text{out}},$$

$$\sum_{v\in\mathcal{N}_n^{\text{in}}} \frac{\vartheta_{n,m}^{(i)(c)}}{\left(\sum_{d\in\mathcal{C}}\left((\theta_{n,v}^{(i)(d)} - \vartheta_{n,v}^{(i)(d)})(1-P_n^{(i)(d)^*}) + \vartheta_{n,v}^{(i)(d)} Q_n^{(i)(d)^*}\right)\right)^{\alpha}} = \lambda_n^{(i)} - \mu_n^{(i)(c)}, \tag{7.51}$$

$$\lambda_n^{(i)}\left(\sum_{d\in\mathcal{C}} \sum_{v\in\mathcal{N}_n^{\text{out}}} p_{nv}^{(i)(d)^*} + Q_n^{(i)(d)^*} - 1\right) = 0, \tag{7.52}$$

$$\delta_{nm}^{(i)(c)} p_{nm}^{(i)(c)^*} = 0, \qquad \mu_n^{(i)(c)} Q_n^{(i)(c)^*} = 0, \tag{7.53}$$

$$p_{nm}^{(i)(c)^*} \geq 0, \qquad Q_n^{(i)(c)^*} \geq 0, \tag{7.54}$$

$$P_n^{(i)(c)^*} = \sum_{v\in\mathcal{N}_n^{\text{out}}} p_{nv}^{(i)(c)^*}, \tag{7.55}$$

where $\lambda_n^{(i)}$, $\delta_{nm}^{(i)(c)}$, and $\mu_n^{(i)(c)}$ are the Lagrange multipliers corresponding to the constraints in (7.30). On the other hand, by definition (cf. [15, pp. 194]), any stationary point in $\mathcal{S}_{SCR}$ satisfies all KKT conditions of problem (NUM-S). Since the objective functions in (NUM-S) and (LOCAL-NUM-S) are the same and the set of constraints in (NUM-S) is the union of the sets of constraints in (LOCAL-NUM-S) for all NICs, the KKT conditions for non-convex optimization

problem (NUM-S) are indeed the union of the conditions in (7.50)-(7.55) for all $i \in \mathcal{I}_n$ and any $n \in \mathcal{N}$. Thus, any fixed point $\langle \boldsymbol{p}^*, \boldsymbol{Q}^* \rangle \in \mathcal{F}_{SCR}$ is also a stationary point. This implies that, $\mathcal{F}_{SCR} \subseteq \mathcal{S}_{SCR}$. Following a similar argument, we can show that, $\mathcal{S}_{SCR} \subseteq \mathcal{F}_{SCR}$. Since $\mathcal{F}_{SCR} \subseteq \mathcal{S}_{SCR}$ and $\mathcal{S}_{SCR} \subseteq \mathcal{F}_{SCR}$, we have: $\mathcal{F}_{SCR} = \mathcal{S}_{SCR}$. ■

7.11 Proof of Theorem 26

Part (a): It is easy to verify that for any node $n \in \mathcal{N}$ and any NIC $i \in \mathcal{I}_n$, we can select $p_{nm}^{(i)(c)} = \frac{1}{2CL_n}$ and $P_n^{(i)(c)} = Q_n^{(i)(c)} = \frac{1}{2C}$ for all $c \in \mathcal{C}$ and any $m \in \mathcal{L}_n^{\text{out}}$ as a *feasible* (not necessarily optimal) solution for problem (NUM-S). Similarly, we can select $p_{nm}^{(i)(c)} = \frac{1}{2CL_n}$ and $P_n^{(i)(c)} \frac{1}{2C}$ for all $c \in \mathcal{C}$ and $m \in \mathcal{L}_n^{\text{out}}$ as a *feasible* solution for problem (NUM-M). Thus, $|\Phi| \geq 1$ and $|\Psi| \geq 1$. Since both problems (NUM-S) and (NUM-M) are feasible problems, they have at least one stationary point [15, pp. 194]. From this, together with Theorem 25, we have: $|\mathcal{F}_{SCR}| = |\mathcal{S}_{SCR}| \geq 1$ and $|\mathcal{F}_{MCR}| = |\mathcal{S}_{MCR}| \geq 1$.

Part (b): From [98, Lemma 2], in a single-channel network, problems (NUM-S) and (NUM-M) can be transformed to *equivalent* convex optimization problems using the logarithmic change of variables. Since a convex problem has a unique stationary point, we have: $|\mathcal{S}_{SCR}| = |\mathcal{S}_{MCR}| = 1$. From this, together with Theorem 25, $|\mathcal{F}_{SCR}| = |\mathcal{F}_{MCR}| = 1$ and the fixed points are unique.

Part (c): We prove this part by contradiction. Consider Algorithm 8 and assume that it has a *unique* fixed point $\langle \boldsymbol{p}, \boldsymbol{Q} \rangle$. From Theorem 25, $\langle \boldsymbol{p}, \boldsymbol{Q} \rangle$ is also a stationary point of optimization problem (NUM-S). Now consider another point $\langle \bar{\boldsymbol{p}}, \bar{\boldsymbol{Q}} \rangle$, where for all nodes $n \in \mathcal{N}$, any NIC $i \in \mathcal{I}_n$, any channel $c \in \mathcal{C}$ and any node $m \in \mathcal{L}_n^{\text{out}}$, we have:

$$\bar{p}_{nm}^{(i)(c)} = p_{nm}^{(i)(C-c+1)}, \qquad \bar{Q}_n^{(i)(c)} = Q_n^{(i)(C-c+1)}. \tag{7.56}$$

For example, if the number of frequency channels $C = 4$, then we have: $\bar{p}_{nm}^{(i)(1)} = p_{nm}^{(i)(4)}$, $\bar{p}_{nm}^{(i)(2)} = p_{nm}^{(i)(3)}$, $\bar{p}_{nm}^{(i)(3)} = p_{nm}^{(i)(2)}$, and $\bar{p}_{nm}^{(i)(4)} = p_{nm}^{(i)(1)}$. From (7.10) and (7.19), it is easy to verify that: $\boldsymbol{r}(\boldsymbol{p}, \boldsymbol{Q}) = \boldsymbol{r}(\bar{\boldsymbol{p}}, \bar{\boldsymbol{Q}})$. Thus, $\langle \bar{\boldsymbol{p}}, \bar{\boldsymbol{Q}} \rangle$ is a stationary point of optimization problem (NUM-S). From Theorem 25, it is also a fixed point of Algorithm 8. However, this contradicts the assumption that $\langle \boldsymbol{p}, \boldsymbol{Q} \rangle$ is a *unique* fixed point. The proof for Algorithm 9 is similar. ■

7.12 Proof of Theorem 27

Part (a): From (7.10), (7.14), (7.19), and (7.21), for each $(n, m) \in \mathcal{L}$,

$$r_{nm} \leq \sum_{i \in \mathcal{I}_n} \sum_{c \in \mathcal{C}} \gamma_{nm}^c \leq \sum_{c \in \mathcal{C}} I^{\max} \gamma^{\max} = C I^{\max} \gamma^{\max}.$$

Thus, the utility of each wireless link $(n, m) \in \mathcal{L}$ is upper bounded by $u(C\, I^{\max}\, \gamma^{\max})$ and the aggregate network utility is upper bounded by $L\, u(C\, I^{\max}\, \gamma^{\max})$.

Part (b): We prove this part by contradiction. First consider the case where the NICs implement multiple-channel reception model and assume that at some time slot $t \in [2, T]$, $U_{SCR}(t-1) > U_{SCR}(t)$. In that case, there exist a node n and an NIC $i \in \mathcal{I}_n$ such that $t \in \mathcal{T}_n^{(i)}$ and solving problem (LOCAL-NUM-S) at NIC i *reduces* the network utility (i.e., the objective function in problem (NUM-S)). However, this is impossible as the objective functions in problems (LOCAL-NUM-S) and (NUM-S) are the same. Thus, $U_{SCR}(t-1) \leq U_{SCR}(t)$ and (7.40) holds. The proof for (7.41) is similar.

Part (c): The existence of the limitations in (7.42) and (7.43) is directly resulted from parts (a) and (b). Notice that any bounded non-decreasing sequence of real numbers always converges to a fixed point. ■

Chapter 8

Two-Fold Pricing to Guarantee Individual Profits and Maximum Social Welfare in Multi-Hop Wireless Access Networks

Various pricing schemes have recently been proposed either to encourage collaboration among the network elements or to utilize the network resources efficiently. Pricing as a tool for *resource allocation* was first proposed in [44, 46] for congestion control among elastic traffic sources. In this regard, the network is designed to solve a network utility maximization (NUM) problem across all traffic sources, subject to the link capacity constraints. The corresponding Lagrange multipliers are interpreted as the congestion prices. Each source which uses a link resource is *charged* with the link's congestion price. The transmission rates and the congestion prices are iteratively updated using the gradient projection method until the global optimal network utility is achieved. The work in [44] has been extended to other resource allocation problems such as medium access control, power control, frequency channel assignment, and spectrum sharing

[30, 47, 116, 117, 118, 119, 120]. Recently, it has also been shown that the gradient updates can be replaced by the best-response updates to achieve faster convergence and more robust performance [54].

Another thread of research focuses on using pricing to encourage *collaboration* among the nodes [121, 122, 123, 124, 125, 126, 127, 128, 129]. In a multi-hop network, where the nodes need to forward packets for other nodes, the optimal network performance might be at the cost of performance degradation for some intermediate relay nodes. When the intermediate nodes have no incentive to collaborate, the well-known *forwarder's dilemma* (cf. [130]) can occur, where no node forwards the packets for other nodes. To resolve this problem, incentives can be offered to the relay nodes in the form of payments or rewards in turn for their help in forwarding other nodes's traffic. In general, achieving the *optimal* network performance may *not* be always guaranteed in the incentive-based strategies as they mainly take the individual profit objectives into consideration. The problem of designing pricing models for Internet service providers (ISPs) in a fixed *wired* network has been studied in [122, 123, 124, 125]. In [122], He and Walrand proposed a pricing scheme which encourages the ISPs to collaborate to achieve a *fair* revenue sharing. In [123], Shen and Basar studied the pricing problem in wired access networks using game theory. The ISPs are modeled as strategic players to maximize their revenue and the users are modeled as natural players to maximize their utilities. The interaction between each ISP and its associated users is modeled as a Stackelberg game. In [124], Davoli *et al.* considered the pricing problem where the ISPs do not have any knowledge about users' utility functions. Pricing for multicast wired Internet has also been studied in [125].

The pricing models for wired networks cannot be easily extended to *wireless* access networks. In general, there are two main challenging issues that need to be addressed in wireless access

networks: *channel imperfection* (e.g., wireless fading), and *interference.* In [126], Neely proposed an economic model for wireless ad-hoc networks, with *stochastic* channel states, within the general framework of *back-pressure* algorithms [131, 132]. The relay prices are used to encourage packet forwarding. However, it is essentially assumed that the network is *interference-free.* That is, each node can transmit with an arbitrary high transmission power without interfering with other nodes. Interference-free pricing has also been addressed in [127] and [128].

In general, most of the previously proposed pricing models in the literature have one or more of the following performance bottlenecks: (1) network resources are not efficiently (i.e., optimally) allocated, (2) individual profits are not taken into consideration, and (3) interference among the wireless transmissions is not taken into account. In this chapter, we address these performance bottlenecks in all three aspects. In particular, we extend the work by Neely [126] and propose a market-based network model with *two-fold pricing* (TFP) which fully incorporates the effect of interference. Our model uses *relay-pricing* to encourage nodes to collaborate and forward each other's packets. We also use *interference-pricing* to encourage the wireless relay nodes to properly share the common network resources. Together, the relay and interference prices incorporate both *cooperative* and *competitive* interactions among the nodes. We analytically prove that for a wide range of pricing functions, our proposed TFP scheme leads to a guaranteed *positive* profit for each individual node. The profit increases as the node forwards more packets. This better pays off the collaborative nodes. Finally, assuming the presence of slow-fading channels, we obtain the relay and interference pricing functions in a code division multiple access (CDMA) network such that not only the positive individual profits are guaranteed, but also the network social welfare and the network utility are maximized. Compared with the single-fold pricing (SFP) model in [126], simulation results show that our TFP scheme can increase the social welfare and the network

throughput by 24.6% and 14.7%, respectively. It also leads to more fair revenue sharing among the nodes as it results in 18.3% higher fairness index.

The rest of this chapter is organized as follows. Our proposed two-fold pricing model is described in Section 8.1. The key properties of our model are analytically proved in Section 8.2. Simulation results are presented in Section 8.3. A summary is given in Section 8.4.

8.1 Market-Based Wireless Access Network Model with Two-Fold Pricing

8.1.1 System Model

Consider a stationary wireless access network. Let $\mathcal{N}$, with size $|\mathcal{N}| = N$, denote the set of *wireless relay nodes* and $\mathcal{L}$, with size $|\mathcal{L}| = L$, denote the set of unidirectional *wireless links.* For each node $n \in \mathcal{N}$, the set of all *incoming* and *outgoing* links are denoted by $\mathcal{L}_n^{\text{in}} \subset \mathcal{L}$ and $\mathcal{L}_n^{\text{out}} \subset \mathcal{L}$, respectively. We also define $\mathcal{N}_n^{\text{in}} = \{m : (m,n) \in \mathcal{L}_n^{\text{in}}\}$ and $\mathcal{N}_n^{\text{out}} = \{m : (n,m) \in \mathcal{L}_n^{\text{out}}\}$ as the set of *in-neighbors* and the set of *out-neighbors* of node n, respectively. Wireless relay nodes are assumed to be *independent* commercial entities. Together, they form a *wireless backbone* to provide connectivity among *wireless users* in a multi-hop manner. The set of users is denoted by $\mathcal{D}$, with size $|\mathcal{D}| = D$. Each relay node $n \in \mathcal{N}$ offers connectivity only to a subset of users, denoted by $\mathcal{D}_n \subset \mathcal{D}$. Each user is offered connectivity from exactly one wireless relay node. All users $i, j \in \mathcal{D}_n$ are able to communicate directly with each other. However, if any user $i \in \mathcal{D}_n$ wants to send data to another user $k \in \mathcal{D}_c$, where $c \in \mathcal{N} \backslash \{n\}$, it should first transfer the data to node n, and the data are then transferred to node c via the intermediate wireless relay nodes before delivering to user k. In turn, node n charges user i for its offered connectivity service. We assume

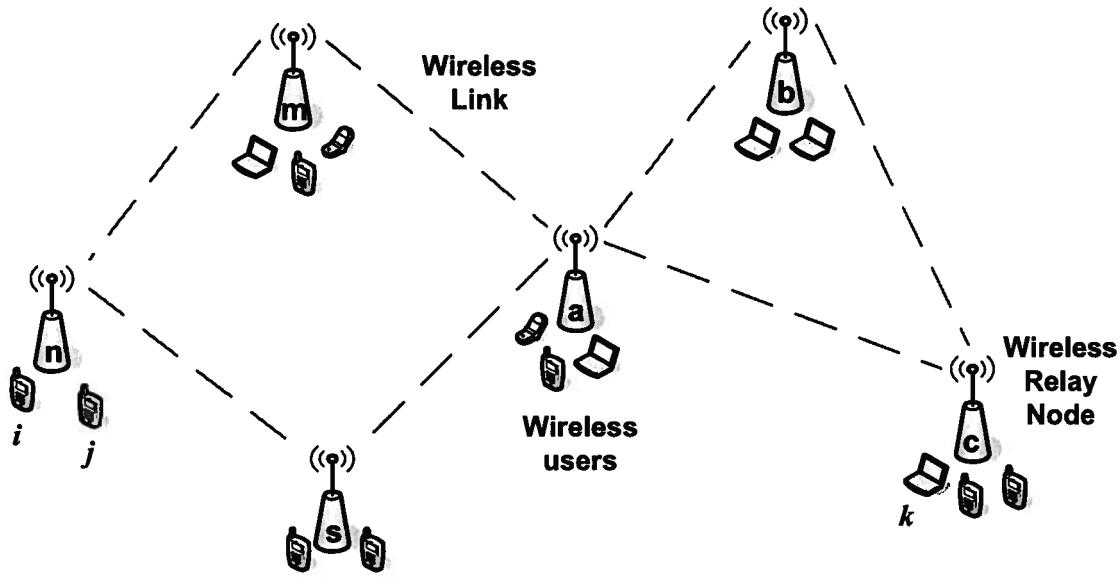


Figure 8.1: A sample multi-hop wireless access network with six wireless relay nodes, labeled as n, m, s, a, b, c, and fifteen wireless users. Here $\mathcal{D}_n = \{i, j\}$ and $k \in \mathcal{D}_c$. Users i and j can directly communicate with each other. However, if user i (or user j) wants to send data to user k, it should first transfer data to its associated wireless relay node (i.e., node n), and the data are then transferred to wireless relay node c via the intermediate nodes (e.g., wireless relay nodes s and a) in a multi-hop manner before being delivered to wireless user k. In turn for the provided connectivity service, wireless relay node n and all the intermediate wireless relay nodes are paid according to their offered relay prices.

that all wireless relay nodes communicate over the same frequency band which is different from those frequency bands used by the users to communicate with each other and their associated wireless relay nodes. This avoids interference between access and relay transmissions. However, the transmissions among the wireless relay nodes can still interfere with each other. A sample wireless access network is shown in Fig. 8.1. In this figure, there are $N = 6$ wireless relay nodes, labeled as n, m, s, a, b, and c. There are also $D = 15$ wireless users.

Each wireless relay node $n \in \mathcal{N}$ is assumed to have $N-1$ separate queues to store the incoming data according to their *final* destination. All data that are destined to any of the users of relay

node $c \in \mathcal{N} \backslash \{n\}$ are stored in the cth queue. The contents of the cth queue are called *commodity c data.* For each commodity c data, node n maintains a set $\mathcal{H}_n^{(c)} \subseteq \mathcal{N}_n^{\text{out}}$, which includes its neighboring relay nodes with *minimum hop-counts* to node c and can *relay* commodity c data towards node c. For example, $\mathcal{H}_n^{(c)} = \{m, s\}$, $\mathcal{H}_m^{(c)} = \{a\}$, $\mathcal{H}_s^{(c)} = \{a\}$, and $\mathcal{H}_a^{(c)} = \{c\}$ in Fig. 8.1.

Time is divided into equal-length slots $\mathcal{T} = \{0, 1, 2, \ldots\}$. For each link $(n, m) \in \mathcal{L}$, let Ω_{nm} denote the set of all possible *channel states.* Channel states can vary (e.g., due to wireless fading). At each time slot $t \in \mathcal{T}$, the current channel state is denoted by $\omega_{nm}(t) \in \Omega_{nm}$. We stack up the channel states of all links at time t and denote the obtained $L \times 1$ vector by $\boldsymbol{\omega}(t)$. That is, $\boldsymbol{\omega}(t) = (w_{nm}(t),\ \forall\, n, m \in \mathcal{N},\ (n, m) \in \mathcal{L})$. Let $\mathcal{T}_\omega \subseteq \mathcal{T}$ denote the set of time slots at which the channel state vector $\boldsymbol{\omega}$ changes. We assume that $\boldsymbol{\omega}$ has an independent and identical distribution (i.i.d.) over time slots $\mathcal{T}_\omega$. We also assume the *slow-fading* scenario such that:

$$|t_2 - t_1| \geq \Lambda, \qquad \forall\, t_1, t_2 \in \mathcal{T}_\omega, \tag{8.1}$$

where $\Lambda \gg 1$. That is, there are at least Λ time slots between any two consecutive changes in channel states. We will consider the *fast fading* case (i.e., when $\Lambda \to 1$) in Section 8.3.

For each wireless relay node $n \in \mathcal{N}$ and any of its neighboring nodes $m \in \mathcal{N}_n^{\text{out}}$, let $\mu_{nm}^{(c)}(\boldsymbol{p}(t), \boldsymbol{w}(t)) \geq 0$ denote the transmission rate offered to commodity c data over link (n, m) during time slot t. Here, $\boldsymbol{p}(t) = \left(p_{nm}^{(c)}(t),\ \forall\, n, m \in \mathcal{N},\ \forall\, c \in \mathcal{N} \backslash \{n\},\ (n, m) \in \mathcal{L}\right)$ denotes the $L(N-1) \times 1$ vector of *transmission powers* for all links and all commodities. The scalar $p_{nm}^{(c)}(t) \geq 0$ denotes the transmission power corresponding to the transmission of commodity c data over link (n, m). In this chapter, we assume that all nodes use CDMA technology. At each time slot $t \in \mathcal{T}$ and for each wireless relay node $n \in \mathcal{N}$, the commodity $c \in \mathcal{N} \backslash \{n\}$ data transmission rate over wireless link

$(n,m) \in \mathcal{L}_n^{\text{out}}$ can be modeled as [133]:

$$\mu_{nm}^{(c)}(\boldsymbol{p}(t), \boldsymbol{w}(t)) = A_s \log\left(1 + \frac{K\, h_{nm}\, \omega_{nm}(t)\, p_{nm}^{(c)}(t)}{I_{nm}(\boldsymbol{p}_{-n}(t)) + \eta_m}\right), \tag{8.2}$$

where A_s denotes the channel symbol rate, K is the processing gain, η_m denotes the noise power at the receiver node m, h_{nm} is the *channel power gain* from relay node n to relay node m, $\boldsymbol{p}_{-n}(t) = \left(p_{ms}^{(d)}(t),\ \forall m \in \mathcal{N}\backslash\{n\},\ s \in \mathcal{N}_m^{\text{out}},\ d \in \mathcal{N}\backslash\{m\}\right)$ denotes the transmission power of all nodes *other than* node n, and $I_{nm}(\boldsymbol{p}_{-n}(t))$ is the aggregate *interference power* on link (n,m). Notice that the term $K\, h_{nm}\, \omega_{nm}(t)\, p_{nm}^{(c)}(t)/(I_{nm}(\boldsymbol{p}_{-n}(t)) + \eta_m)$ is the *signal to interference plus noise ratio* (SINR) for commodity c data transmissions over link (n,m). We have:

$$I_{nm}(\boldsymbol{p}_{-n}(t)) = \textstyle\sum_{a \in \mathcal{N}\backslash\{n\}} h_{am} \left(\sum_{d \in \mathcal{N}\backslash\{n\}} \sum_{b \in \mathcal{N}_a^{\text{out}}} p_{ab}^{(d)}(t)\right). \tag{8.3}$$

Each node $n \in \mathcal{N}$ limits its total transmission power such that $\sum_{c \in \mathcal{N}\backslash\{n\}} \sum_{m \in \mathcal{N}_n^{\text{out}}} p_{nm}^{(c)} \leq P_n^{\max}$, where $P_n^{\max} > 0$ is fixed. Thus, the transmission rates are always *bounded.* We can define:

$$\mu_n^{\max,\ \text{in}} = \max_{\boldsymbol{p},\boldsymbol{\omega}} \textstyle\sum_{c \in \mathcal{N}\backslash\{n\}} \sum_{m \in \mathcal{N}_n^{\text{in}}} \mu_{mn}^{(c)}(\boldsymbol{p}, \boldsymbol{w}), \tag{8.4}$$

and

$$\mu_n^{\max,\ \text{out}} = \max_{\boldsymbol{p},\boldsymbol{\omega}} \textstyle\sum_{c \in \mathcal{N}\backslash\{n\}} \sum_{m \in \mathcal{N}_n^{\text{out}}} \mu_{nm}^{(c)}(\boldsymbol{p}, \boldsymbol{w}), \tag{8.5}$$

as the maximum data rate on any incoming and any outgoing link of node $n \in \mathcal{N}$, respectively.

8.1.2 Two-Fold Relay and Interference Pricing

Pricing among the wireless relay nodes

In our market-based model, at any time slot $t \in \mathcal{T}$, if wireless relay node $n \in \mathcal{N}$ transmits commodity c data with rate $\mu_{nm}^{(c)}(\boldsymbol{p}(t), \boldsymbol{w}(t))$ to its neighboring wireless relay node $m \in \mathcal{N}_n^{\text{out}}$, then it pays $\mu_{nm}^{(c)}(\boldsymbol{p}(t), \boldsymbol{w}(t))\, \phi_m^{(c)}(t)$ units of currency to node m as *relay service charge.* Here $\phi_m^{(c)}(t) \geq 0$

denotes the *relay price* corresponding to commodity c data, advertised by wireless relay node m. In total, at time slot t, node n pays:

$$\sum_{c\in\mathcal{N}\setminus\{n\}} \mu_{nm}^{(c)}(\boldsymbol{p}(t),\boldsymbol{w}(t))\,\phi_m^{(c)}(t) \tag{8.6}$$

units of currency to any neighboring wireless relay node $m \in \mathcal{N}_n^{\text{out}}$ as relay service charge. Similarly, in total, node n receives $\left(\sum_{c\in\mathcal{N}\setminus\{n\}} \mu_{mn}^{(c)}(\boldsymbol{p}(t),\boldsymbol{w}(t))\right)\phi_n^{(c)}(t)$ units of currency from any wireless relay node $m \in \mathcal{N}_n^{\text{in}}$ for its *offered* relay service.

Besides the mutual relay services that the neighboring wireless relay nodes offer to each other, relay nodes also affect each other's operation through *interference power* as shown in (8.2) and (8.3). From (8.3), for each wireless relay node $n \in \mathcal{N}$, the higher the total transmission power $\sum_{c\in\mathcal{N}\setminus\{n\}}\sum_{m\in\mathcal{N}_n^{\text{out}}} p_{nm}^{(c)}(t)$, the greater is the interference power that wireless relay node n causes on other nodes. In our pricing model, at each time slot t, wireless relay node n pays:

$$\left(\sum_{c\in\mathcal{N}\setminus\{n\}}\sum_{m\in\mathcal{N}_n^{\text{out}}} p_{nm}^{(c)}(t)\right)\psi_a^{(n)}(t) \tag{8.7}$$

units of currency to each wireless relay node $a \in \mathcal{N}\setminus\{n\}$ as *interference compensation charge.* Here $\psi_a^{(n)}(t) \geq 0$ denotes the *interference price* informed by relay node a to node n. Unlike the relay prices which vary depending on the commodity data, the interference prices are the same for all commodities as the contents of the transmissions do not affect their interference level. Instead, the interference prices may vary depending on the node locations. The closer two relay nodes are located, the higher is the corresponding channel power gain. This results in higher interference power and consequently higher interference price. Similar to (8.7), at each time slot $t \in \mathcal{T}$, node n receives $(\sum_{d\in\mathcal{N}\setminus\{a\}}\sum_{b\in\mathcal{N}_a^{\text{out}}} p_{ab}^{(d)}(t))\,\psi_a^{(n)}(t)$ units of currency from node a as the compensation for the interference node a causes on the transmissions of node n.

For each wireless relay node $n \in \mathcal{N}$ and at any time slot $t \in \mathcal{T}$, let $U_n^{(c)}(t)$ denote the current

commodity $c \in \mathcal{N}\backslash\{n\}$ queue backlog. We define $\boldsymbol{U}(t) = \left(U_n^{(c)}(t),\ \forall n \in \mathcal{N},\ \forall c \in \mathcal{N}\backslash\{n\}\right)$ as the vector of *queue backlogs* in all wireless relay nodes at time slot t. The relay and the interference prices are assumed to be set as follows:

$$\phi_n^{(c)}(t) = \Phi_n^{(c)}(\boldsymbol{U}(t-\Upsilon), \ldots, \boldsymbol{U}(t), \boldsymbol{p}(t-\Upsilon), \ldots, \boldsymbol{p}(t)), \qquad \forall\, c \in \mathcal{N}\backslash\{n\}, \tag{8.8}$$

$$\psi_n^{(a)}(t) = \Psi_n^{(a)}(\boldsymbol{U}(t-\Upsilon), \ldots, \boldsymbol{U}(t), \boldsymbol{p}(t-\Upsilon), \ldots, \boldsymbol{p}(t)), \qquad \forall\, a \in \mathcal{N}\backslash\{n\}, \tag{8.9}$$

where $\Upsilon \geq 1$ is a design parameter and $\Phi_n^{(c)}(\cdot)$ and $\Psi_n^{(a)}(\cdot)$ are two non-negative real scalar *pricing functions* of all queue backlogs and all transmission powers at time slots $\{t-\Upsilon, t-\Upsilon+1, \ldots, t\}$. The above pricing functions are *general* and can model *various* relay and interference adjustment schemes. We only make a few technical assumptions. First, if $U_n^{(c)}(t) > 0$, then $\Phi_n^{(c)}(\cdot) > 0$. That is, if relay node n already has some backlogged commodity c data, it will *not* offer free relay service. Second, if $\sum_{c \in \mathcal{N}\backslash\{n\}} U_n^{(c)}(t) > 0$ and $\sum_{c \in \mathcal{N}\backslash\{n\}} \sum_{m \in \mathcal{N}_n^{\text{out}}} p_{nm}^{(c)}(t) > 0$, then $\Psi_n^{(m)}(\cdot) > 0$. That is, if relay node n has any backlog and it is currently transmitting some data on at least one of its outgoing links, it will *not* set its advertised interference prices to zero. Third, $\Phi_n^{(c)}(\cdot)$ is an *increasing* function of $U_n^{(c)}(t)$. Finally, there exists a large enough but bounded constant $V_n^{\max}$ such that for any commodity $c \in \mathcal{N}\backslash\{n\}$ and any time slot $t \in \mathcal{T}$, $\phi_n^{(c)}(t) \leq V_n^{\max} U_n^{(c)}(t)$. In general, the unbounded sets of time slots at which the vector of relay prices $\boldsymbol{\phi}(t) = \left(\phi_n^{(c)}(t),\ \forall n \in \mathcal{N},\ c \in \mathcal{N}\backslash\{n\}\right)$ and the vector of interference prices $\boldsymbol{\psi}(t) = \left(\psi_n^{(a)}(t),\ \forall n \in \mathcal{N},\ a \in \mathcal{N}\backslash\{n\}\right)$ are being updated are denoted by $\mathcal{T}_\phi \subset \mathcal{T}$ and $\mathcal{T}_\psi \subset \mathcal{T}$, respectively.

Pricing between each wireless relay node and its users

In our market-based model, each wireless relay node $n \in \mathcal{N}$ provides relay service for its associated wireless users according to its relay prices. At each time slot $t \in \mathcal{T}$, if wireless user $i \in \mathcal{D}_n$ wants

to send data to another user $k \in \mathcal{D}_c$ (for $c \neq n$) at rate $r_i^{(k)}(t)$, it needs to pay $r_i^{(k)}(t)\,\phi_n^{(c)}(t)$ units of currency to wireless relay node n as relay service charge. At time slot t, in total, user i pays:

$$\textstyle\sum_{c\in\mathcal{N}\setminus\{n\}} \sum_{k\in\mathcal{D}_c} r_i^{(k)}(t)\,\phi_n^{(c)}(t). \tag{8.10}$$

We assume that wireless relay node n assigns all its users with a maximum allowed sending rate $R_n^{\max}$ according to its processing capacity. Each user $i \in \mathcal{D}_n$ also maintains a non-negative, increasing, and strictly concave *utility function* $g_i^{(k)}(r_i^{(k)}(t))$ for any $k \in \mathcal{D}\setminus\mathcal{D}_n$ which indicates a *monetary measure* of user i's level of satisfaction from sending rate $r_i^{(k)}(t)$. Thus, user i adjusts its rates $\boldsymbol{r}_i = (r_i^{(k)}(t),\ \forall k \in \mathcal{D}\setminus\mathcal{D}_n)$ by solving the following *local* optimization problem:

$$\begin{aligned} \max_{\boldsymbol{r}_i(t)\succeq\mathbf{0}} \ & \textstyle\sum_{c\in\mathcal{N}\setminus\{n\}}\sum_{k\in\mathcal{D}_c} \left(g_i^{(k)}(r_i^{(k)}(t)) - r_i^{(k)}(t)\phi_n^{(c)}(t)\right) \\ \text{s.t.} \quad & \textstyle\sum_{c\in\mathcal{N}\setminus\{n\}} \sum_{k\in\mathcal{D}_c} r_i^{(k)}(t) \leq R_n^{\max}. \end{aligned} \tag{8.11}$$

Notice that the objective function of the above optimization problem is always *non-negative* as at least for $\boldsymbol{r}_i = \mathbf{0}$, it is equal to zero. We define user i's *profit* at each time slot $t \in \mathcal{T}$ as:

$$\vartheta_i(t) = \left(\textstyle\sum_{c\in\mathcal{N}\setminus\{n\}} \sum_{k\in\mathcal{D}_c} g_i^{(k)}(r_i^{(k)}(t))\right) - \left(\textstyle\sum_{c\in\mathcal{N}\setminus\{n\}}(\sum_{k\in\mathcal{D}_c} r_i^{(k)}(t))\phi_n^{(c)}(t)\right). \tag{8.12}$$

From (8.11), user i adjusts $\boldsymbol{r}_i(t)$ to maximize its profit subject to the total rate constraint. Unlike the network model in [126], where each relay node can only support at most *one* user, here we allow each relay node to offer the connectivity service to an *arbitrary* number of users.

8.1.3 Resource Allocation

At each time slot $t \in \mathcal{T}$, given the advertised relay prices from all its out-neighbors, node $n \in \mathcal{N}$ can compute *differential relay price* for any $m \in \mathcal{N}_n^{\text{out}}$ and each $c \in \mathcal{N}\setminus\{n\}$ as [126]:

$$\delta_{nm}^{(c)}(t) = \phi_n^{(c)}(t) - \phi_m^{(c)}(t) - \phi^{\max}, \tag{8.13}$$

where $\phi^{\max} = V^{\max} U^{\max}$ denotes the largest possible change in *any* relay price during one time slot. Here, $V^{\max} = \max_n V_n^{\max}$ and $U^{\max} = \max_n\{\mu_n^{\max,\text{ out}}, \mu_n^{\max,\text{ in}} + R_n^{\max}\}$ represent the largest possible change in any queue backlog, where $\mu_n^{\max,\text{ in}}$ and $\mu_n^{\max,\text{ out}}$ are defined in (8.4) and (8.5), respectively. At the beginning of each time slot $t \in \mathcal{T}$, relay node n measures $\omega_{nm}(t)$ for all of its outgoing wireless links $(n,m) \in \mathcal{L}_n^{\text{out}}$ and adjusts its transmission powers $\boldsymbol{p}_n(t) = (p_{nm}^{(c)}(t), \forall c \in \mathcal{N}\backslash\{n\},\ m \in \mathcal{N}_n^{\text{out}})$ by solving the following *local* optimization problem:

$$\begin{aligned}
\max_{\boldsymbol{p}_n(t)\succeq \mathbf{0}} \quad & \textstyle\sum_{c\in\mathcal{N}\backslash\{n\}}\sum_{m\in\mathcal{N}_n^{\text{out}}} \mu_{nm}^{(c)}(\boldsymbol{p}(t),\boldsymbol{\omega}(t))\, \delta_{nm}^{(c)}(t) \\
& - \left(\textstyle\sum_{c\in\mathcal{N}\backslash\{n\}}\sum_{m\in\mathcal{N}_n^{\text{out}}} p_{nm}^{(c)}(t)\right)\left(\textstyle\sum_{a\in\mathcal{N}\backslash\{n\}} \psi_a^{(n)}(t)\right) \\
\text{s.t.} \quad & \textstyle\sum_{c\in\mathcal{N}\backslash\{n\}}\sum_{m\in\mathcal{N}_n^{\text{out}}} p_{nm}^{(c)}(t) \le P_n^{\max}, \\
& p_{nm}^{(c)}(t) = 0,\ \forall\, c \in \mathcal{N}\backslash\{n\},\ m \notin \mathcal{H}_n^{(c)} \text{ or } c \neq c_{nm}^{\star}(t) \text{ or } \delta_{nm}^{(c)}(t) \le 0,
\end{aligned} \tag{8.14}$$

where $\mu_{nm}^{(c)}$ is as in (8.2), $\mathcal{H}_n^{(c)}$ is defined in Section 8.1.1, and we have:

$$c_{nm}^{\star}(t) = \arg\max_{c:m\in\mathcal{H}_n^{(c)}} \delta_{nm}^{(c)}(t), \qquad \forall n \in \mathcal{N},\ m \in \mathcal{N}_n^{\text{out}}. \tag{8.15}$$

The *optimal* objective function in (8.14) is always *non-negative* since at least when $\boldsymbol{p}_n(t) = \mathbf{0}$, the objective function is equal to zero. Comparing to the resource allocation problem in [126], the objective function in (8.14) has an extra *negative* term:

$$-\left(\textstyle\sum_{c\in\mathcal{N}\backslash\{n\}}\sum_{m\in\mathcal{N}_n^{\text{out}}} p_{nm}^{(c)}(t)\right)\left(\textstyle\sum_{a\in\mathcal{N}\backslash\{n\}} \psi_a^{(n)}(t)\right), \tag{8.16}$$

which denotes the *total* interference compensation charge that wireless relay node n should pay to other relay nodes. By solving (8.14), node n finds the trade-off between *maximizing* $\sum_{c\in\mathcal{N}\backslash\{n\}}\sum_{m\in\mathcal{N}^{\text{out}}} \mu_{nm}^{(c)}(\boldsymbol{p}(t),\boldsymbol{\omega}(t))\delta_{nm}(t)$ (i.e., the original objective function in [126]) and *minimizing* its interference compensation cost. Each node then implements the same routing strategy as in [126]. That is, node n transmits commodity $c_{nm}^{\star}(t)$ data on iink (n,m) as long as $\delta_{nm}^{(c_{nm}^{\star}(t))} > 0$.

No commodity $c \neq c^{\star}_{nm}(t)$ data is sent on link (n, m) at time t.

Theorem 28 *Let $\boldsymbol{p}^{\star}_n(t)$ denote the optimal solution of problem (8.14). Assuming that $K \gg 1$ and all links operate in the high SINR regime (cf. [30, 133]), for each neighboring relay node $m \in \mathcal{N}^{out}_n$ and any commodity $c \in \mathcal{N}\backslash\{n\}$, if $c = c^{\star}_{nm}(t)$, $\delta^{(c)}_{nm}(t) > 0$, and $m \in \mathcal{H}^{(c)}_n$, then:*

$$p^{\star(c)}_{nm}(t) = \min\left\{\frac{\delta^{(c^{\star}_{nm}(t))}_{nm}(t)}{\sum_{a\in\mathcal{N}\backslash\{n\}} \psi^{(n)}_a(t)}, \frac{\delta^{(c^{\star}_{nm}(t))}_{nm}(t)\, P^{\max}_n}{\sum_{a\in\mathcal{N}^{out}_n} \delta^{(c^{\star}_{na}(t))}_{na}}\right\}; \tag{8.17}$$

otherwise, $p^{\star(c)}_{nm}(t) = 0$.

Theorem 28 provides a *closed-form* solution for the constrained optimization problem in (8.14). The proof of Theorem 28 is given in Section 8.5.1. We show that (8.17) satisfies the *necessary* and *sufficient* Karush-Kuhn-Tucker (KKT) optimality conditions (cf. [16]).

8.2 Guaranteed Positive Individual Profits and Maximum Social Welfare

Recall from Section 8.1.1 that each wireless relay node $n \in \mathcal{N}$ is a self-interested independent commercial entity. Thus, it is willing to *make money* out of its offered relay and connectivity services. In this regard, at each time slot $t \in \mathcal{T}$ we define node n's *profit* as:

$$\begin{aligned}\chi_n(t) = &\sum_{c\in\mathcal{N}\backslash\{n\}} \sum_{i\in\mathcal{D}_n} \sum_{k\in\mathcal{D}_c} r^{(k)}_i(t)\, \phi^{(c)}_n(t)\\ &+ \sum_{m\in\mathcal{N}^{\text{in}}_n} \sum_{c\in\mathcal{N}\backslash\{n\}} \mu^{(c)}_{mn}(\boldsymbol{p}(t), \boldsymbol{w}(t))\phi^{(c)}_n(t) - \sum_{m\in\mathcal{N}^{\text{out}}_n} \sum_{c\in\mathcal{N}\backslash\{n\}} \mu^{(c)}_{nm}(\boldsymbol{p}(t), \boldsymbol{w}(t))\phi^{(c)}_m(t)\\ &+ \sum_{a\in\mathcal{N}\backslash\{n\}} \psi^{(a)}_n(t) \sum_{d\in\mathcal{N}\backslash\{a\}} \sum_{b\in\mathcal{N}^{\text{out}}_a} p^{(d)}_{ab}(t) - \sum_{a\in\mathcal{N}\backslash\{n\}} \psi^{(n)}_a(t) \sum_{c\in\mathcal{N}\backslash\{n\}} \sum_{m\in\mathcal{N}^{\text{out}}_n} p^{(c)}_{nm}(t).\end{aligned} \tag{8.18}$$

The first term in (8.18) is the *total* relay charges from all wireless users $i \in \mathcal{D}_n$. The second and the third terms denote the *total* relay charges from and to all the neighboring wireless relay

nodes, respectively. The fourth and the fifth terms denote the *total* interference charges from and to all other nodes $a \in \mathcal{N} \backslash \{n\}$, respectively. We are now ready to present our first key result:

Theorem 29 *For each $T \gg 1$ and any wireless relay node $n \in \mathcal{N}$, we have:*

$$\sum_{t=0}^{T} \chi_n(t) \geq \sum_{t=0}^{T} \sum_{a \in \mathcal{N} \backslash \{n\}} \psi_n^{(a)}(t) \left(\sum_{d \in \mathcal{N} \backslash \{a\}} \sum_{b \in \mathcal{N}_a^{out}} p_{ab}^{(d)}(t) \right). \tag{8.19}$$

The proof of Theorem 29 is given in Section 8.5.2. From Theorem 29, each relay node is guaranteed to obtain a profit which is at least as high as the right-hand side (RHS) of (8.19). All the terms in the RHS of (8.19) are *non-negative.* From the assumptions on the pricing functions in Section 8.1.2, the RHS of (8.19) is zero *only if* for the duration from time $t = 0$ to $t = T$, no relay node in set $\mathcal{N} \backslash \{n\}$ transmits any data and there is no data in any of the $N - 1$ queues in node n. This happens only if *either* $N = 1$ and there is no other relay node in the network *or* node n has set its relay prices too high so that none of its users and neighboring relay nodes are interested in transferring their data to node n. The former is the case when there is no need to relay node n as all users in set $\mathcal{D}_n = \mathcal{D}$ can communicate with each other directly. The latter is the case when node n is reluctant to contribute as a part of the wireless access network.

Corollary 5 *Each wireless relay node that contributes in relaying data is guaranteed to receive a positive-valued profit. The profit increases as the node forwards more packets.*

Theorem 29 and Corollary 5 are general and apply to any choice of user utilities and pricing functions. Next, we determine the pricing functions $\Phi_n^{(c)}$ and $\Psi_n^{(m)}$ for all relay nodes $n \in \mathcal{N}$, any commodity $c \in \mathcal{N} \backslash \{n\}$, and any neighboring relay node $m \in \mathcal{N}_n^{\text{out}}$ to maximize the *network social welfare*; i.e., the aggregate profit across all wireless relay nodes and users:

$$\sum_{t=1}^{T} \sum_{n \in \mathcal{N}} \chi_n(t) + \sum_{t=1}^{T} \sum_{i \in \mathcal{D}} \vartheta_i(t). \tag{8.20}$$

Lemma 7 *The social welfare model in (8.20) is equal to the following:*

$$\sum_{t=1}^{T} \sum_{n\in\mathcal{N}} \sum_{i\in\mathcal{D}_n} \sum_{c\in\mathcal{N}\setminus\{n\}} \sum_{k\in\mathcal{D}_c} g_i^{(k)}(r_i^{(k)}(t)). \tag{8.21}$$

The proof of Lemma 7 is given in Section 8.5.3. From Lemma 7, the monetary exchanges among the relay nodes and the users cancel out each other. Eq. (8.21) is the *aggregate network utility* across all users. Thus, maximizing the social welfare in our TFP model is equivalent to maximizing the network utility. This helps us to present our second key result as follows.

Theorem 30 *Given $\mathcal{T}$ (i.e., the set of time slots), $\mathcal{T}_\omega$ (i.e., the set of time slots at which the vector of channel states $\boldsymbol{\omega}$ changes), and $\Lambda \gg 1$ (i.e., the fading parameter), let:*

$$\mathcal{T}_\phi = \mathcal{T}_\omega, \qquad \mathcal{T}_\psi = \mathcal{T}, \qquad \text{and} \qquad \Upsilon = \Lambda, \tag{8.22}$$

where $\mathcal{T}_\omega$, $\mathcal{T}_\psi$, and Υ are defined in Section 8.1.2. The aggregate network utility and the network social welfare are maximized if each wireless relay node $n\in\mathcal{N}$ at each time slot $t'\in\mathcal{T}_\phi$ sets:

$$\Phi_n^{(c)} = V\, U_n^{(c)}(t'), \qquad \forall c\in\mathcal{N}\setminus\{n\}, \tag{8.23}$$

and also at each time slot $t \in \{t', \ldots, t'+\Upsilon\} \subset \mathcal{T}_\psi$ each node $n \in \mathcal{N}$ sets:

$$\Psi_n^{(a)} = \sum_{m\in\mathcal{N}_n^{out}} h_{am} \frac{\max\{\delta_{nm}^{(c_{nm}^*(t'))}(t'), 0\}}{I_{nm}(\boldsymbol{p}_{-n}(t)) + \eta_m}, \qquad \forall a\in\mathcal{N}\setminus\{n\}, \tag{8.24}$$

where $V > 0$ is an arbitrary design parameter.

The proof of Theorem 30 is given in Section 8.5.4. The key is to show that our proposed two-fold pricing functions result in solving the well-known *maximum weight matching* problem (cf. [131, 132]) *periodically* (i.e., every $\Upsilon = \Lambda$ time slots). Together, Theorems 29 and 30 show that if the transmission powers, relay prices, and interference prices are set according to (8.17), (8.23), and (8.24), respectively, then not only each relay node receives a guaranteed positive profit, but also the social welfare and the network utility are maximized. The pseudocode of the pricing

algorithms that each node and each user need to execute are given in Algorithms 10 and 11, respectively. In lines 5 to 7 and lines 17 to 19 of Algorithm 10, the relay and interference prices are adjusted according to Theorem 30, respectively. On the other hand, in lines 21 to 28, the transmission powers are set according to Theorem 28. Notice that Algorithm 11 simply adjusts the transmission rates of the users based on the optimal solutions of the profit maximization problem in (8.11).

Algorithm 10 *Executed by each wireless relay node $n \in \mathcal{N}$*

1: Randomly choose the relay prices, interference prices, and the transmission powers.
2: **repeat**
3: Transmit commodity $c \in \mathcal{N}\backslash\{n\}$ data to out-neighbor node $m \in \mathcal{H}_n^{(c)}$ with power $p_{nm}^{(c)}$.
4: **if** $t \in \mathcal{T}_\phi$ **then**
5: **for** all commodity $c \in \mathcal{N}\backslash\{n\}$ **do** Set $\phi_n^{(c)} = V\, U_n^{(c)}$. **end**
6: Inform relay prices $\boldsymbol{\phi}_n = (\phi_n^{(c)},\ \forall c \in \mathcal{N}\backslash\{n\})$ to all its in-neighbors and users.
7: **for** all out-neighbors $m \in \mathcal{N}_n^{\text{out}}$ **do**
8: **for** all commodity $c \in \mathcal{N}\backslash\{n\}$ **do** Set $\delta_{nm}^{(c)} = \phi_n^{(c)} - \phi_m^{(c)} - \phi^{\max}$. **end**
9: Set $c_{nm}^\star = \arg\max_{c:m\in\mathcal{H}_n^{(c)}} \delta_{nm}(c)$.
10: **end**
11: **end**
12: **if** $t \in \mathcal{T}_\psi$ **then**
13: **for** all relay nodes $a \in \mathcal{N}\backslash\{n\}$ **do**
14: Set $\psi_n^{(a)} = \sum_{m\in\mathcal{N}_n^{\text{out}}} (h_{am}/(I_{nm} + \eta_m)) \max\{\delta_{nm}^{(c_{nm}^\star)}, 0\}$.
15: **end**
16: Inform interference prices $\boldsymbol{\psi}_n = (\psi_n^{(a)},\ \forall a \in \mathcal{N}\backslash\{n\})$ to all other nodes.
17: Set $\boldsymbol{p}_n = \mathbf{0}$.
18: **for** all out-neighbors $m \in \mathcal{N}_n^{\text{out}}$ **do**
19: **if** $\sum_{m\in\mathcal{N}_n^{\text{out}}} \delta_{nm}^{(c_{nm}^\star)} \le P_n^{\max}(\sum_{a\in\mathcal{N}\backslash\{n\}} \psi_a^{(n)})$ **then**
20: Set $p_{nm}^{(c)} = \delta_{nm}^{(c_{nm}^\star)} / \left(\sum_{a\in\mathcal{N}\backslash\{n\}} \psi_a^{(n)}\right)$.
21: **else**
22: Set $p_{nm}^{(c)} = \left(\delta_{nm}^{(c_{nm}^\star)} P_n^{\max}\right) / \left(\sum_{a\in\mathcal{N}_n^{\text{out}}} \delta_{na}^{(c_{na}^\star)}\right)$.
23: **end**
24: **end**
25: **end**
26: Charge any node $m \in \mathcal{N}_n^{\text{in}}$ and any $i \in \mathcal{D}_n$ for relaying commodity $c \in \mathcal{N}\backslash\{n\}$ data at price $\phi_n^{(c)}$.
27: Pay any node $m \in \mathcal{N}_n^{\text{out}}$ for relaying commodity $c \in \mathcal{N}\backslash\{n\}$ data at price $\phi_m^{(c)}$.
28: Charge any other node $a \in \mathcal{N}\backslash\{n\}$ for the interference it causes on node n at price $\psi_n^{(a)}$.
29: Pay any other node $a \in \mathcal{N}\backslash\{n\}$ for the interference node n causes on it at price $\psi_a^{(n)}$.
30: **until** the wireless relay node n switches off.

Algorithm 11 *Executed by each wireless user $i \in \mathcal{D}_n$*

1: **repeat**
2: Set the transmission rates $\boldsymbol{r}_i$ by solving the convex optimization problem in (8.11).
3: Pay wireless relay node n for relaying any commodity $c \in \mathcal{N}\backslash\{n\}$ data at price $\phi_n^{(c)}$.
4: **until** the wireless user i switches off or leaves the network.

8.3 Simulation Results

In this section, we evaluate the performance of our proposed TFP scheme and compare it with a recently proposed SFP scheme in [126], where only the relay prices are taken into account and the network is assumed to be interference-free. We consider three performance metrics: 1) network social welfare, 2) fairness index, and 3) aggregate throughput. The fairness index is calculated among the profits that the wireless relay nodes achieve [89]: $\frac{\left(\sum_{n\in\mathcal{N}}\sum_{t=1}^{T}\chi_n(t)\right)^2}{N\sum_{n\in\mathcal{N}}\left(\sum_{t=1}^{T}\chi_n(t)\right)^2}$, where $T = 5000$ is the simulation time. Each wireless relay node n provides the connectivity for $|\mathcal{D}_n| = 5$ wireless users. Each wireless user is interested in sending data to two other (randomly selected) users inside the network. We consider ten different *random* topologies. In each topology, the wireless relay nodes are randomly located in a 1 km $\times$ 1 km square field and the communication range is 200 m. There is a link between any two neighboring wireless relay nodes if they are within the communication range of each other. For each wireless relay node $n \in \mathcal{N}$, we have: $P_n^{\max} = 20$ W and $R_n^{\max} = 100$ kbps. The transmission power, relay prices, and the interference prices are set according to (8.17), (8.23), and (8.24), respectively. The unit of currency is selected such that for a unit queue backlog, relaying 1 Mbps data costs 1 *cent*, i.e., 0.01 *dollar*. Unless stated otherwise, we assume the presence of slow-fading channels with the fading parameter $\Lambda = 10$. The impact of fast-fading is also studied in Section 8.3.3.

8.3.1 Performance Comparison with Single-Fold Pricing

The network social welfare and throughput, when the number of relay nodes N varies from 10 to 50, are shown in Fig. 8.2 (a) and (b), respectively. Each point is the average of the measurements for all 10 simulated topologies. We can see that our proposed TFP always outperforms the SFP scheme in [126], in terms of both throughput and social welfare. The performance gain increases as N increases. It is because our interference pricing scheme better leverages optimal resource allocation. As the number of nodes increases, more nodes interfere with each other's transmissions and optimal interference control becomes crucial. In fact, as Theorem 30 shows, TFP leads to *maximum* social welfare. Considering the case where $N = 50$, TFP results in 24.6% higher social welfare and 14.7% higher throughput compared to SFP.

The exact value of the social welfare, fairness index, and throughput for each of the 10 simulated topologies, where $N = 50$, are shown in Fig. 8.3 (a), (b), and (c), respectively. From Fig. 8.3 (a) and (c), TFP always results in higher social welfare as well as higher throughput compared to SFP. From Fig. 8.3 (b), TFP also always acts more fair. Recall from Theorem 29 that TFP guarantees high positive profits for all relay nodes. In fact, having the interference prices helps those relay nodes that do not experience high traffic demand. Instead, they make some money out of the interference charges. This results in a more fair revenue sharing among the nodes. On average, TFP leads to 18.3% higher fairness index compared to SFP.

8.3.2 Maximum Weight Matching

At each time slot $t \in \mathcal{T}$, we define:

$$\Theta(t) = \sum_{n \in \mathcal{N}} \sum_{c \in \mathcal{N} \setminus \{n\}} \sum_{m \in \mathcal{N}_n^{\text{out}}} \mu_{nm}^{(c)}(\boldsymbol{p}(t), \boldsymbol{\omega}(t))\, \delta_{nm}^{(c)}(t'), \tag{8.25}$$

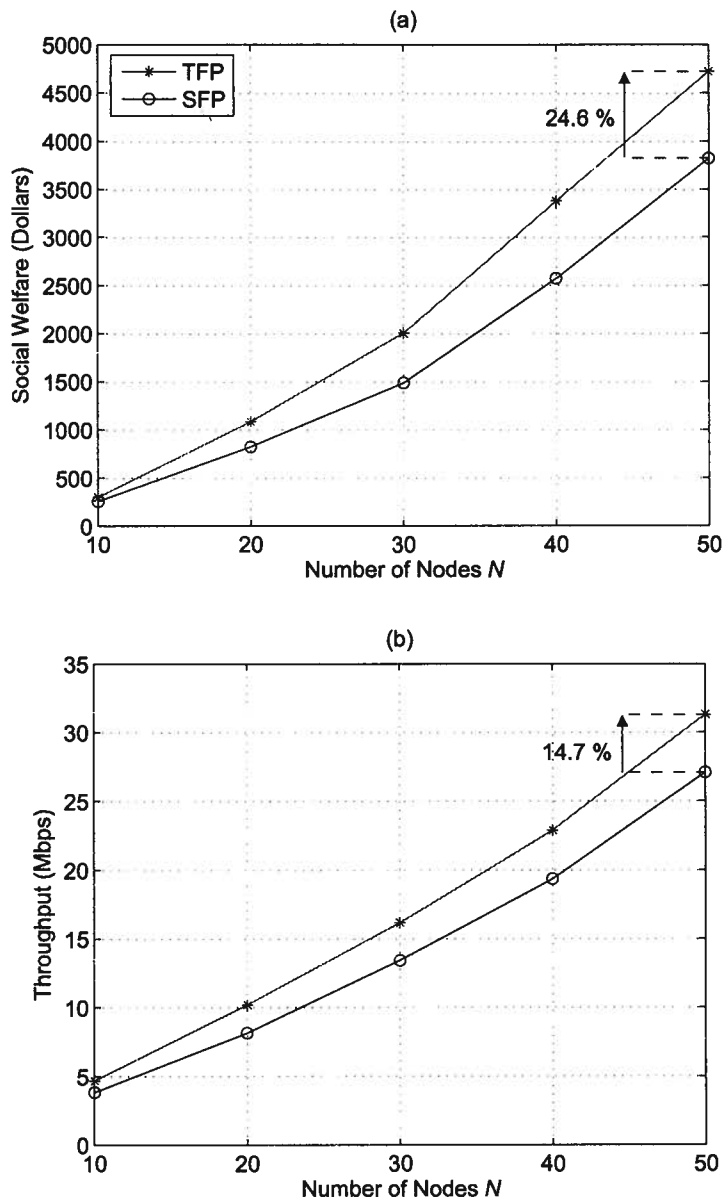


Figure 8.2: Network social welfare and aggregate throughput when the number of wireless relay nodes N varies from 10 to 50. Each wireless relay node provides network connectivity for 5 wireless users. Each point is the average of the measurements for all ten topologies: (a) Network social welfare, (b) Aggregate network throughput.

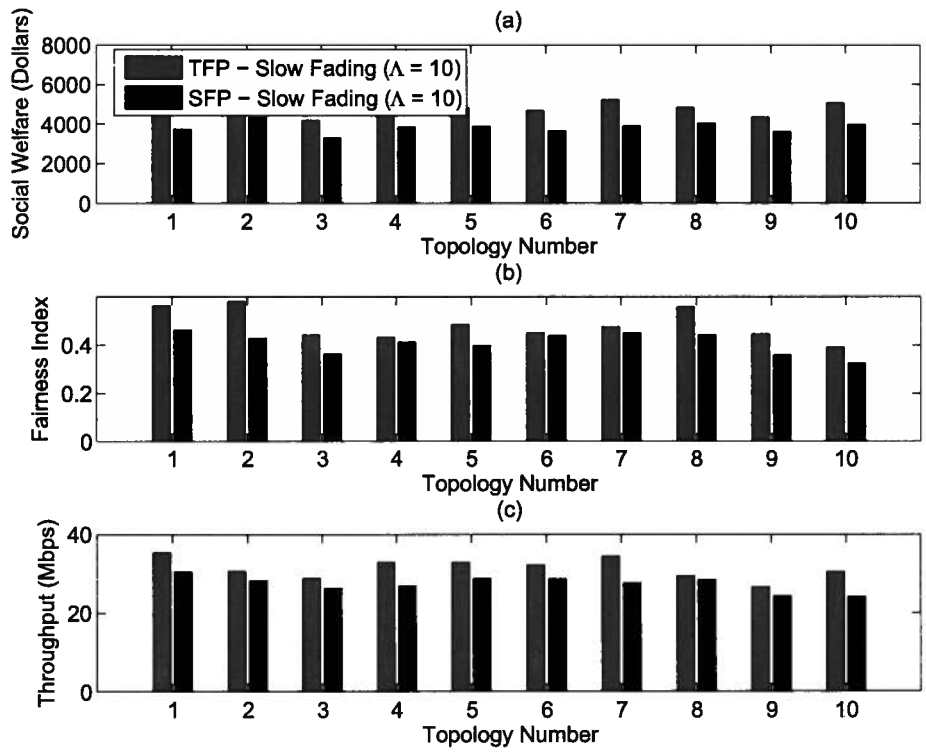


Figure 8.3: Simulation results for each of the 10 random simulated topologies when $N = 50$ and the communication channels experience slow-fading (i.e., $\Lambda = 10$): (a) Network social welfare, (b) Fairness index among the profits achieved by wireless relay nodes, and (c) Aggregate network throughput.

where t' is the smallest time slot in set $\mathcal{T}_\omega$ such that: $t \geq t'$. In other words, t' is the most recent time slot at which the vector of channel states $\boldsymbol{\omega}$ has changed. We notice that, $\Theta(t)$ is indeed the same as the objective function in the maximum weight matching problem in (8.46). From Theorem 30, TFP results in maximum network social welfare and maximum network utility by *periodically* solving optimization problem (8.46), i.e., maximizing the values of $\Theta(t)$. This is illustrated in Fig. 8.4. In this figure, the trend of $\Theta(t)$ for topology number 1 is shown versus the time slots. Notice that, the fading parameter $\Lambda = 10$. Thus, the vector of channel states $\boldsymbol{\omega}$ randomly changes every 10 time slots. This implies that the optimal solution of the maximum

weight matching problem also changes every 10 time slots. For proper operation, TPF needs to converge to the *new* optimal solution accordingly. This is shown in the zoomed area in Fig. 8.4 (b). Clearly, the convergence is fast. From the results in Fig. 8.4 (a) and (b), we can also see that TFP always results in substantially higher maximum weight matching objective $\Theta(t)$, compared to SFP. The higher the maximum weight matching objective, the higher is the aggregate network utility [132]. From Lemma 7, this also implies higher network social welfare.

8.3.3 Impact of Fast-Fading

In the previous experiments, we assumed that the channels experience slow-fading. In this section, we study the impact of fast-fading. Results for all the ten simulated topologies, where the number of wireless relay nodes $N = 50$ and the fading parameter $\Lambda = 2$, are shown in Fig. 8.5. In this scenario, the channel states randomly change every 2 time slots. As a result, our proposed distributed transmission power adjustment mechanism (see lines 22 to 29 in Algorithm 10) does *not* have enough time to converge to the new optimal solution of the maximum weight matching problem after each change in the channel states. Thus, the optimal performance is not always achieved. Nevertheless, from the results in Fig. 8.5 (a) and 8.5 (c), TFP still results in 46.3% higher social welfare and 32.4% higher throughput, compared to SFP. On the other hand, from Fig. 8.5 (b), TFP is 35.2% more fair compared to SFP in this scenario.

In summary, assuming the presence of slow-fading channels, our proposed TFP scheme leads to not only higher aggregate profit across the nodes and users, but also more fair revenue distribution among the relay nodes. It also results in significantly higher network throughput. When the underlying communication channels experience fast-fading, although TFP still results in substantially better performance compared to SFP, optimality may not be always guaranteed.

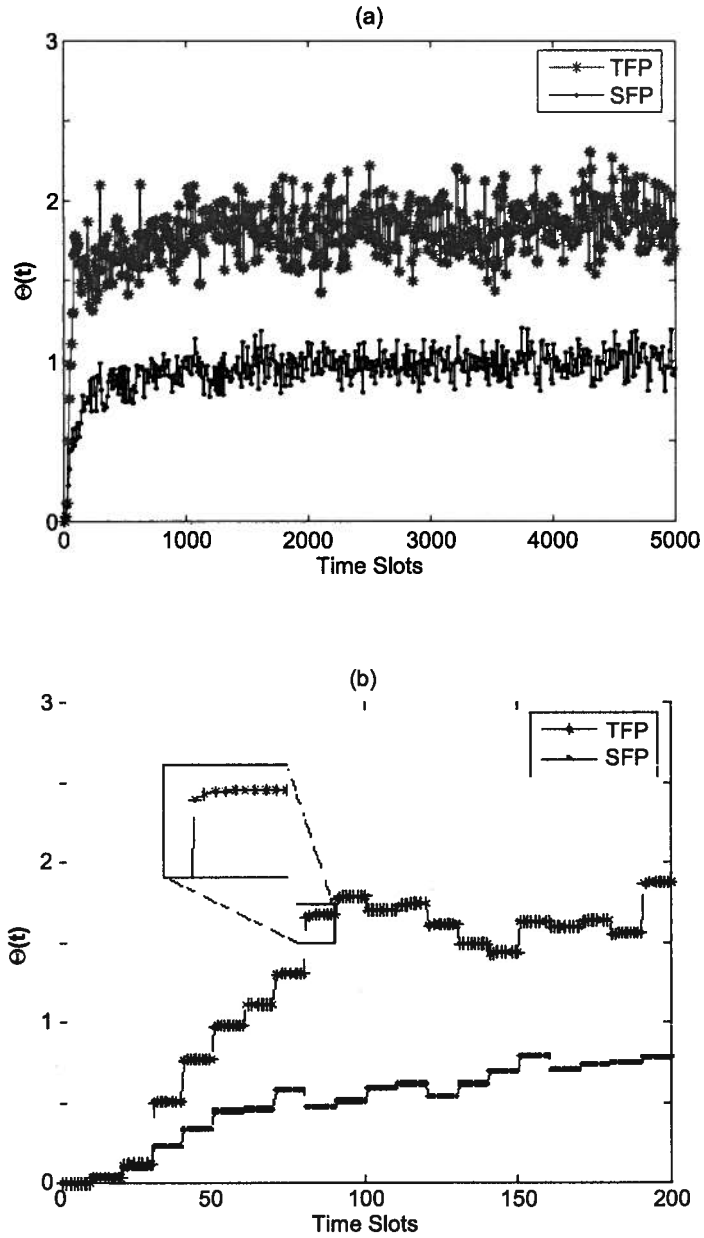


Figure 8.4: Trend of the maximum weight matching objective $\Theta(t)$ versus time slots: (a) During the whole simulation time, i.e., 5000 time slots, (b) During the first 200 time slots. Notice that every $\Upsilon = \Lambda = 10$ time slots, the channel states change randomly and the maximum weight matching objective converges to its new optimal value accordingly.

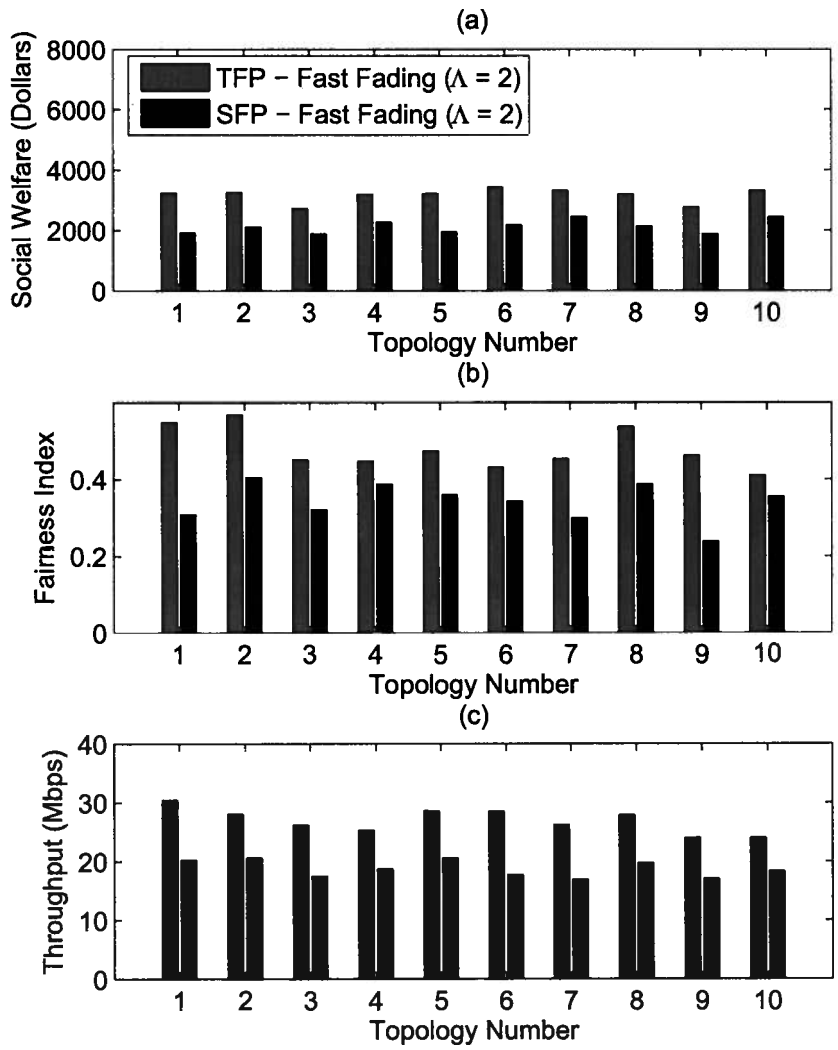


Figure 8.5: Simulation results for each of the 10 random simulated topologies when $N = 50$ and the communication channels experience fast-fading (i.e., $\Lambda = 2$): (a) Network social welfare, (b) Fairness index among the profits achieved by wireless relay nodes, and (c) Aggregate network throughput.

8.4 Summary

In this chapter, we proposed a market-based wireless access network model with two-fold pricing (TFP), where several self-interested wireless relay nodes provide the connectivity service for wireless users. The relay-prices are used as incentives to encourage nodes to collaborate and forward each other's packets. The interference-prices are also used to leverage optimal resource allocation. Together, the relay and interference prices incorporate both cooperative and competitive interactions among the nodes. The positive profit for each individual wireless relay node is guaranteed for a wide range of pricing functions. Assuming that CDMA technology is being used for transmissions, the relay and interference pricing functions are then determined to also maximize the network social welfare and aggregate network utility. Compared with the single-fold price (SFP) scheme in [126], where only the relay prices are taken into account, simulation results show that TFP increases the network social welfare and aggregate throughput by 24.6% and 14.7%, respectively. TFP also leads to significantly more fair revenue sharing and profit distribution among the wireless relay nodes with 18.3% higher fairness index.

For future work, we plan to extend our model to other multiple access technologies, such as the contention-based medium access control scheme in the IEEE 802.11 distributed coordination function. We shall also consider the effect of user mobility and roaming.

8.5 Analytical Proofs

8.5.1 Proof of Theorem 28

Knowing that all the constraints are linear and the objective function is strictly concave, problem (8.14) is a *convex* optimization problem. Therefore, it has a *unique* local (thus global) optimal

solution. In a high SINR regime, the optimal solution should satisfy the following *necessary* and *sufficient* Karush-Kuhn-Tucker (KKT) optimality conditions [15, Proposition 3.3.1]:

$$\delta_{nm}^{(c_{nm}^{\star}(t))}(t)/p_{nm}^{\star(c_{nm}^{\star}(t))}(t)-\sum_{a\in\mathcal{N}\setminus\{n\}}\psi_a^{(n)}(t)=\lambda_n^{\star}-\sigma_{nm}^{\star},\ \forall\, m\in\mathcal{N}_n^{\text{out}}, \tag{8.26}$$

$$\lambda_n^{\star}\left(\sum_{m\in\mathcal{N}_n^{\text{out}}}p_{nm}^{\star(c_{nm}^{\star}(t))}(t)-P_n^{\max}\right)=0, \tag{8.27}$$

$$\sigma_{nm}^{\star}p_{nm}^{\star(c_{nm}^{\star}(t))}(t)=0, \qquad \forall\, m\in\mathcal{N}_n^{\text{out}}, \tag{8.28}$$

$$\lambda_n^{\star}\geq 0,\quad \sigma_{nm}^{\star}\geq 0, \qquad \forall\, m\in\mathcal{N}_n^{\text{out}}, \tag{8.29}$$

where $\lambda_n^{\star}$ denotes the Lagrange multiplier corresponding to constraint $\sum_{m\in\mathcal{N}_n^{\text{out}}}p_{nm}^{\star(c_{nm}^{\star}(t))}(t)\leq P_n^{\max}$ and $\sigma_{nm}^{\star}$ denotes the Lagrange multiplier corresponding to constraint $p_{nm}^{\star(c_{nm}^{\star}(t))}(t)\geq 0$ for each wireless link $(n,m)\in\mathcal{L}_n^{\text{out}}$. We can show that if

$$\sum_{m\in\mathcal{N}_n^{\text{out}}}\delta_{nm}^{(c_{nm}^{\star}(t))}(t)<P_n^{\max}\left(\sum_{a\in\mathcal{N}\setminus\{n\}}\psi_a^{(n)}(t)\right), \tag{8.30}$$

then the KKT conditions (8.26)-(8.29) are satisfied by setting $\lambda_n^{\star}=0$ and $\sigma_{nm}^{\star}=0$ for all links $(n,m)\in\mathcal{L}_n^{\text{out}}$. In this case, for each link $(n,m)\in\mathcal{L}_n^{\text{out}}$ and any commodity $c=c_{nm}^{\star}(t)$ such that $\delta_{nm}^{(c)}(t)>0$, we have $p_{nm}^{\star(c)}(t)=\delta_{nm}^{(c_{nm}^{\star}(t))}/(\sum_{a\in\mathcal{N}\setminus\{n\}}\psi_a^{(n)}(t))$. On the other hand, if

$$\sum_{m\in\mathcal{N}_n^{\text{out}}}\delta_{nm}^{(c_{nm}^{\star}(t))}(t)\geq P_n^{\max}\left(\sum_{a\in\mathcal{N}\setminus\{n\}}\psi_a^{(n)}(t)\right), \tag{8.31}$$

then the KKT conditions are satisfied by setting $\sigma_{nm}^{\star}=0$ for all links $(n,m)\in\mathcal{L}_n^{\text{out}}$ and

$$\lambda_n^{\star}=\left(\sum_{m\in\mathcal{N}_n^{\text{out}}}\delta_{nm}^{(c_{nm}^{\star}(t))}(t)-P_n^{\max}\left(\sum_{a\in\mathcal{N}\setminus\{n\}}\psi_a^{(n)}(t)\right)\right)/P_n^{\max}\overset{\text{by (8.31)}}{\geq}0. \tag{8.32}$$

Notice that, in this case, for each link $(n,m)\in\mathcal{L}_n^{\text{out}}$ and any commodity $c=c_{nm}^{\star}(t)$ such that $\delta_{nm}^{(c)}(t)>0$, we have $p_{nm}^{\star(c)}(t)=\delta_{nm}^{(c_{nm}^{\star}(t))}(t)\,P_n^{\max}/\left(\sum_{a\in\mathcal{N}_n^{\text{out}}}\delta_{na}^{(c_{na}^{\star}(t))}(t)\right)$. Thus, since $\boldsymbol{p}_n(t)$ is the only point that satisfies the KKT conditions in (8.26)-(8.29), it is indeed the unique global optimal solution for the local transmission power control problem in (8.14). ■

8.5.2 Proof of Theorem 29

From (8.13) and (8.18), for any node $n \in \mathcal{N}$ and at any time $t \in \mathcal{T}$, we have:

$$\begin{aligned}
\chi_n(t) &= \chi_n(t) + \textstyle\sum_{c\in\mathcal{N}\setminus\{n\}} \sum_{m\in\mathcal{N}_n^{\text{out}}} \mu_{nm}^{(c)}(\boldsymbol{p}(t),\boldsymbol{\omega}(t))\delta_{nm}^{(c)}(t) \\
&\quad - \textstyle\sum_{c\in\mathcal{N}\setminus\{n\}} \sum_{m\in\mathcal{N}_n^{\text{out}}} \mu_{nm}^{(c)}(\boldsymbol{p}(t),\boldsymbol{\omega}(t)) \left(\phi_n^{(c)}(t) - \phi_m^{(c)}(t) - \phi^{\max}\right) \\
&= \Big[\textstyle\sum_{c\in\mathcal{N}\setminus\{n\}} \sum_{m\in\mathcal{N}_n^{\text{out}}} \mu_{nm}^{(c)}(\boldsymbol{p}(t),\boldsymbol{\omega}(t))\, \delta_{nm}^{(c)}(t) \\
&\quad - \textstyle\sum_{a\in\mathcal{N}\setminus\{n\}} \psi_a^{(n)}(t) \sum_{c\in\mathcal{N}\setminus\{n\}} \sum_{m\in\mathcal{N}_n^{\text{out}}} p_{nm}^{(c)}(t) \Big] \\
&\quad + \Big[\textstyle\sum_{c\in\mathcal{N}\setminus\{n\}} \left(\sum_{i\in\mathcal{D}_n} \sum_{k\in\mathcal{D}_c} r_i^{(k)}(t) + \sum_{m\in\mathcal{N}_n^{\text{in}}} \mu_{mn}^{(c)}(\boldsymbol{p}(t),\boldsymbol{\omega}(t)) \right) \phi_n^{(c)}(t) \\
&\quad - \textstyle\sum_{c\in\mathcal{N}\setminus\{n\}} \sum_{m\in\mathcal{N}_n^{\text{out}}} \mu_{nm}^{(c)}(\boldsymbol{p}(t),\boldsymbol{\omega}(t)) \left(\phi_n^{(c)}(t) - \phi^{\max}\right) \Big] \\
&\quad + \textstyle\sum_{a\in\mathcal{N}\setminus\{n\}} \psi_n^{(m)}(t) \sum_{d\in\mathcal{N}\setminus\{a\}} \sum_{b\in\mathcal{N}_a^{\text{out}}} p_{ab}^{(d)}(t).
\end{aligned} \tag{8.33}$$

Since the optimal objective function in (8.14) is non-negative,

$$\sum_{c\in\mathcal{N}\setminus\{n\}} \sum_{m\in\mathcal{N}_n^{\text{out}}} \mu_{nm}^{(c)}(\boldsymbol{p}(t),\boldsymbol{\omega}(t))\, \delta_{nm}^{(c)}(t) - \sum_{a\in\mathcal{N}\setminus\{n\}} \psi_a^{(n)}(t) \sum_{c\in\mathcal{N}\setminus\{n\}} \sum_{m\in\mathcal{N}_n^{\text{out}}} p_{nm}^{(c)}(t) \geq 0. \tag{8.34}$$

Following the proof of [126, Theorem 2b], we can also show that:

$$\begin{aligned}
&\textstyle\sum_{c\in\mathcal{N}\setminus\{n\}} \left(\sum_{i\in\mathcal{D}_n} \sum_{k\in\mathcal{D}_c} r_i^{(k)}(t) + \sum_{m\in\mathcal{N}_n^{\text{in}}} \mu_{mn}^{(c)}(\boldsymbol{p}(t),\boldsymbol{\omega}(t)) \right) \phi_n^{(c)}(t) \\
&- \textstyle\sum_{c\in\mathcal{N}\setminus\{n\}} \sum_{m\in\mathcal{N}_n^{\text{out}}} \mu_{nm}^{(c)}(\boldsymbol{p}(t),\boldsymbol{\omega}(t)) \left(\phi_n^{(c)}(t) - \phi^{\max}\right) \geq 0.
\end{aligned} \tag{8.35}$$

By replacing (8.34) and (8.35) in (21) we have:

$$\chi_n(t) \geq \textstyle\sum_{a\in\mathcal{N}\setminus\{n\}} \psi_n^{(m)}(t) \sum_{d\in\mathcal{N}\setminus\{a\}} \sum_{b\in\mathcal{N}_a^{\text{out}}} p_{ab}^{(d)}(t). \tag{8.36}$$

Adding up both sides for $t = 1, \ldots, T$, the inequality in (8.19) results. ■

8.5.3 Proof of Lemma 7

Replacing $\vartheta_i(t)$ and $\chi_n(t)$ in (8.20) by (8.12) and (8.18), respectively, we have:

$$
\begin{aligned}
&\textstyle\sum_{t=1}^{T}\sum_{n\in\mathcal{N}}\chi_n(t)+\sum_{t=1}^{T}\sum_{i\in\mathcal{D}}\vartheta_i(t)\\
&\quad=\textstyle\sum_{t=1}^{T}\sum_{n\in\mathcal{N}}\Big(\sum_{c\in\mathcal{N}\setminus\{n\}}\sum_{i\in\mathcal{D}_n}\sum_{k\in\mathcal{D}_c}r_i^{(k)}(t)\phi_n^{(c)}(t)\\
&\qquad+\textstyle\sum_{m\in\mathcal{N}_n^{\text{in}}}\sum_{c\in\mathcal{N}\setminus\{n\}}\mu_{mn}^{(c)}(\boldsymbol{p}(t),\boldsymbol{w}(t))\phi_n^{(c)}(t)-\sum_{m\in\mathcal{N}_n^{\text{out}}}\sum_{c\in\mathcal{N}\setminus\{n\}}\mu_{nm}^{(c)}(\boldsymbol{p}(t),\boldsymbol{w}(t))\phi_m^{(c)}(t)\\
&\qquad+\textstyle\sum_{a\in\mathcal{N}\setminus\{n\}}\psi_n^{(a)}(t)\sum_{d\in\mathcal{N}\setminus\{a\}}\sum_{b\in\mathcal{N}_a^{\text{out}}}p_{ab}^{(d)}(t)-\sum_{a\in\mathcal{N}\setminus\{n\}}\psi_a^{(n)}(t)\sum_{c\in\mathcal{N}\setminus\{n\}}\sum_{m\in\mathcal{N}_n^{\text{out}}}p_{nm}^{(c)}(t)\Big)\\
&\qquad+\textstyle\sum_{t=1}^{T}\sum_{n\in\mathcal{N}}\sum_{i\in\mathcal{D}_n}\Big(\sum_{c\in\mathcal{N}\setminus\{n\}}\sum_{k\in\mathcal{D}_c}g_i^{(k)}(r_i^{(k)}(t))-\sum_{n\in\mathcal{N}\setminus\{n\}}\sum_{k\in\mathcal{D}_c}r_i^{(k)}(t)\phi_n^{(c)}(t)\Big)\\
&\quad=\textstyle\sum_{t=1}^{T}\sum_{n\in\mathcal{N}}\Big(\sum_{c\in\mathcal{N}\setminus\{n\}}\sum_{i\in\mathcal{D}_n}\sum_{k\in\mathcal{D}_c}r_i^{(k)}(t)\phi_n^{(c)}(t)-r_i^{(k)}(t)\phi_n^{(c)}(t)\Big)\\
&\qquad+\textstyle\sum_{t=1}^{T}\Big(\Big(\sum_{n\in\mathcal{N}}\sum_{c\in\mathcal{N}\setminus\{n\}}\sum_{m\in\mathcal{N}_n^{\text{in}}}\mu_{mn}^{(c)}(\boldsymbol{p}(t),\boldsymbol{w}(t))\phi_n^{(c)}(t)\Big)\\
&\qquad\qquad-\Big(\textstyle\sum_{n\in\mathcal{N}}\sum_{c\in\mathcal{N}\setminus\{n\}}\sum_{m\in\mathcal{N}_n^{\text{out}}}\mu_{nm}^{(c)}(\boldsymbol{p}(t),\boldsymbol{w}(t))\phi_m^{(c)}(t)\Big)\Big)\\
&\qquad+\textstyle\sum_{t=1}^{T}\Big(\Big(\sum_{n\in\mathcal{N}}\sum_{a\in\mathcal{N}\setminus\{n\}}\psi_n^{(a)}(t)\sum_{d\in\mathcal{N}\setminus\{a\}}\sum_{b\in\mathcal{N}_a^{\text{out}}}p_{ab}^{(d)}(t)\Big)\\
&\qquad\qquad-\Big(\textstyle\sum_{n\in\mathcal{N}}\sum_{a\in\mathcal{N}\setminus\{n\}}\psi_a^{(n)}(t)\sum_{c\in\mathcal{N}\setminus\{n\}}\sum_{m\in\mathcal{N}_n^{\text{out}}}p_{nm}^{(c)}(t)\Big)\Big)\\
&\qquad+\textstyle\sum_{t=1}^{T}\sum_{n\in\mathcal{N}}\sum_{i\in\mathcal{D}_n}\sum_{c\in\mathcal{N}\setminus\{n\}}\sum_{k\in\mathcal{D}_c}g_i^{(k)}(r_i^{(k)}(t))\\
&\quad=\textstyle\sum_{t=1}^{T}\sum_{n\in\mathcal{N}}\sum_{i\in\mathcal{D}_n}\sum_{c\in\mathcal{N}\setminus\{n\}}\sum_{k\in\mathcal{D}_c}g_i^{(k)}(r_i^{(k)}(t)).
\end{aligned}
\tag{8.37}
$$

The last line in (8.37) is indeed the same as (8.21). Notice that we have:

$$
\sum_{n\in\mathcal{N}}\sum_{c\in\mathcal{N}\setminus\{n\}}\sum_{m\in\mathcal{N}_n^{\text{in}}}\mu_{mn}^{(c)}(\boldsymbol{p}(t),\boldsymbol{w}(t))\phi_n^{(c)}(t)=\sum_{n\in\mathcal{N}}\sum_{c\in\mathcal{N}\setminus\{n\}}\sum_{m\in\mathcal{N}_n^{\text{out}}}\mu_{nm}^{(c)}(\boldsymbol{p}(t),\boldsymbol{w}(t))\phi_m^{(c)}(t), \tag{8.38}
$$

and

$$
\sum_{n\in\mathcal{N}}\sum_{a\in\mathcal{N}\setminus\{n\}}\sum_{d\in\mathcal{N}\setminus\{a\}}\sum_{b\in\mathcal{N}_a^{\text{out}}}\psi_n^{(a)}(t)\,p_{ab}^{(d)}(t)=\sum_{n\in\mathcal{N}}\sum_{a\in\mathcal{N}\setminus\{n\}}\sum_{c\in\mathcal{N}\setminus\{n\}}\sum_{m\in\mathcal{N}_n^{\text{out}}}\psi_a^{(n)}(t)\,p_{nm}^{(c)}(t). \tag{8.39}
$$

In (8.38), the left hand side denotes the aggregate relay price that *all* wireless relay nodes *receive* while the right hand side denotes the aggregate relay price *all* wireless relay nodes *pay*. Similarly,

in (8.39), the left hand side is the aggregate interference price that *all* relay nodes *receive* while the right hand side is the aggregate interference price *all* relay nodes *pay*. ■

8.5.4 Proof of Theorem 30

Given $t' \in \mathcal{T}_\phi$, for each time slot $t \in \{t', \ldots, t' + \Upsilon\}$, consider two arbitrary non-negative valued transmission power vectors $\tilde{\boldsymbol{p}}(t)$ and $\hat{\boldsymbol{p}}(t)$ such that:

$$\tilde{\boldsymbol{p}}(t) \preceq \hat{\boldsymbol{p}}(t), \tag{8.40}$$

where the inequality is interpreted coordinate-wise. That is, for any wireless link $(n, m) \in \mathcal{L}$ and each commodity $c \in \mathcal{N}\backslash\{n\}$, we have $\tilde{p}_{nm}^{(c)}(t) \leq \hat{p}_{nm}^{(c)}(t)$. From (8.3), we can show that:

$$I_{nm}(\tilde{\boldsymbol{p}}_{-n}(t)) \leq I_{nm}(\hat{\boldsymbol{p}}_{-n}(t)), \qquad \forall n \in \mathcal{N},\ m \in \mathcal{N}_n^{\text{out}}, \tag{8.41}$$

$$1/(I_{nm}(\tilde{\boldsymbol{p}}_{-n}(t)) + \eta_m) \leq 1/(I_{nm}(\hat{\boldsymbol{p}}_{-n}(t)) + \eta_m), \qquad \forall n \in \mathcal{N},\ m \in \mathcal{N}_n^{\text{out}}, \tag{8.42}$$

$$\psi_n^{(a)}(\tilde{\boldsymbol{p}}(t)) \geq \psi_n^{(a)}(\hat{\boldsymbol{p}}(t)), \qquad \forall n \in \mathcal{N},\ a \in \mathcal{N}\backslash\{n\}, \tag{8.43}$$

$$1/\textstyle\sum_{a\in\mathcal{N}\backslash\{n\}} \psi_a^{(n)}(\tilde{\boldsymbol{p}}(t)) \leq 1/\sum_{a\in\mathcal{N}\backslash\{n\}} \psi_a^{(n)}(\hat{\boldsymbol{p}}(t)), \qquad \forall n \in \mathcal{N}. \tag{8.44}$$

Replacing (8.44) in (8.17), we have:

$$\tilde{\boldsymbol{p}}(t+1) \preceq \hat{\boldsymbol{p}}(t+1). \tag{8.45}$$

From (8.40) and (8.45), the update formulation in (8.17) forms a *monotone mapping* [18]. Monotone mappings satisfy both *synchronous convergence* and *box* conditions [18, pp. 431]. Thus, from the *asynchronous convergence theorem* [18], the transmission powers will converge to a fixed point, assuming that $\Upsilon = \Lambda$ is large enough. By definition, $\boldsymbol{p}^\star$ should denote the optimal solution of the local optimization problem in (8.14) for *all* relay nodes. Next, we show that $\boldsymbol{p}^\star$ also denotes

the unique optimal solution of the following *maximum weight matching* problem [131, 132]:

$$\begin{aligned} \max_{\boldsymbol{p}(t)\succeq \mathbf{0}} \quad & \textstyle\sum_{n\in\mathcal{N}} \sum_{c\in\mathcal{N}\setminus\{n\}} \sum_{m\in\mathcal{N}_n^{\text{out}}} \mu_{nm}^{(c)}(\boldsymbol{p}(t),\boldsymbol{\omega}(t))\, \delta_{nm}^{(c)}(t') \\ \text{s.t.} \quad & \textstyle\sum_{c\in\mathcal{N}\setminus\{n\}} \sum_{m\in\mathcal{N}_n^{\text{out}}} p_{nm}^{(c)}(t) \le P_n^{\max}, \qquad \forall n\in\mathcal{N} \\ & p_{nm}^{(c)}(t) = 0, \qquad \forall n\in\mathcal{N},\ c\in\mathcal{N}\setminus\{n\},\ m\notin\mathcal{H}_n^{(c)} \text{ or } c\neq c_{nm}^{\star}(t') \text{ or } \delta_{nm}^{(c)}(t')\le 0. \end{aligned} \tag{8.46}$$

Notice that the objective function in (8.46) is different from the objective function in (8.14) as it is the weighted summation of the data rates over *all* links. Problem (8.46) is indeed the key resource allocation problem to be solved by the *back-pressure* algorithms [131, 132]. Using the *logarithmic* change of variables (cf. [30, Theorem 1]), we can transform problem (8.46) to an equivalent *convex* problem. Thus, problem (8.46) has a unique optimal solution (cf. [15]). Let $\boldsymbol{p}^*$ denote the unique optimal solution of problem (8.46). From KKT conditions, we have:

$$\frac{\delta_{nm}^{(c_{nm}^{\star}(t'))}(t')}{p_{nm}^{*(c_{nm}^{\star}(t'))}(t)} - \textstyle\sum_{a\in\mathcal{N}\setminus\{n\}} \sum_{b\in\mathcal{N}_a^{\text{out}}(t)} h_{nb} \frac{\delta_{ab}^{(d_{ab}^{\star}(t'))}(t')}{I_n(\boldsymbol{p}_{-n}^*(t))+\eta_b} = \rho_n^* - \varrho_{nm}^*, \quad \forall\, n\in\mathcal{N},\ m\in\mathcal{N}_n^{\text{out}}, \tag{8.47}$$

$$\rho_n^* \left(\textstyle\sum_{m\in\mathcal{N}_n^{\text{out}}} p_{nm}^{*(c_{nm}^{\star}(t'))}(t) - P_n^{\max}\right) = 0, \qquad \forall\, n\in\mathcal{N} \tag{8.48}$$

$$\varrho_{nm}^* p_{nm}^{*(c_{nm}^{\star}(t'))}(t) = 0, \qquad \forall\, n\in\mathcal{N},\ m\in\mathcal{N}_n^{\text{out}}, \tag{8.49}$$

$$\rho_n^* \ge 0, \quad \varrho_{nm}^* \ge 0, \qquad \forall\, n\in\mathcal{N},\ m\in\mathcal{N}_n^{\text{out}}, \tag{8.50}$$

where for each node $n\in\mathcal{N}$, ρ_n^* denotes the Lagrange multiplier corresponding to constraint $\sum_{m\in\mathcal{N}_n^{\text{out}}} p_{nm}^{*(c_{nm}^{\star}(t'))}(t) \le P_n^{\max}$ and ϱ_{nm}^* denotes the Lagrange multiplier corresponding to constraint $p_{nm}^{*(c_{nm}^{\star}(t'))}(t)\ge 0$. Comparing with (8.27)-(8.29), the KKT conditions (8.48)-(8.50) hold if we set $\boldsymbol{p}^* = \boldsymbol{p}^\star$, $\rho_n^* = \lambda_n^\star$, and $\varrho_{nm}^* = \sigma_{nm}^\star$, for all nodes $n\in\mathcal{N}$ and all links $(n,m)\in\mathcal{L}_n^{\text{out}}$. In this case, since

$$\sum_{a\in\mathcal{N}\setminus\{n\}} \sum_{b\in\mathcal{N}_a^{\text{out}}} h_{nb} \frac{\delta_{ab}^{(d_{ab}^{\star}(t'))}(t')}{I_n(\boldsymbol{p}_{-n}^*)+\eta_b} \overset{\text{by (8.24)}}{=} \sum_{a\in\mathcal{N}\setminus\{n\}} \psi_a^{(n)}(t),$$

the KKT condition in (8.47) is also resulted from (8.26). Thus, $\boldsymbol{p}^\star = \boldsymbol{p}^\star$ is indeed the unique optimal solution of the maximum weight matching problem in (8.46). In other words, given the interference pricing model in (8.24), optimization problem (8.46) is solved every $\Upsilon = \Lambda$ time slots. This, together with (8.23), results in achieving maximum aggregate network utility (cf. [132] and [134, Theorem 4]). From Lemma 7, obtaining the maximum aggregate network utility also implies achieving maximum network social welfare. ■

Chapter 9

Conclusions and Future Work

9.1 Summary of Work Accomplished

In this thesis we have considered various optimal resource management problems in wireless access networks. More specifically, we have considered three problems under the general framework of network utility maximization:

- **Cross-Layer Interface Assignment and Channel Allocation for MC-WMNs**: In Chapters 2, 3, 4, and 7 of the thesis we developed practical cross-layer channel and interface assignment algorithms to optimize the network performance in MC-WMNs. It was a challenging task because of the lack of accurate and tractable capacity models and the inherent discrete properties of the channel and interface assignment problems. We first developed a linear link capacity model using the concepts of multi-channel contention graph and binary linearization (see Chapters 2, 3). We also proposed a novel continuous capacity model in the context of multi-interface multi-channel random access (see Chapter 7). Unlike the prior work, we took into account not only the orthogonal (i.e., non-overlapped) frequency channels, but also the partially overlapped channels (see Chapter 4). This specially enabled us to fully utilize the available frequency spectrum. We then formulated a family of NUM problems which could incorporate the features of the multi-channel WMNs. Using

various techniques from the optimization theory and the theory of parallel and distributed computation, we solved the formulated problems both in centralized and distributed fashions. Compared to the previous channel and interface assignment algorithms, our proposed algorithms achieve better efficiency-fairness tradeoff and have reduced computational complexity. They also closely collaborate with other network resource management algorithms such as MAC, routing, and congestion control.

- **Simple, Robust, and Optimal Random Access**: In Chapters 5 and 6, we considered the NUM problem at MAC layer when the optimization variables are the persistent probabilities of the wireless nodes. Most of the previous NUM-based random MAC algorithms have one or more of the following performance bottlenecks: (1) extensive signaling overhead, (2) requiring synchronization among wireless nodes, (3) slow convergence speed, and (4) supporting a limited range of utility functions under which the NUM is shown to be a convex problem. Thus, most of the previously proposed NUM-based random MAC algorithms are not adequate for practical implementation. Our proposed random MAC algorithm in Chapter 5 overcomes these bottlenecks in all four aspects. First, it only requires a limited message passing among the nodes. The signaling overhead is reduced by a factor of 10. Second, our algorithm is fully asynchronous. This minimizes the coordination overhead and allows maximum heterogeneity among wireless nodes in terms of computational complexity and software implementation. Furthermore, our algorithm is robust to arbitrary large but finite message passing delays as well as message loss. Third, our algorithm can achieve much faster convergence. Finally, our proposed algorithm has provable convergence, optimality, and robustness properties under a wide range of utility functions, even if the corresponding

NUM problems are non-convex. We then modified the algorithm to eliminate the need for frequent message passing in Chapter 6. In the proposed algorithm, each wireless node simply observes the contention history and accordingly assigns its persistent probabilities to maximize the NUM problem. This is of critical importance in practice, since any explicit message passing among wireless users will lead to further contentions in the network and reduce the network performance.

- **Two-Fold Pricing for Multi-Hop Wireless Access Networks**: In Chapter 8, we proposed a market-based model with two-fold pricing for wireless access networks. In our model, the relay-pricing is used to encourage nodes to forward each other's packets. That is, each node receives a payment for the relay service it provides. We also considered interference-pricing to leverage optimal resource allocation. Together, the relay and the interference prices incorporate both cooperative and competitive interactions among the nodes. We proved that two-fold pricing guarantees positive profit for each individual wireless node under a wide range of pricing functions. The profit increases as the node forwards more packets. Thus, the cooperative nodes are well rewarded. We then determined the relay and the interference pricing functions such that the network social welfare and the aggregate network utility are maximized. Simulation results show that, compared to a recently proposed single-fold pricing model where only the relay prices are considered, our proposed two-fold pricing scheme significantly increases the total network profit and the network throughput. It also leads to more fair revenue sharing and profit distribution among the wireless nodes.

9.2 Future Work

This thesis can be extended in several aspects. Here, we present overview some possible directions for future work.

- **Tackling Non-Convex Optimization Problems**: The analysis techniques we proposed in Chapter 5 are quite general and can be used to tackle various other non-convex optimization problems in communications and networking. The key idea is to break down the original problem into several local and myopic problems from each node's point of view. In many cases (as in the case for random access as well as multi-interface multi-channel random access), the local problem turns out to be convex. This implies that each node can easily solve its local resource management problem. In some cases, the closed-form solution can also be obtained as in Theorems 10 and 11. Then, we can apply several mapping analysis techniques (see Section 1.2.3) to derive the behavior of the system when each node only solves its own local and myopic optimization problem.

- **Algorithms with No Explicit Frequent Message Passing**: The novel distributed learning strategies in Chapter 6 can also be applied to other resource management problems to achieve signalling-free protocols. In particular, it is easy to extend the multi-interface multi-channel random access algorithm in Chapter 7.

- **Partially Overlapped Channel Assignment**: The mathematical formulations in Chapter 4 can be used to extend a wide range of frequency channel assignment algorithms in the literature that only use orthogonal channels. In fact, different from the general theme of this thesis which uses optimization as the main solution tool, the proposed models in

Chapter 4 can even be used for related heuristic channel assignment algorithms.

- **Channel Assignment Using Convex Optimization**: The continuous capacity model that we have provided in Chapter 7 can be used in different problems, other than the formulated NUM problem. In many cases, the local and myopic problem of each wireless node remain convex, which enables us to take the advantages of the existing wide range of convex programming tools.

- **Two-Fold Pricing for Other Access Technologies**: Here in this thesis, we limited our two-fold pricing model to the case where the CDMA technology is being used. However, our interference pricing model in Chapter 8 can easily be replaced by the corresponding models for other access technologies such as random multiple access as in Chapter 5.

Bibliography

[1] S. Low, F. Paganini, and J. Doyle, "Internet congestion control," *IEEE Control Systems Magazine*, pp. 28–43, Feb. 2002.

[2] I. Akyildiz, X. Wang, and X. Wang, "Wireless mesh networks: A survey," *Elsevier Journal on Computer Networks*, vol. 47, pp. 445–487, Mar. 2005.

[3] R. Bruno, M. Conti, and E. Gregori, "Mesh networks: Commodity multihop ad hoc networks," *IEEE Communications Magazine*, vol. 43, pp. 123–131, Mar. 2005.

[4] "Tropos Networks," http://www.tropos.com.

[5] "BelAir Networks," http://www.belairnetworks.com.

[6] "MIT Roofnet," http://pdos.csail.mit.edu/roofnet/doku.php.

[7] "Meshdynamics Inc." http://www.meshdynamics.com.

[8] "Status of Project IEEE 802.11s ESS Mesh Networking," http://grouper.ieee.org/groups/802/11/Reports/tgs_update.htm, July 2007.

[9] "IEEE 802.11a Standard," http://standards.ieee.org/getieee802/download/802.11-1999.pdf.

[10] "IEEE 802.11b Standard," http://standards.ieee.org/getieee802/download/802.11b-1999.pdf.

[11] P. Bahl, A. Adya, J. Padhye, and A. Wolman, "Reconsidering wireless systems with multiple radios," *ACM Computer Communication Review*, vol. 34, pp. 39–46, Oct. 2004.

[12] J. Kurose and K. Ross, *Computer Networking: A Top-Down Approach*, 4th ed. Addison Wesley, 2007.

[13] X. Lin, N. B. Shroff, and R. Srikant, "A tutorial on cross-layer optimization in wireless networks," *IEEE J. on Selected Areas in Communications*, vol. 24, pp. 1452–1463, Aug. 2006.

[14] S. Shakkottai, T. S. Rappaport, and P. C. Karlsson, "Cross-layer design for wireless networks," *IEEE Communications Magazine*, vol. 41, pp. 74–80, Oct. 2003.

[15] D. P. Bertsekas, *Nonlinear Programming*, 2nd ed. Athena Science, 2004.

[16] S. Boyd and L. Vandenberghe, *Convex Optimization.* Cambridge University Press, 2004.

[17] Z. Q. Luo and W. Yu, "An introduction to convex optimization for communications and signal processing," *IEEE J. on Selected Areas in Communications*, vol. 24, pp. 1426–1438, Aug. 2006.

[18] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods.* Prentice Hall, 1989.

[19] M. Chiang, S. H. Low, A. R. Calderbank, and J. C. Doyle, "Layering as optimization decomposition: A mathematical theory of network architectures," *Proceedings of IEEE*, vol. 95, pp. 255 – 312, Jan. 2007.

[20] P. E. Gill and W. Murray, *Numerical Methods for Constrained Optimization.* Academic Press, 1974.

[21] S. J. Wright, *Primal-Dual Interior-Point Methods*, 1st ed. Society for Industrial and Applied Mathematics, 1997.

[22] D. Bertsimas and J. N. Tsitsiklis, *Introduction to Linear Optimization.* Athena Scientific, 1997.

[23] S. I. Gass, *Linear Programming: Methods and Applications*, 5th ed. Dover Publications, 2003.

[24] R. Dorfman, P. A. Samuelson, and R. M. Solow, *Linear Programming and Economic Analysis*, 1st ed. Dover Publications, 1987.

[25] D. P. Bertsekas, A. Nedic, and A. Ozdaglar, *Convex Analysis and Optimization*, 1st ed. Athena Scientific, 2003.

[26] M. Chiang, "Geometric programming for communication systems," *Short monograph in Foundations and Trends in Communications and Information Theory*, vol. 2, pp. 1–154, Aug. 2005.

[27] S. Boyd, S. J. Kim, L. Vandenberghe, and A. Hassibi, "A tutorial on geometric programming," *Optimization and Engineering*, vol. 8, pp. 67–127, Mar. 2007.

[28] C. W. Tan, D. P. Palomer, and M. Chiang, "Exploring hidden convexity for flexible and robust resource allocation in cellular networks," in *Proc. of IEEE INFOCOM*, Anchorage, Alaska, May 2007.

[29] L. Xiao, M. Johansson, and S. P. Boyd, "Simultaneous routing and resource allocation via dual decomposition," *IEEE Trans. on Communications*, vol. 52, pp. 1136–1144, July 2004.

[30] M. Chiang, "To layer or not to layer: Balancing transport and physical layers in wireless multihop networks," in *Proc. of IEEE INFOCOM*, Hong Kong, China, Mar. 2004.

[31] "ILOG CPLEX." http://www.ilog.com/products/cplex/, 2007.

[32] "MOSEK." http://www.mosek.com, 2007.

[33] A. H. Land and A. G. Doig, "An automatic method for solving discrete programming problems," *Econometrica*, vol. 28, pp. 497–520, 1960.

[34] H. Taha, *Operations Research: An introduction*, 7th ed. Prentice Hall, 2003.

[35] L. J. Watters, "Reduction of integer polynomial programming problems to zero-one linear programming problems," *Operations Research*, vol. 15, pp. 1171 – 1174, Nov. 1967.

[36] F. Glover, "Improved linear integer programming formulations of nonlinear integer problems," *Management Science*, vol. 22, pp. 455 – 460, Dec. 1975.

[37] C. Chang and C. Chang, "A linearization method for mixed 0-1 polynomial programs," *Computers and Operations Research*, vol. 27, pp. 1005 – 1016, Sept. 2000.

[38] C. C. Peterson, "A note on transforming the product of variables to linear form in linear programs," Working Paper, Purdue University, 1971.

[39] F. Glover and E. Woolsey, "Further reduction of zero-one polynomial programming problems to zero-one linear programming," *Operations Research*, vol. 21, pp. 156 – 161, Jan. 1973.

[40] ——, "Converting the 0-1 polynomial programming problem to a 0-1 linear problem," *Operations Research*, vol. 22, pp. 180 – 182, Jan. 1974.

[41] W. Adams and H. Sherali, "A tight linearization and an algorithm for zero-one quadratic programming problems," *Management Science*, vol. 32, pp. 1274 – 1290, Oct. 1986.

[42] ——, "Linearization strategies for a class of zero-one mixed integer programming problems," *Operations Research*, vol. 38, pp. 217 – 226, Mar. 1990.

[43] M. Oral and O. Kettani, "A linearization procedure for quadratic and cubic mixed-integer problems," *Operations Research*, vol. 40, pp. 109 – 116, Jan. 1992.

[44] F. P. Kelly, "Charging and rate control for elastic traffic," *European Trans. on Telecommunication*, vol. 8, pp. 33–37, Jan. 1997.

[45] P. C. Fishburn, *Utility Theory for Decision Making*, 1st ed. Robert E. Krieger Publishing Co, 1970.

[46] S. Low and D. Lapsley, "Optimization flow control, I: Basic algorithms and convergence," *IEEE/ACM Trans. on Networking*, vol. 7, pp. 861–874, Dec. 1999.

[47] L. Chen, S. Low, and J. Doyle, "Joint congestion control and media access control design for ad hoc wireless networks," in *Proc. of IEEE INFOCOM*, Miami, FL, Mar. 2005.

[48] L. Chen, S. Low, M. Chiang, and J. Doyle, "Optimal cross-layer congestion control, routing and scheduling design in ad hoc wireless networks," in *Proc. of IEEE INFOCOM*, Barcelona, Spain, Apr. 2006.

[49] M. Chiang, "Balancing transport and physical layers in wireless multihop networks: Jointly optimal congestion control and power control," *IEEE J. on Selected Areas in Communications*, vol. 23, pp. 104 – 116, Jan. 2005.

[50] J. Mo and J. Walrand, "Fair end-to-end window-based congestion control," *IEEE/ACM Trans. on Networking*, vol. 8, pp. 556–567, Oct. 2000.

[51] X. Wang and K. Kar, "Cross-layer rate control for end-to-end proportional fairness in wireless networks with random access," *IEEE J. on Selected Areas in Comm.*, vol. 24, pp. 1548–1559, Aug. 2006.

[52] J. Wang, L. Li, S. Low, and J. Doyle, "Cross-layer optimization in TCP/IP networks," *IEEE/ACM Trans. on Networking*, vol. 13, pp. 582 – 595, June 2005.

[53] Z. Fang and B. Bensaou, "Fair bandwidth sharing algorithm based on game theory framework for wireless ad-hoc networks," in *Proc. of IEEE INFOCOM*, Hong Kong, China, Mar. 2004.

[54] J. Huang, R. A. Berry, and M. L. Honig, "Distributed interference compensation for wireless networks," *IEEE J. on Selected Areas in Communications*, vol. 24, pp. 1074–1084, May 2006.

[55] X. Huang and B. Bensaou, "On max-min fairness and scheduling in wireless ad-hoc networks: Analytical framework and implementation," in *Proc. of ACM MobiHoc*, Long Beach, CA, Oct. 2001.

[56] X. Wang and K. Kar, "Distributed algorithms for max-min fair rate allocation in aloha networks," in *Proc. of Annual Allerton Conference on Communication, Control, and Computing*, Urbana-Champaign, IL, Sept. 2004.

[57] Z. Fang and B. Bensaou, "Design and implementation of a MAC scheme for wireless mesh networks based on a cooperative game framework," in *Proc. of IEEE ICC*, Paris, France, June 2004.

[58] F. P. Kelly, A. K. Maulloo, and D. K. H. Tan, "Rate control in communication networks: shadow prices, proportional fairness and stability," *Journal of the Operational Research Society*, vol. 49, pp. 237–252, Mar. 1998.

[59] D. P. Palomer and M. Chiang, "A tutorial on decomposition methods for network utility maximization," *IEEE J. on Selected Areas in Communications*, vol. 24, pp. 1439–1451, Aug. 2006.

[60] B. Johansson, P. Soldati, and M. Johansson, "Mathematical decomposition techniques fir distributed cross-layer optimization of data networks," *IEEE J. on Selected Areas in Communications*, vol. 24, pp. 1535–1547, Aug. 2006.

[61] A. Raniwala and T. Chiueh, "Architecture and algorithms for an IEEE 802.11-based multi-channel wireless mesh network," in *Proc. of IEEE INFOCOM*, Miami, Florida, Mar. 2005.

[62] M. Marina and S. Das, "A topology control approach for utilizing multiple channels in multi-radio wireless mesh networks," in *Proc. of Broadnet*, Boston, MA, Oct. 2005.

[63] "IEEE 802.1D Standard - MAC Bridges," http://standards.ieee.org/getieee802/download/802.1D-1998.pdf.

[64] K. Jain, J. Padhye, V. Padmanabhan, and L. Qiu, "Impact of interference on multi-hop wireless network performance," in *Proc. of ACM MobiCom*, San Diego, CA, Sept. 2003.

[65] J. Tang, G. Xue, and W. Zhang, "Interference-aware topology control and QoS routing in multi-channel wireless mesh networks," in *Proc. of ACM MobiHoc*, Urbana-Champaign, IL, May 2005.

[66] A. Raniwala, K. Gopalan, and T. Chiueh, "Centralized channel assignment and routing algorithms for multi-channel wireless mesh networks," *ACM Mobile Computing and Communications Review*, vol. 8, pp. 50–65, Apr. 2004.

[67] M. Alicherry, R. Bhatia, and L. Li, "Joint channel assignment and routing for throughput optimization in multi-radio wireless mesh networks," in *Proc. of ACM MobiCom*, Cologne, Germany, Sept. 2005.

[68] A. Brezezinski, G. Zussman, and E. Modiano, "Enabling distributed throughput maximization in wireless mesh networks - a partitioning approach," in *Proc. of ACM MobiCom*, Los Angeles, CA, Sept. 2006.

[69] B. Ramin, "Channel allocation in 802.11-based mesh networks," in *Proc. of IEEE INFOCOM*, Barcelona, Spain, Apr. 2006.

[70] A. K. Das, H. M. K. Alazemi, R. Vijayakumar, and S. Roy, "Optimization models for fixed channel assignment in wireless mesh networks with multiple radios," in *Proc. of IEEE SECON*, Santa Clara, CA, Sept. 2005.

[71] Y. Y. Chen, S. C. Liu, and C. Chen, "Channel assignment and routing for multi-channel wireless mesh networks using simulated annealing," in *Proc. of IEEE Globecom*, San Francisco, CA, Nov. 2006.

[72] A. K. Das, R. Vijayakumar, and S. Roy, "Static channel assignment in multi-radio multi-channel 802.11 wireless mesh networks: Issues, metrics, and algorithms," in *Proc. of IEEE Globecom*, San Francisco, CA, Nov. 2006.

[73] R. Draves, J. Padhye, and B. Zill, "Routing in multi-radio, multi-hop wireless mesh networks," in *Proc. of ACM MobiCom*, Philadelphia, Pennsylvania, Sept. 2004.

[74] P. Bahl, R. Chandra, and J. Dunagan, "SSCH: slotted seeded channel hopping for capacity improvement in IEEE 802.11 ad hoc wireless networks," in *Proc. of ACM MobiCom*, Philadelphia, PA, Oct. 2004.

[75] V. Bhandari and N. Vaidya, "Capacity of multi-channel wireless networks with random (c, f) assignment," in *Proc. of ACM MobiHoc*, Montreal, Canada, Sept. 2007.

[76] M. Kodialam and T. Nandagopal, "Characterizing the capacity region in multi-radio multi-channel wireless mesh networks," in *Proc. of ACM MobiCom*, Cologne, Germany, Sept. 2005.

[77] A. H. Mohsenian Rad and V. W. S. Wong, "Joint optimal channel assignment and congestion control in multi-radio wireless mesh networks," in *Proc. of IEEE ICC*, Istanbul, Turkey, June 2006.

[78] X. Meng, K. Tan, and Q. Zhang, "Joint routing and channel assignment in multi-radio wireless mesh networks," in *Proc. of IEEE ICC*, Istanbul, Turkey, June 2006.

[79] P. Kyasanur and N. Vaidya, "Routing and interface assignment in multi-channel multi-interface wireless networks," in *Proc. of IEEE WCNC*, New Orleans, LA, Mar. 2005.

[80] D. P. Bertsekas and R. Gallager, *Data Communications*, 2nd ed. Prentice Hall, 1992.

[81] S. Iyer, S. Bhattacharyya, N. Taft, and C. Diot, "An approach to alleviate link overload as observed on an IP backbone," in *Proc. of IEEE INFOCOM*, San Francisco, CA, Apr. 2003.

[82] K. Papagiannaki, S. Moon, C. Fraleigh, P. Thiron, F. Tobagi, and C. Diot, "Analysis of measured single-hop delay from an operational backbone network," in *Proc. of IEEE INFOCOM*, New York, NY, June 2002.

[83] A. Narula-Tam and E. Modiano, "Dynamic load balancing in WDM packet networks with and without wavelength constraints," *IEEE J. on Selected Areas in Communications*, vol. 18, pp. 1972–1979, Oct. 2000.

[84] J. M. Jaffe, "Bottleneck flow control," *IEEE Trans. on Communications*, vol. 29, pp. 954–962, July 1981.

[85] J. Gao and L. Zhang, "Tradeoffs between stretch factor and load balancing ratio in routing on growth restricted graphs," in *Proc. of ACM Symposium on Principles of Distributed Computing*, St. Johns, Canada, July 2005.

[86] A. Lucena and J. E. Beasley, "Branch and cut algorithms," in *Advanced in Linear and Integer Programming*, J. E. Beasley, Ed. Oxford University Press, 1996.

[87] C. Blum and A. Roli, "Metaheuristics in combinatorial optimization: Overview and conceptual comparison," *ACM Computing Surveys*, vol. 35, pp. 268–308, Sept. 2003.

[88] H. R. Lourenco, O. Martin, and T. Stutzle, "Iterated local search," in *Handbook of Metaheuristics*, F. Glover and G. Kochenberger, Eds. Kluwer Academic Publishers, 2002, pp. 321–353.

[89] R. Jain, W. Hawe, and D. Chiu, "A quantitative measure of fairness and discrimination for resource allocation in shared computer systems," Tech. Rep. DEC-TR-301, Sept. 1984.

[90] A. H. Mohsenian Rad and V. W. S. Wong, "Logical topology design and interface assignment for multi-channel wireless mesh networks," in *Proc. of IEEE Globecom*, San Francisco, CA, Dec. 2006.

[91] R. Ramachandran, E. Belding, K. Almeroth, and M. Buddhikot, "Interference-aware channel assignment in multi-radio wireless mesh networks," in *Proc. of IEEE INFOCOM*, Barcelona, Spain, Apr. 2006.

[92] Y. Xi and E. Yeh, "Distributed algorithms for spectrum allocation, power control, routing, and congestion control in wireless networks," in *Proc. of ACM MobiHoc*, Montreal, Canada, Sept. 2007.

[93] L. Yu, Y. Xiang, and M. Shi, "A novel channel assignment algorithm based on topology simplification in multi-radio wireless mesh networks," in *Proc. of IEEE IPCCC*, Phoenix, AZ, Apr. 2006.

[94] A. Bar-noy, A. Mayer, B. Schieber, and M. Sudan, "Guaranteeing fair service to persistent dependent tasks," *SIAM Journal on Computing*, vol. 27, pp. 1168–1189, Aug. 1998.

[95] J. Proakis, *Digital Communications*, 4th ed. McGraw-Hill, 2000.

[96] R. Horn and C. Johnson, *Topics in Matrix Analysis.* Cambridge University Press, 1994.

[97] J. Lee, A. Tang, J. Huang, M. Chiang, and A. Calderbank, "Reverse engineering MAC: A game-theoretic model," *IEEE J. on Selected Areas in Communications*, vol. 6, pp. 2741–2751, July 2007.

[98] J. Lee, M. Chiang, and R. Calderbank, "Utility-optimal random-access control," *IEEE Trans. on Wireless Communications*, vol. 25, pp. 1135–1147, Aug. 2007.

[99] P. Gupta and A. Stolyar, "Optimal throughput allocation in general random-access networks," in *Proc. of CISS*, Princeton, NJ, Mar. 2006.

[100] L. Chen, S. H. Low, and J. C. Doyle, "Contention control: A game theoretic approach," in *Proc. of IEEE CDC*, New Orleans, LA, Dec. 2007.

[101] A. B. MacKenzie and S. B. Wicker, "Stability of multipacket slotted Aloha with selfish users and perfect information," in *Proc. of IEEE INFOCOM*, San Francisco, CA, Mar. 2003.

[102] M. Cagalj, S. Ganeriwal, I. Aad, and J. P. Hubaux, "On selfish behavior in CSMA/CA networks," in *Proc. of IEEE INFOCOM*, Miami, FL, Mar. 2005.

[103] Y. Sagduyu and A. Ephremides, "Power control and rate adaptation as stochastic games for random access," in *Proc. of IEEE CDC*, Maui, HI, Dec. 2003.

[104] A. H. Mohsenian Rad, J. Huang, M. Chiang, and V. W. S. Wong, "Utility-optimal random access: Reduced complexity, fast convergence, and robust performance," *IEEE Trans. on Wireless Communications (submitted)*, Dec. 2007.

[105] F. Calì, M. Conti, and E. Gregori, "Dynamic tuning of the IEEE 802.11 protocol to achieve a theoretical throughput limit," *IEEE/ACM Trans. on Net.*, vol. 8, no. 6, pp. 785–799, Dec. 2000.

[106] V. S. Borkar, "Asynchronous Stochastic Approximation," *SIAM Journal on Control and Optimization*, vol. 36, pp. 840–851, May 1998.

[107] A. H. Mohsenian Rad and V. W. S. Wong, "Joint optimal channel allocation, interface assignment, and MAC design for multi-channel wireless mesh networks," in *Proc. of IEEE INFOCOM*, Anchorage, AK, May 2007.

[108] ——, "Joint logical topology design, interface assignment, channel allocation, and routing for multi-channel wireless mesh networks," *IEEE Trans. on Wireless Communications*, vol. 6, pp. 4432–4440, Dec. 2007.

[109] E. Biglieri, R. Calderbank, A. Constantinides, A. Goldsmith, A. Paulraj, and H. V. Poor, *MIMO Wireless Communications.* Cambridge University Press, 2007.

[110] A. H. Mohsenian Rad and V. W. S. Wong, "Partially overlapped channel assignment for multi-channel wireless mesh networks," in *Proc. of IEEE ICC*, Glasgow, Scotland, June 2007.

[111] J. Lee, M. Chiang, and R. Calderbank, "Jointly optimal congestion and contention control in wireless ad hoc networks," *IEEE Communications Letters*, vol. 10, no. 3, pp. 216–218, Mar. 2006.

[112] H. S. Chiu, K. L. Yeung, and K. S. Lui, "Maximizing broadcast load in multi-channel multi-interface wireless mesh networks," in *Proc. of IEEE Globecom*, New Orleans, LA, Nov. 2008.

[113] E. Aryafar, O. Gurewitz, and E. W. Knightly, "Distance-1 constrained channel assignment in single radio wireless mesh networks," in *Proc. of IEEE INFOCOM*, Phoenix, AZ, May 2008.

[114] J. So and N. Vaidya, "Multi-channel MAC for ad hoc networks: Handling multi-channel hidden terminals using a single transceiver," in *Proc. of ACM MobiHoc*, Tokyo, Japan, May 2004.

[115] D. Poole, *Linear Algebra: A Modern Introduction*, 2nd ed. Brooks Cole, 2005.

[116] D. Niyato and E. Hossain, "Competitive pricing for spectrum sharing in cognitive radio networks: Dynamic game, inefficiency of nash equilibrium, and collusion," *IEEE J. on Selected Areas in Comm.*, vol. 26, pp. 192–202, Jan. 2008.

[117] J. H. Huang and Y. F. Kao, "Price-based resource allocation strategies for wireless ad hoc networks with transmission rate and energy constraints," in *Proc. of IEEE ICCCN'07*, Honolulu, HI, Aug. 2007.

[118] A. H. Mohsenian Rad and V. W. S. Wong, "Joint channel allocation, interface assignment, and mac design for multi-channel wireless mesh networks," in *Proc. of IEEE INFOCOM*, Anchorage, Alaska, May 2007.

[119] J. Lee, M. Chiang, and R. Calderbank, "Utility-optimal random-access control," *IEEE Trans. on Wireless Communications*, vol. 25, pp. 1135–1147, Aug. 2007.

[120] D. Niyato and E. Hossain, "Integration of WiMAX and WiFi: Optimal pricing for bandwidth sharing," *IEEE Communications Magazine*, vol. 45, pp. 140–146, May 2007.

[121] O. Ileri, S. Mau, and N. Mandayam, "Pricing for enabling forwarding in self-configuring ad-hoc networks," *IEEE J. on Selected Areas in Communications*, vol. 23, pp. 151–162, Jan. 2005.

[122] L. He and J. Walrand, "Pricing and revenue sharing strategies for internet service providers," *IEEE J. on Selected Areas in Communications*, vol. 24, pp. 942–951, May 2006.

[123] H. X. Shen and T. Basar, "Incentive-based pricing for network games with complete and incomplete information," *Annals of the Int. Soc. of Dynamic Games - Advances in Dynamic Game Theory: Numerical Methods, Algorithms, and Applications to Ecology and Economics (S. Jorgensen, M. Quincampoix and T. L. Vincent)*, vol. 9, pp. 431–458, 2007.

[124] F. Davoli, M. Marchese, and M. Mongelli, "Adaptive pricing without explicit knowledge of users traffic demands and utility functions," in *Proc. of IEEE GLOBECOM'07*, Washington, DC, Nov. 2007.

[125] J. Zhao, P. Zhu, and X. Lu, "Pricing models of inter-domain multicasting applications," in *Proc. of IEEE CCNC'07*, Las Vegas, NV, Jan. 2007.

[126] M. J. Neely, "Optimal pricing in a free market wireless network," in *Proc. of IEEE INFOCOM*, Anchorage, Alaska, May 2007.

[127] J. Lau and B. Liang, "Optimal pricing for selfish users and prefetching in heterogeneous wireless networks," in *Proc. of IEEE ICC'07*, Glasgow, UK, June 2007.

[128] S. Mandal, D. Saha, and M. Chatterjee, "Pricing wireless network services using smart market models," in *Proc. of IEEE CCNC'06*, Las Vegas, NV, Jan. 2006.

[129] P. Marbach and Y. Qiu, "Cooperation in wireless ad hoc networks: A market-based approach," *IEEE/ACM Trans. on Networking*, vol. 13, pp. 1325–1338, Dec. 2005.

[130] A. MacKenzie, *Game Theory for Wireless Engineers*, 1st ed. Morgan & Claypool Publishers, 2006.

[131] L. Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks," *IEEE Trans. on Automatic Control*, vol. 13, pp. 582–595, June 1992.

[132] M. J. Neely, E. Modiano, and C. P. Li, "Fairness and opimal stochastic control for heterogeneous networks," in *Proc. of IEEE INFOCOM*, Miami, Florida, Mar. 2005.

[133] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication.* Cambridge University Press, 2004.

[134] Y. Xi and E. Yeh, "Throughput optimal distributed control of stochastic wireless networks," in *Proc. of WiOpt*, Boston, MA, Apr. 2006.