Optimal Random Access Protocols for Wireless Networks

by

Man Hon Cheung

B.Eng., The Chinese University of Hong Kong, Hong Kong, China, 2005
M.Phil., The Chinese University of Hong Kong, Hong Kong, China, 2007

A THESIS SUBMITTED IN PARTIAL FULFILLMENT 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)

March 2012

Abstract

In this thesis, we present several random access algorithms for medium access control in wireless networks. Optimization theory, game theory, and dynamic programming are applied in the analysis and the design of these algorithms.

First, we study the problem of multi-channel random access using the signal-to-interference-plus-noise-ratio (SINR) model in cognitive radio networks. We formulate it as a network utility maximization (NUM) problem, and propose a distributed algorithm that converges to a near-optimal solution. Moreover, we apply coalitional game theory to study the incentive issues of rational user cooperation in a given channel under the SINR model.

Next, we consider a wireless local area network (WLAN) with rational users, who may strategically declare their access categories (ACs) not intended for their applications in order to gain some unfair shares of the network resources. We propose to use the Vickrey-Clarke-Groves (VCG) mechanism to motivate each user to declare truthfully its actual AC to the access point (AP). In order to implement the VCG mechanism with concave, step, and quasi-concave utility functions, we propose an enumeration algorithm to obtain the global optimal solution of the formulated non-convex NUM problem.

To extend the aforementioned work on single-channel random access in WLANs, we focus on sigmoidal utility functions. We propose a subgradient algorithm to solve the formulated NUM problem using the dual decomposition method. If the sufficient conditions on link capacities are satisfied, the algorithm obtains the optimal solution.

Finally, we consider the vehicular ad hoc networks. We study the problem of random access in a drive-thru scenario, where roadside APs are installed on a highway to provide temporary Internet access for vehicles. We first consider the single-AP scenario with random vehicular traffic, and propose a dynamic optimal random access (DORA) algorithm that aims to minimize the total transmission cost of a vehicle. We determine the conditions under which the optimal transmission policy has a threshold structure, and propose an algorithm with a lower computational complexity. Then, we consider the multiple-AP scenario with deterministic vehicular traffic arrival due to traffic estimation. A joint DORA is proposed to obtain the optimal transmission policy.

Preface

I am the first author and principal contributor of all chapters. All chapters are co-authored with Dr. Vincent W.S. Wong, who supervised the research. Chapters 2, 3, and 4 are co-authored with Dr. Robert Schober, who provided valuable comments for the works. Chapters 3 and 4 are co-authored with Dr. Amir-Hamed Mohsenian-Rad, who contributed in the formulation of the network utility maximization problems. In particular, in Chapter 3, Dr. Amir-Hamed Mohsenian-Rad contributed in proving a necessary condition related to the feasibility of the optimization problem. Chapter 5 are co-authored with Dr. Fen Hou and Dr. Jianwei Huang, who provided valuable comments for the work.

The following publications describe the work completed in this thesis. In some cases, the conference papers contain materials overlapping with the journal papers.

Journal Papers Accepted/Published

- Man Hon Cheung, Amir-Hamed Mohsenian-Rad, Vincent W.S. Wong, and Robert Schober, "Random Access for Elastic and Inelastic Traffic in WLANs," *IEEE Trans. on Wireless Communications*, vol. 9, no. 6, pp. 1861–1866, June 2010.

- Man Hon Cheung, Vincent W.S. Wong, and Robert Schober, "SINR-based Random Access for Cognitive Radio: Distributed Algorithm and Coalitional Game," *IEEE Trans. on Wireless Communications*, vol. 10, no. 11, pp. 3887–3897, Nov. 2011.

- Man Hon Cheung, Fen Hou, Vincent W.S. Wong, and Jianwei Huang, "DORA: Dynamic Optimal Random Access for Vehicle-to-Roadside Communications," accepted for publication in *IEEE Journal on Selected Areas in Communications*, 2012.

Journal Paper Submitted

- Man Hon Cheung, Amir-Hamed Mohsenian-Rad, Vincent W.S. Wong, and Robert Schober, "Utility-Optimal Random Access for Wireless Multimedia Networks," submitted, 2012.

Conference Papers Published

- Man Hon Cheung, Amir-Hamed Mohsenian-Rad, Vincent W.S. Wong, and Robert Schober, "Random Access Protocols for WLANs based on Mechanism Design," in *Proc. of IEEE International Conference on Communications (ICC)*, Dresden, Germany, June 2009.

- Man Hon Cheung, Fen Hou, Vincent W.S. Wong, and Jianwei Huang, "Dynamic Optimal Random Access for Vehicle-to-Roadside Communications," in *Proc. of IEEE International Conference on Communications (ICC)*, Kyoto, Japan, June 2011.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

AC	Access category
ACK	Acknowledgement
AP	Access point
BS	Base station
CR	Cognitive radio
CRN	Cognitive radio network
CSMA/CA	Carrier sense multiple access with collision avoidance
DCF	Distributed coordination function
DORA	Dynamic optimal random access
DP	Dynamic programming
DSRC	Dedicated short range communications
IEEE	Institute of Electrical and Electronics Engineers
ITS	Intelligent transportation system
JDORA	Joint dynamic optimal random access
KKT	Karush-Kuhn-Tucker (optimality conditions)
MAC	Medium access control
MS	Mobile station
NE	Nash equilibrium
NUM	Network utility maximization

OBU	Onboard unit
QoS	Quality of service
RSU	Roadside unit
SINR	Signal-to-interference-plus-noise ratio
SNR	Signal-to-noise ratio
VANET	Vehicular ad hoc network
VCG	Vickrey-Clarke-Groves (mechanism)
V2R	Vehicle-to-roadside
V2V	Vehicle-to-vehicle
WAVE	Wireless access in vehicular environments
WLAN	Wireless local area network

Acknowledgments

First and foremost, I wish to express my deep gratitude to my supervisor, Dr. Vincent Wong, for his supervision. His advice and comments have greatly improved the quality of my research. He has provided me with a lot of opportunities to experience different research areas in wireless networking. He has also taught me about technical writing and has helped me improve the presentation of my works.

I am particularly grateful to Dr. Robert Schober, Dr. Victor Leung, Dr. Jianwei Huang, Dr. Amir-Hamed Mohsenian-Rad, and Dr. Fen Hou for their collaborations and useful comments that help improve the quality of my research work. I would also like to thank the members of my doctoral committee for their time and effort.

I am grateful to the colleagues in my lab: Dr. Joo-Han Song, Dr. Hongbo Guo, Dr. Enrique Stevens-Navarro, Dr. Yuxia Lin, Dr. Vahid Shah-Mansouri, Derrick Wing Kwan Ng, Keivan Ronasi, Pedram Samadi, Binglai Niu, Mani Malekesmaeili, Ehsan Vahedi, Enxin Yao, Bojiang Ma, Chi Sun, Jiaqi Gui, Wei Bao, Xiaolei Hao, Wenbo Shi, Jinbiao Xu, Shaobo Mao, and Suyang Duan.

I wish to record my gratitude to my dad, mom, and sister for their love and support when I am studying in Canada. The life as a Ph.D student would be tough and lonely without their care.

I hope to thank my friends in my church in Vancouver. They have helped me adapt to a new environment in Vancouver. Their prayers are important to me, especially when the deadlines are approaching. I am also grateful to my friends in my church in Hong Kong for their prayers and support.

Last by not the least, I would like to thank my Lord for accompanying me in my life in Vancouver and every stages of my research. He is “a lamp to my feet and a light for my path” (Psalm 119:105).

This work was supported by the AUTO21, a member of the Network of Centres of Excellence of Canada program.

Chapter 1

Introduction

In a wireless network, a medium access control (MAC) protocol is used to coordinate the access of the users to the shared wireless medium. In general, there are two main classes of MAC protocols: *scheduling* and *random access* [2]. In a scheduling-based MAC, the transmissions of the users are scheduled orderly in an attempt to prevent packet collisions among the users. On the other hand, in a random access MAC, the users need to contend for the channel for transmission, so packet collisions are likely to occur. However, contention-based random access protocols are scalable and flexible, and are widely used in wireless networks.

In this thesis, we design random access protocols for different types of wireless networks, which includes cognitive radio networks (CRNs), wireless local area networks (WLANs), and vehicular ad hoc networks (VANETs). Mathematical tools, such as optimization theory, game theory, and dynamic programming, are applied in the design and analysis of these protocols. The rest of this chapter is organized as follows. In Section 1.1, we first provide an overview of the wireless network settings that we consider in this thesis. We then introduce the mathematical tools that we use in the thesis in Section 1.2. In Section 1.3, we summarize the main contributions and results in this thesis. Finally, the organization of the thesis is described in Section 1.4.

1.1 Random Access in Wireless Networks

1.1.1 Cognitive Radio Networks

With the licensed radio spectrum being under-utilized [3], cognitive radio (CR) [4, 5] has emerged as the solution to the spectrum scarcity problem. To address the problem of spectrum sharing between the primary (licensed) users and secondary (unlicensed) users, the *commons model* and the *property model* [6] have been proposed. In the commons model, the primary users act as if the secondary users are not present. The secondary users access the spectrum holes *opportunistically* so that they do not cause interference to the primary users. In the property model, the primary users are allowed to *trade* some of their temporarily unused spectrum to the secondary users in exchange for monetary return. In Chapter 2, we consider the setting where a set of channels from the primary network is available to the secondary CRN, e.g., in the form of *dynamic spectrum leasing* [7–9] in the property model or by *spectrum sensing* [10, 11] in the commons model. In order to implement a scalable system that is adaptive to the dynamic network changes in a CRN, a distributed MAC protocol is proposed for the secondary users. The multi-channel signal-to-interference-plus-noise-ratio (SINR) model is consider in the chapter.

1.1.2 Wireless Local Area Networks

In a WLAN, an access point (AP) is usually set up to offer connectivity to the users in the network. Due to the network topology and limited transmission range in a WLAN, it is usually reasonable to assume that all the users are one-hop neighbours to each other. We consider a single channel model, where each user randomly attempts to access the shared channel with a certain transmission probability. We consider that the users support applications with both *elastic* and *inelastic* traffic [12]. The elastic traffic is generated by non-real-time applications such as traditional file transfer and electronic mail. The inelastic traffic is generated by real-time applications including real-time voice and video streaming, which usually have tight quality of service (QoS) requirements. The random access problem is formulated mathematically using the framework of network utility maximization (NUM). In Chapter 3, we model the utilities of the users with elastic traffic using concave functions, and the utilities of the users with inelastic traffic using quasi-concave and step functions. *Global* optimal solution is obtained for the NUM problem. In Chapter 4, we extend the work in Chapter 3 by considering concave utility functions for elastic traffic sources, and sigmoidal utility functions for inelastic traffic sources. Optimal solution is obtained if a sufficient condition on link capacities is satisfied. Otherwise, an approximate solution with the lower and upper bounds of the objective value are obtained.

1.1.3 Vehicular Ad Hoc Networks

The development of intelligent transportation system (ITS) has gained significant momentum in recent years, especially after the Federal Communications Commission (FCC) in the United States allocated 75 MHz licensed spectrum in the 5.9 GHz band in 1999 for this purpose. There are two types of application in an ITS. *Safety* applications, such as cooperative forward collision warning, lane change warning, and left turn assistant (e.g.,[13, 14]), have been proposed to improve the safety of the passengers by informing the vehicles of potential dangers ahead of time. *Non-safety* applications, such as traffic management, instant messaging, and media content delivery, have been designed to avoid traffic congestion and improve the experience of driving.

Vehicular ad hoc networks, which are designed to provide reliable communications among vehicles and roadside APs, are playing an important role in the development of ITS. VANETs support the ITS applications through different types of communication patterns, including vehicle-to-roadside (V2R) and vehicle-to-vehicle (V2V) communications [15]. V2R communications involve data transmissions between vehicular nodes and roadside APs, and V2V communications involve data exchange among vehicular nodes only. For both types, we can further classify the communications as either single-hop or multi-hop. Some of the characteristics of VANETs are briefly described as follows:

- Rechargeability: Because the power of the vehicles can be recharged easily and readily, power constraint is usually not a prime design issue in VANETs.
- Frequent change in network topology: In urban areas, the average speed of the vehicles is around 50 to 60km/h. While in highways, the vehicles move with high average speed of around 80 to 110 km/h. As a result, the network topology changes rapidly.
- Variable network density: In urban areas with traffic jams during rush hours, the network density is very high. On the other hand, it can be very low in rural areas.
- Non-random trajectory: Because of the fixed road topology, vehicles move in an organized instead of a completely random manner. The position of a vehicle is confined to the road network. Also, the mobility of the vehicles is influenced by human behaviours and some pre-defined traffic rules.
- Frequent fragmentation of network: Due to the mobility of the vehicles and the limited transmission ranges of the antenna, a network is frequently segmented in regions where the network density is not too high. As a result, some of the nodes are isolated and are not connected to the network.
- Large scale: During the rush hours in urban areas, the number of vehicles involved in a VANET can be very large.

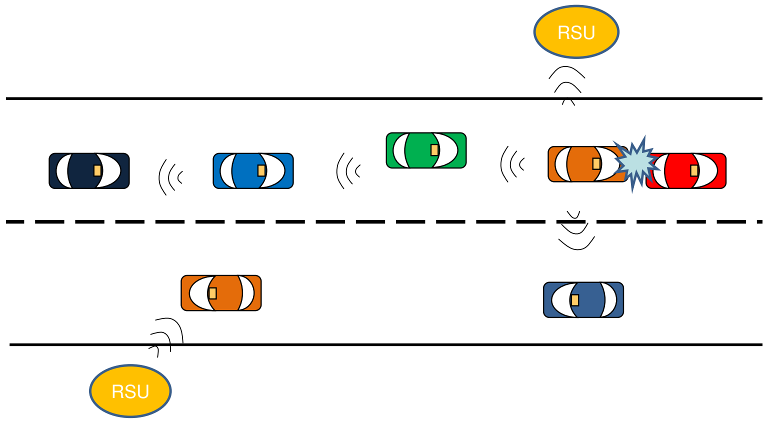

Figure 1.1: V2V and V2R communications in a VANET.

In recent years, the dedicated short range communications (DSRC) was proposed to provide a short to medium range communication service that supports both the safety and non-safety applications in V2R and V2V communication environments. DSRC is meant to be a complement to the cellular communications by providing high data rate in some circumstances while minimizing the latency of the communication links. The wireless access in vehicular environments (WAVE) standard is a core part of DSRC, and it includes the IEEE 802.11p and IEEE 1609.x standards. The IEEE 802.11p standard specifies the physical and MAC layer of the wireless communications, and the IEEE 1609.1, 1609.2, 1609.3, and 1609.4 standards are related to the resource manager, security, networking, and multichannel operations, respectively [16–18].

Since the IEEE 802.11p MAC protocol is a heuristic, we aim to design distributed and optimal uplink random access schemes for VANETs analytically. In Chapter 5, we study random access for V2R single-hop uplink transmissions from the vehicles to the APs in a drive-thru scenario. We propose algorithms for single AP optimization with random vehicular traffic and for joint AP optimization with traffic pattern estimation.

1.2 Mathematical Foundation

In this thesis, we extensively employed several useful mathematical tools for the design and analysis of the random access protocols, which include the optimization theory, game theory, and dynamic programming. Optimization theory [19–23] is related to the problem of maximization or minimization of a real-valued function by systematically choosing the variables from a feasible set and computing the values of the objective and constraint functions. Game theory [24–27] is a useful technique for studying the interactions among the users with strategic interdependence and characterizing the outcome of the game based on some well-defined solution concepts. Dynamic programming [28, 29] is a set of mathematical and computational tools that is usually applied for the study of sequential decision problems by taking into account both the short-term and long-term consequences of a chosen action. In this section, we provide an overview of each of these three techniques.

1.2.1 Optimization Theory

In some problems, we are given a well-defined objective function, some constraints, and some optimization variables. We may want to obtain an optimal or a near-optimal solution of the problem systematically based on some results in optimization theory. Specifically, we consider the following optimization problem

$$\begin{array}{ll} \underset{\boldsymbol{x}}{\text{minimize}} & f_0(\boldsymbol{x}) \\ \text{subject to} & f_i(\boldsymbol{x}) \leq 0, \quad i = 1, \ldots, m, \\ & \boldsymbol{x} \in \mathcal{X}, \end{array} \tag{1.1}$$

where $f_0(\boldsymbol{x})$ is the objective function, and $f_i(\boldsymbol{x}) \leq 0$ is the i^{th} constraint. The vector $\boldsymbol{x}$ contains all the optimization variables, and $\mathcal{X}$ is its feasible set.

The *Lagrangian function* is given by

$$L(\boldsymbol{x}, \boldsymbol{\lambda}) = f_0(\boldsymbol{x}) + \sum_{i=1}^{m} \lambda_i f_i(\boldsymbol{x}), \tag{1.2}$$

where λ_i is the *Lagrange multiplier* associated with the i^{th} constraint $f_i(\boldsymbol{x}) \le 0$.

The *Lagrangian dual function* is given by

$$g(\boldsymbol{\lambda}) = \inf_{\boldsymbol{x} \in \mathcal{X}} L(\boldsymbol{x}, \boldsymbol{\lambda}). \tag{1.3}$$

The *dual problem* is

$$\begin{aligned} \underset{\boldsymbol{\lambda}}{\text{maximize}} \quad & g(\boldsymbol{\lambda}) \\ \text{subject to} \quad & \boldsymbol{\lambda} \succeq \mathbf{0}. \end{aligned} \tag{1.4}$$

It should be noted that the dual problem (1.4) is a convex optimization problem. This is the case whether or not the primal problem (1.1) is convex [21, pp. 223]. We assume that $g(\boldsymbol{\lambda})$ has a finite value for all $\boldsymbol{\lambda} \succeq \mathbf{0}$.

By applying the Danskin's Theorem [19, pp. 737], the subdifferential $\partial g(\boldsymbol{\lambda})$ (i.e., the set of all subgradients of $g(\boldsymbol{\lambda})$) is given by

$$\partial g(\boldsymbol{\lambda}) = \text{conv}\big\{ \nabla_{\boldsymbol{\lambda}} L(\boldsymbol{x}, \boldsymbol{\lambda}) : \boldsymbol{x} \in \boldsymbol{x}^*(\boldsymbol{\lambda}) \big\}, \tag{1.5}$$

where

$$\nabla_{\boldsymbol{\lambda}} L(\boldsymbol{x}, \boldsymbol{\lambda}) = \left(\frac{\partial L(\boldsymbol{x}, \boldsymbol{\lambda})}{\partial \lambda_1}, \ldots, \frac{\partial L(\boldsymbol{x}, \boldsymbol{\lambda})}{\partial \lambda_m} \right)^T, \tag{1.6}$$

and

$$\boldsymbol{x}^*(\boldsymbol{\lambda}) = \arg\min_{\boldsymbol{x} \in \mathcal{X}} L(\boldsymbol{x}, \boldsymbol{\lambda}). \tag{1.7}$$

$\text{conv}\{\mathcal{H}\}$ is the convex hull of set $\mathcal{H}$ and the notation $(\cdot)^T$ denotes vector transpose operator.

Since $g(\boldsymbol{\lambda})$ is concave, it can be shown that $\partial g(\boldsymbol{\lambda})$ is a nonempty, convex, and compact set [19, pp. 732]. Let $\boldsymbol{s}(\boldsymbol{\lambda}) \in \partial g(\boldsymbol{\lambda})$ be the *subgradient* of g at $\boldsymbol{\lambda}$. $g(\boldsymbol{\lambda})$ is *differentiable* at $\boldsymbol{\lambda}$ with *gradient* $\boldsymbol{s}(\boldsymbol{\lambda}) = \nabla g(\boldsymbol{\lambda}) = \nabla_{\boldsymbol{\lambda}} L(\boldsymbol{x}^*(\boldsymbol{\lambda}), \boldsymbol{\lambda})$, if and only if there is only one element $\nabla g(\boldsymbol{\lambda})$ in $\partial g(\boldsymbol{\lambda})$ (i.e., $\partial g(\boldsymbol{\lambda}) = \{\nabla g(\boldsymbol{\lambda})\}$ is a singleton) [19, pp. 732]. Otherwise, $g(\boldsymbol{\lambda})$ is *non-differentiable* at $\boldsymbol{\lambda}$.

Using the *subgradient projection method* (or more specifically, the gradient projection method if $g(\boldsymbol{\lambda})$ is a differentiable function), we update $\boldsymbol{\lambda}$ according to the following equation:

$$\boldsymbol{\lambda}(t+1) = \Big[\boldsymbol{\lambda}(t) + \alpha(t)\boldsymbol{s}\big(\boldsymbol{\lambda}(t)\big)\Big]^+, \tag{1.8}$$

where t is the index of the iteration, $\alpha(t)$ is the step size, and $[z]^+ = \max\{z, 0\}$. Note that for gradient projection method, $\boldsymbol{s}(\boldsymbol{\lambda})$ is an improving feasible direction [23, pp. 590]. However, this may not be true for the subgradient projection method [22, pp. 461]. The convergence of the subgradient projection method is due to the fact that the distance of the current solution $\boldsymbol{\lambda}(t)$ to the optimal solution $\boldsymbol{\lambda}^*$ decreases for sufficiently small step size [22, pp. 462].

To study the convergence of the subgradient projection method, we need to assume that the subgradients are bounded [22, pp. 471]. With the use of diminishing step size $\alpha(t) \geq 0$ such that $\lim_{t\to\infty} \alpha(t) = 0$ and $\sum_{t=0}^{\infty} \alpha(t) = \infty$ (e.g., we can choose $\alpha(t) = \frac{1+k}{k+t}$, where k is a positive constant [19, pp. 624]), it can be shown that the subgradient projection method can obtain the optimal solution $\boldsymbol{\lambda}^*$ of problem (1.4) [22, pp. 478]. However, with the use of fixed step size $\alpha(t) = \alpha$, it is only guaranteed to converge to within some bounds of the optimal value [22, pp. 473]. For differentiable function $g(\boldsymbol{\lambda})$, with the use of fixed step size, it is guaranteed to converge to the optimal solution $\boldsymbol{\lambda}^*$, provided that α is sufficiently small and the gradient satisfies a Lipschitz continuity condition [19, pp. 240].

Let $p^\star$ and $d^\star$ be the optimal values of the primal problem (1.1) and the dual problem

(1.4), respectively. We can establish the lower bound on the optimal value of the primal problem by the weak duality theorem [19, pp. 495]).

Theorem 1.1 *(Weak Duality Theorem)* $d^\star \leq p^\star$.

After we have obtained the optimal solution $\boldsymbol{\lambda}^*$ of the dual problem (1.4), we may want to check if we can use $\boldsymbol{\lambda}^*$ to obtain the optimal solution of the primal problem (1.1). This is done possible by the following theorem.

Theorem 1.2 *If* $g(\boldsymbol{\lambda})$ *is differentiable at* $\boldsymbol{\lambda}^*$ *and* $\boldsymbol{x}^*$ *is the unique minimum in* $\boldsymbol{x}$ *of* $L(\boldsymbol{x}, \boldsymbol{\lambda})$*, then* $(\boldsymbol{x}^*, \boldsymbol{\lambda}^*)$ *is a saddle point of the primal problem (1.1). Thus,* $\boldsymbol{x}^*$ *is the global optimal solution of the primal problem (1.1).*

The results follow from [20, Property 6.5(c), Theorem 5.3]. Even if we cannot obtain the primal optimal solution from the dual optimal solution, we can still find a good *approximate* solution to the primal problem based on the dual solution. Consider the following *perturbed* problem

$$\begin{aligned} \underset{\boldsymbol{x}}{\text{minimize}} \quad & f_0(\boldsymbol{x}) \\ \text{subject to} \quad & f_i(\boldsymbol{x}) \leq f_i\big(\boldsymbol{x}^*(\boldsymbol{\lambda})\big), \quad i = 1, \dots, m, \\ & \boldsymbol{x} \in \mathcal{X}, \end{aligned} \tag{1.9}$$

where $\boldsymbol{x}^*(\boldsymbol{\lambda})$ is defined in (1.7).

Theorem 1.3 *For any* $\boldsymbol{\lambda} \succeq \mathbf{0}$*,* $\boldsymbol{x}^*(\boldsymbol{\lambda})$ *is a global optimal solution of the perturbed problem (1.9).*

The results follow from [20, Property 6.6]. As a result, if the term $f_i\big(\boldsymbol{x}^*(\boldsymbol{\lambda})\big)$ is not much larger than zero for all i (i.e., there is only a small violation of the inequality constraints), we may obtain an acceptable practical solution using the dual algorithm [19, pp. 602].

1.2.2 Game Theory

In some problems, we need to model the interaction among multiple rational users with *strategic interdependence* [27] in a wireless network. Game theory is a useful mathematical tool for analyzing the strategic interactions of the players and predicting the final outcome. Here, we introduce two main branches in game theory, namely the *non-cooperative game theory* and *coalitional game theory* (also known as cooperative game theory). However, their names may be misleading in that the former does not apply exclusively to situations where the interests of different players conflict (e.g., the result of using cooperative strategies in the prisoner's dilemma in a repeated game), and the latter does not consider only situations where the interests of players align with each other (e.g., the grand coalition may not be stable in some cases that the players choose not to cooperate as one coalition). In other words, it is possible that the players cooperate in non-cooperative game theory and does not cooperate in coalitional game theory. The essential differences between these two branches lies in the modeling unit, where the modeling unit for the former is an individual player, while that for the latter is a group of players [24]. In addition, we discuss the theory of *mechanism design*, which the players have private preferences on the available options, and the preferences are not publicly observable.

Non-cooperative Game Theory

In a non-cooperative game, there are four ingredients that characterize the game [27]: the players, the rules (e.g., the order of the players in making a move), the outcome (i.e., the result after the players have chosen their actions), and the payoff (i.e., the utility function that ranks the preference of a player towards different outcomes).

One way of classifications in non-cooperative game theory is to classify the games into two categories: strategic game (also known as static game) or extensive game (also known as dynamic game). In a *static game*, each player chooses his action once and for all, and all the players make their moves *simultaneously*. Given the assumptions about the preference, rationality, and information available of each player, *solution concepts* are defined that characterize the possible outcomes of a game. Examples of solution concepts include Nash equilibrium, Pareto optimality, dominant strategy, Bayesian-Nash equilibrium, maxmin strategy, minmax strategy, minimax regret, correlated equilibrium [24–27].

In contrast, in a *dynamic game*, the events are ordered *sequentially* that the players may choose their actions at different time instants. The players are interacted sequentially, which the strategy of one player is conditioned on the strategies of the other players. Solution concepts, such as subgame perfect Nash equilibrium and perfect Bayesian-Nash equilibrium, are commonly used in a dynamic game.

Coalitional Game Theory

Different from noncooperative game theory, coalitional game theory focuses on on what a group or some groups of players can achieve rather than on what individual players can achieve. Formally, a coalitional game $\mathcal{G}$ is defined as a pair $(\mathcal{N}, v)$, where

- $\mathcal{N}$ is the set of players or the *grand coalition*.
- v is the *value* of a coalition $\mathcal{S} \subseteq \mathcal{N}$. For a *transferable utility* (TU) game [24], we have $v(\mathcal{S}) \in \mathbb{R}$, which is a scalar. Otherwise, for a *nontransferable utility* (NTU) game, $v(\mathcal{S}) \in \mathbb{R}^{|\mathcal{S}|}$, which is a vector that represents the payoff of each player in coalition $\mathcal{S}$.

In fact, there are different categories of coalitional game, which includes the *canonical coalitional game* and *coalition formation game* [30]. For the canonical coalitional game, a *superadditive* game is considered. For a TU game, it is defined as follows:

Definition 1.1 *A coalitional game $\mathcal{G}$ is superadditive if $v(\mathcal{S}_1 \cup \mathcal{S}_2) \geq v(\mathcal{S}_1) + v(\mathcal{S}_2), \forall\, \mathcal{S}_1, \mathcal{S}_2 \subset \mathcal{N}$ with $\mathcal{S}_1 \cap \mathcal{S}_2 = \phi$.*

It means that the formation of a large coalition by cooperation out of disjoint coalitions achieves at least the sum of the value achieved by the disjoint coalitions individually. Solution concepts, such as the core and the Shapley value, are usually used to analyze the coalitional game. The *core* is used to analyze the stability of the grand coalition. Because of the superadditivity of the game, the players have an incentive to form the grand coalition $\mathcal{N}$ that consists of all the players. The core characterizes if it is possible that a subset of players may opt out of the grand coalition to form a smaller coalition, where the players in the smaller coalition receive higher utilities than when they participate in the grand coalition. The *Shapley value* is related to *fairly* dividing the payoff, achieved by the grand coalition, among the players. It is fair in the sense that the division of the payoff among the players is related to the contribution of each player to a coalition. Other useful solution concepts for superadditive game include the nucleolus, the kernel, the bargaining set, and the Nash bargaining solution [24].

For games that are not superadditive, we may consider the problem of *coalition formation* [30]. We define a coalition structure as a partition of the set of players $\mathcal{N}$ into a number of disjoint coalitions. Let the value of a coalition structure be the sum of the values of all the coalitions in the coalition structure. If the game is superadditive, coalition structure generation is trivial because the coalition structure with the largest value is the grand coalition. For the formation of coalitions based on maximizing the value of the coalition structure, the *merge-and-split* algorithm can be used to find a stable coalition structure [30, 31]. Stability concepts such as $\mathbb{D}_{hp}$-stability and $\mathbb{D}_c$-stability can be applied to study the stability of the coalition structure after the merge-and-split operations. For the formation of coalitions based on the individual preferences of the players, *hedonic coali-*

tion formation game can be used [30, 32, 33]. We can characterize the stability of the final coalition structure by testing if it is *Nash-stable* or *individually stable*.

Mechanism Design Theory

Mechanism design is a sub-field in microeconomics and game theory that considers how to implement a desirable solutions with self-interested players, each with *private information* about their preferences [25, 27, 34]. Since the players' actual preferences are not publicly known, it is important to elicit this information from the rational players so that a socially favourable outcome can be implemented. Famous examples of mechanism design includes the Vickrey-Clarke-Groves (VCG) mechanism and the dAGVA mechanism due to d'Aspremont, Gerard-Varet, and Arrow. There are a number of properties that can be used to characterize the operation of a mechanism. They include incentive compatibility (i.e., whether the players will reveal their true preferences), efficiency (i.e., whether the allocation results in the maximum aggregate utility in the system), budget balance (i.e., whether there is a net transfer of payment from or to the mechanism), and individual rationality (i.e., whether a player has the intention to participate in the system).

1.2.3 Dynamic Programming

In some problems, we need to make decisions *sequentially*, which have both short-term and long-term consequences. In order to achieve the optimal performance, we need to take into account the relationship between the current and future decisions, and that between the current and future outcomes. Dynamic programming is a collection of mathematical and computational tools for analyzing this type of *sequential decision problems*. Specifically, there are five key ingredients in a sequential decision problem [28]:

1. *Decision epoch*: $t \in \mathcal{T}$, where $\mathcal{T}$ represents the set of time points at which decisions

can be made. Notice that $\mathcal{T}$ can be either finite or infinite, and either discrete or continuous. In this chapter, we focus on a discrete-time finite-horizon problem with $\mathcal{T} = \{1, \ldots, T\}$, where T is the total number of decision time points.

2. *State*: $s \in \mathcal{S}$, where $\mathcal{S}$ represents the set of states. It summarizes the past information that is relevant for future optimization.

3. *Action*: $a \in \mathcal{A}$, where $\mathcal{A}$ represents the set of actions that the decision maker can take.

4. *State transition probability*: $p_t(s'|s, a)$ represents that the probability that the system will be in state s' at time $t + 1$ if action a is taken in state s at time t.

5. *Cost*: $c_t(s, a)$ represents the *immediate* cost incurred by choosing action a in state s at time $t \in \mathcal{T}$. For a discrete-time finite-horizon problem with $\mathcal{T} = \{1, \ldots, T\}$, we define $\hat{c}_{T+1}(s)$ as the *terminal* cost when the system is in state s at time $t = T + 1$ (i.e., after all the decision epochs).

Let $\delta_t : \mathcal{S} \to \mathcal{A}$ be the *decision rule* that specifies the decision at state s at time $t \in \mathcal{T}$. We define a *policy* $\boldsymbol{\pi} = (\delta_t(s), \forall\, s \in \mathcal{S}, t \in \mathcal{T})$ as a set of decision rules covering all the states at all the time. We denote $s_t^{\boldsymbol{\pi}}$ as the state at time t if policy $\boldsymbol{\pi}$ is used, and we let Π be the feasible set of $\boldsymbol{\pi}$. The decision maker aims to find an optimal policy that minimizes the total expected cost, which can be formulated as the following optimization problem

$$\min_{\boldsymbol{\pi} \in \Pi} E_{\boldsymbol{\pi}, S} \sum_{\tau=1}^{T} c_t\Big(s_t^{\boldsymbol{\pi}}, \delta_t(s_t^{\boldsymbol{\pi}})\Big) + \hat{c}_{T+1}(s_{T+1}^{\boldsymbol{\pi}}), \tag{1.10}$$

where $E_{\boldsymbol{\pi}, S}$ denotes the expectation with respect to the probability distribution by policy $\boldsymbol{\pi}$ with an initial state S at time $t = 1$. Besides the finite-horizon decision problem described in this chapter, it is also possible to consider a infinite-horizon problem with discounted cost or average cost [28].

Finite-Horizon Dynamic Programming

Let $v_t(s)$ be the minimal expected total cost that the decision maker has to pay from time t to time $T+1$, given that the system is in state s immediately before the decision at time slot $t \in \mathcal{T}$. The *optimality equation* [28, pp. 83] relating the minimal expected total cost at different states for $t \in \mathcal{T}$ is

$$v_t(s) = \min_{a \in \mathcal{A}} \{\psi_t(s,a)\}, \tag{1.11}$$

where

$$\psi_t(s,a) = c_t(s,a) + \sum_{s' \in \mathcal{S}} p_t\big(s' \mid s,a\big) v_{t+1}(s') \tag{1.12}$$

The first and second terms on the right hand side of the equation above are the *immediate cost* and the *expected future cost* in the remaining decision epochs for choosing action a, respectively. For time $t = T+1$, we have the boundary condition that

$$v_{T+1}(s) = \hat{c}_{T+1}(s). \tag{1.13}$$

By evaluating the optimality equation (1.11) recursively from $t = T$ to $t = 1$ starting from the boundary condition in (1.13) (i.e., using *backward induction* [28, pp. 92]), we can obtain the *optimal policy* $\boldsymbol{\pi}^* = (\delta_t^*(s), \forall\, s \in \mathcal{S}, t \in \mathcal{T})$, where

$$\delta_t^*(s) = \arg\min_{a \in \mathcal{A}} \{\psi_t(s,a)\}. \tag{1.14}$$

The policy $\boldsymbol{\pi}^*$ obtained is the optimal solution of problem (1.10) [29, pp. 18]. In fact, $\boldsymbol{\pi}^*$ is a *contingency plan* that contains information about the optimal decisions in *all* possible states s at all time $t \in \mathcal{T}$.

Threshold Optimal Policy

In some special cases, the optimal policy $\boldsymbol{\pi}^*$ may have a threshold structure. Consider an example with an action set $\mathcal{A} = \{a_1, a_2\}$, where a_1 and a_2 are distinct actions. The optimal policy is said to have a *threshold* structure, if for all $t \in \mathcal{T}$, we have

$$\delta_t^*(s) = \begin{cases} a_1, & \text{if } s \leq s_t^*, \\ a_2, & \text{otherwise.} \end{cases} \tag{1.15}$$

where s_t^* is a control limit. Establishing a threshold policy is appealing, because the decision rule is easier to implement and the computational complexity in obtaining the optimal policy can be reduced significantly [28, pp. 103]. Moreover, memory can be saved, because we do not need to store the decision rules in all states at all time (i.e., $\delta_t^*(s)$, $\forall\, s \in \mathcal{S}, t \in \mathcal{T}$), but just the set of thresholds ($s_t^*, \forall\, t \in \mathcal{T}$) in order to characterize the optimal policy $\boldsymbol{\pi}^*$. In addition, it may provide useful insight into the structure of the problem.

To establish a threshold policy, one useful step is to check if the expected total cost $\psi_t(s, a)$ is *supermodular* or *submodular*.

Definition 1.2 *A function $\psi_t(s, a)$ is supermodular in $\mathcal{S} \times \mathcal{A}$ if for $\hat{s}, \check{s} \in \mathcal{S}$ and $\hat{a}, \check{a} \in \mathcal{A}$, where $\hat{s} \geq \check{s}$ and $\hat{a} \geq \check{a}$, we have*

$$\psi_t(\hat{s}, \hat{a}) + \psi_t(\check{s}, \check{a}) \geq \psi_t(\hat{s}, \check{a}) + \psi_t(\check{s}, \hat{a}). \tag{1.16}$$

If the reverse inequality holds, then $\psi_t(s, a)$ is submodular*.*

If we can establish the supermodularity or submodularity of the function $\psi_t(s,a)$, we can show directly that the optimal policy $\boldsymbol{\pi}^*$ has a threshold structure by the following theorem:

Theorem 1.4 *a) If $\psi_t(s,a)$ is a supermodular function in $\mathcal{S} \times \mathcal{A}$, then $\delta_t^*(s)$ is a nonincreasing function in $\mathcal{S}$. b) If $\psi_t(s,a)$ is a submodular function in $\mathcal{S} \times \mathcal{A}$, then $\delta_t^*(s)$ is a nondecreasing function in $\mathcal{S}$.*

The result follows from [28, pp. 104, 115]. With a threshold structure, we can apply the monotone backward induction [28, pp. 111] with a lower computational complexity to obtain the optimal policy $\boldsymbol{\pi}^*$.

1.3 Summary of Results and Contributions

The thesis covers several analytical MAC design problems in different types of wireless networks, which includes CRNs, WLANs, and VANETs. The results are divided into four chapters. The contributions in each chapter are as follows:

- Chapter 2 considers the problem of multi-channel random access in CRNs. While most of the previously proposed MAC protocols for CRNs are heuristic and are based on the simplistic protocol model, we design a distributed MAC protocol using the more accurate SINR model. First, we assume that the secondary users are *cooperative* and formulate the problem of assigning transmission and listening probabilities for random access as a non-convex NUM problem. We propose a three-phase algorithm that converges to a near-optimal solution after solving a number of convex optimization problems distributively. Simulation results show that our proposed algorithm based on the SINR model achieves a higher aggregate throughput than other schemes which are based on the protocol model. Then, we consider the case that the secondary users are *rational*. We use coalitional game theory to study the incentive issues of user cooperation in a given channel for the SINR model. In particular, we use the solution concept of the core to analyze the stability of the grand coalition, and the solution concept of the Shapley value to fairly divide the payoff among the users. We show that the Shapley value lies in the core when all the users are one-hop neighbours of each other. We illustrate the Shapley value and the core with a numerical example. The work in Chapter 2 is published in [35].

- Chapter 3 considers the incentive issues of access category (AC) declaration in a WLAN, where any rational station can strategically declare an AC which is not intended for its application in order to gain an unfair share of the network resources. We first apply game theory to analyze the behaviour of the rational stations. We then propose to use the VCG mechanism to motivate each station to declare truthfully the required AC of its application to the AP. The AP will then inform each station about its transmission probability and the price that the station needs to pay for the offered service. Furthermore, we consider the implementation of the VCG mechanism in a WLAN with both elastic (e.g., file transfer) and inelastic (e.g., real-time video) traffic. By modeling the utilities of mobile stations with concave, step, and quasi-concave utility functions, we show that implementing the VCG mechanism involves solving a non-convex network utility maximization problem *optimally*. We propose an enumeration algorithm to obtain the global optimal solution by solving a number of convex optimization problems. Simulation results show that a truthful mechanism prevents selfish users from gaining an unfair share of the network bandwidth and supports adequate service differentiation among different ACs. The work in Chapter 3 is published in [36].

- Chapter 4 extends the work of random access in a WLAN with both elastic and inelastic traffic in Chapter 3. The utilities of the applications generating elastic and inelastic traffic are modeled by concave and *sigmoidal* functions, respectively. We formulate a NUM problem, where the optimization variables are the transmission probabilities of the stations. By applying the dual decomposition method, we propose a subgradient algorithm to solve the formulated NUM problem. We also develop *closed-form* solutions for the dual subproblems involving sigmoidal functions that have to be solved in each iteration of the proposed algorithm. Furthermore, we obtain a sufficient condition on the link capacities which guarantees achieving the global optimal solution when our proposed algorithm is being used. If this condition is not satisfied, then we can still guarantee that the optimal value of the objective function is within some lower and upper bounds. We perform various simulations to validate our analytical models when the available link capacities meet or do not meet the sufficient optimality condition. The work in Chapter 4 is published in [37].

- Chapter 5 considers the problem of random access in a drive-thru scenario, where roadside APs are installed on a highway to provide temporary Internet access for vehicles. We consider V2R communications for a vehicle that aims to upload a file when it is within the APs' coverage ranges, where both the channel contention level and transmission data rate vary over time. The vehicle will pay a fixed amount each time it tries to access the APs, and will incur a penalty if it cannot finish the file uploading when leaving the APs. First, we consider the problem of finding the optimal transmission policy in an AP with random vehicular traffic arrival. We formulate it as a finite-horizon sequential decision problem, solve it using dynamic programming (DP), and design a general dynamic optimal random access (DORA) algorithm. We determine the conditions under which the optimal transmission policy has a threshold structure. A monotone DORA algorithm with a lower computational complexity is proposed for this special case. Next, we consider the problem of finding the optimal transmission policy in multiple APs with deterministic vehicular traffic arrival due to traffic estimation. The optimal transmission policy is obtained using DP and a joint DORA algorithm is proposed. Simulation results based on realistic vehicular traffic model show that our algorithms achieve the minimal total cost and the highest upload ratio as compared with some other heuristic schemes. The work in Chapter 5 is published in [38, 39].

1.4 Thesis Organization

The rest of the thesis is organized as follows. In Chapter 2, we study the multi-channel random access problem using the SINR model, and propose a distributed algorithm to obtain a near-optimal solution. We also study the interactions of the rational users in single channel under the SINR model using coalitional game theory. In Chapter 3, we study the incentive issues of AC declaration of the rational users. We apply the VCG mechanism to motivate the users to declare their ACs truthfully. To implement the VCG-based random access with concave, quasi-concave, and step utility functions, an enumeration algorithm is proposed to obtain the optimal solution of the NUM problem. In Chapter 4, we focus on users with sigmoidal and concave utility functions. A subgradient algorithm based on the dual method is proposed that can obtain the optimal solution if a sufficient condition on link capacities is satisfied. Otherwise, it can be used to obtain a near-optimal solution with the lower and upper bounds of the optimal value of the objective function. Finally, in Chapter 5, we study uplink random access in VANETs. Both the single-AP and multiple-AP scenarios are studied, and DORA algorithms that minimizes the total transmission cost in a finite horizon are proposed. Each of the main chapters in this thesis is self-contained and included in separate journal articles or conference papers. A review of the related work is given for each chapter accordingly. The notations are defined separately for each chapter.

Chapter 2

Multi-channel SINR-based Random Access and Coalitional Game

In the throughput analysis of multi-channel MAC protocols in CRNs, such as [40, 41], the *protocol model* or *unit disk model* [42] is widely used to account for the effect of multi-user interference due to its simplicity in characterizing the physical layer. Under the protocol model, a transmission is successful if the receiver is within the transmission range of its intended transmitter and outside the interference range of other transmitters. However, in reality, the interference at the receiver is the *cumulative* power received from other nodes that are concurrently transmitting. As a result, the *signal-to-interference-plus-noise-ratio (SINR) model* or *physical model* [42] characterizes the effect of interference more accurately. Under the SINR model, a transmission is successful if and only if the SINR at the intended receiver is above a predefined threshold that depends on the adopted modulation and coding schemes. Despite its higher complexity, the SINR model is getting more attention in recent years due to its higher practicality and accuracy in modeling. Some recent works have investigated contention-based *random access* protocols [43, 44] and collision-free *scheduling* protocols [45, 46] using the SINR model.

Since most of the proposed multi-channel MAC protocols for CRNs are *heuristic* in nature and apply the simplistic protocol model, in this chapter, we propose a *distributed* random access protocol for CRNs, which is based on the *multi-channel SINR model* and the mathematical framework of network utility maximization (NUM). In particular, we extend the mathematical models in [44] and [47], where the former focused on the single-channel SINR model, while the latter focused on the multi-channel protocol model. The resulting *non-convex* optimization problem is more difficult to solve than the problems in [44] and [47]. In particular, our problem involves the dimension of channel selection which was absent in [44], and entails a more accurate and complex interaction among the users due to the SINR model which was absent in [47]. We propose a distributed three-phase algorithm using convex optimization and the coordinate ascent method to obtain a near-optimal solution for the non-convex NUM problem. Simulation results show that the proposed scheme based on the SINR model achieves a higher aggregate throughput than other schemes which are based on the protocol model.

In the formulation of the NUM problem, we assume that all the secondary users are *cooperative*. Thus, an interesting question is what happens if the users are *rational* and they aim to maximize their own utilities? Previous works, such as [48, 49] use *non-cooperative game theory* to analyze the behaviour of rational users in CRNs. However, this approach is more appropriate for analyzing the behaviour of *individual* rational users. To analyze what a *group* of rational users can achieve under the SINR model, where the effect of interference is cumulative, *coalitional game theory* [24] is a more suitable tool. Coalitional game theory has found many applications in communication networks [10, 30, 50]. In [50], it was applied to study the behaviour of users under the SINR model. However, [50] investigated cooperative communications, whereas we consider random access in CRNs. In summary, the contributions of this chapter are as follows:

- We first assume that the secondary users are cooperative. We formulate the problem of random access with multiple channels as a NUM problem using the SINR model, where the optimization variables are the transmission and listening probabilities of the users.

- We propose a distributed three-phase algorithm using convex optimization and the coordinate ascent method to obtain a near-optimal solution for the non-convex NUM problem.

- We then study the case where the secondary users are rational. We formulate the problem as a coalitional game to analyze the interactions among the users under the SINR model. We apply the solution concepts of the *core* and the *Shapley value* [24] to characterize the stability and fair allocation of the aggregate utility among the rational users, respectively. We show that the Shapley value lies in the core when all users are one-hop neighbours.

- Simulation results show that the proposed scheme based on the SINR model achieves a higher aggregate throughput than other schemes which are based on the protocol model. A numerical example is given to illustrate both the Shapley value and the core.

The rest of this chapter is organized as follows. We present the related work in Section 2.1. The system model is described in Section 2.2. We formulate the random access problem in Section 2.3 and present our distributed algorithm in Section 2.4. The coalitional game is discussed in Section 2.5 and simulation results are presented in Section 2.6. A summary is given in Section 2.7.

2.1 Related Work

Several multi-channel MAC protocols have been proposed for CRNs. In [51], Cordeiro *et al.* proposed a distributed cognitive MAC protocol that includes a slotted beaconing period for nodes to negotiate on the channel usage. A rendezvous channel is used to coordinate nodes tuned to different channels. Su *et al.* proposed in [40] two sensing policies for

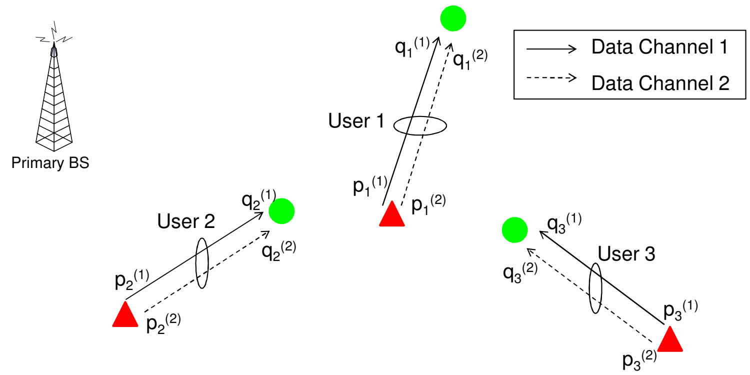

Figure 2.1: A CRN with set of users $\mathcal{N} = \{1, 2, 3\}$, where the triangles and circles represent the transmitters and receivers, respectively. The set of available orthogonal data channels $\mathcal{C} = \{1, 2\}$ is provided by the primary BS. $p_i^{(c)}$ and $q_i^{(c)}$ denote the transmission and listening probabilities for user i in channel c, respectively.

the physical layer and a packet scheduling algorithm for the MAC layer of a distributed CRN. Queueing theory was used to evaluate the throughput and delay in saturated and non-saturated networks under the proposed sensing policies. Jia *et al.* proposed in [52] a hardware-constrained cognitive MAC protocol that coordinates the contention and spectrum usage among the secondary users. Practical constraints related to hardware, sensing, and transmission were considered. Timmers *et al.* proposed in [41] an energy-efficient distributed multi-channel MAC protocol for a multi-hop CRN, which is based on the timing structure of the power-saving mode used in the IEEE 802.11 standard.

2.2 System Model

As shown in Fig. 2.1, we consider a CRN with several secondary nodes located in a neighbourhood, where a set of orthogonal data channels $\mathcal{C}$ and one control channel are obtained from the primary base station (BS), e.g, in the form of spectrum leasing. The data channels are used for data transmissions, and the control channel is used for the exchange of control messages. The total number of data channels is $C = |\mathcal{C}|$. We consider only single-hop transmissions between the secondary nodes. We define $\mathcal{N}$ as the set of one-hop transmitter/receiver pairs or links in the CRN, and we refer to each transmitter/receiver pair as a *user*. The total number of users is $N = |\mathcal{N}|$. We adopt a slotted MAC protocol, where time is divided into equal time slots. The users attempt to access the shared channel at the beginning of each time slot according to their *transmission probabilities* in each channel. That is, each user $i \in \mathcal{N}$ can access a channel c with a certain transmission probability $p_i^{(c)}$, and we define a vector $\boldsymbol{p} = (p_i^{(c)},\ \forall\, i \in \mathcal{N}, c \in \mathcal{C})$. Also, we introduce a vector $\boldsymbol{q} = (q_i^{(c)},\ \forall\, i \in \mathcal{N}, c \in \mathcal{C})$, where $q_i^{(c)}$ is the *listening probability* of receiver i in channel c. We have the following constraints:

$$\sum_{c \in \mathcal{C}} p_i^{(c)} \leq 1 \quad \text{and} \quad \sum_{c \in \mathcal{C}} q_i^{(c)} \leq 1,\ \forall\, i \in \mathcal{N}. \tag{2.1}$$

For the *SINR model*, if user $i \in \mathcal{N}$ chooses to transmit in channel $c \in \mathcal{C}$, then the SINR at receiver i is given by

$$\theta_i^{(c)} = \frac{P_i G_{ii}^{(c)}}{\iota_i^{(c)} + n_i^{(c)}}, \tag{2.2}$$

where P_i is the transmit power of user i. $G_{ij}^{(c)}$ is the channel gain from the transmitter of user i to the receiver of user j in channel c. $\iota_i^{(c)}$ and $n_i^{(c)}$ are the interference and noise powers received by user i in channel c, respectively. Given that receiver i has tuned to channel c for reception, the communication of user i is successful if

$$\theta_i^{(c)} \geq \theta_i^{th} \quad \Leftrightarrow \quad \iota_i^{(c)} \leq \frac{P_i G_{ii}^{(c)}}{\theta_i^{th}} - n_i^{(c)}, \tag{2.3}$$

where θ_i^{th} is the SINR threshold. Let $\mathbb{N}_i$ be the power set (i.e., the set of all subsets) of $\mathcal{N} \backslash \{i\}$. As an example, for $\mathcal{N} = \{1, 2, 3\}$, $\mathbb{N}_2 = \{\{\}, \{1\}, \{3\}, \{1, 3\}\}$. Assuming that the transmit powers $(P_i, \forall\, i \in \mathcal{N})$ are fixed, we define $\mathbb{M}_i^{(c)}$ as a set where each element is a set of users that can transmit simultaneously with user i without affecting the reception of receiver i in channel c (i.e., θ_i^{th} can be achieved). The set $\mathbb{M}_i^{(c)}$ obtained with the SINR model is given by

$$\mathbb{M}_{i,SINR}^{(c)} = \Big\{\mathcal{M} \in \mathbb{N}_i : \sum_{m\in\mathcal{M}} P_m G_{mi}^{(c)} \leq \frac{P_i G_{ii}^{(c)}}{\theta_i^{th}} - n_i^{(c)}\Big\}. \tag{2.4}$$

If the *protocol model* is used, only pairwise interference is considered. User m is an *interferer* or *one-hop neighbour* to user i if the SINR due to the interference from user m only is below the SINR threshold. That is,

$$\frac{P_i G_{ii}^{(c)}}{P_m G_{mi}^{(c)} + n_i^{(c)}} < \theta_i^{th}. \tag{2.5}$$

The set $\mathbb{M}_i^{(c)}$ obtained with the protocol model is given by

$$\mathbb{M}_{i,PTC}^{(c)} = \Big\{\mathcal{M} \in \mathbb{N}_i : P_m G_{mi}^{(c)} \leq \frac{P_i G_{ii}^{(c)}}{\theta_i^{th}} - n_i^{(c)}, \forall\, m \in \mathcal{M}\Big\}. \tag{2.6}$$

Intuitively, more users are allowed to transmit simultaneously in the protocol model than in the SINR model. This is confirmed by the following lemma.

Lemma 2.1 $\mathbb{M}_{i,SINR}^{(c)} \subseteq \mathbb{M}_{i,PTC}^{(c)}$.

Proof: Observing the fact that $\sum_{m\in\mathcal{M}} P_m G_{mi}^{(c)} \leq \frac{P_i G_{ii}^{(c)}}{\theta_i^{th}} - n_i^{(c)}$ in (2.4) implies $P_m G_{mi}^{(c)} \leq \frac{P_i G_{ii}^{(c)}}{\theta_i^{th}} - n_i^{(c)}, \forall\, m \in \mathcal{M}$ in (2.6), it follows directly that $\mathbb{M}_{i,SINR}^{(c)} \subseteq \mathbb{M}_{i,PTC}^{(c)}$. ■

Example 2.1 *We consider Fig. 2.1 where the transmit powers of all users are the same. Assuming that all users have selected channel 1, at a certain transmit power level P, we can observe the following: Since transmitter 1 is close to receivers 2 and 3, user 1 interferes with users 2 and 3. However, since transmitters 2 and 3 are far away from receiver 1, users 2 and 3 do not interfere with user 1 as long as they do not transmit* *simultaneously. Users 2 and 3 are far from each other and do not interfere with each other. For the protocol model, we have* $\mathbb{M}^{(1)}_{1,PTC} = \{\{\}, \{2\}, \{3\}, \{2,3\}\}$, $\mathbb{M}^{(1)}_{2,PTC} = \{\{\}, \{3\}\}$, *and* $\mathbb{M}^{(1)}_{3,PTC} = \{\{\}, \{2\}\}$. *However, the protocol model does not take into account that user 1 may be interfered when both users 2 and 3 transmit* simultaneously. *In this case, we have* $\mathbb{M}^{(1)}_{1,SINR} = \{\{\}, \{2\}, \{3\}\} \subset \mathbb{M}^{(1)}_{1,PTC}$, $\mathbb{M}^{(1)}_{2,SINR} = \mathbb{M}^{(1)}_{2,PTC}$, *and* $\mathbb{M}^{(1)}_{3,SINR} = \mathbb{M}^{(1)}_{3,PTC}$.

The probability of successful transmission of user i in channel c is given by

$$p_i^{succ,(c)} = p_i^{(c)} q_i^{(c)} \sum_{\mathcal{M} \in \mathbb{M}_i^{(c)}} \left(\prod_{m \in \mathcal{M}} p_m^{(c)} \right) \left(\prod_{k \in \mathcal{N} \setminus \mathcal{M}, k \neq i} (1 - p_k^{(c)}) \right). \tag{2.7}$$

We define $\mathsf{M}_i = \left\{ \mathbb{M}_i^{(c)}, \forall\, c \in \mathcal{C} \right\}$. The average data rate of user i is given by

$$r_i(\boldsymbol{p}, \boldsymbol{q}_i, \mathsf{M}_i) = \sum_{c \in \mathcal{C}} \mu_i^{(c)} p_i^{succ,(c)}, \tag{2.8}$$

where $\mu_i^{(c)}$ is the peak data rate for user i in channel c, and vector $\boldsymbol{q}_i = \left(q_i^{(c)}, \forall\, c \in \mathcal{C} \right)$ contains the listening probabilities of receiver i in all the channels. Given $\boldsymbol{p}$ and $\boldsymbol{q}_i$, we have the following lemma, which states that the average data rate r_i is over-estimated when the protocol model is used.

Lemma 2.2 $r_i(\boldsymbol{p}, \boldsymbol{q}_i, \mathsf{M}_{i,PTC}) \geq r_i(\boldsymbol{p}, \boldsymbol{q}_i, \mathsf{M}_{i,SINR})$.

Proof: From (2.8) and Lemma 2.1, we have

$$\begin{aligned} r_i(\boldsymbol{p}, \boldsymbol{q}_i, \mathsf{M}_{i,PTC}) &= r_i(\boldsymbol{p}, \boldsymbol{q}_i, \mathsf{M}_{i,SINR}) + \sum_{c \in \mathcal{C}} \mu_i^{(c)} p_i^{(c)} q_i^{(c)} \\ &\quad \times \sum_{\mathcal{M} \in \mathbb{M}^{(c)}_{i,PTC} \setminus \mathbb{M}^{(c)}_{i,SINR}} \left(\prod_{m \in \mathcal{M}} p_m^{(c)} \right) \left(\prod_{k \in \mathcal{N} \setminus \mathcal{M}, k \neq i} (1 - p_k^{(c)}) \right) &(2.9) \\ &\geq r_i(\boldsymbol{p}, \boldsymbol{q}_i, \mathsf{M}_{i,SINR}), &(2.10) \end{aligned}$$

which completes the proof. ■

For the rest of this chapter, we assume that sets $\mathbb{M}_i^{(c)}$ in (2.4) and (2.6) are given, so we write $r_i(\boldsymbol{p}, \boldsymbol{q}_i, \mathsf{M}_i)$ as $r_i(\boldsymbol{p}, \boldsymbol{q}_i)$ for simplicity.

2.3 Network Utility Maximization

We now formulate the multi-channel random access problem as a NUM problem with vectors $\boldsymbol{p}$ and $\boldsymbol{q}$ as the optimization variables. The NUM problem is given by

$$\begin{aligned} \underset{\boldsymbol{p},\, \boldsymbol{q}}{\text{maximize}} \quad & \sum_{i \in \mathcal{N}} U_i\big(r_i(\boldsymbol{p}, \boldsymbol{q}_i)\big) \\ \text{subject to} \quad & \textstyle\sum_{c \in \mathcal{C}} p_i^{(c)} \leq 1, \ \sum_{c \in \mathcal{C}} q_i^{(c)} \leq 1, \quad \forall\, i \in \mathcal{N}, \\ & 0 \leq p_i^{(c)}, q_i^{(c)} \leq 1, \qquad \forall\, i \in \mathcal{N}, c \in \mathcal{C}, \end{aligned} \tag{2.11}$$

where $U_i\big(r_i(\boldsymbol{p}, \boldsymbol{q}_i)\big)$ is a concave and non-decreasing function in $r_i(\boldsymbol{p}, \boldsymbol{q}_i)$. However, due to the product form of the variables in (2.7), problem (2.11) is *non-convex*, even if the utility functions are concave. As a result, the problem is difficult to solve in general. An example of a concave utility function useful for resource allocation is the α-fair function [53] defined as

$$U_i(r_i) = \begin{cases} (1-\alpha_i)^{-1} r_i^{1-\alpha_i}, & \text{if } \alpha_i \in [0,1) \cup (1, \infty), \\ \ln r_i, & \text{if } \alpha_i = 1, \end{cases} \quad \forall\, i \in \mathcal{N}. \tag{2.12}$$

Intuitively, r_i increases when $p_i^{(c)}$ increases or when $p_j^{(c)}$ decreases, $j \neq i$. This is confirmed by the following lemma:

Lemma 2.3 *For $i \in \mathcal{N}$, we have: (a) $r_i(\boldsymbol{p}, \boldsymbol{q}_i)$ is a non-decreasing function of $p_i^{(c)}$, $\forall\, c \in \mathcal{C}$. (b) $r_i(\boldsymbol{p}, \boldsymbol{q}_i)$ is a non-increasing function of $p_j^{(c)}$, $\forall\, j \in \mathcal{N} \backslash \{i\}, c \in \mathcal{C}$.*

Proof: (a) From (2.8), r_i can be written in the form $r_i(\boldsymbol{p}, \boldsymbol{q}_i) = \sum_{c\in\mathcal{C}} x_i^{(c)} p_i^{(c)}$, where

$$x_i^{(c)} = \mu_i^{(c)} q_i^{(c)} \sum_{\mathcal{M}\in\mathbb{M}_i^{(c)}} \left(\prod_{m\in\mathcal{M}} p_m^{(c)}\right) \left(\prod_{k\in\mathcal{N}\backslash\mathcal{M}, k\neq i} (1 - p_k^{(c)})\right). \tag{2.13}$$

Since $x_i^{(c)} \geq 0$ and it is independent of $p_i^{(c)}$, $r_i(\boldsymbol{p}, \boldsymbol{q}_i)$ is a non-decreasing function of $p_i^{(c)}$, $\forall\, c \in \mathcal{C}$.

(b) Let $j \in \mathcal{N}\backslash\{i\}$ be given. We first define two sets of users that exclude users i and j:

$$\tilde{\mathcal{S}}_{i,j}^{(c)} = \left\{\mathcal{S} : \mathcal{S} \in \mathcal{N}\backslash\{i, j\}, \mathcal{S} \in \mathbb{M}_i^{(c)}, \mathcal{S} \cup \{j\} \in \mathbb{M}_i^{(c)}\right\} \tag{2.14}$$

and

$$\hat{\mathcal{S}}_{i,j}^{(c)} = \left\{\mathcal{S} : \mathcal{S} \in \mathcal{N}\backslash\{i, j\}, \mathcal{S} \in \mathbb{M}_i^{(c)}, \mathcal{S} \cup \{j\} \notin \mathbb{M}_i^{(c)}\right\}. \tag{2.15}$$

From (2.8), we can write r_i as

$$\begin{aligned} r_i(\boldsymbol{p}, \boldsymbol{q}_i) \;=\; & \sum_{c\in\mathcal{C}} \mu_i^{(c)} p_i^{(c)} q_i^{(c)} \Bigg[\sum_{\mathcal{S}\in\tilde{\mathcal{S}}_{i,j}^{(c)}} \left(\prod_{s\in\mathcal{S}} p_s^{(c)}\right) \left(\prod_{k\in\mathcal{N}\backslash\mathcal{S}, k\neq i,j} (1 - p_k^{(c)})\right) + \\ & \sum_{\mathcal{S}\in\hat{\mathcal{S}}_{i,j}^{(c)}} \left(\prod_{s\in\mathcal{S}} p_s^{(c)}\right) \left(\prod_{k\in\mathcal{N}\backslash\mathcal{S}, k\neq i,j} (1 - p_k^{(c)})\right) (1 - p_j^{(c)}) \Bigg], \end{aligned} \tag{2.16}$$

which is a non-increasing function of $p_j^{(c)}$, $\forall\, c \in \mathcal{C}$. ■

Although it is possible that the users may occupy more than one channel at an optimal solution, we can show based on Lemma 2.3 that we can always find another optimal solution where each user occupies only one channel.

Theorem 2.1 *A* global *optimal solution of problem (2.11), $(\boldsymbol{p}^*, \boldsymbol{q}^*)$, is in the form:*

$$p_i^{(c)*} \begin{cases} \in [0,1], & \textit{if } c = c_i, \\ = 0, & \textit{otherwise,} \end{cases} \quad \textit{and} \quad q_i^{(c)*} = \begin{cases} 1, & \textit{if } c = c_i, \\ 0, & \textit{otherwise,} \end{cases} \tag{2.17}$$

where c_i is the channel chosen by user i.

Proof: Assume that $(\boldsymbol{p}, \boldsymbol{q})$ is feasible in problem (2.11), but $\boldsymbol{p}$ and $\boldsymbol{q}$ are not in the form of (2.17). From (2.8), we have

$$r_i(\boldsymbol{p}, \boldsymbol{q}_i) = \sum_{c \in \mathcal{C}} s_i^{(c)}(\boldsymbol{p}) q_i^{(c)}, \tag{2.18}$$

where

$$s_i^{(c)}(\boldsymbol{p}) = \mu_i^{(c)} p_i^{(c)} \sum_{\mathcal{M} \in \mathbb{M}_i^{(c)}} \left(\prod_{m \in \mathcal{M}} p_m^{(c)} \right) \left(\prod_{k \in \mathcal{N} \setminus \mathcal{M}, k \neq i} (1 - p_k^{(c)}) \right). \tag{2.19}$$

We define

$$c_i = \arg\max_{c \in \mathcal{C}} s_i^{(c)}(\boldsymbol{p}), \ \forall\, i \in \mathcal{N}, \tag{2.20}$$

and $\boldsymbol{q}_i^* = (q_i^{(c)*}, \ \forall\, i \in \mathcal{N}, c \in \mathcal{C})$, where

$$q_i^{(c)*} = \begin{cases} 1, & \text{if } c = c_i, \\ 0, & \text{otherwise.} \end{cases} \tag{2.21}$$

We have

$$r_i(\boldsymbol{p}, \boldsymbol{q}_i) = \sum_{c \in \mathcal{C}} s_i^{(c)}(\boldsymbol{p}) q_i^{(c)} \leq s_i^{(c_i)}(\boldsymbol{p}) = r_i(\boldsymbol{p}, \boldsymbol{q}_i^*), \forall\, i \in \mathcal{N}, \tag{2.22}$$

where the inequality in the middle is due to the definition of c_i in (2.20) and the fact that

$\sum_{c\in\mathcal{C}} q_i^{(c)*} \leq 1$. Since $U_i(r_i)$ is a non-decreasing function in r_i, $\forall\, i \in \mathcal{N}$, we have

$$\sum_{i\in\mathcal{N}} U_i\big(r_i(\boldsymbol{p}, \boldsymbol{q}_i)\big) \leq \sum_{i\in\mathcal{N}} U_i\big(r_i(\boldsymbol{p}, \boldsymbol{q}_i^*)\big). \tag{2.23}$$

Given $\boldsymbol{q}^*$, we have

$$r_i(\boldsymbol{p}, \boldsymbol{q}_i^*) = \mu_i^{(c_i)} p_i^{(c_i)} \sum_{\mathcal{M}\in\mathbb{M}_i^{(c_i)}} \left(\prod_{m\in\mathcal{M}} p_m^{(c_i)}\right)\left(\prod_{k\in\mathcal{N}\setminus\mathcal{M}, k\neq i} (1 - p_k^{(c_i)})\right). \tag{2.24}$$

Since $\boldsymbol{p}$ is not in the form as shown on the left hand side of (2.17), there exists $c \neq c_i$ such that $p_i^{(c)} > 0$. We define $\boldsymbol{p}^* = (p_i^{(c)*},\ \forall\, i\ \in \mathcal{N}, c\ \in \mathcal{C})$, where

$$p_i^{(c)*} = \begin{cases} p_i^{(c)}, & \text{if } c = c_i, \\ 0, & \text{otherwise.} \end{cases} \tag{2.25}$$

Notice that r_i in (2.24) is independent of $p_i^{(c)}$ for $c \neq c_i$, and it is a non-increasing function of $p_j^{(c)}$, $\forall\, j \in \mathcal{N}\setminus\{i\}, c \in \mathcal{C}$ as shown in Lemma 2.3(b). Thus, we have

$$r_i(\boldsymbol{p}, \boldsymbol{q}_i^*) \leq r_i(\boldsymbol{p}^*, \boldsymbol{q}_i^*),\ \forall\, i \in \mathcal{N}. \tag{2.26}$$

Since $U_i(r_i)$ is a non-decreasing function in r_i, $\forall\, i \in \mathcal{N}$, we have

$$\sum_{i\in\mathcal{N}} U_i\big(r_i(\boldsymbol{p}, \boldsymbol{q}_i^*)\big) \leq \sum_{i\in\mathcal{N}} U_i\big(r_i(\boldsymbol{p}^*, \boldsymbol{q}_i^*)\big). \tag{2.27}$$

Combining (2.23) and (2.27), we have

$$\sum_{i\in\mathcal{N}} U_i\big(r_i(\boldsymbol{p}, \boldsymbol{q}_i)\big) \leq \sum_{i\in\mathcal{N}} U_i\big(r_i(\boldsymbol{p}, \boldsymbol{q}_i^*)\big) \leq \sum_{i\in\mathcal{N}} U_i\big(r_i(\boldsymbol{p}^*, \boldsymbol{q}_i^*)\big). \tag{2.28}$$

To sum up, given any feasible point $(\boldsymbol{p}, \boldsymbol{q})$, we can always find another feasible point $(\boldsymbol{p}^*, \boldsymbol{q}^*)$ in the form of (2.25) and (2.21), which yields an objective value that is not smaller than that for $(\boldsymbol{p}, \boldsymbol{q})$ and each user occupies only one channel. The result thus follows. ■

2.4 Three-Phase Distributed Algorithm using Sequential Convex Optimization

In this section, our goal is to solve non-convex NUM problem (2.11). We propose a low-complexity three-phase algorithm where the transmitters and receivers have to solve a number of convex optimization problems distributively. Convergence of the solution is guaranteed.

2.4.1 Transmission Probability Optimization

We define the vector $\boldsymbol{p}_i = (p_i^{(c)}, \forall\, c \in \mathcal{C})$. Transmitter $i \in \mathcal{N}$ needs to solve the following *local* optimization problem, which has the same objective function as problem (2.11):

$$\begin{aligned} \underset{\boldsymbol{p}_i}{\text{maximize}} \quad & U_i\left(\sum_{c\in\mathcal{C}} o_i^{(c)} p_i^{(c)}\right) + \sum_{j\in\mathcal{N}\setminus\{i\}} U_j\left(\sum_{c\in\mathcal{C}}\left(v_{ji}^{(c)} p_i^{(c)} + w_{ji}^{(c)}\left(1-p_i^{(c)}\right)\right)\right) \\ \text{subject to} \quad & \textstyle\sum_{c\in\mathcal{C}} p_i^{(c)} \le 1, \quad 0 \le p_i^{(c)} \le 1,\ \forall\, c\in\mathcal{C}, \end{aligned} \tag{2.29}$$

where

$$o_i^{(c)} = \mu_i^{(c)} q_i^{(c)} \sum_{\mathcal{M}\in\mathbb{M}_i^{(c)}} \left(\prod_{m\in\mathcal{M}} p_m^{(c)}\right)\left(\prod_{k\in\mathcal{N}\setminus\mathcal{M},k\neq i}\left(1-p_k^{(c)}\right)\right), \tag{2.30}$$

$$v_{ji}^{(c)} = \mu_j^{(c)} p_j^{(c)} q_j^{(c)} \sum_{\mathcal{M}\in\mathbb{M}_j^{(c)}:\, i\in\mathcal{M}} \left(\prod_{m\in\mathcal{M}\setminus\{i\}} p_m^{(c)}\right)\left(\prod_{k\in\mathcal{N}\setminus\mathcal{M},k\neq j}\left(1-p_k^{(c)}\right)\right), \tag{2.31}$$

and

$$w_{ji}^{(c)} = \mu_j^{(c)} p_j^{(c)} q_j^{(c)} \sum_{\mathcal{M} \in \mathbb{M}_j^{(c)} :\, i \notin \mathcal{M}} \left(\prod_{m \in \mathcal{M}} p_m^{(c)} \right) \left(\prod_{k \in \mathcal{N} \setminus \mathcal{M}, k \neq j, i} \left(1 - p_k^{(c)} \right) \right). \tag{2.32}$$

The coefficients $o_i^{(c)}$, $v_{ji}^{(c)}$, and $w_{ji}^{(c)}$ should be computed by transmitter i based on the broadcast messages from other transmitters and receivers.

Theorem 2.2 *Problem (2.29) is a* convex *optimization problem in* $\boldsymbol{p}_i$.

Proof: First, the constraints in problem (2.29) are linear. Also, as $o_i^{(c)}$, $v_{ji}^{(c)}$, and $w_{ji}^{(c)}$ are independent of $p_i^{(c)}$, and since the arguments within the utility functions are linear in $\boldsymbol{p}_i$, the objective function is concave in $\boldsymbol{p}_i$ [21, pp. 79]. Thus, problem (2.29) is a convex optimization problem. ■

Hence, we can solve problem (2.29) by using the *interior point method* [21].

2.4.2 Listening Probability Optimization

Receiver $i \in \mathcal{N}$ needs to solve the following *local* optimization problem with the same objective function as problem (2.11).

$$\begin{aligned} \underset{\boldsymbol{q}_i}{\text{maximize}} \quad & U_i \left(\sum_{c \in \mathcal{C}} a_i^{(c)} q_i^{(c)} \right) + \sum_{j \in \mathcal{N} \setminus \{i\}} U_j \left(r_j(\boldsymbol{p}, \boldsymbol{q}_j) \right) \\ \text{subject to} \quad & \textstyle\sum_{c \in \mathcal{C}} q_i^{(c)} \leq 1, \quad 0 \leq q_i^{(c)} \leq 1, \ \forall\, c \in \mathcal{C}, \end{aligned} \tag{2.33}$$

where

$$a_i^{(c)} = \mu_i^{(c)} p_i^{(c)} \sum_{\mathcal{M} \in \mathbb{M}_i^{(c)}} \left(\prod_{m \in \mathcal{M}} p_m^{(c)} \right) \left(\prod_{k \in \mathcal{N} \setminus \mathcal{M}, k \neq i} (1 - p_k^{(c)}) \right). \tag{2.34}$$

Theorem 2.3 *Let* $c_i = \arg\max_{c\in\mathcal{C}} a_i^{(c)}$*. A* closed-form *solution of problem (2.33) is*

$$q_i^{(c)*} = \begin{cases} 1, & \text{if } c = c_i, \\ 0, & \text{otherwise.} \end{cases} \tag{2.35}$$

Proof: First, notice that $a_i^{(c)}$ and $\sum_{j\in\mathcal{N}\setminus\{i\}} U_j\left(r_j(\boldsymbol{p},\boldsymbol{q}_j)\right)$ are independent of $q_i^{(c)}$. Since U_i is a non-decreasing function, problem (2.33) is equivalent to the following linear programming problem

$$\begin{aligned} \underset{\boldsymbol{q}_i}{\text{maximize}} \quad & \sum_{c\in\mathcal{C}} a_i^{(c)} q_i^{(c)} \\ \text{subject to} \quad & \textstyle\sum_{c\in\mathcal{C}} q_i^{(c)} \le 1, \\ & 0 \le q_i^{(c)} \le 1, \quad \forall\, c\in\mathcal{C}, \end{aligned} \tag{2.36}$$

the solution of which is given by (2.35). ■

2.4.3 Three-Phase Distributed Algorithm

Having introduced the local optimization problems for the transmitter and receiver of user $i \in \mathcal{N}$, we are now ready to present Algorithm 2.1 for obtaining a near-optimal solution of problem (2.11) based on the *coordinate ascent method* [54, pp. 207]. Let $\boldsymbol{p}_{-i} = (\boldsymbol{p}_1,\ldots,\boldsymbol{p}_{i-1},\boldsymbol{p}_{i+1},\ldots,\boldsymbol{p}_N)$ and $\boldsymbol{q}_{-i} = (\boldsymbol{q}_1,\ldots,\boldsymbol{q}_{i-1},\boldsymbol{q}_{i+1},\ldots,\boldsymbol{q}_N)$. Considering transmitter i, the basic idea of this method is that we fix $\boldsymbol{p}_{-i}$ and $\boldsymbol{q}$, and maximize the aggregate utility $\sum_{i\in\mathcal{N}} U_i\big(r_i(\boldsymbol{p},\boldsymbol{q}_i)\big)$ with respect to $\boldsymbol{p}_i$ (i.e., problem (2.29)). Similarly, for receiver i, we fix $\boldsymbol{p}$ and $\boldsymbol{q}_{-i}$, and maximize the aggregate utility $\sum_{i\in\mathcal{N}} U_i\big(r_i(\boldsymbol{p},\boldsymbol{q}_i)\big)$ with respect to $\boldsymbol{q}_i$ (i.e., problem (2.33)). The updates of the solutions are carried out *successively*. Notice that the solution of problem (2.33) as stated in Theorem 2.3 represents a channel selection. Once the channel is selected by the receiver, the transmitter will not attempt to transmit in other channels, to which the receiver is not listening. As a result, the receivers should defer their decisions of selecting a channel until after the transmitters have coordinated their transmission probabilities.

With this idea, we propose our Algorithm 2.1 with three phases. In phase I, the receivers are initialized to listen to each channel with a certain probability. The transmitters then probe the channels by adjusting their transmission probabilities until the aggregate utility converges. In phase II, each transmitter/receiver pair selects the channel that results in the highest average data rate. The reason for choosing only one channel is given by Theorem 2.1. In phase III, based on this channel selection, the transmitters adjust their transmission probabilities again until the aggregate utility converges. After the execution of Algorithm 2.1, the transmitters and receivers can proceed to transmit in and listen to the data channels according to $\boldsymbol{p}^*$ and $\boldsymbol{q}^*$, respectively. In other words, in the *control stage*, Algorithm 2.1 is executed to determine $\boldsymbol{p}^*$ and $\boldsymbol{q}^*$. In the *transmission stage*, data transmissions take place based on $\boldsymbol{p}^*$ and $\boldsymbol{q}^*$.

In Algorithm 2.1, $\mathcal{T}_i$ is the set of time slots in which user $i \in \mathcal{N}$ solves the local optimization problem in the control stage. Also, we use variable u to keep track of the aggregate utility achieved in the previous iteration, and we let $u^*(t)$ be the aggregate utility achieved in iteration t. The algorithm transitions from phase I to phase II and from phase III to the exit if the difference $\Delta = u^*(t) - u$ is less than the predefined convergence threshold ϵ. The complexity of Algorithm 2.1 is relatively low because it involves solving only convex problem (2.29) and evaluating closed-form equation (2.35). Thus, it is reasonable to assume that users in CRNs have the computational capabilities to perform these mathematical operations.

Algorithm 2.1 *Three-Phase Distributed Algorithm to Obtain a Near-optimal Solution for Problem (2.11).*

1: Initialize $\boldsymbol{p}^*$ such that $\sum_{c\in\mathcal{C}} p_i^{(c)*} \leq 1, \forall\, i\, \in \mathcal{N}$, and $0 \leq p_i^{(c)*} \leq 1, \forall\, i \in \mathcal{N}, c \in \mathcal{C}$
2: Initialize $\boldsymbol{q}^*$ such that $\sum_{c\in\mathcal{C}} q_i^{(c)*} \leq 1, \forall\, i\, \in \mathcal{N}$, and $0 \leq q_i^{(c)*} \leq 1, \forall\, i \in \mathcal{N}, c \in \mathcal{C}$
3: Set the convergence threshold $\epsilon > 0$
4: Set the iteration counter $t := 1$
5: Set $u := -\infty$ and $\Delta := \infty$
6: <u>Phase I: Channel Probing</u>
7: **while** $\Delta > \epsilon$
8:   **for** each transmitter $i \in \mathcal{N}$
9:     **If** $t \in \mathcal{T}_i$ **then**
10:       Calculate $o_i^{(c)}, \forall\, c \in \mathcal{C}$ using (2.30)
11:       Calculate $v_{ji}^{(c)}$, $\forall\, j \in \mathcal{N}\backslash\{i\}, c \in \mathcal{C}$ using (2.31)
12:       Calculate $w_{ji}^{(c)}$, $\forall\, j \in \mathcal{N}\backslash\{i\}, c \in \mathcal{C}$ using (2.32)
13:       Solve problem (2.29) to obtain the solution $\boldsymbol{p}_i^*$
14:       Broadcast $\boldsymbol{p}_i^*$ to other users using the control channel
15:       Set $u^*(t) := \sum_{i\in\mathcal{N}} U_i\big(r_i(\boldsymbol{p}^*, \boldsymbol{q}_i^*)\big)$
16:       Set $t := t + 1$
17:     **end if**
18:   **end for**
19:   Set $\Delta := u^*(t) - u$ and $u := u^*(t)$
20: **end while**
21: <u>Phase II: Channel Selection</u>
22: **for** each receiver $i \in \mathcal{N}$
23:   **If** $t \in \mathcal{T}_i$ **then**
24:     Calculate $a_i^{(c)}, \forall\, c \in \mathcal{C}$ using (2.34)
25:     Set $c_i := \arg\max_{c\in\mathcal{C}} a_i^{(c)}$
26:     Set $q_i^{(c)*}, \forall\, c \in \mathcal{C}$ using (2.35)
27:     Broadcast $\boldsymbol{q}_i^*$ to other users using the control channel
28:     Set $p_i^{(c)*} := 0$, if $c \neq c_i,\ \forall\, c \in \mathcal{C}$
29:     Set $u^*(t) := \sum_{i\in\mathcal{N}} U_i\big(r_i(\boldsymbol{p}^*, \boldsymbol{q}_i^*)\big)$
30:     Set $t := t + 1$
31:   **end if**
32: **end for**
33: <u>Phase III: Transmission Probability Allocation</u>
34: Set $\Delta := \infty$
35: Repeat Lines 7 to 20 once

For the message exchanges, after solving for the corresponding local optimization problems, transmitter i and receiver i need to broadcast the solutions $\boldsymbol{p}_i^*$ and $\boldsymbol{q}_i^*$ in (2.29) and (2.35) using the control channel, respectively. Thus, the signalling overhead grows linearly with the number of users N in the system. The exchange of $\boldsymbol{p}_i^*$ and $\boldsymbol{q}_i^*$ can be achieved by using broadcast protocols, such as limited-scope message flooding [55]. However, it should be noted that interference also affects these message exchanges. For the case with a high level of interference in the CRN, broadcast protocols with high reliability can be considered, such as [56]. Alternatively, the transmitters or receivers that cannot receive the control messages correctly are not required to solve their corresponding local optimization problems. In our system model, since the secondary nodes are located in a neighbourhood, the broadcast of each control message to the whole CRN can be completed in a few hops. In this way, the duration of a time slot in the control stage should take into account both the amount of time required for solving a local optimization problem and for broadcasting a control message to all the secondary nodes in the CRN.

We have the following theorem that shows the convergence of Algorithm 2.1. Notice that even in a centralized setting, there is no guarantee that we can obtain the globally optimal solution of problem (2.11) due to its non-convexity.

Theorem 2.4 *The aggregate utility $u^*(t)$ converges to a fixed point u^*. That is, $\lim_{t\to\infty} u^*(t) = u^*$. Moreover, $u^*(t)$ is a non-decreasing sequence in t. That is, $u^*(t) \leq u^*(t+1)$ for all $t \geq 0$.*

Proof: In both phases I and III, because we fix $\boldsymbol{p}_{-i}^*$ and $\boldsymbol{q}^*$ to solve problem (2.29) for $\boldsymbol{p}_i^*$, and update the solution of transmission probabilities $\boldsymbol{p}^*$ in the Gauss-Seidel manner [54, pp. 185], we can show by [54, Proposition 3.9, pp. 219] that $u^*(t)$ converges to a fixed point. In each iteration t, since we are maximizing the objective function $\sum_{i\in\mathcal{N}} U_i\big(r_i(\boldsymbol{p}^*, \boldsymbol{q}_i^*)\big)$ over some variables, while the other variables are fixed, we must have $u^*(t) \leq u^*(t+1)$ for all $t \geq 0$. ■

2.5 Coalitional Game Theory for SINR Model

In the previous section, we have assumed that all the secondary nodes cooperate to maximize the aggregate utility. This gives rise to the question of what happens if the users are *rational* and aim to maximize their own utilities. In fact, if user i is rational and there is no coordination among the users, user i may choose to transmit in a particular channel c_i by setting $p_i^{(c_i)} = q_i^{(c_i)} = 1$ and $p_i^{(c)} = q_i^{(c)} = 0$ for $c \neq c_i$ in order to maximize U_i as suggested by the proof of Theorem 2.1. Hence, a significant amount of interference will be generated. In the worst case (e.g., when the number of channels C is small), it is possible that the utilities of all the users will be zero. To prevent this problem, the users may coordinate among themselves in the form of a *coalition*. The users belonging to the same coalition coordinate their transmission and listening probabilities to maximize the aggregate utility, which is then divided among themselves.

Example 2.2 *We continue with Example 2.1, and assume that the three users have selected channel 1 and transmit with power P. We assume that their peak data rates are $\mu_1 = 5$, $\mu_2 = 2$, and $\mu_3 = 1$. If all the users are willing to coordinate their transmission probabilities $\boldsymbol{p}$, the optimal transmission probabilities based on throughput maximization (i.e., $\alpha = 0$ in problem (2.11)) for both the SINR and protocol models are given by $p_1^* = 1$, $p_2^* = 0$, and $p_3^* = 0$. From (2.12), the corresponding utilities are $U_1 = 5$ and $U_2 = U_3 = 0$. However, if users 2 and 3 are rational, they may not be satisfied with zero utility. For the protocol model, users 2 and 3 have no* bargaining power *with user 1 to increase their utilities. On the other hand, the SINR model reveals that users 2 and 3 can threaten user 1 to transmit simultaneously and jam user 1's transmission. This effect is not captured by the protocol model. In the following, we apply coalitional game theory to study the incentives of rational user cooperation and the payoff distribution among the users. We note that coalitions can also be formed in the protocol model. However, in this case, the significance* *of the formation of coalitions may be undermined by the fact that the protocol model does not capture the cumulative effect of interference.*

2.5.1 Coalitional Game

Since the channels are orthogonal, we focus on one particular channel, and refer to the set of users that have selected that channel by $\mathcal{N}$ for notational simplicity. In this case, the average data rate of user i in (2.8) can be simplified to

$$r_i(\boldsymbol{p}) = \mu_i p_i \sum_{\mathcal{M} \in \mathbb{M}_i} \Bigg(\prod_{m \in \mathcal{M}} p_m \Bigg) \Bigg(\prod_{k \in \mathcal{N} \backslash \mathcal{M}, k \neq i} (1 - p_k) \Bigg), \tag{2.37}$$

where we drop the superscript for channel c and the term for the listening probability. We further restrict our attention to non-decreasing concave utility functions $U_i(r_i)$, where $U_i(0) = 0$.

We define the coalitional game $\mathcal{G}$ with *transferable utility* [24] as a pair $(\mathcal{N}, v)$, where $\mathcal{N}$ is the set of players or the *grand coalition*, and $v : 2^{\mathcal{N}} \to \mathbb{R}$ is the *value* of a coalition $\mathcal{S} \subseteq \mathcal{N}$ that the members of the coalition can distribute among themselves. In our problem, this value is defined as

$$\begin{aligned} v(\mathcal{S}) = \underset{\boldsymbol{p}}{\text{maximize}} \quad & \sum_{i \in \mathcal{S}} U_i\big(r_i(\boldsymbol{p})\big) \\ \text{subject to} \quad & 0 \le p_i \le 1, \qquad \forall\, i \in \mathcal{S}, \\ & p_j = 1, \qquad\quad\;\; \forall\, j \in \mathcal{N} \backslash \mathcal{S}. \end{aligned} \tag{2.38}$$

That is, $v(\mathcal{S}) = \sum_{i \in \mathcal{S}} U_i\big(r_i(\boldsymbol{p}^*(\mathcal{S}))\big)$, where $\boldsymbol{p}^*(\mathcal{S})$ is the optimal solution of problem (2.38). The users within coalition $\mathcal{S}$ coordinate among themselves to maximize the aggregate utility, subject to the *worst-case interference* from users $j \in \mathcal{N} \backslash \mathcal{S}$ when they choose transmission probabilities $p_j = 1$. All users in set $\mathcal{N} \backslash \mathcal{S}$ are not coordinating with the users within coalition $\mathcal{S}$. Instead, each user $j \in \mathcal{N} \backslash \mathcal{S}$ transmits with $p_j = 1$ in order to maximize its own utility, because $U_j(r_j(\boldsymbol{p}))$ is a non-decreasing function in p_j from Lemma 2.3(a). $v(\mathcal{S})$ can be obtained from Algorithm 2.1 with a few minor changes: Choose $C = 1$. Run line 13 if $i \in \mathcal{S}$, but replace it with "Set $p_i^* := 1$" if $i \in \mathcal{N} \backslash \mathcal{S}$. It should be noted that game $\mathcal{G}$ is a one-shot game. Also, we assume that the communication overhead is negligible when a coalition is formed.

The property of *superadditivity* [24] is often observed in coalitional games, including game $\mathcal{G}$. It is defined as follows:

Definition 2.1 *A game is superadditive if* $v(\mathcal{S}_1 \cup \mathcal{S}_2) \geq v(\mathcal{S}_1) + v(\mathcal{S}_2), \forall\, \mathcal{S}_1, \mathcal{S}_2 \subset \mathcal{N}$ *with* $\mathcal{S}_1 \cap \mathcal{S}_2 = \phi$.

Theorem 2.5 *Game* $\mathcal{G}$ *is superadditive.*

Proof: Let $\boldsymbol{p}^*(\mathcal{S}_1)$, $\boldsymbol{p}^*(\mathcal{S}_2)$, and $\boldsymbol{p}^*(\mathcal{S}_1 \cup \mathcal{S}_2)$ be the optimal probabilities maximizing $v(\mathcal{S}_1)$, $v(\mathcal{S}_2)$, and $v(\mathcal{S}_1 \cup \mathcal{S}_2)$, respectively, as defined in (2.38). For $\mathcal{S}_1 \cap \mathcal{S}_2 = \phi$, we construct a vector $\boldsymbol{p}(\mathcal{S}_1 \cup \mathcal{S}_2)$, where the i^{th} element is

$$p_i(\mathcal{S}_1 \cup \mathcal{S}_2) \triangleq \begin{cases} p_i^*(\mathcal{S}_1), & \text{if } i \in \mathcal{S}_1, \\ p_i^*(\mathcal{S}_2), & \text{if } i \in \mathcal{S}_2, \\ 1, & \text{otherwise.} \end{cases} \tag{2.39}$$

So $\boldsymbol{p}(\mathcal{S}_1 \cup \mathcal{S}_2)$ is feasible in problem (2.38) with $\mathcal{S} = \mathcal{S}_1 \cup \mathcal{S}_2$. From (2.38), we have $p_i^*(\mathcal{S}_1) = 1$ if $i \in \mathcal{N} \backslash \mathcal{S}_1$. Thus, we have

$$p_i^*(\mathcal{S}_1) \begin{cases} = p_i(\mathcal{S}_1 \cup \mathcal{S}_2), & \text{if } i \in \mathcal{S}_1, \\ \geq p_i(\mathcal{S}_1 \cup \mathcal{S}_2), & \text{if } i \in \mathcal{N} \backslash \mathcal{S}_1. \end{cases} \tag{2.40}$$

From Lemma 2.3(b), we have

$$r_i(\boldsymbol{p}^*(\mathcal{S}_1)) \leq r_i(\boldsymbol{p}(\mathcal{S}_1 \cup \mathcal{S}_2)), \forall\, i \in \mathcal{S}_1. \tag{2.41}$$

Since U_i is a non-decreasing function of r_i, we have

$$U_i\big(r_i(\boldsymbol{p}^*(\mathcal{S}_1))\big) \leq U_i\big(r_i(\boldsymbol{p}(\mathcal{S}_1 \cup \mathcal{S}_2))\big), \forall\, i \in \mathcal{S}_1, \tag{2.42}$$

which implies that

$$\sum_{i\in\mathcal{S}_1} U_i\big(r_i(\boldsymbol{p}^*(\mathcal{S}_1))\big) \leq \sum_{i\in\mathcal{S}_1} U_i\big(r_i(\boldsymbol{p}(\mathcal{S}_1 \cup \mathcal{S}_2))\big). \tag{2.43}$$

Similarly, we have

$$\sum_{i\in\mathcal{S}_2} U_i\big(r_i(\boldsymbol{p}^*(\mathcal{S}_2))\big) \leq \sum_{i\in\mathcal{S}_2} U_i\big(r_i(\boldsymbol{p}(\mathcal{S}_1 \cup \mathcal{S}_2))\big). \tag{2.44}$$

Overall, we have

$$\begin{aligned}
&\sum_{i\in\mathcal{S}_1} U_i\big(r_i(\boldsymbol{p}^*(\mathcal{S}_1))\big) + \sum_{i\in\mathcal{S}_2} U_i\big(r_i(\boldsymbol{p}^*(\mathcal{S}_2))\big) \\
\leq &\sum_{i\in\mathcal{S}_1\cup\mathcal{S}_2} U_i\big(r_i(\boldsymbol{p}(\mathcal{S}_1 \cup \mathcal{S}_2))\big) \leq \sum_{i\in\mathcal{S}_1\cup\mathcal{S}_2} U_i\big(r_i(\boldsymbol{p}^*(\mathcal{S}_1 \cup \mathcal{S}_2))\big),
\end{aligned} \tag{2.45}$$

which concludes the proof. ■

2.5.2 The Core

To determine the *stability* of the grand coalition, we use the solution concept of the *core* [24]. It is possible that a subset of users may opt out of the grand coalition to form a smaller coalition, if the users in the smaller coalition receive higher utilities than when they participate in the grand coalition. In that case, the core is *empty*. The core is formally defined as follows:

Definition 2.2 *The core is the set of feasible utility allocation vectors* $\boldsymbol{U} = (U_i, \forall\, i \in \mathcal{N})$ *where*

$$\mathcal{U}_{core} = \Big\{\boldsymbol{U} : \sum_{i\in\mathcal{N}} U_i = v(\mathcal{N}),\, \sum_{i\in\mathcal{S}} U_i \geq v(\mathcal{S}), \forall\, \mathcal{S} \subset \mathcal{N}\Big\}. \tag{2.46}$$

In some special cases, it can be shown that the core is non-empty. One such special case is when all the users are one-hop neighbours to each other (i.e., user i is a one-hop neighbour to user j, $\forall\, i, j \in \mathcal{N}$, $i \neq j$, where *one-hop neighbour* is defined in (2.5)). In this case, since $\mathbb{M}_{i,SINR}$ and $\mathbb{M}_{i,PTC}$ are null sets $\forall\, i \in \mathcal{N}$ from (2.4) and (2.6), the SINR model is identical to the protocol model. So, the average data rate of user i in (2.37) can further be simplified as

$$r_i(\boldsymbol{p}) = \mu_i p_i \prod_{j\in\mathcal{N}\backslash\{i\}} (1 - p_j). \tag{2.47}$$

Theorem 2.6 *If all the users are one-hop neighbours to each other, then the core is non-empty.*

Proof: Since $p_j = 1$, $\forall\, j \in \mathcal{N}\backslash\mathcal{S}$ from (2.38), we have $r_i = 0$, $\forall\, i \in \mathcal{S} \subset \mathcal{N}$ from (2.47) if all the users are one-hop neighbours to each other, which implies that $v(\mathcal{S}) = 0$, $\forall\, \mathcal{S} \subset \mathcal{N}$. Notice that any vector $\boldsymbol{U} = (U_i, \forall\, i \in \mathcal{N} : \sum_{i\in\mathcal{N}} U_i = v(\mathcal{N}),\, U_i \geq 0, \forall\, i \in \mathcal{N})$ satisfies $\sum_{i\in\mathcal{S}} U_i \geq v(\mathcal{S}) = 0, \forall\, \mathcal{S} \subset \mathcal{N}$. So $\boldsymbol{U} \in \mathcal{U}_{core}$, and the core is thus non-empty. ■

2.5.3 Shapley Value

As a solution concept, the core has a few drawbacks. It can be *empty* and the allocation of payoff according to the core may be *unfair*. In Example 2.2, with the use of the SINR model, we can show that $v(\{1, 2\}) = v(\{1, 3\}) = v(\{1, 2, 3\}) = 5$ using (2.38). The only allocation of utilities that lies in the core is $U_1 = 5$, $U_2 = U_3 = 0$. This allocation is stable since no smaller coalitions can be formed where the members can receiver a higher payoff than when they are in the grand coalition. However, it is unfair in the division of the payoff among the users as it does not take into account the contribution of each user to a coalition. In the following, we propose to use the *Shapley value* [24] to *fairly* divide the payoff among the players. Let the total number of users in coalition $\mathcal{S}$ be $S = |\mathcal{S}|$.

Definition 2.3 *The Shapley value is the payoff allocation vector* $\boldsymbol{\phi}(v) = \big(\phi_1(v), \ldots, \phi_N(v)\big)$, *where*

$$\phi_i(v) = \sum_{\mathcal{S} \subseteq \mathcal{N} \setminus \{i\}} \frac{S!(N-S-1)!}{N!} \Big[v(\mathcal{S} \cup \{i\}) - v(\mathcal{S}) \Big]. \tag{2.48}$$

In fact, $\phi_i(v)$ represents the expected *marginal contribution* of user i to different coalitions $\mathcal{S}$ without user i. The Shapley value has a number of nice properties. First, we have $\sum_{i \in \mathcal{N}} \phi_i(v) = v(\mathcal{N})$. Moreover, it is fair in the sense that users who make the same contribution to different coalitions receive the same payoff. Mathematically, if $v(\mathcal{S} \cup \{i\}) = v(\mathcal{S} \cup \{j\})$, $\forall\, \mathcal{S} \in \mathcal{N} \setminus \{i, j\}$, then $\phi_i(v) = \phi_j(v)$. As we have discussed in Example 2.2, with the use of the SINR model, users 2 and 3 can threaten to leave the coalition to jointly jam user 1's transmission. The Shapley value in this case is $\boldsymbol{\phi}(v) = (3.33, 0.83, 0.83)$ and both users 2 and 3 receive positive utilities. It is worth mentioning that since users 2 and 3 have no bargaining power in the protocol model, we can show that the Shapley value in this case is $\boldsymbol{\phi}(v) = (5, 0, 0)$ and both users 2 and 3 receive zero utility.

In general, the Shapley value is not directly related to the core. However, the Shapley value lies in the core in some special cases, including the case where all the users are one-hop neighbours to each other for our problem.

Theorem 2.7 *If all the users are one-hop neighbours to each other, then (a)* $\phi_i(v) = \frac{v(\mathcal{N})}{N}, \forall\, i \in \mathcal{N}$, *and (b)* $\boldsymbol{\phi}(v) \in \mathcal{U}_{core}$.

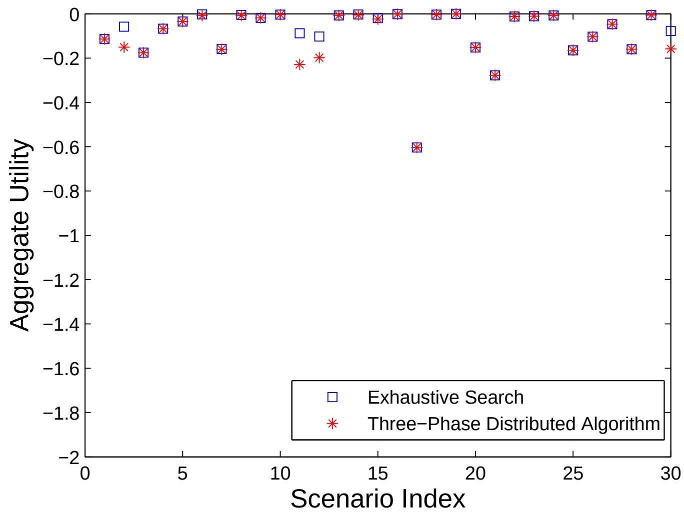

Figure 2.2: Aggregate utility obtained using an exhaustive search and the three-phase distributed algorithm (i.e., Algorithm 2.1) based on the multi-channel SINR model. We can see that Algorithm 2.1 achieves a near-optimal solution.

Proof: (a) If all the users are one-hop neighbours to each other, we have $v(\mathcal{S}) = 0$, $\forall\, \mathcal{S} \subset \mathcal{N}$, from the proof of Theorem 2.6. From (2.48), notice that the only non-zero term in the summation is given by $\mathcal{S} = \mathcal{N} \backslash \{i\}$. Therefore, we have $\phi_i(v) = \frac{(N-1)!(N-(N-1)-1)!}{N!}[v(\mathcal{N}) - v(\mathcal{N}\backslash\{i\})] = \frac{v(\mathcal{N})}{N}$.

(b) From part (a), we have $\sum_{i \in \mathcal{N}} \phi_i(v) = v(\mathcal{N})$. Also, $\sum_{i \in \mathcal{S}} \phi_i(v) = Sv(\mathcal{N})/N > 0 = v(\mathcal{S})$, $\forall\, \mathcal{S} \subset \mathcal{N}$. From (2.46), we know that $\boldsymbol{\phi}(v) \in \mathcal{U}_{core}$. ■

Thus, in this case, the payoff allocation vector $\boldsymbol{\phi}(v)$, which distributes the total payoff equally among the users, is in the core. Empirical investigations regarding the core and the Shapley value in the general setting, where not all the users are one-hop neighbours, are provided in the next section.

2.6 Performance Evaluations

In this section, we evaluate the performances of Algorithm 2.1 for the SINR and protocol models, and compare with that of a heuristic scheme. We also illustrate the significance of the core and the Shapley value. Unless specified otherwise, we assume that the secondary nodes are randomly placed in a 50 m $\times$ 50 m area. The peak data rate of a user is randomly selected to be between 1 Mbps and 10 Mbps. For simplicity, we do not take into account the effect of fading and model the channel gain as $G_{i,j}^{(c)} = 1/d_{i,j}^{\gamma}$, where $d_{i,j}$ is the distance between the transmitter of user i and the receiver of user j, and γ is the path loss exponent. We adopt $\gamma = 2$. When the effect of channel fading is considered, Algorithm 2.1 is still applicable. In this case, after estimating the channel gain $G_{i,j}^{(c)}$ in every coherence interval, we rerun Algorithm 2.1 to obtain an updated solution. The transmit powers of all the users are equal and set to a value which yields a minimum signal-to-noise-ratio (SNR) of 10 dB at the receivers. The SINR threshold is $\theta_i^{th} = \theta^{th}$, $\forall\, i \in \mathcal{N}$, and is set to 0 dB. The convergence threshold ϵ is set to 10^{-4}. All the users have the same α-fair utility functions with $\alpha_i = \alpha$, $\forall\, i \in \mathcal{N}$. For initialization, we use $p_i^{(c)*} = q_i^{(c)*} = 1/C, \forall\, i \in \mathcal{N}, c \in \mathcal{C}$ in lines 1 and 2 in Algorithm 2.1.

We first evaluate the optimality of the solution obtained with Algorithm 2.1. We consider the case of five users, two orthogonal channels with identical channel conditions, and $\alpha = 5$. The optimal solution under the SINR model is obtained with an exhaustive search. As shown in Fig. 2.2, the solution obtained with Algorithm 2.1 is near-optimal. In Fig. 2.3, we evaluate the convergence of Algorithm 2.1 for $N = 10$, $C = 3$, and $\alpha = 0$. From Theorem 2.4, the algorithm converges to a fixed point $\lim_{t \to \infty} u^*(t) = u^*$. Also, the aggregate utility $u^*(t)$ obtained in iteration t is a non-decreasing sequence, i.e., $u^*(t) \leq u^*(t+1)$. The improvement in $u^*(t)$ in phase III is more significant than that in phase I. In phase I, the transmitters may transmit in all channels, i.e., a significant amount of

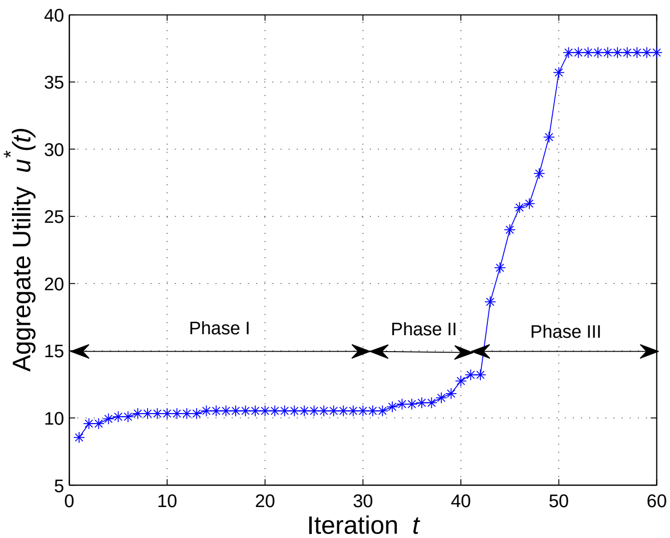

Figure 2.3: Convergence of the aggregate utility $u^*(t)$ using the three-phase distributed algorithm (i.e., Algorithm 2.1). Notice that the aggregate utility obtained in each iteration is non-decreasing. The users probe the channels in phase I and select the best channel in phase II. In phase III, the transmission probabilities are adjusted based on the channels selected in phase II.

interference is generated. However, in phase III, since the users have selected to transmit and listen to only one channel, the number of potential interferers in each channel is reduced. As a result, the improvement in $u^*(t)$ is more significant.

In Fig. 2.4, we consider the case $N = 10$, $C = 4$, and $\alpha = 0$ when the set of data channels $\mathcal{C}$ changes due to dynamic spectrum leasing. Specifically, we assume that two channels are removed from $\mathcal{C}$ when the lease expires, and one new channel is leased and added back to $\mathcal{C}$ later. As we can see, by running Algorithm 2.1 based on $\boldsymbol{p}^*$ from the previous solution after each change in set $\mathcal{C}$, the solution converges quickly to a fixed point again and adapts to these dynamic network changes.

Next, we compare the aggregate throughput achieved with Algorithm 2.1 for the SINR model (using $\mathsf{M}_i = \mathsf{M}_{i,SINR}$, $\forall\, i \in \mathcal{N}$) and protocol model (using $\mathsf{M}_i = \mathsf{M}_{i,PTC}$, $\forall\, i \in \mathcal{N}$), and the multi-channel MAC (MMAC) protocol [1] for $N = 10$ and $\alpha = 0$ averaged over 100

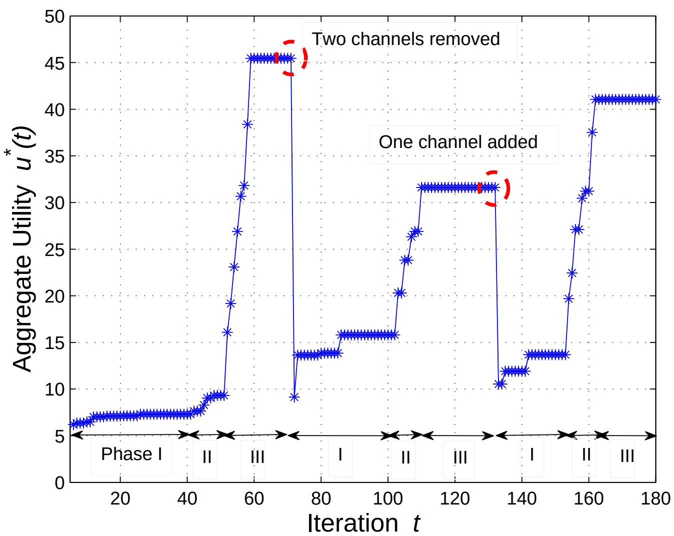

Figure 2.4: The change in aggregate utility $u^*(t)$ when the set of data channels $\mathcal{C}$ changes due to dynamic spectrum leasing. Initially, we assume that there are four data channels available. We assume that two data channels are removed from $\mathcal{C}$ and then one data channel is added back to $\mathcal{C}$. We run the three-phase distributed algorithm (i.e., Algorithm 2.1) based on the previous solution after set $\mathcal{C}$ has changed. We can see that $u^*(t)$ converges again quickly to a fixed point when the set $\mathcal{C}$ changes.

different random topologies when the number of orthogonal channels C varies. The MMAC protocol is a multi-channel extension of the IEEE 802.11 distributed coordination function and is suitable for the spectrum leasing model in CRNs. In MMAC, the users first select the channel with the least scheduled traffic, and then contend for it by using the carrier sense multiple access with collision avoidance (CSMA/CA) protocol. We assume that the channel is sensed busy if any one-hop neighbour transmits. In other words, the sensing is based on the protocol model. Since Algorithm 2.1 obtains a locally optimal solution from a given starting point, we execute Algorithm 2.1 from thirteen randomly generated feasible starting points $(\boldsymbol{p}^*, \boldsymbol{q}^*)$, and record the solution that yields the maximum aggregate utility to obtain a solution that is close to the globally optimal one. As shown in Fig. 2.5, when C increases, less interference is experienced by each user, so the overall system throughput is increased. Also, we notice that the design based on the SINR model always achieves a

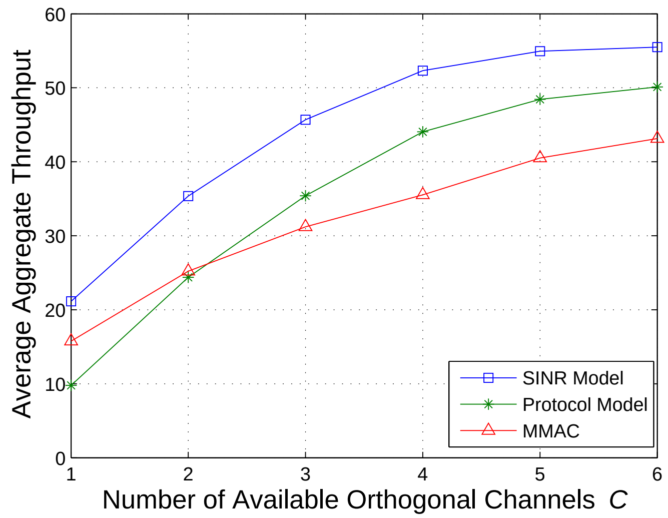

Figure 2.5: Average aggregate throughput versus the number of orthogonal channels available for Algorithm 2.1 using the SINR model, the protocol model, and the MMAC [1]. Notice that the design based on the SINR model achieves the highest aggregate throughput.

higher throughput than that using the protocol model and the MMAC protocol.

Finally, we investigate the payoff distribution for the Shapley value and the existence of the core for the network scenario shown in Fig. 2.6. Eight secondary users are randomly placed in a 75 m $\times$ 75 m open area, $\alpha = 0$, and the peak data rate of each user is fixed to 10 Mbps. The minimum SNR is guaranteed to be at least 20 dB and we consider different SINR thresholds θ^{th}, e.g., for different bit error rate requirements. The aggregate utility of the grand coalition $v(\mathcal{N}) = \sum_{i \in \mathcal{N}} \phi_i(v)$ and the Shapley value $\boldsymbol{\phi}(v)$ are shown in Fig. 2.7. By increasing θ^{th}, the receivers become less tolerant to interference from other users, so the interference range is increased and the spatial reuse factor is reduced. As a result, $v(\mathcal{N})$ is reduced as shown in Fig. 2.7. From (2.5), when θ^{th} is increased up to a certain value, all users are one-hop neighbours to each other. This holds true in this setting when $\theta^{th} \geq 15$ dB. As expected from Theorem 2.7, all the users equally share the aggregate utility in this case. Also, notice that user 5 generates and receives the least amount of interference due to

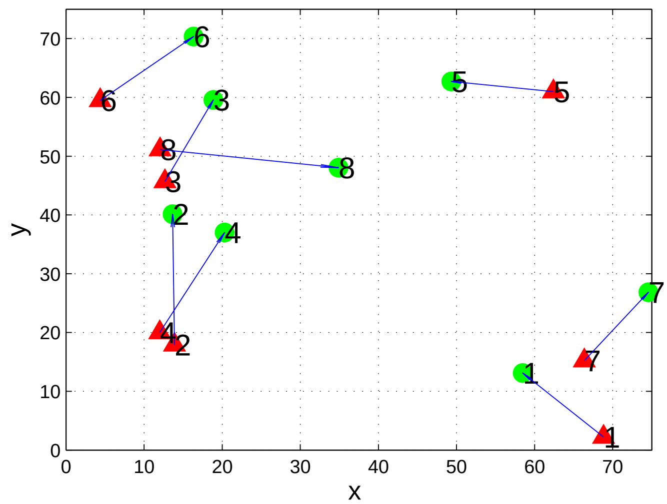

Figure 2.6: A CRN with eight users. User 5 generates and receives the least amount of interference due to its isolated position.

its isolated position. Thus, it has a large marginal contribution to different coalitions, and receives the largest proportion of the payoff for $\theta^{th} < 15$ dB. Moreover, it can be shown that the constraints in (2.46) can be satisfied and the core exists in this example for all the values of θ^{th} that we have studied. However, the Shapley value lies only in the core for $\theta^{th} \geq 10$ dB, which includes the cases $\theta^{th} \geq 15$ dB where all the users are one-hop neighbours to each other as stated in Theorem 2.7.

2.7 Summary

In this chapter, we have studied random access in CRNs using the SINR model. For cooperative users in a multi-channel model, a three-phase distributed algorithm has been proposed to obtain a near-optimal solution for the formulated non-convex NUM problem. It converges readily to a close-to-optimal value even when the set of data channels changes due to dynamic spectrum leasing. For rational users in a single-channel model, we have used the core and the Shapley value to characterize the stability and fair allocation of the

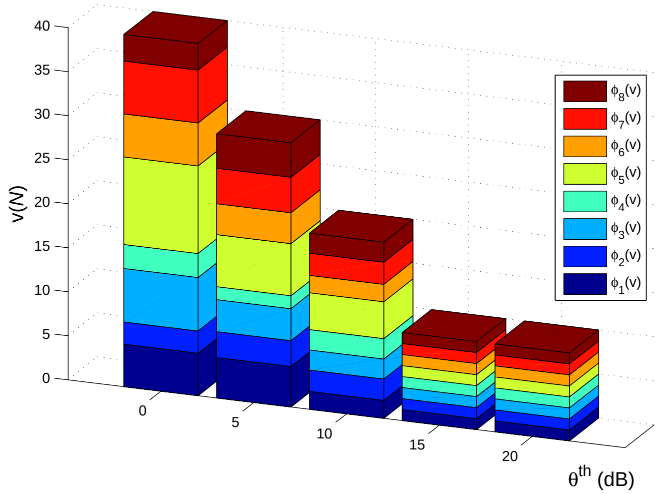

Figure 2.7: Aggregate utility of the grand coalition $v(\mathcal{N}) = \sum_{i \in \mathcal{N}} \phi_i(v)$ and the Shapley value $\boldsymbol{\phi}(v)$ for the secondary users in Fig. 2.6. When θ^{th} is increased, $v(\mathcal{N})$ is decreased because the interference range is increased. When θ^{th} is increased to 15 dB, $v(\mathcal{N})$ is equally shared among these one-hop neighbours as stated in Theorem 2.7. Notice that user 5 has the largest share of payoff for $\theta^{th} < 15$ dB due to its large marginal contribution to different coalitions.

payoff among the users, respectively. In our system model, we have assumed that (a) the set of users $\mathcal{N}$ is fixed and (b) the transmission between the transmitter and receiver of each user is only single-hop. For (a), it can be shown that by running Algorithm 2.1 starting from the previous solution after $\mathcal{N}$ has changed, the solution converges readily again to a fixed point. For (b), we may consider the multi-hop setting by introducing binary routing variables and flow conservation constraints as in [57].

Chapter 3

VCG-based Truthful Random Access Protocols for WLANs

In this chapter, we consider a WLAN with several users associated with an AP. The users support applications with both *elastic* and *inelastic* traffic [12]. The elastic traffic is generated by applications such as traditional file transfer and electronic mail. The inelastic traffic is generated by applications including real-time voice and video streaming, which usually have tight QoS requirements. We consider the setting where the users first declare their ACs (or utility functions) to the AP. In return, the AP assigns transmission probabilities to the users for random access such that the aggregate utility of all the users is maximized. If all the users accurately declare their AC information, the aforementioned setting can lead to adequate QoS and service differentiation based on application needs. However, some users may cheat on their declared ACs to obtain some larger shares of bandwidth. We first use non-cooperative game theory to analyze the behaviour of users in the described wireless system. Then, we apply the Vickrey-Clarke-Groves (VCG) mechanism [25, 26, 34] to motivate the users to be truthful through *pricing*. In fact, pricing is important for the efficient allocation of network resources among users [58]. The VCG mechanism has been successfully applied in various other areas, e.g., resource allocation for wireless multimedia applications [59] and multipath traffic assignment [60].

To implement the VCG mechanism in the described wireless system, we need to obtain the *exact* optimal solution of a formulated NUM problem, where the optimization variables are the transmission probabilities of all the users in the system. Note that replacing the optimal solution with even a *near-optimal* solution will make the mechanism useless and untruthful [61]. Previous works, such as [62, 63], have focused on solving the NUM problem to achieve efficiency and fairness for wireless random access for elastic traffic only, where the utility functions of the applications are *concave* [64]. However, here we include the more challenging case of inelastic traffic. The application requirements of users with inelastic traffic are modeled using *step* or *quasi-concave* utility functions. As a result, the formulated NUM problem in this chapter is *non-convex* and is difficult to solve in general. We propose an algorithm to obtain the optimal solution for the formulated non-convex problem, which is based on solving a number of *convex* optimization problems. In summary, the main contributions of this chapter are listed as follows.

- We formulate a wireless random access game using non-cooperative game theory and analyze the behaviour of rational users, where service differentiation for QoS support is implemented.
- We apply the VCG mechanism to encourage the users to truthfully reveal their ACs.
- We address several challenging computational issues in implementing the VCG mechanism which requires solving a complicated non-convex optimization problem. We propose a low-complexity enumeration algorithm to obtain the global optimal solution by iteratively solving a chain of convex optimization problems.
- Simulation results show that our scheme ensures that both service differentiation and maximum network utility can be achieved. Moreover, we demonstrate the low computational complexity of our scheme and its performance gain over a CSMA scheme in terms of the system utility.

The rest of this chapter is organized as follows. We present the related work in Section

3.1. The system model and the non-cooperative random access game are described in Section 3.2. The proposed VCG-based scheme is presented in Section 3.3. The algorithm required for implementing the VCG mechanism for random access is discussed in Section 3.4. Simulation results are presented in Section 3.5. A summary is given in Section 3.6.

3.1 Related Work

A number of previous works addressed non-cooperative random access from the *players*' viewpoint using game theory. Network equilibrium points were characterized and strategies were proposed to counteract the selfish behaviours of players. In [65], Cagalj *et al.* modeled carrier sense multiple access with collision avoidance (CSMA/CA) using game theory. Both normal-form and repeated-form CSMA/CA games were formulated and the existence of a Nash equilibrium (NE) was shown for each game. In [66], game theory was applied to analyze the behaviour of selfish nodes in a one-shot random access game. Necessary and sufficient conditions for the NE were proposed, and the asymptotic properties of the system were studied. Chen *et al.* proposed in [67] an analytical framework for random access using game theory. Distributed algorithms were proposed to achieve the NE. In [68], a cartel maintenance repeated game framework was proposed. A trigger-punishment rule was designed so that it is always in each user's best interest to cooperate.

On the other hand, it is possible to take a *proactive* approach from the *system designer*'s point of view and introduce some *mechanisms* to *prevent* players from misbehaving. Wang *et al.* proposed in [69] a strategyproof mechanism for wireless multicast routing. An agent's profit is maximized when it truthfully reports its cost. Nuggehalli *et al.* proposed in [70] an incentive mechanism to avoid selfishness. They showed that the users are encouraged to always be truthful on declaring their ACs in an attempt to increase throughput under some conditions. Bae *et al.* studied in [71] the design of a dynamic auction for wireless spectrum sharing between the high and low transmit power users. A mechanism was proposed that maximizes the incentives for truthful bidding. Huang *et al.* proposed in [72] two auction mechanisms, namely signal-to-noise ratio (SNR) auction and power auction, for distributed relay selection and relay power allocation in cooperative communications. The best response bid updates globally converge to the unique NE asynchronously. Ko *et al.* proposed in [73] a mechanism to gather the private traffic information of selfish users in two-tier Orthogonal Frequency Division Multiple Access (OFDMA) femtocell networks. It was shown that the resource allocation achieves weighted max-min fairness, weighted proportional fairness, and Pareto efficiency.

3.2 System Model

Consider a WLAN with one AP and N mobile stations (MSs)[1]. The set of MSs is denoted by $\mathcal{N} = \{1, 2, \ldots, N\}$. All MSs are one-hop neighbors to the AP. Time is divided into equal-length slots. We only consider the uplink scenario[2], where each MS $i \in \mathcal{N}$ attempts to access the shared wireless channel at the beginning of each time slot with *transmission probability* p_i. Note that the choice of transmission probabilities can be transformed into equivalent contention window sizes that can be implemented in IEEE 802.11 WLANs [74]. Let p_i^{succ} denote the probability that a transmission from station $i \in \mathcal{N}$ is successful, i.e., does not experience collision, in a time slot. We have

$$p_i^{\mathrm{succ}}(\boldsymbol{p}) \;=\; p_i \prod_{j \in \mathcal{N} \setminus \{i\}} (1 - p_j), \qquad \forall\, i \in \mathcal{N}, \tag{3.1}$$

[1]In this chapter, we use the terms *mobile stations*, *users*, and *players* interchangeably.

[2]We notice that selfish users may only affect the performance of the uplink transmissions. In fact, for downlink transmissions, the AP can simply perform scheduling with adequate QoS provisioning and there is no need for random access.

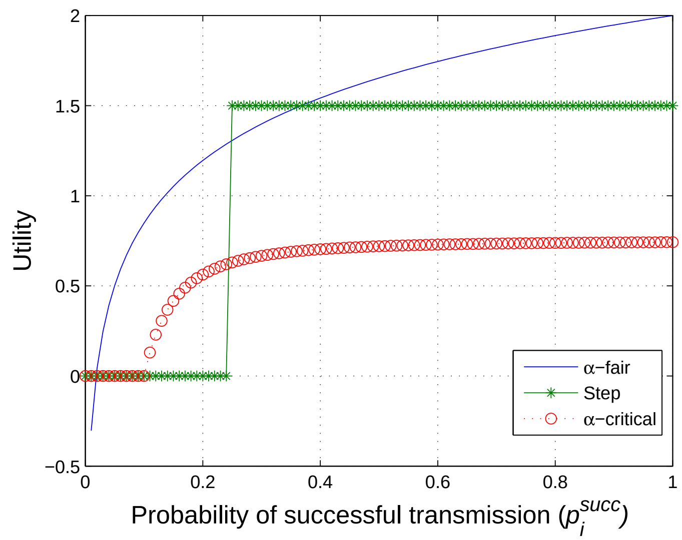

Figure 3.1: Three different types of utility functions considered in this work: (1) Concave function: α-fair function ($K_i = 0.1$, $\alpha_i = 1$, and $L_i = 4$), (2) Step function ($K_i = 1.5$ and $p_i^{\text{critical}} = 0.25$) , and (3) Quasi-concave function: α-critical function ($K_i = 0.015$, $\alpha_i = 3$, and $p_i^{\text{critical}} = 0.1$).

where vector $\boldsymbol{p} = (p_i,\ i \in \mathcal{N})$. Given the *nominal* data rate φ (e.g., 54 Mbps in IEEE 802.11g), the *average* data rate for user i is obtained as $\varphi\ p_i^{\text{succ}}(\boldsymbol{p})$. The MSs are assumed to run different types of applications, where each application may have different QoS requirements.

3.2.1 Utility Functions

For each MS $i \in \mathcal{N}$, we use a utility function $u_i\left(p_i^{\text{succ}}(\boldsymbol{p})\right)$ to model the level of satisfaction that station i experiences from its application when it attains success probability $p_i^{\text{succ}}(\boldsymbol{p})$. The utility functions are assumed to be nondecreasing.

We consider a system with MSs having both elastic and inelastic traffic. Let $\mathcal{N}_{\mathcal{E}}$ and $\mathcal{N}_{\mathcal{I}}$ denote the sets of users with elastic and inelastic traffic, respectively. We refer to these users as elastic and inelastic users in the following. Note that $\mathcal{N}_{\mathcal{E}} \cap \mathcal{N}_{\mathcal{I}} = \emptyset$ and $\mathcal{N}_{\mathcal{E}} \cup \mathcal{N}_{\mathcal{I}} = \mathcal{N}$, where $\emptyset$ is the null set. For each user $j \in \mathcal{N}_{\mathcal{E}}$ with an *elastic* application (e.g., file transfer and electronic mail), we can use a *concave* function to model the utility [64]. A common class of concave utility functions is α-fair utility function, which is defined as [53]

$$u_i(p_i^{\text{succ}}(\boldsymbol{p}), \alpha_i, K_i, L_i) = \begin{cases} K_i\Big(\ln\big(p_i^{\text{succ}}(\boldsymbol{p})\big) + L_i\Big), & \text{if } \alpha_i = 1, \\ K_i\Big(\frac{p_i^{\text{succ}}(\boldsymbol{p})^{(1-\alpha_i)}}{1-\alpha_i} + L_i\Big), & \text{if } \alpha_i > 1, \end{cases} \tag{3.2}$$

where u_i is the utility that user i receives, $\alpha_i \geq 1$ is a fixed utility parameter, and $K_i \geq 0$ is an amplitude parameter. L_i is a parameter that adjusts the vertical position of the utility curve. On the other hand, the applications supported by each user $i \in \mathcal{N}_{\mathcal{I}}$, such as voice and video streaming, may have tight QoS requirements and require some minimum level of available bandwidth. If the available bandwidth drops below the required threshold, then the connection will become useless, leading to zero utility for the corresponding user. In this chapter, we use two types of utility functions to model inelastic traffic: *step* functions and *quasi-concave* functions. A step utility function is characterized by parameters K_i and p_i^{critical}. Parameter $p_i^{\text{critical}} \geq 0$ refers to the *minimum* required $p_i^{\text{succ}}(\boldsymbol{p})$ for the application to run properly in station $i \in \mathcal{N}_{\mathcal{I}}$. Parameter K_i determines the amplitude of the utility function as long as the required p_i^{critical} is achieved. Step utility functions are used to mathematically model various *hard real-time* applications, such as audio and voice applications, which cannot operate if the minimum required data rate is not provided [12]. That is,

$$u_i\left(p_i^{\text{succ}}(\boldsymbol{p}), K_i, p_i^{\text{critical}}\right) = \begin{cases} K_i\,, & \text{if } p_i^{\text{succ}}(\boldsymbol{p}) \geq p_i^{\text{critical}}, \\ 0, & \text{if } p_i^{\text{succ}}(\boldsymbol{p}) < p_i^{\text{critical}}. \end{cases} \tag{3.3}$$

Furthermore, for *rate-adaptive* video, audio, and other applications with minimum bandwidth requirements, we can model the utility functions to be quasi-concave. We introduce a new quasi-concave utility function [21, pp. 95], which we refer to as the α-*critical* utility function, by modifying the α-fair utility function in (3.2). If $\alpha_i = 1$, we have

$$u_i\left(p_i^{\text{succ}}(\boldsymbol{p}), \alpha_i, K_i, p_i^{\text{critical}}\right) = \begin{cases} K_i \ln\left(\frac{p_i^{\text{succ}}(\boldsymbol{p})}{p_i^{\text{critical}}}\right), & \text{if } p_i^{\text{succ}}(\boldsymbol{p}) \geq p_i^{\text{critical}}, \\ 0, & \text{if } p_i^{\text{succ}}(\boldsymbol{p}) < p_i^{\text{critical}}. \end{cases} \tag{3.4}$$

If $\alpha_i > 1$, then the α-critical utility function is modeled as

$$u_i(p_i^{\text{succ}}(\boldsymbol{p}), \alpha_i, K_i, p_i^{\text{critical}}) = \begin{cases} \frac{K_i}{1-\alpha_i}\left[(p_i^{\text{succ}}(\boldsymbol{p}))^{(1-\alpha_i)} - \left(p_i^{\text{critical}}\right)^{(1-\alpha_i)}\right], & \text{if } p_i^{\text{succ}}(\boldsymbol{p}) \geq p_i^{\text{critical}}, \\ 0, & \text{if } p_i^{\text{succ}}(\boldsymbol{p}) < p_i^{\text{critical}}. \end{cases} \tag{3.5}$$

Clearly, α-critical and step utility functions are *non-concave* and *non-differentiable*. Some examples of the utility functions that we consider in this chapter are shown in Fig. 3.1.

For simplicity of presentation, we denote the set of utility parameters for each user $i \in \mathcal{N}$ by θ_i. Using the terminology of game theory, we refer to θ_i as a *type* for user i. Notice that there is a one-to-one correspondence between a type and an AC. If utility function u_i is a concave α-fair function as in (3.2), we have $\theta_i = \{K_i, \alpha_i, L_i\}$. If utility function u_i is a step function as in (3.3), then $\theta_i = \{K_i, p_i^{\text{critical}}\}$. Finally, if it is an α-critical function as in (3.4) and (3.5), then $\theta_i = \{K_i, \alpha_i, p_i^{\text{critical}}\}$.

3.2.2 Network Utility Maximization Problem

Given complete knowledge of all system parameters (i.e., $\theta_i, \forall\, i \in \mathcal{N}$) and centralized control of the WLAN, an efficient choice of all transmission probabilities $\boldsymbol{p}$ is characterized as an optimal solution of the following NUM problem across all users [12, 62, 75]:

$$\underset{\boldsymbol{p} \in \mathcal{P}}{\text{maximize}} \quad \sum_{i \in \mathcal{N}} u_i\left(p_i^{\text{succ}}(\boldsymbol{p}), \theta_i\right), \tag{3.6}$$

where $\mathcal{P} = \{\boldsymbol{p} : 0 \leq p_i \leq 1, \forall\, i \in \mathcal{N}\}$ represents the set of all feasible transmission probabilities. The objective function in optimization problem (3.6) is also called *network social welfare* [27]. As the ACs of the applications running on the MSs are private information, they are *not* known to the AP. That is, the AP is not aware of the MSs' utility functions. Thus, the AP may not be able to solve NUM problem (3.6) unless each MS $i \in \mathcal{N}$ declares its true type θ_i to the AP. Clearly, if all the stations are *truthful*, then the obtained vector of optimal transmission probabilities leads to the optimal network performance. However, if a user $i \in \mathcal{N}$ is *selfish*, then it may declare its type to be $\hat{\theta}_i \neq \theta_i$ to obtain a higher utility. In that case, the obtained transmission probabilities are not optimal. In fact, the network performance can be very poor in the latter case, as shown in Section 3.5.

3.2.3 Non-cooperative Random Access Game

Using game theory, we next formulate the described N-user random access system as a finite N-person non-cooperative normal-form game $(\mathcal{N}, \Omega, \boldsymbol{u})$, where $\mathcal{N}$ is the set of players, $\Omega = \Theta_1 \times \Theta_2 \times \cdots \times \Theta_N$ is the Cartesian product of the action sets of all players, Θ_i is the action set of player i, and $\boldsymbol{u} = (u_1, u_2, \ldots, u_N)$ is the vector of utility functions for all the stations. In this chapter, we assume for simplicity that all players have the same action set Θ. That is, $\Theta_i = \Theta$, $\forall\, i \in \mathcal{N}$. The action of each station $i \in \mathcal{N}$ is to *strategically* select its declared type $\hat{\theta}_i$ (which is not necessarily the same as its true type θ_i) to maximize its own utility. That is, given $\hat{\boldsymbol{\theta}}_{-i} = (\hat{\theta}_1, \ldots, \hat{\theta}_{i-1}, \hat{\theta}_{i+1}, \ldots, \hat{\theta}_N)$ as the vector of declared types for all stations other than station i, where $\hat{\boldsymbol{\theta}}_{-i} \in \Omega_{-i} = \Theta_1 \times \cdots \times \Theta_{i-1} \times \Theta_{i+1} \times \cdots \times \Theta_N$, station i selects $\hat{\theta}_i$ to solve the following *local* problem related to its *actual* utility function $u_i\,(p_i^{\text{succ}}, \theta_i)$:

$$\underset{\hat{\theta}_i \in \Theta_i}{\text{maximize}} \quad u_i \left(p_i^{\text{succ}}(\hat{\boldsymbol{p}}(\hat{\theta}_i, \hat{\boldsymbol{\theta}}_{-i})), \theta_i \right), \tag{3.7}$$

which is based on the prior knowledge that the AP will determine the vector of the players' transmission probabilities $\hat{\boldsymbol{p}}$ by solving the following *global* optimization problem, according to the application type declarations $\hat{\boldsymbol{\theta}}$ of all the players:

$$\hat{\boldsymbol{p}}(\hat{\theta}_i, \hat{\boldsymbol{\theta}}_{-i}) = \hat{\boldsymbol{p}}(\hat{\boldsymbol{\theta}}) = \underset{\boldsymbol{p} \in \mathcal{P}}{\arg\max} \sum_{i \in \mathcal{N}} u_i \left(p_i^{\text{succ}}(\boldsymbol{p}), \hat{\theta}_i \right). \tag{3.8}$$

Notice that the difference between problems (3.6) and (3.8) is that in (3.8) we have replaced the *true* type θ_i with the *declared* type $\hat{\theta}_i$ for each $i \in \mathcal{N}$. That is, the solution of problem (3.6) is what the system aims to achieve, while that of problem (3.8) is what the system actually achieves.

The complete analysis of game ($\mathcal{N}$, Ω, $\boldsymbol{u}$) is very difficult in general. Nevertheless, we can show the following interesting theorem, which states that player i is interested in declaring $\hat{K}_i$ to be as large as possible in order to have a higher success probability $p_i^{\text{succ}}(\boldsymbol{p})$, and thus a higher utility u_i.

Theorem 3.1 *For any α-fair, step, or α-critical utility functions in game ($\mathcal{N}$, Ω, $\boldsymbol{u}$), if a player $i \in \mathcal{N}$ declares $\hat{\theta}_i$ such that $\hat{K}_i > K_i \geq 0$ and the declarations of other players $\hat{\boldsymbol{\theta}}_{-i}$ remain the same, we have*

$$p_i^{succ}(\hat{\boldsymbol{p}}(\hat{\theta}_i, \hat{\boldsymbol{\theta}}_{-i})) \geq p_i^{succ}(\hat{\boldsymbol{p}}(\theta_i, \hat{\boldsymbol{\theta}}_{-i})), \;\; \forall\, \hat{\theta}_i \neq \theta_i, \hat{\boldsymbol{\theta}}_{-i} \in \Omega_{-i}, \tag{3.9}$$

where $p_i^{succ}(\hat{\boldsymbol{p}}(\hat{\theta}_i, \hat{\boldsymbol{\theta}}_{-i}))$ and $p_i^{succ}(\hat{\boldsymbol{p}}(\theta_i, \hat{\boldsymbol{\theta}}_{-i}))$ are the probabilities of successful transmission obtained when player i declares $\hat{K}_i$ and K_i, respectively.

Proof: We notice that parameters $\hat{K}_i, i \in \mathcal{N}$, act as weighting parameters in problem (3.8). Together with the fact that α-fair, step, and α-critical utility functions are all nondecreasing functions in $p_i^{\text{succ}}(\boldsymbol{p})$, it follows that the higher the value of $\hat{K}_i$, the higher the value of $p_i^{\text{succ}}(\boldsymbol{p})$ would be at optimality. Thus, for $\hat{K}_i > K_i$, we have $p_i^{\text{succ}}(\hat{\boldsymbol{p}}(\hat{\theta}_i, \hat{\boldsymbol{\theta}}_{-\boldsymbol{i}})) \geq p_i^{\text{succ}}(\hat{\boldsymbol{p}}(\theta_i, \hat{\boldsymbol{\theta}}_{-\boldsymbol{i}}))$. ■

From Theorem 3.1, if there is a range of values that $\hat{K}_i$ can be chosen from, then declaring $\hat{K}_i$ to its maximum possible value is a *dominant* strategy [25, 34] of player i. A dominant strategy is a strategy that is chosen regardless of the strategies of other players. Clearly, playing such an untruthful dominant strategy results in a significant degradation of the network performance and prevents adequate service distinction among different ACs.

In our system model, we assume that the players can possibly declare *all* the utility parameters (i.e., their types) in such a way that increases their own utilities. The analysis related to the strategic declarations of $\hat{\alpha}_i$ and $\hat{p}_i^{\text{critical}}$ are not as straightforward as that of $\hat{K}_i$ stated in Theorem 3.1 and we leave it for future work. Next, we show that by using a VCG mechanism, the players are encouraged to declare their types truthfully for their own good.

3.3 Truthful Mechanism Design for WLANs

The results in Theorem 3.1 reveal that it is crucial to develop efficient schemes to motivate the stations to be truthful, i.e., to declare their true types. In this section, we use *mechanism design* [25, 26, 34] for this purpose. Mechanism design is a sub-field in microeconomics and game theory that studies the problem of optimal resource allocation in the presence of selfish players, who aim to maximize only their own payoffs. Mechanisms are responsible for the allocation of resources and incur *payment* to the players, so as to provide them with *incentives* to declare their private information (i.e., their types) truthfully. Groves mechanism and its subfamily, named Vickrey-Clarke-Groves (VCG) mechanism, are among the most efficient mechanisms that not only prevent the dishonesty of players, but also guarantee achieving the maximum network social welfare. The latter implies achieving the optimal performance in terms of solving the NUM problem in (3.6).

The VCG mechanism consists of two main components [34]: an *allocation rule* and a *payment rule*. For the *allocation* rule, given the declared types of all players $\hat{\boldsymbol{\theta}} = (\hat{\theta}_1, \ldots, \hat{\theta}_N)$, the AP selects the transmission probabilities according to the optimal solution of problem (3.8). Moreover, according to the payment rule, it also imposes a *payment* $t_i(\hat{\boldsymbol{\theta}})$ on each MS $i \in \mathcal{N}$ such that

$$t_i(\hat{\boldsymbol{\theta}}) = \sum_{j \in \mathcal{N} \setminus \{i\}} u_j \left(p_j^{\text{succ}}(\tilde{\boldsymbol{p}}_i(\hat{\boldsymbol{\theta}})), \hat{\theta}_j \right) - \sum_{j \in \mathcal{N} \setminus \{i\}} u_j \left(p_j^{\text{succ}}(\hat{\boldsymbol{p}}(\hat{\boldsymbol{\theta}})), \hat{\theta}_j \right), \tag{3.10}$$

where

$$\tilde{\boldsymbol{p}}_i(\hat{\boldsymbol{\theta}}) = \underset{\boldsymbol{p} \in \mathcal{P},\, p_i = 0}{\arg\max} \sum_{j \in \mathcal{N} \setminus \{i\}} u_j \left(p_j^{\text{succ}}(\boldsymbol{p}), \hat{\theta}_j \right), \tag{3.11}$$

and $\hat{\boldsymbol{p}}(\hat{\boldsymbol{\theta}})$ is as in (3.8). The above payment values are calculated based on the *declared* types $\hat{\boldsymbol{\theta}}$, *not* the true types $\boldsymbol{\theta}$, as the AP is not aware of the true types of the MSs. Also notice that in (3.10), the first term, i.e., $\sum_{j \in \mathcal{N} \setminus \{i\}} u_j(p_j^{\text{succ}}(\tilde{\boldsymbol{p}}_i(\hat{\boldsymbol{\theta}})), \hat{\theta}_j)$, *charges* player i with the aggregate utility achieved when player i is *removed* from the network, and the second term, i.e., $\sum_{j \in \mathcal{N} \setminus \{i\}} u_j(p_j^{\text{succ}}(\hat{\boldsymbol{p}}(\hat{\boldsymbol{\theta}})), \hat{\theta}_j)$, *pays* player i with the aggregate utility achieved excluding that of player i when player i is *added* to the network. Thus, player i is made to pay its *social cost*, which is the aggregate impact that its participation has on other players' utilities [25]. Note that there are various ways to implement such payments in practice. For example, one can assume that there is a trusted charging entity which maintains a billing account for every player in the system (e.g., as in [76]).

Given the vector of payment rules $\boldsymbol{t}(\hat{\boldsymbol{\theta}}) = (t_1(\hat{\boldsymbol{\theta}}), \ldots, t_N(\hat{\boldsymbol{\theta}}))$, each station needs to pay $t_i(\hat{\boldsymbol{\theta}})$ to the AP for relaying its transmitted packets. Intuitively, VCG selects the payment values such that it is the *best* choice for the players to be honest and declare their true types. In fact, since each player needs to pay the AP for the packets it transmits, player $i \in \mathcal{N}$ needs to declare its type $\hat{\theta}_i$ such that its *surplus* (i.e., its utility minus payment) is maximized. In other words, when VCG mechanism is used, instead of solving problem

(3.7), player i has to solve the following problem to maximize its own payoff:

$$\underset{\hat{\theta}_i \in \Theta_i}{\text{maximize}} \;\; u_i\left(p_i^{\text{succ}}(\hat{\boldsymbol{p}}(\hat{\theta}_i, \hat{\boldsymbol{\theta}}_{-i})), \theta_i\right) - t_i(\hat{\theta}_i, \hat{\boldsymbol{\theta}}_{-i}). \tag{3.12}$$

In fact, VCG mechanism forces all players to be *honest* as shown by the following theorem.

Theorem 3.2 *Assume that the allocation and payment rules are implemented as in (3.8) and (3.10), respectively, then declaring* $\hat{\theta}_i = \theta_i$ *is a dominant strategy for player* $i \in \mathcal{N}$*.*

Proof: Please refer to [34, pp. 42]. ■

From Theorem 3.2, if VCG mechanism is used, the most beneficial action for all players is to declare their true types such that $\hat{\theta}_i = \theta_i$ for all $i \in \mathcal{N}$, irrespective of the declarations by other players. Thus, solving problem (3.8) based on the declared types suffices to achieve optimal performance, i.e., the maximum network utility as described in (3.6). This is the key property of the VCG mechanism.

We are now ready to propose our VCG-based mechanism for QoS provisioning in WLANs with random access. It includes the following key steps:

1. *Type declaration*: Before starting transmission, all stations declare their types to the AP.

2. *VCG mechanism*: Given the declared types $\hat{\boldsymbol{\theta}}$, the AP calculates the transmission probability $\hat{\boldsymbol{p}}(\hat{\boldsymbol{\theta}})$ as in (3.8). It also calculates the payments $\boldsymbol{t}(\hat{\boldsymbol{\theta}})$ using (3.10) and (3.11).

3. *Resource allocation and payment*: The obtained vectors of transmission probabilities and payments are broadcast by the AP to all stations. Stations may only transmit based on the transmission probabilities assigned by the AP; otherwise, they will be

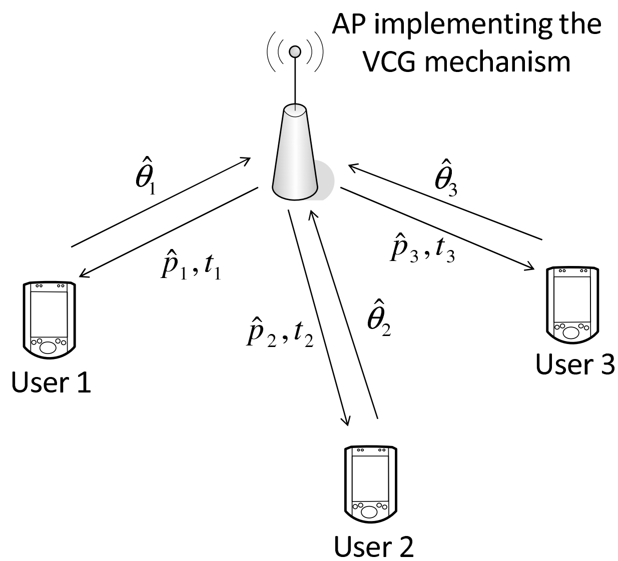

Figure 3.2: A WLAN with a set of users $\mathcal{N} = \{1, 2, 3\}$. In our system model, user i first declares to the AP the utility parameter (or type) $\hat{\theta}_i$, which characterizes the AC of the application of user i. After receiving $\hat{\theta}_i$, $\forall\, i \in \mathcal{N}$, the AP assigns the transmission probabilities $\hat{\boldsymbol{p}} = (\hat{p}_1, \hat{p}_2, \hat{p}_3)$ for random access and charges the users $\boldsymbol{t} = (t_1, t_2, t_3)$ according to the allocation and payment rules of the VCG mechanism described in (3.8) and (3.10), respectively. By using the VCG mechanism, it is shown in Theorem 3.2 that a rational player i should declare its true utility parameter θ_i in order to maximize its own utility.

refused service[3].

The system model of the VCG-based random access scheme is shown in Fig. 3.2. Next, we try to answer the following key question: *How can we solve the computationally challenging optimization problems (3.8) and (3.11)?*

3.4 Implementation of the VCG Mechanism

In order to implement the VCG mechanism, we need to compute $\hat{\boldsymbol{p}}(\hat{\boldsymbol{\theta}})$ in (3.8) and $\tilde{\boldsymbol{p}}(\hat{\boldsymbol{\theta}})$ in (3.11). However, in general, solving the optimization problems in (3.8) and (3.11) is not an easy task due to the *non-convexity* of the product forms in (3.1) and the *non-differentiability*

[3]It is easy for the AP to check whether the stations are indeed transmitting according to the assigned transmission probabilities by listening to the shared communication medium as explained in [75].

of step and α-critical utility functions. In this section, we propose an algorithm to obtain the globally optimal solutions of problems (3.8) and (3.11) by iteratively solving a number of convex optimization problems. Here, we will focus on problem (3.8) because similar techniques can be applied to solve problem (3.11).

In this chapter, we assume that the declared utility function for each elastic user $j \in \mathcal{N}_{\mathcal{E}}$ is an α-fair utility function as in (3.2). We also assume that the declared utility function for each inelastic user $i \in \mathcal{N}_{\mathcal{I}}$ is either a step function as described in (3.3) or an α-critical function as defined in (3.4) and (3.5). Although we only consider α-fair functions for concave functions and α-critical functions for quasi-concave functions, our approach can be applied to any similar continuous nondecreasing function as long as its concave part satisfies the following condition on the curvature of the utility function [62]:

$$\frac{d^2 u(p^{\text{succ}}, \hat{\theta})}{d(p^{\text{succ}})^2} p^{\text{succ}} \leq -\frac{du(p^{\text{succ}}, \hat{\theta})}{dp^{\text{succ}}}. \tag{3.13}$$

With both elastic and inelastic users in the system, problem (3.8) can be written as

$$\underset{\boldsymbol{p} \in \mathcal{P}}{\text{maximize}} \sum_{j \in \mathcal{N}_{\mathcal{E}}} u_j(p_j^{\text{succ}}(\boldsymbol{p}), \hat{\theta}_j) + \sum_{i \in \mathcal{N}_{\mathcal{I}}} u_i(p_i^{\text{succ}}(\boldsymbol{p}), \hat{\theta}_i), \tag{3.14}$$

where $p_i^{\text{succ}}(\boldsymbol{p})$ is defined as in (3.1) for all users $i \in \mathcal{N}$. Notice that problem (3.14) is a *non-convex* and *non-differentiable* optimization problem due to the non-convexity and non-differentiability of α-critical or step utility functions as we discussed in Section 3.2.1.

Let $\boldsymbol{p}^*$ denote the optimal solution of problem (3.14). Also let $p_i^{\text{succ}}(\boldsymbol{p}^*) = p_i^* \prod_{j \in \mathcal{N} \setminus \{i\}}(1 - p_j^*)$ denote the corresponding optimal success probability for station $i \in \mathcal{N}$. We can show the following lemma which helps us compute the optimal allocation of transmission probabilities.

Lemma 3.1 *At any optimal solution of problem (3.14), for all inelastic users $i \in \mathcal{N}_{\mathcal{I}}$, we* *have either* $p_i^{succ}(\boldsymbol{p}^*) \geq \hat{p}_i^{critical}$ *or* $p_i^{succ}(\boldsymbol{p}^*) = 0$.

Proof: We prove by contradiction. Assume that at optimality, we have $0 < p_i^{\text{succ}}(\boldsymbol{p}^*) < \hat{p}_i^{\text{critical}}$ for some user $i \in \mathcal{N}_{\mathcal{I}}$. Since the minimum required success probability is not satisfied for user $i \in \mathcal{N}_{\mathcal{I}}$, we have $u_i = 0$. Thus, the objective function of problem (3.14) at optimality becomes

$$\sum_{j \in \mathcal{N}_{\mathcal{E}}} u_j(p_j^{\text{succ}}(\boldsymbol{p}^*), \hat{\theta}_j) + \sum_{k \in \mathcal{N}_{\mathcal{I}} \setminus \{i\}} u_k(p_k^{\text{succ}}(\boldsymbol{p}^*), \hat{\theta}_k). \tag{3.15}$$

On the other hand, from (3.1), the success probability $p_j^{\text{succ}}(\boldsymbol{p})$ is a decreasing function of p_i for any $j \neq i$. Therefore, the summation in (3.15) is decreasing in p_i^* and at optimality we have $p_i^* = 0$. This implies that $p_i^{\text{succ}}(\boldsymbol{p}^*) = 0$ which contradicts our assumption that $0 < p_i^{\text{succ}}(\boldsymbol{p}^*) < \hat{p}_i^{\text{critical}}$. ■

From Lemma 3.1, when VCG mechanism is being used, the AP either does not admit an inelastic user i, or if it does admit user i, then it guarantees to provide it with its minimum required success probability $\hat{p}_i^{\text{critical}}$. Thus, we can obtain the optimal value of problem (3.14) by considering all subsets of users $\mathcal{M} \subseteq \mathcal{N}_{\mathcal{I}}$ admitted:

$$\underset{\mathcal{M} \subseteq \mathcal{N}_{\mathcal{I}}}{\text{maximize}} \quad v(\mathcal{M}), \tag{3.16}$$

where

$$\begin{aligned} v(\mathcal{M}) \triangleq \underset{\boldsymbol{x},\, \boldsymbol{p} \in \mathcal{P}}{\text{maximize}} \quad & \sum_{j \in \mathcal{N}_{\mathcal{E}}} u_j(x_j, \hat{\theta}_j) + \sum_{i \in \mathcal{N}_{\mathcal{I}}} u_i(x_i, \hat{\theta}_i) \\ \text{subject to} \quad & 0 \leq x_i \leq p_i^{\text{succ}}(\boldsymbol{p}), && \forall\, i \in \mathcal{N}_{\mathcal{E}} \cup \mathcal{M}, \\ & \hat{p}_i^{\text{critical}} \leq p_i^{\text{succ}}(\boldsymbol{p}), && \forall\, i \in \mathcal{M}, \\ & p_i = 0, && \forall\, i \in \mathcal{N}_{\mathcal{I}} \setminus \mathcal{M}. \end{aligned} \tag{3.17}$$

In problem (3.17), the first constraint is introduced for the auxiliary variable x_i [62]. We divide the set of inelastic users $\mathcal{N}_{\mathcal{I}}$ into two subsets: subset $\mathcal{M}$ and subset $\mathcal{N}_{\mathcal{I}}\backslash\mathcal{M}$. Here, set $\mathcal{M}$ denotes the set of those users which are admitted to the system, and acts as an auxiliary set to model *admission control*. For each inelastic user $i \in \mathcal{M}$, problem (3.17) includes the extra constraint $p_i^{\text{succ}}(\boldsymbol{p}) \geq \hat{p}_i^{\text{critical}}$ such that all admitted inelastic users achieve their minimum required success probabilities $\hat{p}_i^{\text{critical}}$. On the other hand, for each inelastic user $i \in \mathcal{N}_{\mathcal{I}}\backslash\mathcal{M}$, which is not admitted, we include the constraint $p_i = 0$ to make sure that no transmission probability is allocated to it.

By taking the logarithm of both sides of the first and second constraints in (3.17) and a logarithm change of variables $u_i'(x_i', \hat{\theta}_i) = u_i(e^{x_i'}, \hat{\theta}_i)$ and $x_i' = \ln x_i$, we can reformulate problem (3.17) as

$$\begin{aligned} v(\mathcal{M}) = \underset{\boldsymbol{x'},\, \boldsymbol{p} \in \mathcal{P}}{\text{maximize}} \quad & \textstyle\sum_{j \in \mathcal{N}_{\mathcal{E}}} u_j'(x_j', \hat{\theta}_j) + \sum_{i \in \mathcal{N}_{\mathcal{I}}} u_i'(x_i', \hat{\theta}_i) \\ \text{subject to} \quad & x_i' \leq \ln p_i + \sum_{j \in \mathcal{N}\backslash\{i\}} \ln(1-p_j), \qquad \forall\, i \in \mathcal{N}_{\mathcal{E}} \cup \mathcal{M}, \\ & \ln \hat{p}_i^{\text{critical}} \leq \ln p_i + \sum_{j \in \mathcal{N}\backslash\{i\}} \ln(1-p_j), \quad \forall\, i \in \mathcal{M}, \\ & p_i = 0, \qquad \forall\, i \in \mathcal{N}_{\mathcal{I}}\backslash\mathcal{M}. \end{aligned} \tag{3.18}$$

From [62], problem (3.18) is *convex*, so it can be solved using the *interior point method* [21].

Let $N_I = |\mathcal{N}_{\mathcal{I}}|$. Notice that we need to evaluate 2^{N_I} possible subsets $\mathcal{M}$ of $\mathcal{N}_{\mathcal{I}}$ in problem (3.16). However, there are many redundant computations that can be eliminated. Let $\mathcal{M}(\theta) = \left(i \in \mathcal{M} : \hat{\theta}_i = \theta\right)$ be the subset of users in set $\mathcal{M} \subseteq \mathcal{N}_{\mathcal{I}}$ with declared type θ. We define the *equivalent AC sets* as follows:

Definition 3.1 *A pair of sets $\mathcal{M}_1, \mathcal{M}_2 \subseteq \mathcal{N}_\mathcal{I}$ are equivalent AC sets if*

$$|\mathcal{M}_1(\theta)| = |\mathcal{M}_2(\theta)|, \forall\, \theta \in \Theta. \tag{3.19}$$

In other words, sets $\mathcal{M}_1$ and $\mathcal{M}_2$ have the same number of users in every ACs.

Given the above definition, we can show the following lemma for equivalent AC sets:

Lemma 3.2 *If $\mathcal{M}_1, \mathcal{M}_2 \subseteq \mathcal{N}_\mathcal{I}$ are equivalent AC sets, then we necessarily have $v(\mathcal{M}_1) = v(\mathcal{M}_2)$.*

Proof: By definition, $\mathcal{M}_1$ and $\mathcal{M}_2$ have the same number of users in every AC. Thus, the optimization problems resulting for $v(\mathcal{M}_1)$ and $v(\mathcal{M}_2)$ as defined in (3.17) have the same objective functions and constraints. Thus, we have $v(\mathcal{M}_1) = v(\mathcal{M}_2)$. ■

Let Θ_I be the set of types that are present in set $\mathcal{N}_\mathcal{I}$. That is, $\theta_i \in \Theta_I \Leftrightarrow i \in \mathcal{N}_\mathcal{I}$. Let N_I^θ be the number of users in set $\mathcal{N}_\mathcal{I}$ with type $\theta \in \Theta_I$. We have $\sum_{\theta\in\Theta_I} N_I^\theta = N_I$. The total number of subsets of set $\mathcal{N}_\mathcal{I}$ which are not equivalent AC sets is given by $\prod_{\theta\in\Theta_I}(N_I^\theta + 1)$ [77, pp. 197]. From Lemma 3.2, after solving problem (3.18) for $v(\mathcal{M}_1)$ in problem (3.16), we do not have to solve it again for $v(\mathcal{M}_2)$ if $\mathcal{M}_1$ and $\mathcal{M}_2$ are equivalent AC sets. Thus, in problem (3.16), we actually need to solve problem (3.18) for only $\prod_{\theta\in\Theta_I}(N_I^\theta + 1)$ different $v(\mathcal{M})$ if we explore the equivalent AC sets. Moreover, in some cases when too many users are admitted, problem (3.17) may become *infeasible*. Let $M = |\mathcal{M}|$. The following lemma helps us in identifying some of the infeasible cases.

Lemma 3.3 *Given set $\mathcal{M} \subseteq \mathcal{N}_\mathcal{I}$ and $\hat{p}_i^{critical}$ for all $i \in \mathcal{M}$, problem (3.17) is feasible* only if

$$\sqrt[M-1]{\prod_{i\in\mathcal{M}} \hat{p}_i^{critical}} \;\leq\; 1 - \sum_{i\in\mathcal{M}} \hat{p}_i^{critical}. \tag{3.20}$$

Proof: Problem (3.17) is feasible if and only if we have

$$p_i \left(\prod_{j \in \mathcal{M} \setminus \{i\}} (1 - p_j) \right) \geq \hat{p}_i^{\text{critical}}, \qquad \forall\, i \in \mathcal{M}. \tag{3.21}$$

From (3.21) and by reordering of the terms we have

$$\frac{p_i}{1 - p_i} \geq \frac{\hat{p}_i^{\text{critical}}}{\prod_{j \in \mathcal{M}} (1 - p_j)} \qquad \Rightarrow \qquad p_i \geq \frac{1}{1 + \frac{\prod_{j \in \mathcal{M}} (1 - p_j)}{\hat{p}_i^{\text{critical}}}}, \qquad \forall\, i \in \mathcal{M}. \tag{3.22}$$

From the last line, for each user $i \in \mathcal{M}$, we have

$$1 - p_i \leq \frac{\frac{\prod_{j \in \mathcal{M}} (1 - p_j)}{\hat{p}_i^{\text{critical}}}}{1 + \frac{\prod_{j \in \mathcal{M}} (1 - p_j)}{\hat{p}_i^{\text{critical}}}} = \frac{\prod_{j \in \mathcal{M}} (1 - p_j)}{\hat{p}_i^{\text{critical}} + \prod_{j \in \mathcal{M}} (1 - p_j)}. \tag{3.23}$$

Multiplying the terms for each user $i \in \mathcal{M}$, we can come up with the following condition

$$\prod_{i \in \mathcal{M}} (1 - p_i) \;\leq\; \prod_{i \in \mathcal{M}} \left(\frac{\prod_{j \in \mathcal{M}} (1 - p_j)}{\hat{p}_i^{\text{critical}} + \prod_{j \in \mathcal{M}} (1 - p_j)} \right). \tag{3.24}$$

We define $A(\boldsymbol{p}) = \prod_{j \in \mathcal{M}} (1 - p_j)$. Clearly, $0 \leq A(\boldsymbol{p}) \leq 1$ is the probability of experiencing an idle time slot. That is, the probability that no user transmits any packet. Replacing $A(\boldsymbol{p})$ in (3.24), problem (3.17) is feasible if there exists a value A between zero and one such that we have

$$A \leq \frac{A^M}{\prod_{i \in \mathcal{M}} (\hat{p}_i^{\text{critical}} + A)} \qquad \Rightarrow \qquad \prod_{i \in \mathcal{M}} (\hat{p}_i^{\text{critical}} + A) \leq A^{M-1}. \tag{3.25}$$

For the rest of the proof, we show that condition (3.20) is a *necessary* condition for the existence of any A such that (3.25) holds. We first notice that from (3.25), we need to have

$$\sqrt[M-1]{\prod_{i\in\mathcal{M}} \hat{p}_i^{\text{critical}}} \;\leq\; A. \tag{3.26}$$

Condition (3.25) can be written in the following extended form

$$A^M + \left(\sum_{i\in\mathcal{M}} \hat{p}_i^{\text{critical}} - 1\right) A^{M-1} + \Gamma(A) \;\leq\; 0, \tag{3.27}$$

where $\Gamma(A)$ is a polynomial in A with degree $M-2$ and only non-negative multipliers. Clearly, $\Gamma(A) \geq 0$. Thus, from (3.27) we also need to have

$$A \leq 1 - \sum_{i\in\mathcal{M}} \hat{p}_i^{\text{critical}}. \tag{3.28}$$

Combining the *lower-bound* in (3.26) and the *upper-bound* in (3.28), optimization problem (3.17) is feasible *only if* there exists an A such that the following holds:

$$\sqrt[M-1]{\prod_{i\in\mathcal{M}} \hat{p}_i^{\text{critical}}} \;\leq\; A \;\leq 1 - \sum_{i\in\mathcal{M}} \hat{p}_i^{\text{critical}}. \tag{3.29}$$

Clearly, the above condition holds as long as the upper bound is greater than or equal to the lower bound. This directly results in condition (3.20). ■

Notice that the condition in (3.20) is a *necessary condition* for the feasibility of the constraints in problem (3.17).

We are now ready to propose Algorithm 3.1 to find the exact global optimal solution of problem (3.8) when elastic users have α-fair utility functions and inelastic users have α-critical or step utility functions. A similar algorithm can be used to solve problem (3.11). In Algorithm 3.1, from lines 3 to 7, we iterate through all possible subsets of $\mathcal{N}_\mathcal{I}$ and record the non-equivalent AC sets in Ψ. We only need to consider the non-equivalent AC sets, because we know from Lemma 3.2 that $v(\mathcal{M}) = v(\tilde{\mathcal{M}})$ if $\mathcal{M}$ and $\tilde{\mathcal{M}}$ are equivalent AC

Algorithm 3.1 *Algorithm to solve (3.8) for the mix of α-fair, step, and α-critical utility functions defined in (3.2) to (3.5).*

1: **Input:** $\hat{\theta}_i, \forall\, i \in \mathcal{N}$
2: (Initialization) Set $s := -\infty$, $\boldsymbol{p^*} := \mathbf{0}$, $\mathcal{M}^* := \emptyset$, and $\Psi := \emptyset$
3: **for all** subset $\mathcal{M}$ of $\mathcal{N}_{\mathcal{I}}$ **do**
4:   **if** $\mathcal{M}$ and $\tilde{\mathcal{M}}$ are not equivalent AC sets, $\forall\, \tilde{\mathcal{M}} \in \Psi$, as defined in (3.19), **then**
5:     Set $\Psi := \Psi \cup \mathcal{M}$
6:   **end if**
7: **end for**
8: **for all** $\mathcal{M} \in \Psi$ **do**
9:   Set $M := |\mathcal{M}|$
10:   **if** $\sqrt[M-1]{\prod_{i\in\mathcal{M}} \hat{p}_i^{\text{critical}}} \leq 1 - \sum_{i\in\mathcal{M}} \hat{p}_i^{\text{critical}}$ **then**
11:     Solve problem (3.18) for $v(\mathcal{M})$ and the optimal solution $\boldsymbol{p}$ using the interior point method
12:     **if** $v(\mathcal{M}) > s$, **then**
13:       Set $s := v(\mathcal{M})$, $\boldsymbol{p^*} := \boldsymbol{p}$, and $\mathcal{M}^* := \mathcal{M}$
14:     **end if**
15:   **end if**
16: **end for**
17: **Output:** $\boldsymbol{p^*}$ and $\mathcal{M}^*$

sets. From lines 8 to 16, we iterate through all the sets in Ψ. In line 10, we use Lemma 3.3 to rule out infeasible cases, which reduces the computational complexity of the algorithm. In line 11, the allocated transmission probability $\boldsymbol{p}$ for the given set $\mathcal{M}$ is calculated. Set $\mathcal{M}$ and the corresponding $\boldsymbol{p}$ that result in the largest aggregate utility so far are recorded in lines 12 to 14. In line 17, $\boldsymbol{p}^*$ is the resulting optimal solution of optimization problem (3.8) and $\mathcal{M}^*$ is the resulting set of inelastic users admitted to the system for the optimal admission control solution.

3.5 Performance Evaluations

In this section, we assess the performance of our proposed VCG-based scheme in random access systems using MATLAB. We first illustrate with an example about how our proposed

VCG-based scheme enforces truthfulness of the stations. Then, we show the support of service differentiation of different ACs and maximum aggregate utility in various scenarios with selfish stations, and the low computational complexity of our scheme. We also compare our random access scheme with a CSMA scheme. Unless specified otherwise, we assume a nominal data rate $\varphi = 54$ Mbps.

First, we provide an example to illustrate the operation of our proposed VCG-based pricing and resource allocation scheme by considering a network with one AP and six MSs. The AP supports four different ACs: AC 1 has a step utility function with parameters $K = 0.1$ and $p^{\text{critical}} = 0.1$. AC 2 also has a step utility function but with parameters $K = 1$ and $p^{\text{critical}} = 0.1$. AC 3 has an α-critical utility function with parameters $K = 0.3$, $\alpha = 1$, and $p^{\text{critical}} = 0.001$. AC 4 has an α-fair utility function with parameters $K = 5$, $\alpha = 1$, and $L = 4$. We assume that MS 1 belongs to AC 1, MS 2 belongs to AC 2, both MSs 3 and 4 belong to AC 3, and both MSs 5 and 6 belong to AC 3. We consider two cases: In Case I, MS 1 *honestly* declares that it supports applications in AC 1 and all other MSs are honest. In Case II, MS 1 *selfishly* declares that it supports applications in AC 2 (i.e., declares a larger K) while other MSs are still honest. Notice that MS 1 has the motivation to do so as stated in Theorem 3.1. With the use of the VCG mechanism and Algorithm 3.1, we plot the utilities, payments, and surpluses of the six MSs in Fig. 3.3, where Cases I and II are represented by the two bars at the index of each MS. As shown in Fig. 3.3(a), when MS 1 is honest, it is not admitted into the system and it receives zero utility. However, it is not charged with any payment by the VCG mechanism as shown in Fig. 3.3(b). On the other hand, when MS 1 is selfish, it is admitted into the system and receives a positive utility. However, with the use of the VCG mechanism, MS 1 is *punished* with a large payment when it lies. As shown in Fig. 3.3(c), the *surplus* (i.e., utility minus payment) of player 1 indeed decreases if it lies due to the use of the proposed

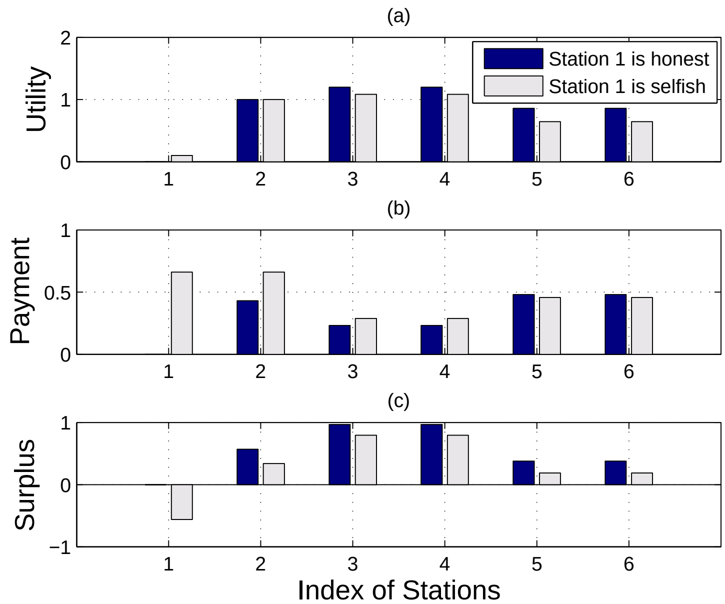

Figure 3.3: Results in a sample network with six stations: (a) Utility, (b) Payment, and (c) Surplus (i.e., utility minus payment) of each station using the proposed VCG-based scheme. With the use of VCG mechanism, station 1 ends up having a lower surplus when it is selfish and is thus motivated to be honest.

VCG mechanism. Clearly, this forces user 1 to be truthful about its type. Notice that in Case II, although MS 1 declares its application to be in AC 2 and thus receives the same transmission probability as MS 2 which has an AC 2 type of application, it receives a lower utility than MS 2 because its application is indeed in AC 1.

Next, we evaluate the performance of our proposed scheme in a larger network with twelve MSs. We assume that four MSs have step utility functions with parameters $K = 0.1$ and $p^{\text{critical}} = 0.1$, four MSs have α-critical utility function with parameters $K = 0.1$, $\alpha = 1$, and $p^{\text{critical}} = 0.001$, and the remaining four MSs have α-fair utility functions with parameters $K = 0.01$, $\alpha = 1$ and $L = 4$. We assume that all the MSs are honest, except some of the MSs with step utility functions which are selfish and may declare a higher amplitude parameter $\hat{K} = 1 > 0.1 = K$ (see Theorem 3.1). The throughput and the network aggregate utility (i.e., the objective function in problem (3.6)) achieved for

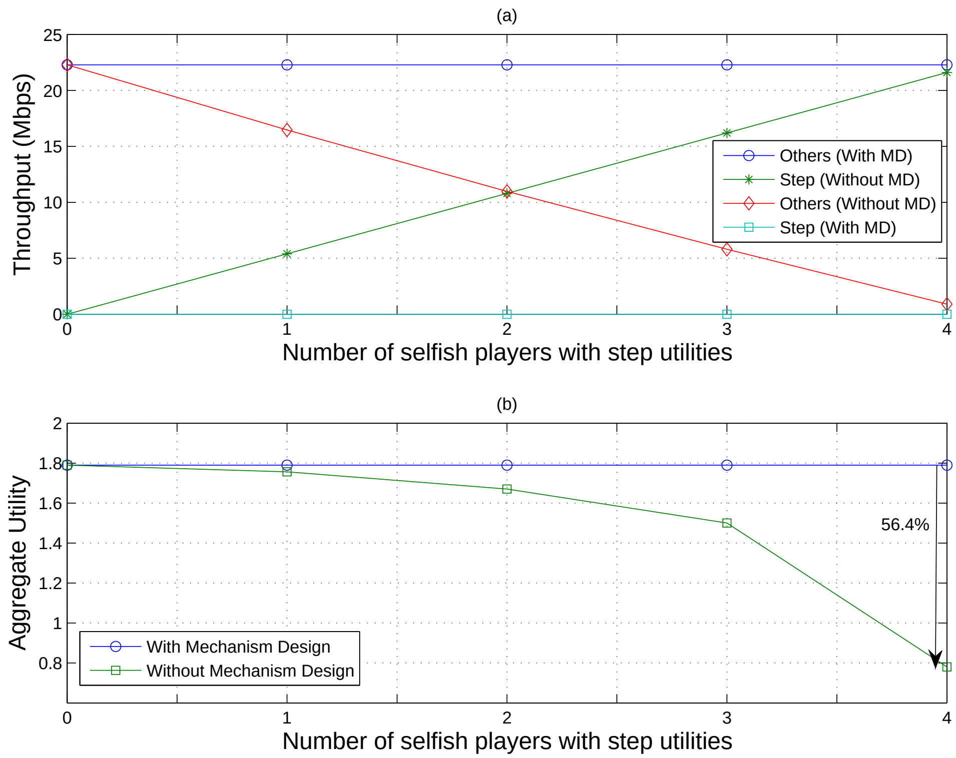

Figure 3.4: (a) Throughput of users with step utility functions and other utility functions and (b) aggregate utility. We assume that the stations with α-fair and α-critical utilities are honest and we vary the number of selfish stations with step utilities. We can see that differentiated QoS and maximum aggregate utility can be maintained by using the VCG-based mechanism design (MD).

different numbers of selfish MSs with step utility functions are shown in Figs. 3.4(a) and (b), respectively. As shown in Fig. 3.4(a), without the use of mechanism design, selfish MSs with step utilities may indeed declare $\hat{K} > K$ in order to gain admission to the system. In this case, since many users are not truthful, the AP's information is inaccurate. Therefore, the AP is not able to provide differentiated QoS. On the other hand, when our proposed VCG-based scheme is used, the throughput of MSs with other types of utility functions is guaranteed that the differentiated QoS is supported. In Fig. 3.4(b), we can see that the dishonest declaration of the MSs with step utilities causes a deviation from the optimal

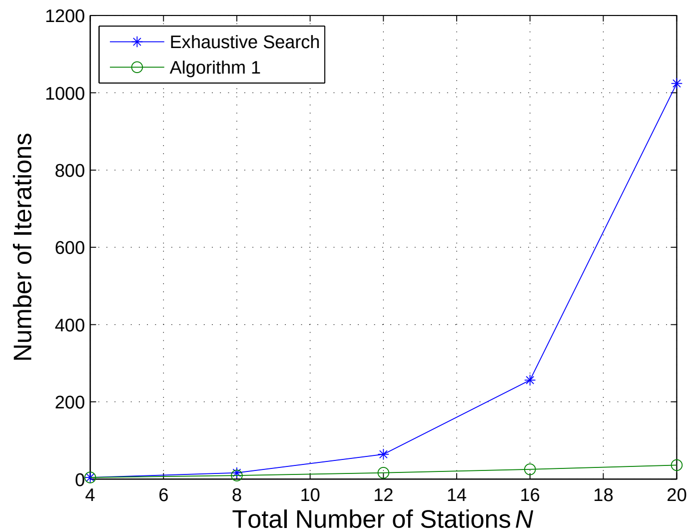

Figure 3.5: Number of iterations required to obtain the optimal solution by Algorithm 3.1 and an exhaustive search. We can see that Algorithm 3.1 has a much lower computational complexity than an exhaustive search.

network aggregate utility. The performance reduction becomes more severe as the number of selfish MSs increases, e.g., resulting in more than 56.4% efficiency loss in the presence of four selfish MSs. Thus, using the VCG-based mechanism results in a significantly better network performance.

In Fig. 3.5, we compare the computational complexity of Algorithm 3.1 with an *exhaustive search*. Specifically, to solve problem (3.8), we compare the number of iterations that $v(\mathcal{M})$ in problem (3.16) is evaluated by the two schemes. In other words, we compare the total number of times that problem (3.18) is solved for different $\mathcal{M}$. For the exhaustive search, we modify Algorithm 3.1 by removing lines 3 to 7 and initializing Ψ to be the power set (i.e., the set of all subsets) of $\mathcal{N}_{\mathcal{I}}$ in line 2 instead. That is, we do not include the result of Lemma 3.2 in the exhaustive search algorithm. In our evaluation, we assume that there are four ACs, where two ACs are for inelastic traffic and the other two ACs are for elastic traffic. We assume that the number of MSs in each AC is the same and we vary

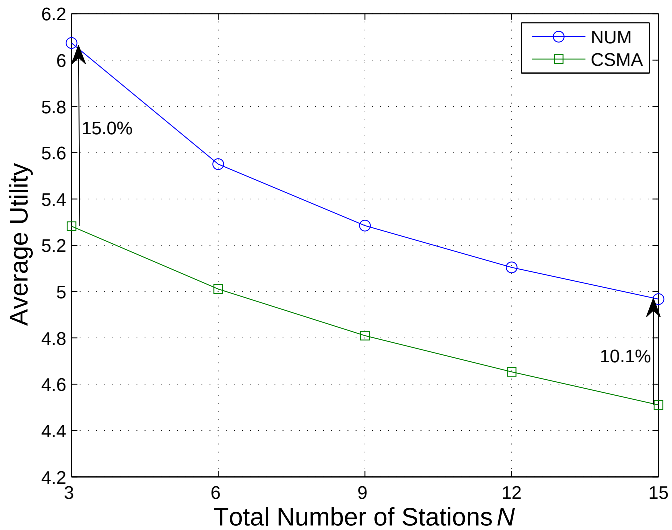

Figure 3.6: Average utility in the system achieved by our NUM-based random access scheme and a CSMA scheme versus the total number of stations N in the system.

the number of MSs N in the system. As we can see, by eliminating a significant number of redundant computations due to the equivalent AC sets from Lemma 3.2, Algorithm 3.1 results in a much lower computational complexity than that of the exhaustive search.

In Fig. 3.6, we compare our random access scheme with a CSMA scheme similar to the one used in the IEEE 802.11e with different contention window sizes for different ACs. Let aCW_{min} and aCW_{max} be two parameters related to the contention window sizes. For the CSMA scheme, we assume that three ACs are available with different minimum and maximum contention window sizes [78, pp. 131]: aCW_{min} and aCW_{max} for AC 1 (best effort), $(aCW_{min} + 1)/2 - 1$ and aCW_{min} for AC 2 (video), and $(aCW_{min} + 1)/4 - 1$ and $(aCW_{min} + 1)/2 - 1$ for AC 3 (voice). We assume that when a user initiates a transmission, it keeps the channel for μ time slots. For simplicity, we do not implement the interframe space in the IEEE 802.11e standard. We assume that the number of MSs in each ACs is the same. We consider a system with real-time voice, rate-adaptive video, and file transfer applications, which their utility functions are as follows: a step function with parameters $K = 10$ and $p^{\text{critical}} = 0.001$ for the real-time voice application, an α-critical utility function with parameters $K = 1.2$, $\alpha = 1$, and $p^{\text{critical}} = 0.001$ for the rate-adaptive video application, and an α-fair utility function with parameters $K = 0.5$, $\alpha = 1$, and $L = 4$ for the file transfer application. Note that the utility of the step utility function is the highest and that of the α-fair utility function is the lowest. From the nature and utilities of the applications, we map the real-time voice application to AC 3 (voice), the rate-adaptive video application to AC 2 (video), and the file transfer application to AC 1 (best effort). We choose $aCW_{min} = 63$, $aCW_{max} = 1023$ [78, pp. 589], and $\mu = 3$. As shown in Fig. 3.6, the performance improvement of the average utility in the system of our scheme over the CSMA scheme is 15.0% for $N = 3$ and 10.1% for $N = 15$, although the CSMA scheme achieves a higher throughput as suggested in the literature.

3.6 Summary

In this chapter, we studied the problem of assigning transmission probabilities to MSs for random access in a WLAN. Each MS is running an elastic or an inelastic application with different QoS requirements, which are characterized by different ACs and modeled by different utility functions. Specifically, we considered that the utility functions for elastic users are concave, while those for inelastic users are step or quasi-concave. Potentially, a selfish MS may strategically declare its AC to unfairly achieve a larger share of bandwidth, which can drastically degrade the network performance and inhibit adequate service distinction among different ACs. We used game theory to analyze the strategic declarations of the ACs of the rational users, and applied the VCG mechanism in our random access protocol to motivate the MSs to declare the ACs of their applications truthfully. In our proposed scheme, the AP performs admission control, and informs the MSs about their assigned transmission probabilities as well as the required payments. In order to implement the

VCG mechanism, we need to solve a non-convex NUM problem optimally. We proposed a novel enumeration algorithm, which involves solving only a convex optimization problem in each iteration. Analytical results related to the equivalent AC sets were presented that significantly reduce the computational complexity of the proposed algorithm. Simulation results show that a truthful mechanism can prevent selfish users from gaining an unfair share of the network bandwidth, such that both the overall network performance in terms of aggregate utility and service differentiation in terms of necessary throughput in each AC can be supported.

Chapter 4

Random Access with Sigmoidal and Concave Utility Functions

In this chapter, we extend the work in Chapter 3 for single-channel wireless random access with both elastic and inelastic traffic in a WLAN. We still model the utilities of the applications generating elastic traffic with concave utility functions. However, we extend the work in [62] and Chapter 3 by not restricting the utility functions to remain concave after a logarithmic change of variables, but allowing the possibilities of concave, convex, or sigmoidal utility functions. For applications generating inelastic traffic, we model their utilities with sigmoidal utility functions, leading to NUM problems which are usually difficult to solve. NUM problems with sigmoidal utility functions have previously been considered in various networking design problems such as Internet congestion control [79, 80], downlink power allocation [81], power control [82], and radio resource allocation [83]. But no prior work has addressed NUM problems with sigmoidal utility functions in *random access* systems.

In this chapter, since we are not considering a VCG-based random access that requires the computation of a global optimal solution, it suffices to obtain obtain a near-optimal solution. We formulate the problem of random access as a NUM problem, which is non-convex. We use the dual approach and the *subgradient projection method* to tackle the non-convexity of the NUM problem. For sigmoidal utility functions, each iteration in our algorithm involves only updating the dual variables with some *closed-form* expressions. In summary, the contributions of this chapter are as follows:

- We consider solving the primal problem using the dual method and derive the Karush-Kuhn-Tucker (KKT) optimality conditions of the dual problem.
- We propose a centralized algorithm based on the subgradient projection method to solve the formulated non-convex NUM problem.
- We provide a sufficient condition on the wireless link capacities which guarantee our algorithm to find the exact global optimal solution of the NUM problem. If this condition is *not* satisfied, we can still obtain *upper* and *lower* bounds for the optimal objective value. The bounds approach each other when the duality gap is zero.
- Simulations are performed to verify our analytical results.

The rest of this chapter is organized as follows. The system model is described in Section 4.1. We present our centralized algorithm and the optimality conditions for the dual problem in Section 4.2. We study the condition on capacity that results in optimal or sub-optimal solutions in Section 4.3. Simulation results are given in Section 4.4. A summary is given in Section 4.5.

4.1 System Model

Consider a WLAN with a single AP and a set of N mobile stations, denoted by $\mathcal{N} = \{1, 2, \ldots, N\}$. All stations are one-hop neighbors to the AP. We only consider the up-link scenario, where each station $i \in \mathcal{N}$ can access the shared medium with a persistent probability (or transmission probability) p_i. We consider using a slotted Aloha MAC protocol, where time is divided into equal time slots. The stations attempt to access the shared channel at the beginning of each time slot according to their persistent probabilities. Notice that the choice of persistent probabilities can be transformed into equivalent contention window sizes that can be implemented directly in IEEE 802.11 WLANs [74]. Let p_i^{succ} denote the probability that a transmission from station $i \in \mathcal{N}$ is *successful*, i.e., the transmission does not experience any collision. We have

$$p_i^{\text{succ}}(\boldsymbol{p}) \;=\; p_i \prod_{j \in \mathcal{N} \setminus \{i\}} (1 - p_j), \qquad \forall\, i \in \mathcal{N}, \tag{4.1}$$

where $\boldsymbol{p} = (p_i,\; i \in \mathcal{N})$. For the rest of this chapter, we will use bold symbols to denote vectors with components $\forall\, i \in \mathcal{N}$. Given the *capacity* c_i for user i, the *average data rate* for station i is $x_i = c_i\, p_i^{\text{succ}}(\boldsymbol{p})$, which is a function of both c_i and $\boldsymbol{p}$. We denote the utility function of each station $i \in \mathcal{N}$ by $U_i(x_i)$, which is a *non-decreasing* function in x_i. The utility function is used to model the level of satisfaction that station i experiences from its application when it attains average data rate x_i. In particular, we have $U_i(x_i) \geq 0$ if $x_i \geq 0$. Also, we have $U_i(0) = 0$.

Each station may have either *elastic* or *inelastic* traffic. Let $\mathcal{N}_{\mathcal{E}}$ and $\mathcal{N}_{\mathcal{I}}$ denote the sets of stations with elastic and inelastic traffic, respectively. We notice that $\mathcal{N}_{\mathcal{E}} \cap \mathcal{N}_{\mathcal{I}} = \phi$ and $\mathcal{N}_{\mathcal{E}} \cup \mathcal{N}_{\mathcal{I}} = \mathcal{N}$. For each user $v \in \mathcal{N}_{\mathcal{E}}$, we can use a *concave* function to model the utility. A common example is the α-fair utility function (see Fig. 4.1) [53] normalized such that $U_i(0) = 0$:

$$U_v(x_v) = \begin{cases} \ln\left(x_v + 1\right), & \text{if } \alpha_v = 1, \\ (1-\alpha_v)^{-1}\left[(x_v + 1)^{(1-\alpha_v)} - 1\right], & \text{if } \alpha_v \in (0,1) \cup (1,\infty), \end{cases} \tag{4.2}$$

where α_v is a fixed utility parameter. On the other hand, for each user $w \in \mathcal{N}_{\mathcal{I}}$, the utility function depends on the quality of service (QoS) requirements of the running voice and video applications. We can use a sigmoidal utility function $U_w(x_w)$ to model these applications such that $U_w''(x_w) > 0$ for $x_w < x_w^{in}$ and $U_w''(x_w) < 0$ for $x_w > x_w^{in}$, where x_w^{in}

is the *point of inflection*. In particular, we can use the sigmoidal function (see Fig. 4.1) defined as [84]:

$$U_w(x_w) = \frac{x_w^{a_w}}{k_w + x_w^{a_w}}, \tag{4.3}$$

where $x_w \geq 0$, $a_w > 1$, $k_w > 0$, and $x_w^{in} = \sqrt[a_w]{\frac{k_w(a_w-1)}{a_w+1}}$. With the logarithmic change of variables $\bar{x}_i \triangleq \ln x_i$ and $\bar{U}_i(\bar{x}_i) \triangleq U_i(e^{\bar{x}_i})$, the utility functions become *more* convex. That is, the concave part may remain concave or turn convex [62], while the convex part always remains convex. For the concave function $U_v(x_v)$ in (4.2), we can see that $\bar{U}_v(\bar{x}_v)$ is a sigmoidal function with point of inflection $\bar{x}_v^{in} = \ln(\frac{1}{\alpha_v - 1})$ for $\alpha_v > 1$, and a convex function for $0 < \alpha_v \leq 1$. Moreover, we have

$$\bar{U}_w(\bar{x}_w) = \frac{1}{1 + e^{-(a_w \bar{x}_w + b_w)}}, \tag{4.4}$$

which represents a sigmoidal function in standard form with the point of inflection $\bar{x}_w^{in} = -b_w/a_w$, where $k_w = e^{-b_w}$. Note that $\bar{x}_w^{in} \neq \ln x_w^{in}$ in general. In the sequel, we will assume that $\bar{U}_i(\bar{x}_i)$ is sigmoidal for $\bar{x}_i^{min} \leq \bar{x}_i \leq \bar{x}_i^{max}$, $\forall\, i \in \mathcal{N}$. We will omit the cases where $\bar{U}_i(\bar{x}_i)$ is either a convex or a concave function for brevity, because the dual problem is straightforward in these cases. It should be noted that the solution approach discussed in the following sections can be applied to concave, convex, and sigmoidal utility functions in general.

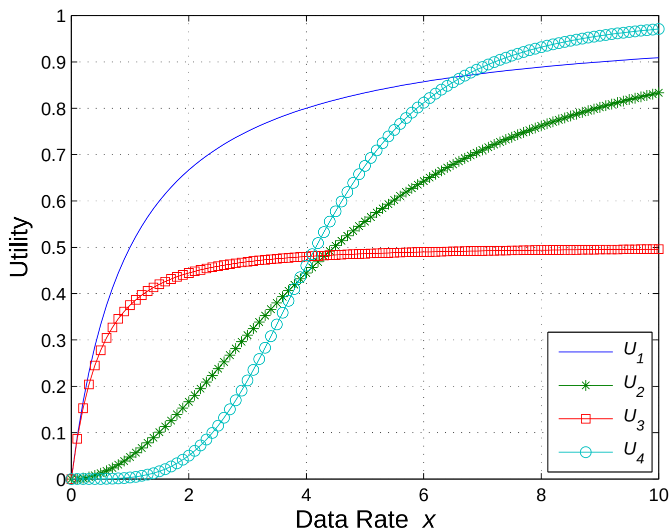

Figure 4.1: Utility functions U_i versus data rate x for utility functions $U_1(x) = 1-(x+1)^{-1}$, $U_2(x) = \frac{x^2}{x^2+20}$, $U_3(x) = \frac{1}{2}[1-(x+1)^{-2}]$, and $U_4(x) = \frac{x^4}{x^4+300}$. Notice that U_1 and U_3 are concave functions, and U_2 and U_4 are sigmoidal functions. We address both concave and sigmoidal utility functions in this chapter.

4.2 Random Access with Sigmoidal and Concave Utilities

4.2.1 NUM for Random Access

In this chapter, we consider the following NUM problem:

$$\begin{array}{lll}
\underset{\boldsymbol{p},\boldsymbol{x}}{\text{maximize}} & \sum_{i\in\mathcal{N}} U_i(x_i) = \sum_{v\in\mathcal{N}_\mathcal{E}} U_v(x_v) + \sum_{w\in\mathcal{N}_\mathcal{I}} U_w(x_w) & \\
\text{subject to} & x_i \le c_i p_i^{\text{succ}} = c_i p_i \prod_{j\in\mathcal{N}\setminus\{i\}}(1-p_j), & \forall i \in \mathcal{N}, \\
& x_i^{min} \le x_i \le x_i^{max}, & \forall i \in \mathcal{N}, \\
& 0 \le p_i \le 1, & \forall i \in \mathcal{N},
\end{array} \tag{4.5}$$

where x_i^{min} and x_i^{max} are the constraints on the minimum and maximum data rates for the transmission of user i, respectively. Notice that problem (4.5) is a non-convex optimization problem, because the objective function is non-concave in general, and the first constraint is non-convex.

4.2.2 Dual Method

Using the logarithmic change of variables $\bar{x}_i \triangleq \ln x_i$, $\bar{x}_i^{min} \triangleq \ln x_i^{min}$, $\bar{x}_i^{max} \triangleq \ln x_i^{max}$, $\bar{U}_i(\bar{x}_i) \triangleq U_i(e^{\bar{x}_i})$, and $\bar{c}_i \triangleq \ln c_i$, we can reformulate optimization problem (4.5) as

$$\begin{aligned}
\underset{\boldsymbol{p},\bar{\boldsymbol{x}}}{\text{maximize}} \quad & \textstyle\sum_{i\in\mathcal{N}} \bar{U}_i(\bar{x}_i) = \sum_{v\in\mathcal{N}_\mathcal{E}} \bar{U}_v(\bar{x}_v) + \sum_{w\in\mathcal{N}_\mathcal{I}} \bar{U}_w(\bar{x}_w) \\
\text{subject to} \quad & \bar{c}_i + \ln p_i + \textstyle\sum_{j\in\mathcal{N}\setminus\{i\}} \ln(1-p_j) - \bar{x}_i \geq 0, && \forall\, i \in \mathcal{N}, \\
& \bar{x}_i^{min} \leq \bar{x}_i \leq \bar{x}_i^{max}, && \forall\, i \in \mathcal{N}, \\
& 0 \leq p_i \leq 1, && \forall\, i \in \mathcal{N}.
\end{aligned} \tag{4.6}$$

Here, the Lagrangian function is derived as

$$L(\boldsymbol{p}, \bar{\boldsymbol{x}}, \boldsymbol{\lambda}) = \sum_{i\in\mathcal{N}} \left(\bar{U}_i(\bar{x}_i) - \lambda_i \bar{x}_i \right) + \sum_{i\in\mathcal{N}} \lambda_i \left(\bar{c}_i + \ln p_i + \sum_{j\in\mathcal{N}\setminus\{i\}} \ln(1-p_j) \right), \tag{4.7}$$

and the Lagrangian dual function becomes

$$g(\boldsymbol{\lambda}) = \sum_{i\in\mathcal{N}} \sup_{\bar{x}_i\in\bar{\mathcal{X}}_i} \left(\bar{U}_i(\bar{x}_i) - \lambda_i \bar{x}_i \right) + \sup_{\boldsymbol{p}\in\mathcal{P}} \sum_{i\in\mathcal{N}} \lambda_i \left(\ln p_i + \sum_{j\in\mathcal{N}\setminus\{i\}} \ln(1-p_j) \right) + \sum_{i\in\mathcal{N}} \lambda_i \bar{c}_i, \tag{4.8}$$

where $\mathcal{P} = \{\boldsymbol{p} : 0 \leq p_i \leq 1, \forall\, i \in \mathcal{N}\}$ and $\bar{\mathcal{X}}_i = \{\bar{x}_i : \bar{x}_i^{min} \leq \bar{x}_i \leq \bar{x}_i^{max}\}$. The dual problem is

$$\begin{aligned}
\underset{\boldsymbol{\lambda}}{\text{minimize}} \quad & g(\boldsymbol{\lambda}) \\
\text{subject to} \quad & \boldsymbol{\lambda} \succeq \mathbf{0}.
\end{aligned} \tag{4.9}$$

In order to solve optimization problem (4.9), we need to solve two subproblems for each

$i \in \mathcal{N}$:

$$\max_{\bar{x}_i \in \bar{\mathcal{X}}_i} \left(\bar{U}_i(\bar{x}_i) - \lambda_i \bar{x}_i \right), \quad \text{and} \quad \max_{\boldsymbol{p} \in \mathcal{P}} \sum_{i \in \mathcal{N}} \lambda_i \left(\ln p_i + \sum_{j \in \mathcal{N} \setminus \{i\}} \ln(1 - p_j) \right). \tag{4.10}$$

4.2.3 First Dual Subproblem

To solve the first dual subproblem, we define $s_i(\bar{x}_i, \lambda_i) = \bar{U}_i(\bar{x}_i) - \lambda_i \bar{x}_i$ and

$$\bar{x}_i^*(\lambda_i) = \arg \max_{\bar{x}_i \in \bar{\mathcal{X}}_i} s_i(\bar{x}_i, \lambda_i). \tag{4.11}$$

Notice that s_i is also a sigmoidal function in $\bar{x}_i$ with point of inflection $\bar{x}_i^{in}$.

Lemma 4.1 *If $\bar{x}_i^{min} \leq \bar{x}_i^{in} \leq \bar{x}_i^{max}$, we have*

$$\bar{x}_i^*(\lambda_i) = \arg \max_{\bar{x}_i \in \{\bar{x}_i^{min},\, \bar{x}_i^{v}(\lambda_i)\}} s_i(\bar{x}_i, \lambda_i), \tag{4.12}$$

where

$$\bar{x}_i^{v}(\lambda_i) \triangleq \arg\max_{\bar{x}_i^{in} \leq \bar{x}_i \leq \bar{x}_i^{max}} s_i(\bar{x}_i, \lambda_i). \tag{4.13}$$

Proof: It is always true that

$$\max_{\bar{x}_i \in \bar{\mathcal{X}}_i} s_i(\bar{x}_i, \lambda_i) = \max \left\{ \max_{\bar{x}_i^{min} \leq \bar{x}_i \leq \bar{x}_i^{in}} s_i(\bar{x}_i, \lambda_i), \max_{\bar{x}_i^{in} \leq \bar{x}_i \leq \bar{x}_i^{max}} s_i(\bar{x}_i, \lambda_i) \right\}. \tag{4.14}$$

Also notice that $s_i(\bar{x}_i, \lambda_i)$ is a *convex* function in $\bar{x}_i$ for $\bar{x}_i^{min} \leq \bar{x}_i \leq \bar{x}_i^{in}$. Thus, we have $\arg\max_{\bar{x}_i^{min} \leq \bar{x}_i \leq \bar{x}_i^{in}} s_i(\bar{x}_i, \lambda_i) = \{\bar{x}_i^{min}, \bar{x}_i^{in}\}$. This concludes the proof. ∎

Notice that problem (4.13) is convex. In fact, we can obtain a closed-form solution for (4.13) when $\bar{U}_w(\bar{x}_w)$ is as in (4.4):

Lemma 4.2 *For $\bar{U}_w(\bar{x}_w)$ in (4.4), we have*

$$\bar{x}_w^v(\lambda_w) = \begin{cases} \left[\frac{-\ln\left(\left(a_w - 2\lambda_w - \sqrt{a_w^2 - 4a_w\lambda_w}\right)/2\lambda_w\right) - b_w}{a_w}\right]_{\bar{x}_w^{in}}^{\bar{x}_w^{max}}, & \text{if } a_w \geq 4\lambda_w, \\ \bar{x}_w^{in}, & \text{otherwise,} \end{cases} \tag{4.15}$$

where $[z]_w^y = \min\{\max\{z, w\}, y\}$.

Proof: Since $s_w(\bar{x}_w, \lambda_w)$ is concave for $\bar{x}_w^{in} \leq \bar{x}_w \leq \bar{x}_w^{max}$, by taking the derivative, we have

$$\bar{U}_w'(\bar{x}_w^v) - \lambda_w = \frac{a_w e^{-(a_w \bar{x}_w^v + b_w)}}{[1 + e^{-(a_w \bar{x}_w^v + b_w)}]^2} - \lambda_w = 0. \tag{4.16}$$

Let $y_w = e^{-(a_w \bar{x}_w^v + b_w)}$, we obtain $\lambda_w y_w^2 + (2\lambda_w - a_w) y_w + \lambda_w = 0$. We can consider two cases:

Case I: If $a_w \geq 4\lambda_w$, since $0 \leq y_w \leq 1$, we can take the root $y_w = \frac{a_w - 2\lambda_w - \sqrt{a_w^2 - 4a_w\lambda_w}}{2\lambda_w}$, so

$$\bar{x}_w^v(\lambda_w) = \left[\frac{-\ln\left(\left(a_w - 2\lambda_w - \sqrt{a_w^2 - 4a\lambda_w}\right)/2\lambda_w\right) - b_w}{a_w}\right]_{\bar{x}_w^{in}}^{\bar{x}_w^{max}}. \tag{4.17}$$

Case II: If $a_w < 4\lambda_w$, we have $s_w'(\bar{x}_w, \lambda_w) < 0$ and $s_w(\bar{x}_w, \lambda_w)$ is decreasing in $\bar{x}_w$. Thus,

$$\bar{x}_w^v(\lambda_w) = \underset{\bar{x}_w^{in} \leq \bar{x}_w \leq \bar{x}_w^{max}}{\arg\max}\ s_w(\bar{x}_w, \lambda_w) = \bar{x}_w^{in}. \tag{4.18}$$

Considering the two cases above, we can fully characterize $\bar{x}_w^v(\lambda_w)$ as in (4.17) and (4.18).

■

4.2.4 Second Dual Subproblem

For the second dual subproblem, given $\boldsymbol{\lambda}$, we have

$$\begin{aligned}&\max_{\boldsymbol{p}\in\mathcal{P}}\sum_{i\in\mathcal{N}}\lambda_i\left(\ln p_i+\sum_{j\in\mathcal{N}\setminus\{i\}}\ln(1-p_j)\right)\\&=\sum_{i\in\mathcal{N}}\max_{0\le p_i\le 1}\left(\lambda_i\ln p_i+\left(\sum_{j\in\mathcal{N}\setminus\{i\}}\lambda_j\right)\ln(1-p_i)\right).\end{aligned}\tag{4.19}$$

Since the problem at the right hand side in (4.19) is convex, we can apply the first order *necessary* and *sufficient* optimization condition to obtain the given optimal solution [62] as follows

$$p_i^*(\boldsymbol{\lambda})=\frac{\lambda_i}{\sum_{j\in\mathcal{N}}\lambda_j},\qquad \text{if}\sum_{j\in\mathcal{N}}\lambda_j\neq 0.\tag{4.20}$$

4.2.5 Centralized Algorithm for Random Access

We define

$$\lambda_i^c=\min\{\lambda\ge 0: s_i(\bar{x}_i^{min},\lambda)=\max_{\bar{x}_i^{in}\le\bar{x}_i\le\bar{x}_i^{max}}s_i(\bar{x}_i,\lambda)\}.\tag{4.21}$$

Thus, $\bar{x}_i^*(\lambda_i^c)$ has two solutions: $\bar{x}_i^*(\lambda_i^c)=\bar{x}_i^{min}$ and $\bar{x}_i^{in}\le\bar{x}_i^*(\lambda_i^c)\le\bar{x}_i^{max}$. As shown in Fig. 4.2, $\bar{x}_i^*(\lambda_i)$ is discontinuous at λ_i^c.

Consider $g(\boldsymbol{\lambda})=\sup_{\boldsymbol{p}\in\mathcal{P},\bar{\boldsymbol{x}}\in\bar{\mathcal{X}}}L(\boldsymbol{p},\bar{\boldsymbol{x}},\boldsymbol{\lambda})$, we apply Danskin's Theorem [19] to find the subdifferential $\partial g(\boldsymbol{\lambda})$ (i.e., the set of all subgradients of $g(\boldsymbol{\lambda})$)

$$\partial g(\boldsymbol{\lambda})=\mathrm{conv}\{\nabla_{\boldsymbol{\lambda}}L(\boldsymbol{p},\bar{\boldsymbol{x}},\boldsymbol{\lambda}):\boldsymbol{p}\in\boldsymbol{p}^*(\boldsymbol{\lambda}),\bar{\boldsymbol{x}}\in\bar{\boldsymbol{x}}^*(\boldsymbol{\lambda})\},\tag{4.22}$$

where $\mathrm{conv}\{\mathcal{H}\}$ is the convex hull of set $\mathcal{H}$ and $\nabla_{\boldsymbol{\lambda}}L(\boldsymbol{p},\bar{\boldsymbol{x}},\boldsymbol{\lambda})=\left(\frac{\partial L(\boldsymbol{p},\bar{\boldsymbol{x}},\boldsymbol{\lambda})}{\partial\lambda_1},\ldots,\frac{\partial L(\boldsymbol{p},\bar{\boldsymbol{x}},\boldsymbol{\lambda})}{\partial\lambda_N}\right)^T$ denotes the gradient of L with respect to $\boldsymbol{\lambda}$, and the notation $(\cdot)^T$ denotes vector transpose operator. Moreover, $\boldsymbol{p}^*(\boldsymbol{\lambda})$ and $\bar{\boldsymbol{x}}^*(\boldsymbol{\lambda})$ are the solutions of (4.20) and (4.11) at $\boldsymbol{\lambda}$ for all

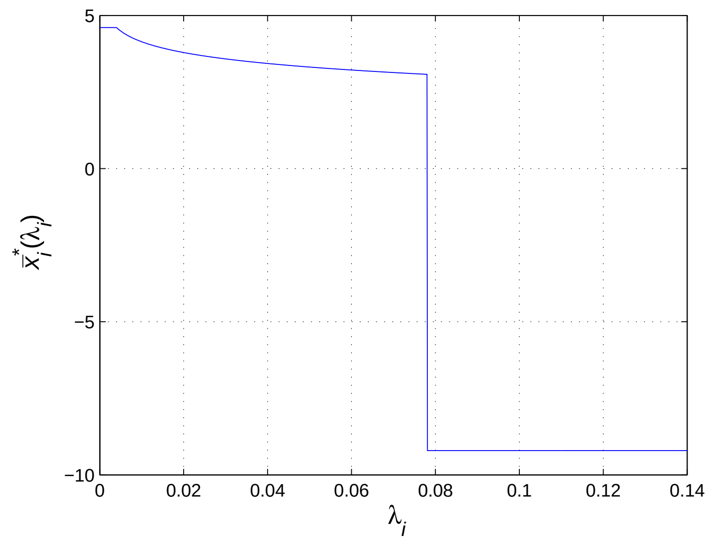

Figure 4.2: The solution of the first dual subproblem $\bar{x}_i^*(\lambda_i)$ versus λ_i for sigmoidal utility function $U_i(x) = \frac{x^2}{x^2+20}$. We can see that $\bar{x}_i^*(\lambda_i)$ is discontinuous at $\lambda_i = \lambda_i^c = 0.0780$.

$i \in \mathcal{N}$, respectively. We note that $g(\boldsymbol{\lambda})$ is differentiable at $\boldsymbol{\lambda}$, because there is only one element in both sets $\boldsymbol{p}^*(\boldsymbol{\lambda})$ and $\bar{\boldsymbol{x}}^*(\boldsymbol{\lambda})$. This is always true unless $\exists\, i \in \mathcal{N}$ such that $\lambda_i = \lambda_i^c$, because in that case there are two possible solutions for $\bar{x}_i^*(\lambda_i^c)$ as discussed above. Using the subgradient projection method, we update $(\lambda_i, \forall i \in \mathcal{N})$ according to the following equation:

$$\lambda_i(t+1) = \left[\lambda_i(t) - \alpha(t)\left(\bar{c}_i + \ln p_i^*(\boldsymbol{\lambda}(t)) + \sum_{j \in \mathcal{N}\setminus\{i\}} \ln(1 - p_j^*(\boldsymbol{\lambda}(t))) - \bar{x}_i^*(\lambda_i(t))\right)\right]^+, \tag{4.23}$$

where $[z]^+ = \max\{z, 0\}$ and t is the index of the iteration. With a diminishing step size $\alpha(t) \geq 0$ such that $\lim_{t\to\infty} \alpha(t) = 0$ and $\sum_{t=1}^{\infty} \alpha(t) = \infty$ (e.g., we can choose $\alpha(t) = m/t$, where m is a positive constant), it can be shown that $\lambda_i(t)$ converges to the dual optimal solution λ_i^* as $t \to \infty$ [19]. The algorithm to solve problem (4.6) is shown in Algorithm 4.1. In the algorithm, we initialize the variables in lines 1 to 3, and update the variables $\boldsymbol{p}^*$, $\bar{\boldsymbol{x}}^*$ and $\boldsymbol{\lambda}$ in lines 4 to 10. Notice that the subgradient method is usually used without any

Algorithm 4.1 *Centralized Algorithm to Solve Problem (4.6).*

Input: $c_i, x_i^{min}, x_i^{max}, \; \forall\, i \in \mathcal{N}$
Calculate $\bar{x}_i^{in}, \; \forall\, i \in \mathcal{N}$
Set $t := 1$ and initialize $\lambda_i(t) > 0, \; \forall\, i \in \mathcal{N}$
while $t < MAXITER$
  Set $p_i^*(\boldsymbol{\lambda}(t)) := \frac{\lambda_i(t)}{\sum_{j \in \mathcal{N}} \lambda_j(t)}, \; \forall i \in \mathcal{N}$
  Set $\bar{x}_i^v(\lambda_i(t)) := \underset{\bar{x}_i^{in} \leq \bar{x}_i \leq \bar{x}_i^{max}}{\arg\max} \; s_i(\bar{x}_i, \lambda_i(t))$
  Set $\bar{x}_i^*(\lambda_i(t)) := \underset{\bar{x}_i \in \{\bar{x}_i^{min}, \bar{x}_i^v(\lambda_i(t))\}}{\arg\max} \; s_i(\bar{x}_i, \lambda_i(t))$
  Set $\lambda_i(t+1)$ as in (4.23) with $\alpha(t) := m/t$, where $m > 0$
  Set $t := t + 1$
end while
Output: $\boldsymbol{p}^*$ and $\bar{\boldsymbol{x}}^*$.

formal stopping criterion, so we run our algorithm for a pre-specified number of iterations $MAXITER$. We will discuss the optimality of its solution in the next section.

4.2.6 General Optimality Conditions

We have the following general optimality condition for the dual problem in (4.9):

Theorem 4.1 *Vector $\boldsymbol{\lambda}^*$ is the solution of problem (4.9) if and only if $\boldsymbol{\lambda}^* \succeq \mathbf{0}$ and*

1. *If $\lambda_i^* = 0$, we have*

$$\bar{c}_i + \ln p_i^*(\boldsymbol{\lambda}^*) + \sum_{j \in \mathcal{N} \setminus \{i\}} \ln(1 - p_j^*(\boldsymbol{\lambda}^*)) - \bar{x}_i^v(\lambda_i^*) \geq 0. \tag{4.24}$$

2. *If $\lambda_i^* > 0$ and $\lambda_i^* \neq \lambda_i^c$, we have*

$$\bar{c}_i + \ln p_i^*(\boldsymbol{\lambda}^*) + \sum_{j \in \mathcal{N} \setminus \{i\}} \ln(1 - p_j^*(\boldsymbol{\lambda}^*)) - \bar{x}_i^v(\lambda_i^*) = 0. \tag{4.25}$$

3. If $\lambda_i^* > 0$ and $\lambda_i^* = \lambda_i^c$, we have

$$\bar{c}_i + \ln p_i^*(\boldsymbol{\lambda}^*) + \sum_{j \in \mathcal{N} \setminus \{i\}} \ln(1 - p_j^*(\boldsymbol{\lambda}^*)) - \bar{x}_i^v(\lambda_i^*) \leq 0, \tag{4.26}$$

and

$$\bar{c}_i + \ln p_i^*(\boldsymbol{\lambda}^*) + \sum_{j \in \mathcal{N} \setminus \{i\}} \ln(1 - p_j^*(\boldsymbol{\lambda}^*)) - \bar{x}_i^{min} \geq 0. \tag{4.27}$$

Proof: First, for dual feasibility, we have $\boldsymbol{\lambda}^* \succeq \mathbf{0}$. For cases 1 and 2, $g(\boldsymbol{\lambda})_i$ is differentiable at $\lambda_i = \lambda_i^*$, where $g(\boldsymbol{\lambda})_i$ is the i^{th} element in $g(\boldsymbol{\lambda})$. The i^{th} entry of the derivative becomes

$$\nabla g(\boldsymbol{\lambda}^*)_i = \bar{c}_i + \ln p_i^*(\boldsymbol{\lambda}^*) + \sum_{j \in \mathcal{N} \setminus \{i\}} \ln(1 - p_j^*(\boldsymbol{\lambda}^*)) - \bar{x}_i^v(\lambda_i^*). \tag{4.28}$$

Since problem (4.9) is convex, the result directly follows from [21, pp. 142]. For case 3, $g(\boldsymbol{\lambda})_i$ is non-differentiable at $\lambda_i = \lambda_i^*$; therefore, the KKT condition is $0 \in \partial g(\boldsymbol{\lambda}^*)_i$. From (4.22), we have

$$\begin{aligned} \partial g(\boldsymbol{\lambda}^*)_i &= \operatorname{conv}\{\bar{c}_i + \ln p_i^*(\boldsymbol{\lambda}^*) + \textstyle\sum_{j \in \mathcal{N} \setminus \{i\}} \ln(1 - p_j^*(\boldsymbol{\lambda}^*)) - \bar{x}_i^* : \bar{x}_i^* \in \bar{x}_i^*(\lambda_i^*)\} \\ &= \big[\bar{c}_i + \ln p_i^*(\boldsymbol{\lambda}^*) + \textstyle\sum_{j \in \mathcal{N} \setminus \{i\}} \ln(1 - p_j^*(\boldsymbol{\lambda}^*)) - \bar{x}_i^v(\lambda_i^*), \\ &\qquad \bar{c}_i + \ln p_i^*(\boldsymbol{\lambda}^*) + \textstyle\sum_{j \in \mathcal{N} \setminus \{i\}} \ln(1 - p_j^*(\boldsymbol{\lambda}^*)) - \bar{x}_i^{min}\big], \end{aligned} \tag{4.29}$$

which is simply an interval. Considering the lower and upper bounds in the interval, we can directly derive (4.26) and (4.27), respectively. ■

Since the dual problem is convex, the KKT conditions (4.24)-(4.27) are *necessary* and *sufficient* [21, p. 139]. By using the following theorem, we can determine whether Algorithm 4.1 can solve the NUM problem.

Theorem 4.2 *If $\lambda_i^* \neq \lambda_i^c$, $\forall i \in \mathcal{N}$, then Algorithm 4.1 finds the optimal solution of problem (4.6).*

Proof: If $\lambda_i^* \neq \lambda_i^c,\ \forall\, i \in \mathcal{N}$, from the discussion in Section 4.2.5, $\boldsymbol{p}^* = \boldsymbol{p}^*(\boldsymbol{\lambda}^*)$ and $\bar{\boldsymbol{x}}^* = \bar{\boldsymbol{x}}^*(\boldsymbol{\lambda}^*)$ form the unique minimizer of Lagrangian $L(\boldsymbol{p}, \bar{\boldsymbol{x}}, \boldsymbol{\lambda}^*)$. Thus $g(\boldsymbol{\lambda})$ is differentiable at $\boldsymbol{\lambda}^*$ by (4.22). Finally, from [20, Property 6.5(c)], the primal problem (4.6) has a saddle point $(\boldsymbol{p}^*, \bar{\boldsymbol{x}}^*, \boldsymbol{\lambda}^*)$. By [20, Theorem 5.3], $\boldsymbol{p}^*$ and $\bar{\boldsymbol{x}}^*$ are the global optimum of the primal problem (4.6). ■

4.3 Optimality and Sub-optimality

In this section, we assume that $x_i^{max} = c_i,\ \forall\, i \in \mathcal{N}$ such that the optimal value of the objective function is restricted by capacity $\boldsymbol{c} = (c_i,\ i \in \mathcal{N})$ only, but not the data rate bound $\boldsymbol{x}^{max} = (x_i^{max},\ i \in \mathcal{N})$. Next, we discuss certain conditions on vector $\boldsymbol{c}$ which affect optimality and sub-optimality of Algorithm 4.1.

4.3.1 Optimal Solution

We first provide a sufficient condition on the link capacities for optimality of Algorithm 4.1:

Theorem 4.3 *With $x_i^{max} = \infty$, suppose λ_i^c and $\bar{x}_i^v(\lambda_i^c)$ are obtained by (4.21) and (4.13) for any $i \in \mathcal{N}$, respectively. We define*

$$c_i^c = \frac{e^{\bar{x}_i^v(\lambda_i^c)}}{p_i^*(\boldsymbol{\lambda}^c) \prod_{j \in \mathcal{N} \setminus \{i\}} (1 - p_j^*(\boldsymbol{\lambda}^c))}, \qquad \forall\, i \in \mathcal{N}, \tag{4.30}$$

where $p_i^*(\boldsymbol{\lambda}^c)$ is as in (4.20). Here, $\boldsymbol{c}^c$ denotes the vector of critical link capacities. *If $\boldsymbol{c} \succ \boldsymbol{c}^c$, then Algorithm 4.1 can obtain the optimal solution in problem (4.6), and thus that of problem (4.5).*

Proof: Let $\boldsymbol{\lambda}^*$ be the dual optimal solution and assume that $\boldsymbol{c} \succ \boldsymbol{c}^c$. From (4.30), we have $\bar{c}_i + \ln p_i^*(\boldsymbol{\lambda}^c) + \sum_{j \in \mathcal{N}\setminus\{i\}} \ln(1 - p_j^*(\boldsymbol{\lambda}^c)) - \bar{x}_i^v(\lambda_i^c) > 0$, for any $i \in \mathcal{N}$. We can further show that

$$
\begin{aligned}
0 \notin \partial g(\boldsymbol{\lambda}^c)_i \quad &= [\bar{c}_i + \ln p_i^*(\boldsymbol{\lambda}^c) + \textstyle\sum_{j \in \mathcal{N}\setminus\{i\}} \ln(1 - p_j^*(\boldsymbol{\lambda}^c)) - \bar{x}_i^v(\lambda_i^c), \\
&\quad\ \bar{c}_i + \ln p_i^*(\boldsymbol{\lambda}^c) + \textstyle\sum_{j \in \mathcal{N}\setminus\{i\}} \ln(1 - p_j^*(\boldsymbol{\lambda}^c)) - \bar{x}_i^{min}].
\end{aligned} \tag{4.31}
$$

Thus, $\lambda_i^* \neq \lambda_i^c,\ \forall\, i \in \mathcal{N}$. By Theorem 4.2, Algorithm 4.1 finds the optimum of problem (4.5). ■

The key idea in the proof is that if $\boldsymbol{c} \succ \boldsymbol{c}^c$, $\lambda_i^* < \lambda_i^c$, $\forall\, i \in \mathcal{N}$, then the optimality of the solution directly results from Theorem 4.2.

4.3.2 Sub-optimal Solution: Upper and Lower Bounds

Next, assume that $\boldsymbol{c} \preceq \boldsymbol{c}^c$. In this case, Algorithm 4.1 may only obtain a sub-optimal solution. For cases other than $\boldsymbol{c} \succ \boldsymbol{c}^c$ and $\boldsymbol{c} \preceq \boldsymbol{c}^c$, Algorithm 4.1 may obtain an optimal or sub-optimal solution depending on the exact scenario. Notice that $\bar{\boldsymbol{x}}^*(\boldsymbol{\lambda})$ and $\boldsymbol{p}^*(\boldsymbol{\lambda})$ obtained from lines 5 to 7 of Algorithm 4.1 always satisfy the second and third constraints in problem (4.6), respectively. By Theorem 4.1, the first constraint can be satisfied in all the three cases in (4.24), (4.25), and (4.27). That is, for the case $\lambda_i^* > 0$ and $\lambda_i^* = \lambda_i^c$, we will choose $\bar{x}_i^*(\lambda_i) = \bar{x}_i^{min}$. By weak duality [21, pp. 225], we can obtain an *upper bound* for the objective value of problems (4.5) and (4.6) as

$$
\sum_{i \in \mathcal{N}} U_i(x_i^*) = \sum_{i \in \mathcal{N}} \bar{U}_i(\bar{x}_i^*) \leq g(\boldsymbol{\lambda}^*) = L(\boldsymbol{p}^*(\boldsymbol{\lambda}^*), \bar{\boldsymbol{x}}^*(\boldsymbol{\lambda}^*), \boldsymbol{\lambda}^*). \tag{4.32}
$$

The first equality is due to the fact that problems (4.5) and (4.6) have the same objective function. The inequality is due to weak duality, and the last equality is by def-

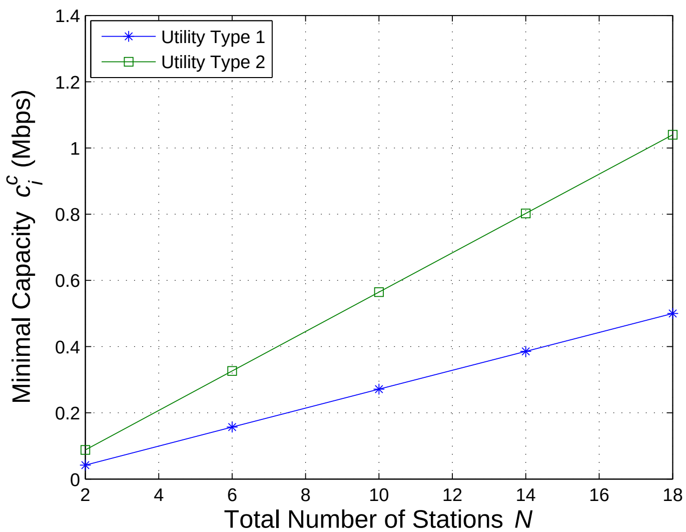

Figure 4.3: The minimal capacities c_1^c and c_2^c for types of utility functions versus the total number of stations N. Type 1 utility functions are concave, while Type 2 utility functions are sigmoidal.

inition. In some cases, we can also obtain a *lower bound* for problem (4.5). If $x_i = c_i p_i^*(\boldsymbol{\lambda}^*) \prod_{j \in \mathcal{N} \setminus \{i\}} (1 - p_j^*(\boldsymbol{\lambda}^*))$ satisfies constraint $x_i^{min} \leq x_i \leq x_i^{max}$, $\forall i \in \mathcal{N}$, by the optimality of $\boldsymbol{x}^*$, we can obtain a lower bound as

$$\sum_{i \in \mathcal{N}} \bar{U}_i(\bar{x}_i^*) = \sum_{i \in \mathcal{N}} U_i(x_i^*) \geq \sum_{i \in \mathcal{N}} U_i \left(c_i p_i^*(\boldsymbol{\lambda}^*) \prod_{j \in \mathcal{N} \setminus \{i\}} (1 - p_j^*(\boldsymbol{\lambda}^*)) \right). \tag{4.33}$$

It should be noted that both the upper and lower bounds are constructed from the same $\boldsymbol{p}^*(\boldsymbol{\lambda}^*)$ and $\bar{\boldsymbol{x}}^*(\boldsymbol{\lambda}^*)$ obtained from Algorithm 4.1. When the duality gap is zero, the upper and lower bounds are both equal to the optimal value of the objective function $\sum_{i \in \mathcal{N}} U_i(x_i^*)$.

4.4 Performance Evaluations

In this section, we consider both cases where $\boldsymbol{c} \succ \boldsymbol{c}^c$ and $\boldsymbol{c} \preceq \boldsymbol{c}^c$ for Algorithm 4.1. We choose $x_i^{min} = 0.0001$ and $x_i^{max} = c_i$, for $\forall\, i \in \mathcal{N}$. Here, we assume that there are two types

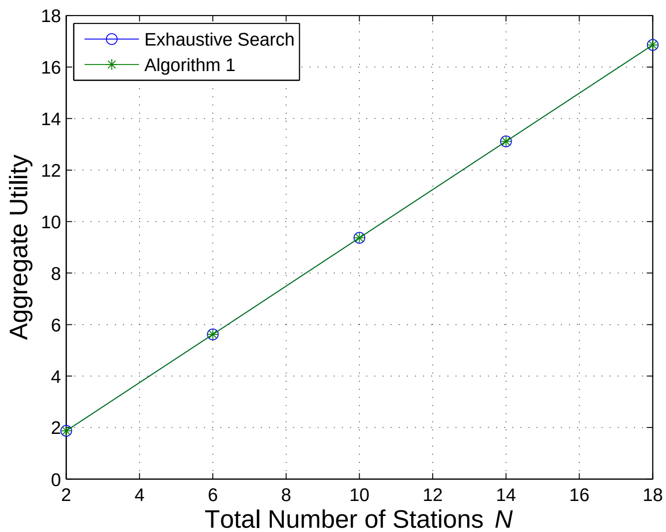

Figure 4.4: Aggregate utility versus the total number of stations N when $\boldsymbol{c} \succ \boldsymbol{c}^c$ using exhaustive search and Algorithm 4.1.

of utility functions used in the network: a concave (Type 1) function $U_1(x_1) = 1-(x_1+1)^{-1}$, and a sigmoidal (Type 2) function $U_2(x_2) = \frac{x_2^2}{x_2^2+20}$, as shown in Fig. 4.1. We can then obtain $\lambda_1^c = 0.0789$ and $\lambda_2^c = 0.0780$ numerically (e.g., using MATLAB as in Fig. 4.2). We assume that there are N stations in the network, where half of them are Type 1, and the other half are Type 2. We plot c_1^c and c_2^c versus the total number of stations N in Fig. 4.3. We can see that only a linear increase in c_1^c and c_2^c is required for Algorithm 4.1 to find the optimal solution when N increases.

Next, we plot the aggregate utility versus the number of stations in Fig. 4.4 to verify the optimality of Algorithm 4.1 when $\boldsymbol{c} \succ \boldsymbol{c}^c$. We can see that the result of the exhaustive search is identical to that of Algorithm 4.1, meaning that Algorithm 4.1 obtains the optimal solution. Then, we consider the case where $\boldsymbol{c} \prec \boldsymbol{c}^c$ by using the capacities $\boldsymbol{c} = 0.5\boldsymbol{c}^c$. Fig. 4.5 shows the upper and lower bounds obtained from Algorithm 4.1.

Next, we focus on the case when $N = 2$, $c_1 = 21$ kbps and $c_2 = 44$ kbps to study the resource allocation when $\boldsymbol{c} \prec \boldsymbol{c}^c$. With the use of diminishing step size $\alpha(t) = 0.01/t$, the

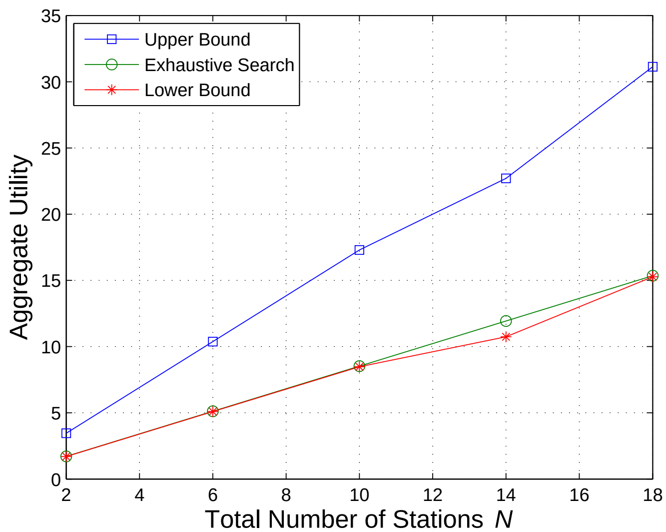

Figure 4.5: Aggregate utility versus the total number of stations N when $\boldsymbol{c} \prec \boldsymbol{c}^c$ using exhaustive search and Algorithm 4.1. The lower and upper bounds are obtained by replacing $\boldsymbol{p}^*(\boldsymbol{\lambda}^*)$ and $\bar{\boldsymbol{x}}^*(\boldsymbol{\lambda}^*)$ (i.e., the results from Algorithm 4.1) into the expressions in (4.33) and (4.32), respectively. We can see that the lower bound is very tight in this case. In fact, except for the case with 14 stations, the lower bound exactly matches the global optimal solution in all other considered cases.

allocations of persistent probabilities converge, as shown in Fig. 4.6. Moreover, we have noticed in the simulation that $\lambda_1(t) \rightarrow \lambda_1^* = \lambda_1^c$ and $\lambda_2(t) \rightarrow \lambda_2^* = \lambda_2^c$ as $t \rightarrow \infty$. It can be verified, by simulation, that the use of a constant step size leads to oscillatory behaviour in the dual variables and the allocation of persistent probabilities.

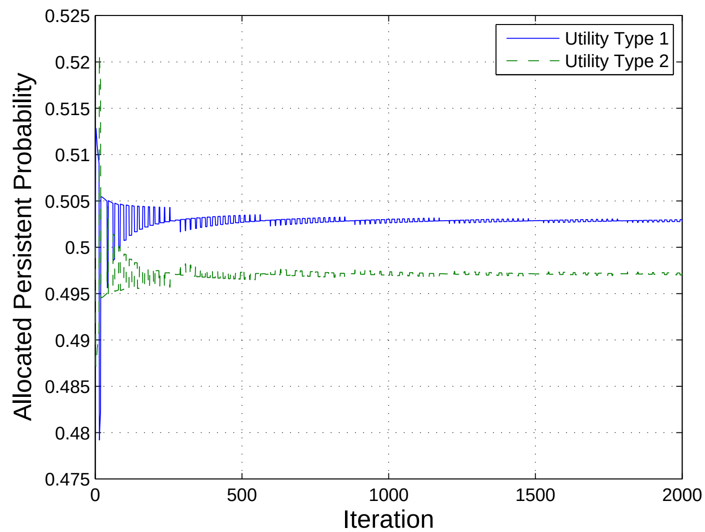

Figure 4.6: Convergence of the allocation of the persistent probabilities with insufficient capacity $\boldsymbol{c} \prec \boldsymbol{c}^c$ using diminishing step size $\alpha(t) = 0.01/t$, even though the allocation may not be globally optimal as discussed in Section 4.3.

4.5 Summary

In this chapter, we proposed a random access algorithm based on the NUM framework for stations with either concave or sigmoidal utilities. We applied the dual method to solve our problem. A sufficient condition on link capacities that guarantee the optimality of the solution is proposed. Simulations have been performed to verify our analytical results.

Chapter 5

Dynamic Optimal Random Access for Vehicle-to-Roadside Communications

In this chapter, we aim to design a *uplink* random access algorithm that is *distributed* in nature, so that it is compatible with the IEEE 802.11p standard that is developed to facilitate the provision of wireless access in vehicular environment [16, 17]. Different from most previous works on *heuristic* distributed uplink V2R communication algorithm design, we aim at designing an *optimal* uplink resource allocation scheme in VANETs analytically.

Specifically, we consider the drive-thru scenario [85], where vehicles pass by several APs located along a highway and obtain Internet access for only a limited amount of time. We assume that a vehicle wants to upload a file when it is within the coverage ranges of the APs, and needs to pay for the attempts to access the channel. As both the channel contention level and achievable data rate vary over time, the vehicle needs to decide when to transmit by taking into account the required payment, the application's QoS requirement, and the level of contention in current and future time slots. Because of the dynamic nature of the problem, we formulate it as a finite-horizon sequential decision problem and solve it using the dynamic programming (DP). The main contributions of this chapter are as follows:

- In the case of a single AP with random vehicular traffic, we propose a general dynamic optimal random access (DORA) algorithm to compute the optimal access policy. We further extend the results to the case of multiple consecutive APs and propose a joint DORA (JDORA) algorithm to compute the optimal policy.

- We consider a special yet practically important case of a single AP with constant data rate. We show that the optimal policy in this case has a threshold structure, which motivates us to propose a low complexity and efficient monotone DORA algorithm.

- Extensive simulation results show that our proposed algorithms achieve the minimal total cost and the highest upload ratio as compared with three other heuristic schemes. In the multi-AP scenario, the performance improvements in upload ratio of the JDORA scheme are 130% and 207% at low and high traffic density, respectively.

The rest of this chapter is organized as follows. We present the related work in Section 5.1. We describe our system model in Section 5.2 and formulate the DP problem in Section 5.3. The general and monotone DORA algorithms for single AP are proposed in Section 5.4.1, and the JDORA algorithm for multiple APs is discussed in Section 5.4.2. Simulation results are given in Section 5.5, and a summary is given in Section 5.6. A list of the key notations used in this chapter is given in Table 5.1.

Table 5.1: List of Key Notations

Notation	Meaning
λ	Arrival rate of the vehicles
ρ, ρ_{max}	Density of the vehicles and density of the vehicles during traffic jam
ν, ν_f	Speed and free-flow speed of the vehicles
j, $\mathcal{J}$	Index of an AP and its feasible set
R_j	Transmission radius of the j^{th} AP
P	Transmit power of the vehicle
γ	Path loss exponent
W	Channel bandwidth
$N_0/2$	Power spectral density of the Gaussian noise
d_t	Distance between the vehicle and the closest AP at time slot t
Δt	Length of a time slot
Δt_{data}	Length of time in a time slot for data transmission
t, $\mathcal{T}_j$, $\mathbb{T}$	Time slot, its feasible set in the j^{th} coverage range, and its overall feasible set
T_j	Total number of time slots that the vehicle stays within the j^{th} coverage range
a, $\mathcal{A}$	Action and its feasible set
S, s, $\mathcal{S}$	Total file size, remaining file size to be uploaded, and its feasible set
p^{succ}, p_t^{succ}, $\mathcal{P}$	Probability of successfully gaining access to the time slot, its value at time slot t, and its feasible set
q_j	Payment per time slot in the j^{th} coverage range
w_t	Data rate at time slot t
l_t	Number of vehicles leaving the j^{th} coverage range at time slot $t \in \mathcal{T}_j$
$\delta_t(\cdot)$	Decision rule at time slot t
$\boldsymbol{\pi}$, Π	Transmission policy and its feasible set
$h(\cdot)$	Self-incurred penalty
σ	Granularity of discrete state element s in the algorithms
n, n_t, $\mathcal{N}_j$	Number of vehicles in the coverage range, its value at time slot t, and its feasible set in the j^{th} coverage range
$N_{max,j}$	Maximum number of vehicles that can be accommodated in the j^{th} coverage range
$g_j(\cdot)$	A function used by the j^{th} AP that maps n to p^{succ}
$\zeta(j,\tau)$	A function that maps the τ^{th} time slot in the j^{th} coverage range to the time line as shown in Fig. 5.2
c_t	Cost at time slot t
$\hat{c}$	Terminal cost
$v_t(s, p^{succ})$	Minimal expected total cost in state (s, p^{succ}) from time t over the planning horizon
$\psi_t(s, p^{succ}, a)$	Expected total cost in state (s, p^{succ}) from time t over the planning horizon if action a is chosen
K, p_r, F	MAC parameters in the MCBC scheme [86]

5.1 Related Work

Several centralized and distributed resource allocation schemes have been proposed for VANETs. In the *centralized* setting, the AP schedules the transmissions from different vehicles based on some predefined criteria. Hadaller *et al.* in [87] proposed a scheduling protocol that grants channel access to a vehicle that achieves the maximum transmission rate. Analytical and simulation results showed significant overall system throughput improvement over a benchmark scheme. Chang *et al.* in [88] proposed a heuristic downlink scheduling algorithm for dedicated short range communication (DSRC) networks. It aims to reduce the handoff rate under some delay constraints of the vehicles. Based on a metric which measures the chance that a vehicle can be served completely within the coverage range of an AP, the algorithm divides the vehicles into two priority classes for scheduling. Zhang *et al.* in [89] considered the case where roadside APs only store the data uploaded by the vehicles. Scheduling priority is determined by two factors: data size and deadline. A request with either a smaller data size or an earlier deadline will be served first. Alcaraz *et al.* in [90] considered both uplink and downlink scheduling of non-real-time traffic for non-safety applications. The scheduling problem was formulated as a constrained linear quadratic regulator design problem that aims to reduce the residual queue backlog for each user. However, because centralized resource allocation is not scalable due to its computational complexity, we focus on distributed resource allocation scheme in this chapter.

In the *distributed* setting, the vehicles contend for the channel for transmission based on the applications' QoS requirements. Yang *et al.* in [91] proposed a cross-layer protocol called coordinated external peer communication for multi-hop V2R communications. The roads are divided into segments such that the vehicles are divided into clusters. Information is aggregated and passed between cluster heads. Shrestha *et al.* in [92] considered the scenario where the data packets are first distributed from the roadside units (RSUs) to the onboard units (OBUs). The OBUs then bargain with each other for the missing data packets, and exchange them using BitTorrent protocol. Jarupan *et al.* in [93] proposed a cross-layer protocol for V2R multi-hop communication. The MAC module collects information of local data traffic, and the routing module finds a path with the minimum delay. Niyato *et al.* in [94] proposed a hierarchical optimization framework for downlink data streaming in V2R communications. The optimal pricing and bandwidth reservation of a service provider is obtained using game theory, and the optimal download policy of an OBU is obtained using constrained Markov decision processes. Sikdar in [95] proposed a V2R protocol, where the AP first obtains an estimate of the number of active nodes in the network, and then uses it to determine the optimal number of data contention slots. An information-theoretic lower bound on the MAC layer overhead was derived related to node reassociations induced by the mobility of the nodes. Tan *et al.* in [96] analyzed the performance of a downlink resource allocation scheme in a V2R communication system with one AP on a road. The resources are equally shared among all the vehicles that are within the coverage range of the AP. The distribution of the number of bytes downloaded per drive-thru was derived using Markov reward processes. Roman *et al.* in [86] proposed a cross-layer protocol in the physical and MAC layers that addresses the issues of channel fading, synchronization, and channel contention. Performance analysis was presented for the channel contention scheme, and a testbed was used to evaluate the proposed protocol.

5.2 System Model

We consider a drive-thru scenario on a highway as shown in Fig. 5.1, where multiple APs are installed and connected to a backbone network to provide Internet services to vehicles within their coverage ranges. We focus on a vehicle that wants to upload a *single* file of size S when it moves through a segment of this highway with a set of APs $\mathcal{J} = \{1, \dots, J\}$,

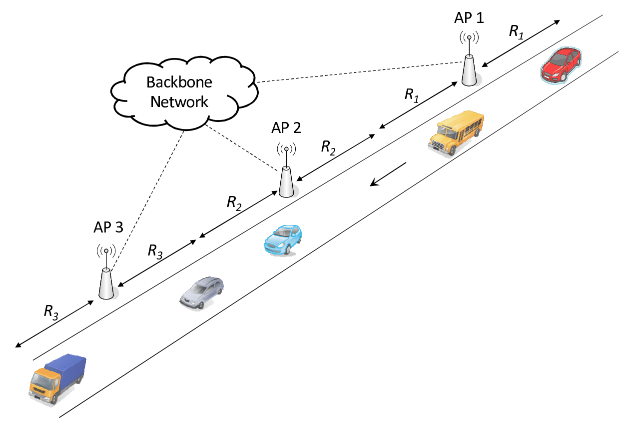

Figure 5.1: Drive-thru V2R communications with multiple APs.

where the vehicles pass through the i^{th} AP before the j^{th} AP for $i < j$ with $i, j \in \mathcal{J}$. We assume that the j^{th} AP has a transmission radius R_j. We also assume that the vehicle is connected to at most one AP at a time. If the coverage areas of the APs are overlapping, then proper handover between the APs will be performed [97]. For the ease of exposition, we assume that the APs are set up in a way that any position in this segment of highway is covered by an AP. Our work can easily be extended to consider the settings where the coverage areas of adjacent APs are isolated from each other.

5.2.1 Traffic Model

Let λ denote the average number of vehicles passing by a fixed AP per unit time. We assume that the number of vehicles moving into this segment of the highway follows a Poisson process [98] with a mean arrival rate λ. Let ρ denote the vehicle density representing the number of vehicles per unit distance along the road segment, and ν be the speed of the vehicles. From [99], we have

$$\lambda = \rho\nu. \tag{5.1}$$

The relation between the vehicle density ρ and speed ν is given by the following equation [99]:

$$\nu = \nu_f(1 - \rho/\rho_{max}), \tag{5.2}$$

where ν_f is the free-flow speed when the vehicle is moving on the road without any other vehicles, and ρ_{max} is the vehicle density during traffic jam.

As we are studying the traffic flow in *steady state*, all the vehicles within the coverage range are assumed to move with the same speed ν in (5.2). Let $\lfloor \cdot \rfloor$ denote the floor function. The maximum number of vehicles that can be accommodated within the coverage range of the j^{th} AP is given by

$$N_{max,j} = \lfloor 2R_j\rho_{max} \rfloor, \quad \forall\, j \in \mathcal{J}. \tag{5.3}$$

5.2.2 Channel Model

Wireless signal propagations suffer from path loss, shadowing, and fading. Since the distance between the vehicle and the AP varies in the drive-thru scenario, we focus on the dominant effect of channel attenuation due to path loss. The data rate at time slot t is given by

$$w_t = W \log_2 \left(1 + \frac{P}{N_0 W d_t^{\gamma}} \right), \tag{5.4}$$

where W is the channel bandwidth, P is the transmit power of the vehicle, d_t is the distance between the vehicle and the closest AP at time slot t, and γ is the path loss exponent. We assume that the additive white Gaussian noise has a zero mean and a power spectral density $N_0/2$. In addition, we also consider a special case with fixed data rate in Section 5.4.1.

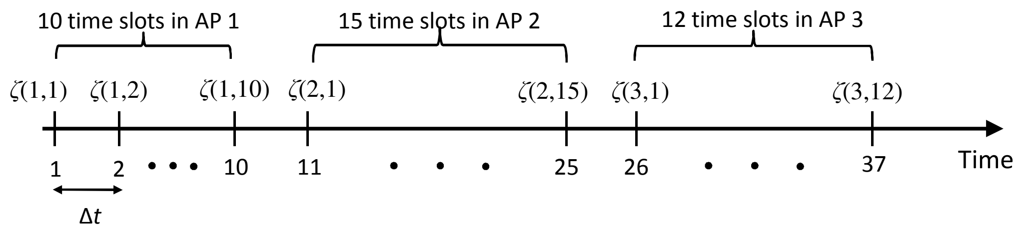

Figure 5.2: An example of the time line representation for the events happened with three APs (i.e., $\mathcal{J} = \{1, 2, 3\}$). Here, we assume that $T_1 = 10$, $T_2 = 15$, and $T_3 = 12$. With respect to the time line, we have $\mathcal{T}_1 = \{1, \ldots, 10\}$, $\mathcal{T}_2 = \{11, \ldots, 25\}$, and $\mathcal{T}_3 = \{26, \ldots, 37\}$. It is clear from the figure that $\zeta(j, \tau) = \sum_{i=0}^{j-1} T_i + \tau$, $\forall\, \tau \in \{1, \ldots, T_j\}$, where $T_0 = 0$.

5.2.3 Distributed Medium Access Control

We consider a slotted MAC protocol, where time is divided into equal time slots of length Δt. We assume that there is perfect synchronization between the APs and the vehicles with the use of global positioning system (GPS) [18]. The total number of time slots that the vehicle stays within the coverage range of the j^{th} AP is $T_j = \left\lfloor \frac{2R_j}{\nu \Delta t} \right\rfloor$. We use the notation $\zeta(j, \tau)$ to denote the τ^{th} time slot when the vehicle is in the coverage area of the j^{th} AP, i.e.,

$$\zeta(j, \tau) = \sum_{i=0}^{j-1} T_i + \tau, \quad \forall\, \tau \in \{1, \ldots, T_j\}, \tag{5.5}$$

where $T_0 = 0$. The set of time slots in the j^{th} AP with respect to this time line representation is $\mathcal{T}_j = \{\zeta(j, 1), \ldots, \zeta(j, T_j)\}$. An example of the time line representation is given in Fig. 5.2.

When the vehicle first *enters the coverage range* of the j^{th} AP, it declares the type of its application to the AP. In return, the j^{th} AP informs the vehicle the channel contention in the coverage range (λ and $p_t^{succ}, \forall\, t \in \mathcal{T}_j$), data rate in all the time slots in the j^{th} coverage range (i.e., $w_t, \forall\, t \in \mathcal{T}_j$), the price q_j, and the estimated number of vehicle departures from the coverage range in all the time slots in the j^{th} coverage range (i.e., $l_t, \forall\, t \in \mathcal{T}_j$). We further elaborate these system parameters as follows:

- p_t^{succ} represents the probability that the vehicle can successfully obtain access in time slot $t \in \mathcal{T}_j$ after contending with all the vehicles in the j^{th} coverage range. p_t^{succ} is estimated by the AP based on the level of system contention and it varies over time. Since p_t^{succ} is related to the number of vehicles n_t currently in the j^{th} coverage range at time slot t, we define $p_t^{succ} = g_j(n_t)$, where g_j is a strictly decreasing function. An AP knows the value of n_t, since vehicles need to establish and terminate their connections when they enter and leave the coverage range, respectively.

- $q_j \geq 0$ denotes the amount (e.g., in virtual currency) that a vehicle needs to pay the AP for each time slot that it sends a transmission request in the j^{th} coverage range, even it fails to access the channel. The value of q_j does not change over time. Moreover, in order to provide QoS support, an AP may charge differently for vehicles running different types of applications.

- l_t represents the number of vehicle departing at time slot $t \in \mathcal{T}_j$ from the j^{th} coverage range. Since all the vehicles move with constant speed ν in the traffic model, we assume that $(l_t, \forall\, t \in \mathcal{T}_j)$ are accurately known by the j^{th} AP, and are sent to the vehicle when it enters the coverage range.

In each time slot $t \in \mathcal{T}_j$ in the j^{th} coverage range, the j^{th} AP first broadcasts the value of p_t^{succ} to all the vehicles in its coverage range. If a vehicle decides to transmit within this time slot, it sends a request to the j^{th} AP at its scheduled mini-slot, where $N_{max,j}$ mini-slots are reserved for transmission requests. The transmissions of requests are thus collision-free. After collecting the requests from all vehicles in its coverage range, the j^{th} AP assigns the time slot to one of these vehicles. The vehicle, which receives the

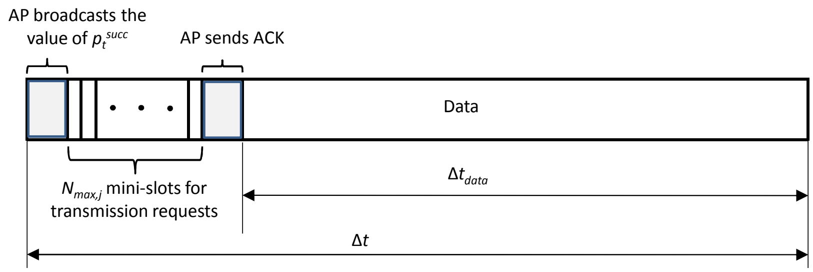

Figure 5.3: The structure of a time slot of the j^{th} AP.

acknowledgement (ACK), can transmit the data packets in the remaining time Δt_{data} of this time slot, where $\Delta t_{data} < \Delta t$. The structure of a time slot is shown in Fig. 5.3.

Meanwhile, regardless of which vehicle is granted the time slot, each vehicle which requested to transmit in the time slot needs to pay q_j to the j^{th} AP. Without such pricing, each vehicle would send a request in every time slot, which unnecessarily increases the contention level and prevents efficient allocation of time slots to the most needed application.

The vehicle aims to achieve a good *tradeoff* between the total uploaded file size and the total payment to the APs according to the QoS requirement of the application. For example, a higher priority may be placed on the total uploaded file size for safety applications, but on the total payment for non-safety applications. The problem is further complicated by the time-varying data rate w_t and channel contention level. Therefore, it is a challenge for the vehicle to decide when to request for data transmission.

5.3 Problem Formulation

In this section, we formulate the optimal transmission problem of *a single vehicle* as a *finite-horizon sequential* decision problem [28]. The *decision epochs* of the vehicle are

$$t \in \mathbb{T} = \bigcup_{j \in \mathcal{J}} \mathcal{T}_j = \bigcup_{j \in \mathcal{J}} \{\zeta(j,1), \ldots, \zeta(j,T_j)\}, \tag{5.6}$$

where $\mathbb{T}$ is the set of all the time slots in the total of J coverage ranges.

The system *state* of a vehicle is defined as (s, p^{succ}), where the state element $s \in \mathcal{S} = [0, S]$ represents the remaining size (in bits) of the single file to be uploaded. If we denote the number of vehicles in the coverage range of the j^{th} AP as $n \in \mathcal{N}_j = \{1, \ldots, N_{max,j}\}$, then $p^{succ} \in \mathcal{P}_j = \{g_j(n) : n \in \mathcal{N}_j\}$.

At any state (s, p^{succ}), the vehicle has two possible *actions*:

$$a \in \mathcal{A} = \{0, 1\}, \tag{5.7}$$

where action $a = 1$ implies that the vehicle decides to request to transmit, and action $a = 0$ otherwise.

The *cost* at state (s, p^{succ}) with action $a \in \mathcal{A}$ at time slot $t \in \mathcal{T}_j$ in the j^{th} coverage range is

$$c_t(s, p^{succ}, a) = aq_j, \quad \forall\, t \in \mathcal{T}_j. \tag{5.8}$$

After the vehicle has left the J^{th} coverage range at time $\zeta(J, T_J + 1)$, we define a *self-incurred penalty* of the vehicle for not being able to complete the file uploading as

$$\hat{c}_{\zeta(J,T_J+1)}(s, p^{succ}) = h(s), \tag{5.9}$$

where $h(s) \geq 0$ is a nondecreasing function of s with $h(0) = 0$. The function depends on the QoS requirement of the application. To sum up, each vehicle is incurred with two costs: the transmission cost in each time slot in (5.8) and the penalty after leaving the J^{th} coverage range in (5.9).

The *state transition probability* $p_t\big((s', p^{succ'}) \,|\, (s, p^{succ}), a\big)$ is the probability that the system will go into state $(s', p^{succ'})$ if action a is taken at state (s, p^{succ}) at time slot $t \in \mathbb{T}$. Since the transition from p^{succ} to $p^{succ'}$ is independent of the value of s but depends on time t, we have

$$p_t\big((s', p^{succ'}) \,|\, (s, p^{succ}), a\big) = p_t\big(s' \,|\, (s, p^{succ}), a\big)\; p_t\big(p^{succ'} \,|\, p^{succ}\big). \tag{5.10}$$

With action $a = 1$, we have

$$p_t\big(s' \,|\, (s, p^{succ}), 1\big) = \begin{cases} p^{succ}, & \text{if } s' = [s - w_t \Delta t_{data}]^+, \\ 1 - p^{succ}, & \text{if } s' = s, \\ 0, & \text{otherwise,} \end{cases} \tag{5.11}$$

where $[x]^+ = \max\{0, x\}$. The first and second cases correspond to the scenarios of successful and unsuccessful packet transmissions, respectively. With action $a = 0$, we have

$$p_t\big(s' \,|\, (s, p^{succ}), 0\big) = \begin{cases} 1, & \text{if } s' = s, \\ 0, & \text{otherwise,} \end{cases} \tag{5.12}$$

where the remaining size of the file to upload does not change. The derivation of $p_t\big(p^{succ'} \,|\, p^{succ}\big)$ will be discussed in detail in Section 5.4.

Let $\delta_t : \mathcal{S} \times \mathcal{P}_j \to \mathcal{A}$ be the *decision rule* that specifies the transmission decision of the vehicle at state (s, p^{succ}) at time slot $t \in \mathcal{T}_j$ in the j^{th} coverage range. We define a *policy* as a set of decision rules covering all the states as $\boldsymbol{\pi} = (\delta_t(s, p^{succ}), \forall\, s \in \mathcal{S}, p^{succ} \in \mathcal{P}_j, t \in \mathcal{T}_j, \forall\, j \in \mathcal{J})$. We denote $(s_t^{\boldsymbol{\pi}}, p_t^{succ,\boldsymbol{\pi}})$ as the state at time slot t if policy $\boldsymbol{\pi}$ is used, and we let Π be the feasible set of $\boldsymbol{\pi}$. The vehicle aims to find an optimal policy that minimizes the total expected cost, which can be formulated as the following optimization problem

$$\min_{\boldsymbol{\pi}\in\Pi} E_{\boldsymbol{\pi},(S,p_1^{succ})}\left[\sum_{j=1}^{J}\left[\sum_{\tau=1}^{T_j} c_{\zeta(j,\tau)}\Big(s^{\boldsymbol{\pi}}_{\zeta(j,\tau)}, p^{succ,\boldsymbol{\pi}}_{\zeta(j,\tau)}, \delta_{\zeta(j,\tau)}(s^{\boldsymbol{\pi}}_{\zeta(j,\tau)}, p^{succ,\boldsymbol{\pi}}_{\zeta(j,\tau)})\Big)\right]\right. \\ \left. + \hat{c}_{\zeta(J,T_J+1)}(s^{\boldsymbol{\pi}}_{\zeta(J,T_J+1)}, p^{succ,\boldsymbol{\pi}}_{\zeta(J,T_J+1)})\right], \tag{5.13}$$

where $E_{\boldsymbol{\pi},(S,p_1^{succ})}$ denotes the expectation with respect to the probability distribution by policy $\boldsymbol{\pi}$ with an initial state (S, p_1^{succ}) at time slot $t = \zeta(1,1) = 1$. In the following section, we will study two scenarios: single AP with random vehicular traffic and multiple APs with traffic pattern estimation.

5.4 Finite-Horizon Dynamic Programming

In this section, we describe how to obtain the optimal transmission policies in both the single-AP and multiple-AP scenarios using *finite-horizon dynamic programming*. We first study the single-AP scenario with random vehicular traffic arrival in Section 5.4.1. In particular, we consider a special case that the optimal policy has a *threshold structure* in Section 5.4.1. When the traffic pattern can be estimated accurately, we consider a joint AP optimization in Section 5.4.2.

5.4.1 Single AP Optimization with Random Vehicular Traffic

Since we are considering one AP (i.e., $\mathcal{J} = \{1\}$) in this subsection, we drop the subscript j for simplicity. Although the exact traffic pattern (i.e., the exact number of vehicles in the coverage range of the AP in each time slot) is not known, the vehicles arrive according to a Poisson process with parameter λ. Meanwhile, the parameters $l_t\,(\forall\, t \in \mathcal{T})$, ρ_{max}, Δt,

R, and the function $g(\cdot)$ are available. The transition probability of p^{succ} is given by

$$p_t\big(p^{succ'} \,|\, p^{succ}\big) = p_t\big(g(n') \,|\, g(n)\big) = p_t\big(n' \,|\, n\big) = \begin{cases} \frac{(\lambda\Delta t)^{n'-n+l_{t+1}}}{(n'-n+l_{t+1})!\,\phi_t(n)}, & \text{if } n - l_{t+1} \leq n' \leq N_{max}, \\ 0, & \text{otherwise}, \end{cases} \tag{5.14}$$

where $\phi_t(n) = \sum_{y=0}^{N_{max}-n+l_{t+1}} \frac{(\lambda\Delta t)^y}{y!}$ is a normalization factor. Because $p^{succ} = g(n)$ is a strictly decreasing function of n, there is a one-to-one mapping between p^{succ} and n as shown in the first two equalities in (5.14). The expression after the third equalities describes the probability with $n' - n + l_{t+1}$ arrivals due to the Poisson process and l_{t+1} deterministic departures at time $t+1$. n' is lower-bounded by $n - l_{t+1} \geq 0$ when there is no vehicle arrival, and is upper-bounded by N_{max}.

In this subsection, since we consider $\mathcal{J} = \{1\}$, we can simplify problem (5.13) as

$$\min_{\boldsymbol{\pi}\in\Pi} E_{\boldsymbol{\pi},(S,p_1^{succ})}\left[\sum_{t=1}^{T} c_t\Big(s_t^{\boldsymbol{\pi}}, p_t^{succ,\boldsymbol{\pi}}, \delta_t(s_t^{\boldsymbol{\pi}}, p_t^{succ,\boldsymbol{\pi}})\Big) + \hat{c}_{T+1}(s_{T+1}^{\boldsymbol{\pi}}, p_{T+1}^{succ,\boldsymbol{\pi}})\right]. \tag{5.15}$$

Let $v_t(s, p^{succ})$ be the minimal expected total cost that the vehicle has to pay from time t to time $T+1$ when it is in the coverage range, given that the system is in state (s, p^{succ}) immediately before the decision at time slot $t \in \mathcal{T}$. The *optimality equation* [28, pp. 83] relating the minimal expected total cost at different states for $t \in \mathcal{T}$ is

$$v_t(s, p^{succ}) = \min_{a\in\mathcal{A}}\{\psi_t(s, p^{succ}, a)\}, \tag{5.16}$$

where

$$\psi_t(s, p^{succ}, a) = c_t(s, p^{succ}, a) + \sum_{s' \in \mathcal{S}} \sum_{p^{succ'} \in \mathcal{P}} p_t\big((s', p^{succ'}) \,|\, (s, p^{succ}), a\big) v_{t+1}(s', p^{succ'}) \quad (5.17)$$

$$= aq + \sum_{p^{succ'} \in \mathcal{P}} p_t\big(p^{succ'} \,|\, p^{succ}\big) \Big[a p^{succ} v_{t+1}\big([s - w_t \Delta t_{data}]^+, p^{succ'}\big) + \big(1 - a p^{succ}\big) v_{t+1}(s, p^{succ'}) \Big]. \quad (5.18)$$

The first and second terms on the right hand side of (5.17) are the *immediate cost* and the *expected future cost* in the remaining time slots in the coverage range for choosing action a, respectively. Equation (5.18) follows directly by evaluating (5.17) using (5.10) - (5.12). For time $t = T + 1$, we have the boundary condition that

$$v_{T+1}(s, p^{succ}) = \hat{c}_{T+1}(s, p^{succ}) = h(s). \quad (5.19)$$

Lemma 5.1 *The value of $\psi_t(s, p^{succ}, a)$, $\forall\, t \in \mathcal{T}$, can be obtained as*

$$\psi_t(s, p^{succ}, a) = aq + \sum_{m=0}^{N_{max} - n + l_{t+1}} \frac{(\lambda \Delta t)^m}{m! \phi_t(n)} \Big[a p^{succ} v_{t+1}\big([s - w_t \Delta t_{data}]^+, g(n + m - l_{t+1})\big) + \big(1 - a p^{succ}\big) v_{t+1}\big(s, g(n + m - l_{t+1})\big) \Big], \quad (5.20)$$

where $n = g^{-1}(p^{succ})$ is the number of vehicles in the coverage range of the AP.

Proof: The result follows directly by evaluating (5.18) using (5.14). ■

Intuitively, the minimal expected cost $v_t(s, p^{succ})$ should be smaller when the remaining file size s to be uploaded is smaller. It is confirmed by the following lemma:

Lemma 5.2 *$v_t(s, p^{succ})$ is a nondecreasing function in s, $\forall\, p^{succ} \in \mathcal{P}, t \in \mathcal{T}$.*

Proof: We prove it by induction. From (5.19), since $v_{T+1}(s, p^{succ}) = h(s)$, $v_{T+1}(s, p^{succ})$ is a nondecreasing function in s, $\forall\, p^{succ} \in \mathcal{P}$. Assume that $v_{t+1}(s, p^{succ})$ is a nondecreasing function in s, $\forall\, p^{succ} \in \mathcal{P}$. Since $p_t\big(p^{succ'} \,|\, p^{succ}\big) \geq 0$ and $0 \leq ap^{succ} \leq 1$, it can be inferred from (5.18) that $\psi_t(s, p^{succ}, a)$ is a nondecreasing function in s, $\forall\, p^{succ} \in \mathcal{P}$. Thus, $v_t(s, p^{succ})$ in (5.16) is a nondecreasing function in s, $\forall\, p^{succ} \in \mathcal{P}$. ■

Using the optimality equation and *backward induction* [28, pp. 92], we propose the general dynamic optimal random access (DORA) algorithm in Algorithm 5.1 to obtain the *optimal policy* $\boldsymbol{\pi}^* = (\delta_t^*(s, p^{succ}), \forall\, s \in \mathcal{S}, p^{succ} \in \mathcal{P}, t \in \mathcal{T})$, where

$$\delta_t^*(s, p^{succ}) = \arg\min_{a \in \mathcal{A}} \{\psi_t(s, p^{succ}, a)\}. \tag{5.21}$$

Theorem 5.1 *The policy $\boldsymbol{\pi}^*$ obtained from Algorithm 5.1 is the optimal solution of problem (5.15).*

Proof: Using the principle of optimality [29, pp. 18], we can show that $\boldsymbol{\pi}^*$ is the optimal solution of problem (5.15). ■

The proposed DORA algorithm consists of two phases: Planning phase and transmission phase. The planning phase starts when the vehicle enters the coverage range. The vehicle then obtains information from the AP and computes the optimal policy $\boldsymbol{\pi}^*$ offline using dynamic programming. In fact, $\boldsymbol{\pi}^*$ is a *contingency plan* that contains information about the optimal decisions at *all* possible states (s, p^{succ}) in the coverage range. In the transmission phase, the transmission decision in each time slot is made according to the optimal policy $\boldsymbol{\pi}^*$, and s is updated depending on whether the time slot is granted to the vehicle for transmission or not. Note that the computational complexity of the algorithm is directly proportional to the dimension of the optimal policy, which is given by $|\mathcal{S}| \times |\mathcal{P}| \times T$.

Algorithm 5.1 *General DORA Algorithm for single AP optimization (i.e., problem (5.15)).*

Planning Phase:
Input the traffic parameters: ν, λ, ρ_{max}, $l_t\,(\forall\, t \in \mathcal{T})$;
Input the system parameters: $h(\cdot)$, S, R, $w_t\,(\forall\, t \in \mathcal{T})$, q, Δt, Δt_{data}, σ, $g(\cdot)$;
Set the boundary condition $v_{T+1}(s, p^{succ})$, $\forall\, s \in \mathcal{S}, \forall\, p^{succ} \in \mathcal{P}$ using (5.19);
$t := T$;
while $t \geq 1$
    for $p^{succ} \in \mathcal{P}$
        $s := 0$;
        while $s \leq S$
            Calculate $\psi_t(s, p^{succ}, a)$, $\forall\, a \in \mathcal{A} = \{0, 1\}$ using (5.20);
            $\delta_t^*(s, p^{succ}) := \arg\min_{a \in \mathcal{A}} \{\psi_t(s, p^{succ}, a)\}$;
            $v_t(s, p^{succ}) := \psi_t\big(s, p^{succ}, \delta_t^*(s, p^{succ})\big)$;
            $s := s + \sigma$;
        end while
    end for
    $t := t - 1$;
end while
Output the optimal policy $\boldsymbol{\pi}^*$ for use in the transmission phase;
Transmission Phase:
$t := 1$ and $s := S$;
while $t \leq T$
    Receive the information of p^{succ} from the AP;
    Set action $a := \delta_t^*(s, p^{succ})$ based on the policy $\boldsymbol{\pi}^*$;
    If action $a = 1$
        Send a request to the AP;
        If ACK is received from the AP
            Transmit packets with total size $w_t \Delta t_{data}$;
            $s := [s - w_t \Delta t_{data}]^+$;
        end if
    end if
    $t := t + 1$;
end while

Special Case: Convex Penalty Function and Fixed Data Rate

In this subsection, we further investigate a special yet practically important case with convex penalty function and non-adaptive data rate [96]. The key idea is that if the self-incurred penalty function $h(s)$ is *convex* and the data rate w_t is fixed within the coverage range (i.e., $w_t = w, \forall\, t \in \mathcal{T}$), we can show that $\psi_t(s, p^{succ}, a)$ is *submodular* [28, pp. 103] on $\mathcal{S} \times \mathcal{A}$, $\forall\, t \in \mathcal{T}$, which is defined as follows.

Definition 5.1 *Given p^{succ}, the function $\psi_t(s, p^{succ}, a)$ is submodular on $\mathcal{S} \times \mathcal{A}$ if for $\hat{s}, \check{s} \in \mathcal{S}$ and $\hat{a}, \check{a} \in \mathcal{A}$, where $\hat{s} \geq \check{s}$ and $\hat{a} \geq \check{a}$, we have*

$$\psi_t(\hat{s}, p^{succ}, \hat{a}) + \psi_t(\check{s}, p^{succ}, \check{a}) \leq \psi_t(\hat{s}, p^{succ}, \check{a}) + \psi_t(\check{s}, p^{succ}, \hat{a}). \tag{5.22}$$

Furthermore, with $\delta_t^*(s, p^{succ})$ as defined in (5.21), we can establish the threshold structure of the optimal policy [28, 100, 101].

The details of the derivation of the threshold policy are as follows. Because $w_t = w, \forall\, t \in \mathcal{T}$, we let $\omega = w\Delta t_{data}$. Since $\delta_t^*(s, p^{succ})$ is defined as in (5.21), we can establish the threshold policy if we can prove that $\psi_t(s, p^{succ}, a)$ is submodular on $\mathcal{S} \times \mathcal{A}$, $\forall\, t \in \mathcal{T}$ [28, pp. 104, 115]. The following results from Lemma 5.3 and 5.4 establish the submodularity of $\psi_t(s, p^{succ}, a)$. First, Lemma 5.3 shows that $v_t(s, p^{succ})$ has a nondecreasing difference in s if $h(s)$ is a convex and nondecreasing function.

Lemma 5.3 *If $h(s)$ is a convex and nondecreasing function in s, then*

$$\begin{aligned} v_t(s, p^{succ}) - v_t\big([s-\omega]^+, p^{succ}\big) \geq v_t\big([s-\sigma]^+, p^{succ}\big) - v_t\big([s-\sigma-\omega]^+, p^{succ}\big), \\ \forall\, s \in \mathcal{S}, p^{succ} \in \mathcal{P}, t \in \mathcal{T} \cup \{T+1\}. \end{aligned} \tag{5.23}$$

Proof: We prove it by induction. Since $h(s)$ is a nondecreasing convex function, we have

$$h(s) - h([s-\omega]^+) \geq h([s-\sigma]^+) - h([s-\sigma-\omega]^+), \quad \forall\, s \in \mathcal{S}. \tag{5.24}$$

Let $s \in \mathcal{S}$, $p^{succ} \in \mathcal{P}$ be given. For $t = T + 1$, we have

$$\begin{aligned} & v_{T+1}(s, p^{succ}) - v_{T+1}\big([s-\omega]^+, p^{succ}\big) = h(s) - h\big([s-\omega]^+\big) \\ & \geq h([s-\sigma]^+) - h([s-\sigma-\omega]^+) = v_{T+1}([s-\sigma]^+, p^{succ}) - v_{T+1}\big([s-\sigma-\omega]^+, p^{succ}\big), \end{aligned} \tag{5.25}$$

where the equalities are due to (5.19) and the inequality is due to (5.24).

Assume that for a given $t \in \mathcal{T}$, we have

$$\begin{aligned} v_{t+1}(s, p^{succ}) - v_{t+1}\big([s-\omega]^+, p^{succ}\big) \geq v_{t+1}\big([s-\sigma]^+, p^{succ}\big) - v_{t+1}\big([s-\sigma-\omega]^+, p^{succ}\big), \\ \forall\, s \in \mathcal{S}, p^{succ} \in \mathcal{P}. \end{aligned} \tag{5.26}$$

From (5.16), let actions a_1, a_2, a_3, and a_4 be defined such that

$$v_t(s, p^{succ}) = \min_{a \in \mathcal{A}} \{\psi_t(s, p^{succ}, a)\} = \psi_t(s, p^{succ}, a_1), \tag{5.27}$$

$$v_t\big([s-\omega]^+, p^{succ}\big) = \min_{a \in \mathcal{A}} \{\psi_t\big([s-\omega]^+, p^{succ}, a\big)\} = \psi_t\big([s-\omega]^+, p^{succ}, a_2\big), \tag{5.28}$$

$$v_t\big([s-\sigma]^+, p^{succ}\big) = \min_{a \in \mathcal{A}} \{\psi_t\big([s-\sigma]^+, p^{succ}, a\big)\} = \psi_t\big([s-\sigma]^+, p^{succ}, a_3\big), \tag{5.29}$$

$$\begin{aligned} v_t\big([s-\sigma-\omega]^+, p^{succ}\big) &= \min_{a \in \mathcal{A}} \{\psi_t\big([s-\sigma-\omega]^+, p^{succ}, a\big)\} \\ &= \psi_t\big([s-\sigma-\omega]^+, p^{succ}, a_4\big). \end{aligned} \tag{5.30}$$

We thus have

$$
\begin{aligned}
& v_t(s,p^{succ}) - v_t\big([s-\omega]^+,p^{succ}\big) - v_t\big([s-\sigma]^+,p^{succ}\big) + v_t\big([s-\sigma-\omega]^+,p^{succ}\big) \\
&= \psi_t(s,p^{succ},a_1) - \psi_t\big([s-\omega]^+,p^{succ},a_2\big) - \psi_t\big([s-\sigma]^+,p^{succ},a_3\big) + \psi_t\big([s-\sigma-\omega]^+,p^{succ},a_4\big) \\
&= \overbrace{\psi_t(s,p^{succ},a_1) - \psi_t\big([s-\sigma]^+,p^{succ},a_1\big)}^{A} + \overbrace{\psi_t\big([s-\sigma]^+,p^{succ},a_1\big) - \psi_t\big([s-\sigma]^+,p^{succ},a_3\big)}^{B} \\
&\quad \overbrace{-\,\psi_t\big([s-\omega]^+,p^{succ},a_2\big) + \psi_t\big([s-\omega]^+,p^{succ},a_4\big)}^{C} \\
&\quad -\Big(\overbrace{\psi_t\big([s-\omega]^+,p^{succ},a_4\big) - \psi_t\big([s-\sigma-\omega]^+,p^{succ},a_4\big)}^{D}\Big) = A+B+C-D.
\end{aligned} \tag{5.31}
$$

We have

$$
\begin{aligned}
A &= \sum_{p^{succ'}\in\mathcal{P}} p_t\big(p^{succ'}\,|\,p^{succ}\big)\Big[a_1 p^{succ}\big[v_{t+1}([s-\omega]^+,p^{succ'}) - v_{t+1}([s-\sigma-\omega]^+,p^{succ'})\big] \\
&\qquad\qquad + \big(1-a_1p^{succ}\big)\big[v_{t+1}(s,p^{succ'}) - v_{t+1}([s-\sigma]^+,p^{succ'})\big]\Big] \\
&\ge \sum_{p^{succ'}\in\mathcal{P}} p_t\big(p^{succ'}\,|\,p^{succ}\big)\big[v_{t+1}([s-\omega]^+,p^{succ'}) - v_{t+1}([s-\sigma-\omega]^+,p^{succ'})\big] \\
&\ge \sum_{p^{succ'}\in\mathcal{P}} p_t\big(p^{succ'}\,|\,p^{succ}\big)\Big[a_4 p^{succ}\big[v_{t+1}([s-2\omega]^+,p^{succ'}) - v_{t+1}([s-\sigma-2\omega]^+,p^{succ'})\big] \\
&\qquad + \big(1-a_4p^{succ}\big)\big[v_{t+1}([s-\omega]^+,p^{succ'}) - v_{t+1}([s-\sigma-\omega]^+,p^{succ'})\big]\Big] = D,
\end{aligned} \tag{5.32}
$$

where the two equalities are obtained by using (5.18) and the two inequalities are due to the induction hypothesis in (5.26). From (5.29) and (5.28), we have $B \ge 0$ and $C \ge 0$, respectively. Overall, from (5.31), we obtain

$$
v_t(s,p^{succ}) - v_t\big([s-\omega]^+,p^{succ}\big) - v_t\big([s-\sigma]^+,p^{succ}\big) + v_t\big([s-\sigma-\omega]^+,p^{succ}\big) \ge 0, \tag{5.33}
$$

which completes the proof. ■

Lemma 5.4 shows that $\psi_t(s, p^{succ}, a)$ is submodular if $v_t(s, p^{succ})$ has a nondecreasing difference in s.

Lemma 5.4 *If* $\forall\, \hat{s}, \check{s} \in \mathcal{S}$, $p^{succ} \in \mathcal{P}$, $t \in \mathcal{T}$ *with* $\hat{s} \geq \check{s}$, *where*

$$v_{t+1}(\hat{s}, p^{succ}) - v_{t+1}\big([\hat{s} - \omega]^+, p^{succ}\big) \geq v_{t+1}(\check{s}, p^{succ}) - v_{t+1}\big([\check{s} - \omega]^+, p^{succ}\big), \tag{5.34}$$

then $\psi_t(s, p^{succ}, a)$ *is submodular on* $\mathcal{S} \times \mathcal{A}$, $\forall\, t \in \mathcal{T}$.

Proof: Let $\hat{s}, \check{s} \in \mathcal{S}$, $\hat{a}, \check{a} \in \mathcal{A}$, $p^{succ} \in \mathcal{P}$, and $t \in \mathcal{T}$ be given, where $\hat{s} \geq \check{s}$ and $\hat{a} \geq \check{a}$. Then

$$\begin{aligned}
&\psi_t(\hat{s}, p^{succ}, \hat{a}) + \psi_t(\check{s}, p^{succ}, \check{a}) - \psi_t(\hat{s}, p^{succ}, \check{a}) - \psi_t(\check{s}, p^{succ}, \hat{a}) \\
= &- \sum_{p^{succ'} \in \mathcal{P}} p_t\big(p^{succ'} \,|\, p^{succ}\big) p^{succ}(\hat{a} - \check{a}) \Big[v_{t+1}(\hat{s}, p^{succ'}) - v_{t+1}\big([\hat{s} - \omega]^+, p^{succ'}\big) \\
&\qquad\qquad - v_{t+1}(\check{s}, p^{succ'}) + v_{t+1}\big([\check{s} - \omega]^+, p^{succ'}\big) \Big] \leq 0,
\end{aligned} \tag{5.35}$$

where the equality is obtained by using (5.18). The inequality at the end is due to the fact that $p_t\big(p^{succ'} \,|\, p^{succ}\big) \geq 0$, $p^{succ} \geq 0$, $\hat{a} \geq \check{a}$, and the given condition in Lemma 5.4. From Definition 5.1 and (5.22), the result follows. ■

Theorem 5.2 *If* $h(s)$ *is a convex and nondecreasing function in* s, *and the data rate* w_t *is fixed such that* $w_t = w, \forall\, t \in \mathcal{T}$, *then we have a threshold optimal policy* $\boldsymbol{\pi}^* = (\delta_t^*(s, p^{succ}), \forall\, s \in \mathcal{S}, p^{succ} \in \mathcal{P}, t \in \mathcal{T})$ *in* s *as follows:*

$$\delta_t^*(s, p^{succ}) = \begin{cases} 1, & \textit{if } s > s_t^*(p^{succ}), \\ 0, & \textit{otherwise}, \end{cases} \tag{5.36}$$

where $s_t^*(p^{succ})$ *is the threshold that depends on both* p^{succ} *and* t.

Proof: Let $\hat{s}, \check{s} \in \mathcal{S}$, $\omega \geq 0$, $p^{succ} \in \mathcal{P}$, and $t \in \mathcal{T}$ be given. Moreover, let $\check{s} = [\hat{s} - k\sigma]^+$, where $k > 0$. If the condition of the theorem is satisfied, by repetitively applying Lemma 5.3, we have

$$\begin{aligned} & v_t(\hat{s}, p^{succ}) - v_t\big([\hat{s} - \omega]^+, p^{succ}\big) \geq v_t\big([\hat{s} - \sigma]^+, p^{succ}\big) - v_t\big([\hat{s} - \sigma - \omega]^+, p^{succ}\big) \geq \cdots \\ & \geq v_t\big([\hat{s} - k\sigma]^+, p^{succ}\big) - v_t\big([\hat{s} - k\sigma - \omega]^+, p^{succ}\big) = v_t\big(\check{s}, p^{succ}\big) - v_t\big([\check{s} - \omega]^+, p^{succ}\big). \end{aligned} \tag{5.37}$$

From Lemma 5.4, $\psi_t(s, p^{succ}, a)$ is submodular on $\mathcal{S} \times \mathcal{A}$, $\forall\, t \in \mathcal{T}$. From [28, pp. 104, 115], $\delta_t^*(s, p^{succ})$ is a monotone nondecreasing function in s. Since $\delta_t^*(s, p^{succ}) \in \mathcal{A} = \{0, 1\}$, $\delta_t^*(s, p^{succ})$ is in the form of (5.36). ■

By modifying Algorithm 5.1, we are ready to propose the *monotone* DORA algorithm with a lower computational complexity in Algorithm 5.2 using monotone backward induction [28, pp. 111]. Let $\tilde{\mathcal{A}} \subseteq \mathcal{A}$ be the set of actions that we need to consider in the minimization in line 11 in Algorithm 5.2. When $\delta_t^*(s, p^{succ}) = 1$ and $flag = 0$ are satisfied (line 13), which means that the threshold $s_t^*(p^{succ})$ is reached, set $\tilde{\mathcal{A}}$ is reduced from $\{0, 1\}$ to $\{1\}$ and the threshold $s_t^*(p^{succ})$ is recorded (line 14). Then the minimization in line 11 is readily known, since set $\tilde{\mathcal{A}} = \{1\}$ is a singleton. The computational complexity is thus reduced. Moreover, memory can be saved, because we do not need to store the complete optimal policy $\boldsymbol{\pi}^* = (\delta_t^*(s, p^{succ}), \forall\, s \in \mathcal{S}, p^{succ} \in \mathcal{P}, t \in \mathcal{T})$. We just need to store the thresholds $(s_t^*(p^{succ}), \forall\, p^{succ} \in \mathcal{P}, t \in \mathcal{T})$, which completely characterize the optimal policy $\boldsymbol{\pi}^*$ as shown in (5.36).

Algorithm 5.2 *Monotone DORA Algorithm for single AP optimization (i.e., problem (5.15)) for the special case with convex penalty function $h(s)$ and fixed data rate w_t.*

Planning Phase:
Input the traffic parameters: ν, λ, ρ_{max}, $l_t\,(\forall\, t \in \mathcal{T})$;
Input the system parameters: $h(\cdot)$, S, R, $w_t\,(\forall\, t \in \mathcal{T})$, q, Δt, Δt_{data}, σ, $g(\cdot)$;
Set the boundary condition $v_{T+1}(s, p^{succ})$, $\forall\, s \in \mathcal{S}, \forall\, p^{succ} \in \mathcal{P}$ using (5.19);
$t := T$;
while $t \geq 1$
  for $p^{succ} \in \mathcal{P}$
    Set $s := 0$, $flag := 0$, and $\tilde{\mathcal{A}} := \{0, 1\}$;
    while $s \leq S$
      Calculate $\psi_t(s, p^{succ}, a)$, $\forall\, a \in \tilde{\mathcal{A}}$ using (5.20);
      $\delta_t^*(s, p^{succ}) := \arg\min_{a \in \tilde{\mathcal{A}}}\{\psi_t(s, p^{succ}, a)\}$;
      $v_t(s, p^{succ}) := \psi_t\big(s, p^{succ}, \delta_t^*(s, p^{succ})\big)$;
      if $\delta_t^*(s, p^{succ}) = 1$ and $flag = 0$
        Set $\tilde{\mathcal{A}} := \{1\}$, $s_t^*(p^{succ}) = s$, and $flag = 1$;
      end if
      $s := s + \sigma$;
    end while
  end for
  $t := t - 1$;
end while
Output the thresholds $(s_t^*(p^{succ}), \forall\, p^{succ} \in \mathcal{P}, t \in \mathcal{T})$ for use in the transmission phase;
Transmission Phase:
$t := 1$ and $s := S$;
while $t \leq T$
  Receive the information of p^{succ} from the AP;
  If $s > s_t^*(p^{succ})$
    Send a request to the AP;
    If ACK is received from the AP
      Transmit packets with total size $w_t \Delta t_{data}$;
      $s := [s - w_t \Delta t_{data}]^+$;
    end if
  end if
  $t := t + 1$;
end while

5.4.2 Joint AP Optimization with Deterministic Vehicular Traffic

In the previous subsection, we consider the optimization problem in a single AP. In this subsection, we extend the result to the case of *multiple* APs, where we assume that the traffic pattern (i.e., the exact number of vehicles in the coverage ranges of the APs in each time slot) can be estimated accurately. The traffic pattern can be estimated in various ways, such as by installing a traffic monitor at a place before the first AP to observe the actual traffic pattern when the vehicles pass by (e.g., using computer vision [102] and pattern recognition [103]). If the traffic flow reaches the steady state (as discussed in Section 5.2.1), the estimation of the number of vehicles n_t at time $t \in \mathbb{T}$ can be reasonably accurate. As a result, the values of $p_t^{succ} = g_j(n_t)$, $\forall\, t \in \mathbb{T}$ can be obtained accurately. As an example, we consider that the traffic model is as described in Section 5.2.1, and all the APs have the same transmission radii. After the traffic monitor has estimated the values of p_τ^{succ}, $\forall\, \tau \in \mathcal{T}_1$ for the first coverage range, it can set $p_{\zeta(j,\tau)}^{succ} := p_\tau^{succ}$ for the remaining coverage ranges $j \in \mathcal{J}\backslash\{1\}$.

The optimality equations relating the minimal expected total cost at different time $t \in \mathbb{T}$ for problem (5.13) are similar to that described in Section 5.4.1, but are simplified because we assume that p_t^{succ}, $\forall\, t \in \mathbb{T}$ are known. At time $t \in \mathcal{T}_j$, we have

$$v_t(s, p_t^{succ}) = \min_{a \in \mathcal{A}} \{\psi_t(s, p_t^{succ}, a)\}, \tag{5.38}$$

where

$$\begin{aligned}\psi_t(s, p_t^{succ}, a) &= c_t(s, p_t^{succ}, a) + \sum_{s' \in \mathcal{S}} p_t\big((s', p_{t+1}^{succ}) \,|\, (s, p_t^{succ}), a\big)\, v_{t+1}(s', p_{t+1}^{succ}) \\ &= aq_j + ap_t^{succ} v_{t+1}\big([s - w_t \Delta t_{data}]^+, p_{t+1}^{succ}\big) \\ &\quad +\big(1 - ap_t^{succ}\big) v_{t+1}(s, p_{t+1}^{succ}).\end{aligned} \tag{5.39}$$

The second line in (5.39) is obtained by using (5.11) and (5.12). After the vehicle has left the Jth coverage range at $t = \zeta(J, T_J + 1)$, the boundary condition is

$$v_{\zeta(J,T_J+1)}(s, p_{\zeta(J,T_J+1)}^{succ}) = \hat{c}_{\zeta(J,T_J+1)}(s, p_{\zeta(J,T_J+1)}^{succ}) = h(s). \tag{5.40}$$

The JDORA algorithm for joint AP optimization is given in Algorithm 5.3.In Algorithm 5.3, the vehicle first needs to obtain the values of p_t^{succ}, $\forall\, t \in \mathbb{T}$, from the traffic monitor. In the planning phase, for each $s \in \mathcal{S}$ and $t \in \mathbb{T}$, the optimal decision rule $\delta_t^*(s, p_t^{succ})$ is the action that minimizes the expected total cost (line 10), where the expected total cost $\psi_t(s, p_t^{succ}, a)$ for all possible actions is calculated (line 9) based on v_{t+1} obtained (line 11) in the previous iteration $t+1$. After the process is repeated for all $t \in \mathbb{T}$ (line 6) and $s \in \mathcal{S}$ (line 8), we obtain the optimal policy $\boldsymbol{\pi}^*$. In the transmission phase, the transmission decision in each time slot is made according to the optimal policy $\boldsymbol{\pi}^*$, and it follows the MAC protocol described in Section 5.2.3.

Theorem 5.3 *The policy* $\boldsymbol{\pi}^* = (\delta_t^*(s, p_t^{succ}), \forall\, s \in \mathcal{S}, t \in \mathbb{T})$ *obtained from Algorithm 5.3 is the optimal solution of problem (5.13) when* p_t^{succ}, $\forall\, t \in \mathbb{T}$ *are accurately known.*

Proof: The result follows directly from the principle of optimality [29, pp. 18]. ■

Algorithm 5.3 *JDORA Algorithm for joint AP optimization (i.e., problem (5.13)).*

Planning Phase:
Input the traffic parameters: ν, $p_t^{succ}\,(\forall\, t \in \mathbb{T})$;
Input the system parameters: $h(\cdot)$, S, $R_j\,(\forall\, j \in \mathcal{J})$, $w_t\,(\forall\, t \in \mathbb{T})$, $q_j(\forall\, j \in \mathcal{J})$, Δt, Δt_{data}, σ;
Set the boundary condition $v_{\zeta(J,T_J+1)}(s, p^{succ}_{\zeta(J,T_J+1)})$, $\forall\, s \in \mathcal{S}$ using (5.40);
$t := \zeta(J, T_J)$;
while $t \geq 1$
  $s := 0$;
  while $s \leq S$
    Calculate $\psi_t(s, p_t^{succ}, a)$, $\forall\, a \in \mathcal{A} = \{0, 1\}$ using (5.39);
    $\delta_t^*(s, p_t^{succ}) := \arg\min_{a \in \mathcal{A}}\{\psi_t(s, p_t^{succ}, a)\}$;
    $v_t(s, p_t^{succ}) := \psi_t\big(s, p_t^{succ}, \delta_t^*(s, p_t^{succ})\big)$;
    $s := s + \sigma$;
  end while
  $t := t - 1$;
end while
Output the optimal policy $\boldsymbol{\pi}^*$ for use in the transmission phase;
Transmission Phase:
$t := 1$ and $s := S$;
while $t \leq \zeta(J, T_J)$
  Set action $a := \delta_t^*(s, p_t^{succ})$ based on the policy $\boldsymbol{\pi}^*$;
  If action $a = 1$
    Send a request to the AP;
    If ACK is received from the AP
      Transmit packets with total size $w_t \Delta t_{data}$;
      $s := [s - w_t \Delta t_{data}]^+$;
    end if
  end if
  $t := t + 1$;
end while

5.5 Performance Evaluations

In this section, we first compare Algorithms 5.1 and 5.3 with three heuristic schemes using the traffic model described in Section 5.2.1 in both the single-AP and multiple-AP scenarios. In particular, we study the performance of Algorithm 5.3 under imperfect

Table 5.2: Simulation Parameters

Parameters	**Values**
Number of APs J	1, 5
AP's transmission radius R	100 m
Free-flow speed ν_f	110 km/hr
Vehicle jam density ρ_{max}	100 veh/km
Duration of a time slot Δt	0.02 sec
Duration for data transmission in a time slot Δt_{data}	0.018 sec
Channel bandwidth W	20 MHz
Transmit signal-to-noise ratio $\frac{P}{N_0 W}$	60 dB
Path loss exponent γ	3
Payment per time slot q	1
Contention window $cw \in [cw_{min}, cw_{max}]$	$[1, 8]$
MCBC parameter K (used in [86])	3
MCBC parameter $[p_1 p_2, p_3]$ (used in [86])	$[0.12, 0.77, 0.86]$
MCBC parameter F (used in [86])	15

estimations of the p_t^{succ} in the multiple-AP scenario. We then study the threshold policies obtained by Algorithm 5.2.

The three heuristic schemes that we consider are as follows. The first heuristic scheme is a *greedy* algorithm, in which each vehicle sends transmission requests at all the time slots if its file upload is not complete. That is, the greedy algorithm aims to maximize the total uploaded file size. The second heuristic scheme is the *exponential backoff* algorithm that is similar to the one used in the IEEE 802.11. We have slightly modified it for the system that we consider as follows. Each vehicle has a counter, which randomly and uniformly chooses an initial integer value *cnt* from the interval $[0, cw)$, where cw is the contention window size. The value of *cnt* is decreased by one after each time slot. When $cnt = 0$, the vehicle will send a request. If the vehicle has sent a request in a time slot, the size of $cw \in [cw_{min}, cw_{max}]$ will change according to the response from the AP: If an ACK is received from the AP, cw is set to cw_{min}. Otherwise, cw is doubled until it reaches cw_{max}. For the DORA, JDORA, greedy, and exponential schemes, we assume that the APs allow the vehicles to share the channel with an equal probability. Therefore, $p_t^{succ} = 1/n_t$. The

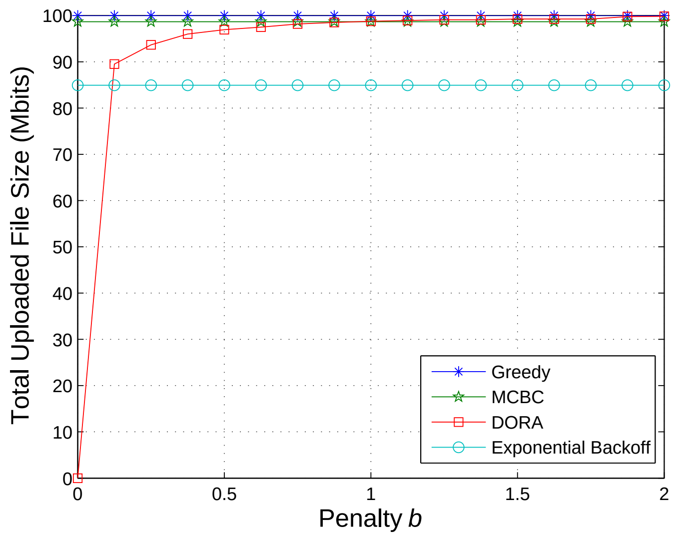

Figure 5.4: Total uploaded file size against the penalty parameter b for $S = 100$ Mbits and $\rho = 20$ veh/km with a single AP. As b increases, a larger file size is uploaded for the DORA scheme.

third heuristic scheme is the MAC protocol in the multi-carrier burst contention (MCBC) scheme [86]. Similar to the greedy scheme, a vehicle will send a request if it has data to send in each time slot. However, the vehicles need to undergo K rounds of contention in each time slot. First, in round r, a vehicle survives the contention with probability p_r. Each of these vehicles will choose a random integer in $\{1, \ldots, F\}$. Vehicles that have chosen the largest number can proceed to round $r+1$. The transmission is successful if there is only one vehicle left in round K. Otherwise, packet collision will occur.

For the evaluations of all the schemes, we use the convex self-incurred penalty function

$$h(s) = bs^2, \quad \forall\, s \in \mathcal{S}, \tag{5.41}$$

where $b \geq 0$ is a constant. The three heuristic schemes are evaluated using a similar transmission phase as in Algorithms 5.1 and 5.3, but with $\boldsymbol{\pi}^*$ in Algorithms 5.1 and 5.3 replaced by the corresponding policies. The simulation parameters are listed in Table 5.2.

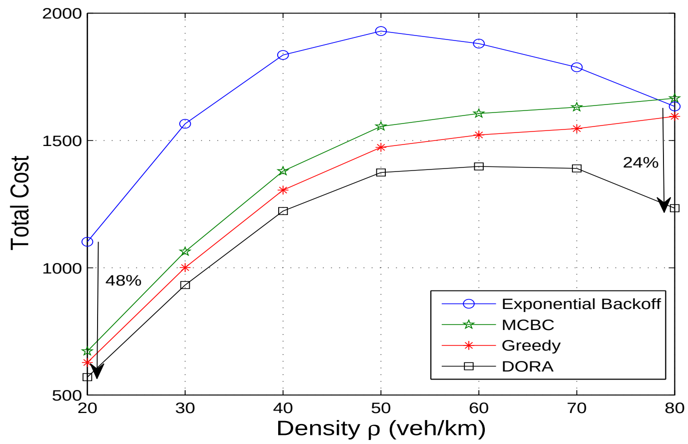

Figure 5.5: Total cost versus traffic density ρ for file size $S = 200$ Mbits with a single AP. The DORA scheme has the minimal total cost.

We first study the impact of penalty parameter b on the total uploaded file size for $S = 100$ Mbits and $\rho = 20$ veh/km in one AP. As shown in Fig. 5.4, by increasing b, a larger penalty is incurred on the size of the file not yet uploaded by using Algorithm 5.1. As a result, a larger file size is uploaded to reduce the penalty. Depending on the QoS requirements of different applications, different values of b should be chosen that tradeoff the total uploaded file size and total payment to the AP by a different degree. Taking safety application as an example, it may be more important to maximize the uploaded file size than to reduce the total payment to the APs, so a large value of b should be be chosen. Also, since the transmission policies of the greedy, MCBC, and exponential backoff schemes do not consider the self-incurred penalty in (5.41), their total uploaded file size are independent of b.

Next, we plot the total cost against the traffic density ρ for $S = 200$ Mbits with $b = 0.1$ for the case of one AP in Fig. 5.5. It is clear that the DORA scheme in Algorithm 5.1 achieves the minimal total cost as stated in Theorem 5.1, with 48% and 24% cost reduction as compared with the exponential backoff scheme at low and high ρ, respectively. To measure the cost effectiveness of the file uploading for the four schemes, we propose a

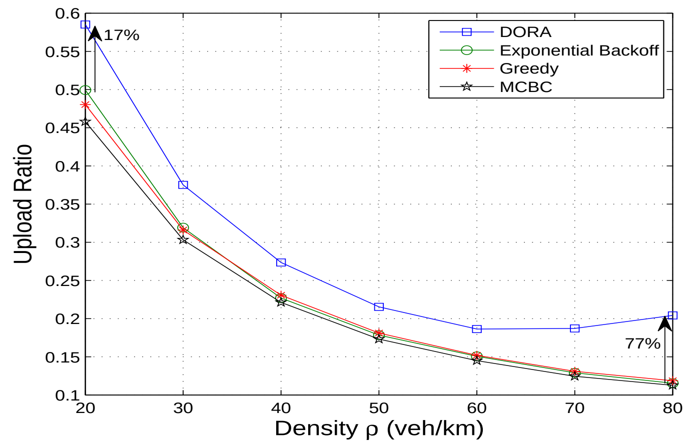

Figure 5.6: Upload ratio (i.e., total uploaded file size / total payment to the APs) versus traffic density ρ for file size $S = 200$ Mbits with a single AP. The DORA scheme achieves the highest upload ratio.

metric called the *upload ratio*, which is defined as the total uploaded file size divided by the total payment to the APs. In other words, it represents the size of the file uploaded per unit payment. As shown in Fig. 5.6, since the DORA algorithm takes into account the varying channel contention level and data rate in determining the transmission policy, it is cost effective and achieves the highest upload ratio. In particular, the performance gains in upload ratio over the exponential backoff scheme are 17% and 77% at low and high ρ, respectively.

Furthermore, we consider the case with five APs, where we assume that all of them have the same transmission radii R and price q. For the JDORA scheme in Algorithm 5.3, we consider that the estimated number of vehicles $\tilde{n}_t$ at time $t \in \mathbb{T}$ is obtained by rounding off a normally distributed random variable with a mean n_t and a variance θ to the nearest non-negative integer. Thus, the lower the variance θ, the higher is the precision of the estimation. The value of p_t^{succ} is obtained by setting $p_t^{succ} = g_j(\tilde{n}_t)$, $\forall\, t \in \mathcal{T}_j, j \in \mathcal{J}$. We plot the total cost and upload ratio in Figs. 5.7 and 5.8 for $S = 500$ Mbits with $b = 0.01$, respectively. In Fig. 5.7, we can see that the JDORA scheme with perfect estimation (i.e.,

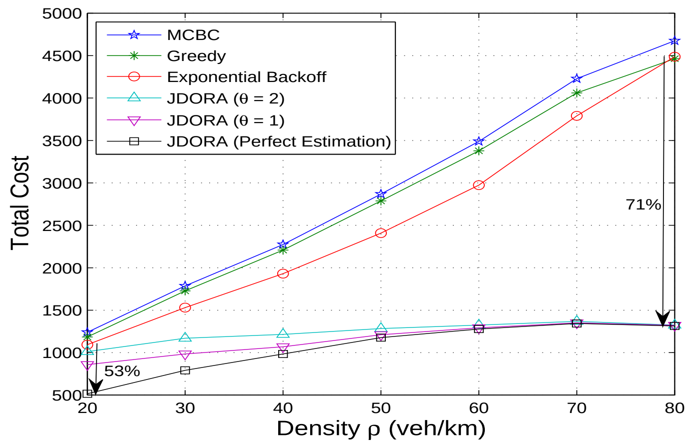

Figure 5.7: Total cost versus traffic density ρ for file size $S = 500$ Mbits with five APs. The JDORA scheme with perfect estimation of p_t^{succ} has the minimal total cost. Moreover, a higher total cost is required when the precision of the estimation reduces (i.e., when the variance of the estimation θ increases).

$\theta = 0$) of p_t^{succ}, $\forall t \in \mathbb{T}$ achieves the minimal total cost as stated in Theorem 5.3, where it achieves 53% and 71% cost reduction as compared with the exponential backoff scheme at low and high traffic density ρ, respectively. In Fig. 5.8, we can see that the JDORA scheme with perfect estimation achieves the highest upload ratio. In particular, it achieves an upload ratio 130% and 207% better than the exponential backoff scheme at low and high traffic density ρ, respectively. As shown in Figs. 5.7 and 5.8, the total cost is increased and the upload ratio is reduced, respectively, when the estimation precision decreases. However, this result based on equal share of bandwidth that $p_t^{succ} = 1/\tilde{n}_t$, $\forall t \in \mathbb{T}$ is less sensitive to the estimation error when the traffic density ρ is high. It suggests that the JDORA algorithm is suitable especially for VANETs with high traffic densities.

Finally, we study the threshold policy in a single AP obtained by Algorithm 5.2 when the penalty function $h(s)$ is convex and data rate w_t is fixed. We consider that $S = 100$ Mbits, $\nu = 100$ km/hr, $w_t = 54$ Mbps, $\forall t \in \mathcal{T}$, and $h(s)$ is defined as in (5.41). From Theorem 5.2, we know that the optimal policy has a threshold structure. In Fig. 5.9, we

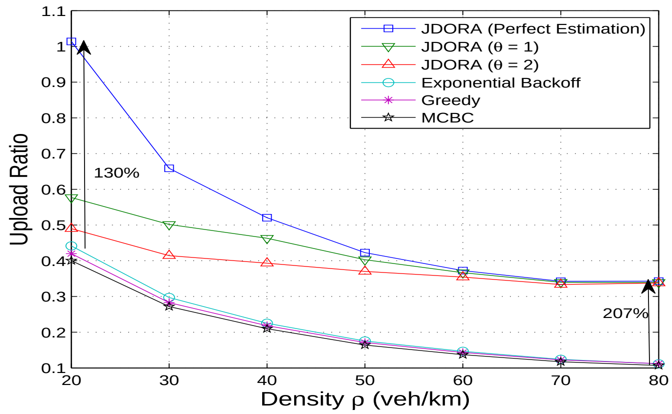

Figure 5.8: Upload ratio versus traffic density ρ for file size $S = 500$ Mbits with five APs. The JDORA scheme with perfect estimation of p_t^{succ} achieves the highest upload ratio as compared with three other heuristic schemes. Moreover, a lower upload ratio is achieved when the precision of the estimation reduces (i.e., when the variance of the estimation θ increases).

plot the thresholds $s_t^*(p^{succ})$ of the optimal policy against the decision epoch t for different values of p^{succ}. With the use of the convex penalty function, we can see that the threshold increases with t. In Fig. 5.9(a), for $b = 0.1$, we can observe that the threshold increases when p^{succ} decreases. It is because a small penalty parameter is chosen, which places a higher priority on the total payment than on the uploaded file size. When p^{succ} is small, the chance of successful transmission is low, so the vehicle chooses a higher threshold and transmits less aggressively to reduce the amount of payment. In Fig. 5.9(b), we choose a larger penalty parameter $b = 10$ such that a higher priority is placed on the uploaded file size than on the total payment. We can observe that the threshold decreases when p^{succ} decreases. It is because when p^{succ} is small, the vehicle needs to transmit more aggressively (i.e., with a lower threshold) to prevent a large penalty. Moreover, we can see that the thresholds presented in Fig. 5.9(b) is lower than that in Fig. 5.9(a) due to the higher incentive to transmit when the penalty is large.

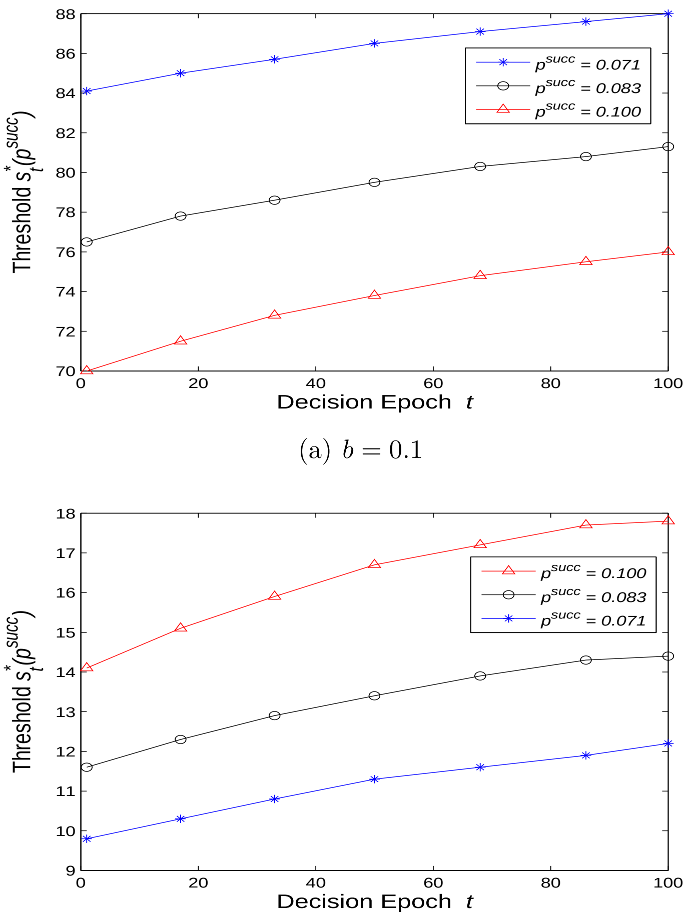

Figure 5.9: The thresholds $s_t^*(p^{succ})$ of the optimal policy against the decision epoch t for different penalty parameters b.

5.6 Summary

In this chapter, we studied the V2R uplink transmission from a vehicle to the APs in a dynamic drive-thru scenario, where both the channel contention level and data rate vary over time. Depending on the applications' QoS requirements, the vehicle can achieve different levels of tradeoff between the total uploaded file size and the total payment to the APs by tuning the self-incurred penalty. For a single AP with random vehicular traffic, we proposed a DORA algorithm based on DP to obtain the optimal transmission policy for the vehicle in a coverage range. We prove that if the self-incurred penalty function $h(s)$ is convex and the data rate w_t is non-adaptive and fixed, then the optimal transmission policy has a threshold structure. A monotone DORA algorithm with a lower computational complexity was proposed for this special case. Next, for multiple APs with known vehicular patterns, we considered the transmission policy in multiple coverage ranges jointly and proposed an optimal JDORA algorithm. Simulation results showed that our schemes achieve the minimal total cost and the highest upload ratio as compared with three other heuristic schemes.

Chapter 6

Conclusions and Future Work

In this chapter, we summarize the results and highlight the contributions of this thesis. We also suggest several topics for future work.

6.1 Research Contributions

In this thesis, we have proposed several optimal or near-optimal random access algorithms for wireless networks in Chapters 2, 3, and 4 using the NUM framework. We have also considered a uplink transmission problem in VANETs with time-varying channel contention level and data rate in Chapter 5.

- In Chapter 2, we considered the random access problem in CR networks using the SINR model. For cooperative users in a multi-channel model, a three-phase distributed algorithm was proposed to obtain a near-optimal solution for the formulated non-convex NUM problem. It converges readily to a close-to-optimal value even when the set of data channels changes due to dynamic spectrum leasing. For rational users in a single-channel model, we used the solution concepts of core and the Shapley value in coalitional game theory to characterize the stability and fair allocation of the payoff among the users, respectively. The performance gain in aggregate throughput of the proposed algorithm based on the SINR model over other schemes based on the protocol model are validated by simulations. The Shapley value and the core were illustrated with a numerical example.

- In Chapter 3, we studied the single-channel random access problem in WLANs, where the users have concave, step, and quasi-concave utility functions. Potentially, a selfish user may strategically declare its utility function or AC to unfairly achieve a larger share of bandwidth, which can drastically degrade the network performance and inhibit adequate service distinction among different ACs. We applied the VCG mechanism in our random access protocol to motivate the users to declare the ACs of their applications truthfully. In order to implement the VCG mechanism, we proposed a low-complexity enumeration algorithm that can obtain the global optimal solution for the formulated non-convex problem. Simulation results show that a truthful mechanism can prevent selfish users from gaining an unfair share of the network bandwidth, such that both the overall network performance in terms of aggregate utility and service differentiation in terms of necessary throughput in each AC can be supported.

- In Chapter 4, we extended the work of random access in WLANs, where we considered that the users have concave or sigmoidal utility functions. Different from Chapter 3, we did not restrict a concave utility function to remain concave after a logarithmic change of variables. By applying the dual decomposition method, we proposed a subgradient algorithm to solve the formulated non-convex NUM problem. Closed-form solutions for the dual subproblems involving sigmoidal functions were obtained. If a sufficient condition on link capacities is satisfied, it is guaranteed that the proposed algorithm can obtain the global optimal solution. Otherwise, lower and upper bounds of the optimal value of the objective function were obtained. Simulations were performed to verify our analytical results.

- In Chapter 5, we studied V2R uplink transmission from a vehicle to the APs in a dynamic drive-thru scenario with time-varying channel contention level and transmission data rate. First, for a single AP with random vehicular traffic, we proposed a DORA algorithm based on DP to obtain the optimal transmission policy. We proved that if the self-incurred penalty function is convex and the data rate is fixed, then we obtain an optimal transmission policy with threshold structure. We proposed a low complexity monotone DORA algorithm for this special case. Then, for multiple APs with known vehicular patterns, we considered joint AP optimization and proposed a JDORA algorithm to obtain the optimal transmission policy. Simulation results showed that our proposed schemes achieve the minimal total cost and the highest upload ratio as compared with three other heuristic schemes.

6.2 Suggestions for Future Work

In the following, we consider several interesting possibilities for extension of the current work.

1. **CSMA-based MAC Protocol**. In Chapters 2, 3, and 4, we have used a slotted Aloha type of MAC protocol, which each user accesses the channel with a certain transmission probability. It is interesting to consider the CSMA-based MAC protocol, where a user attempts to transmit only when the channel is sensed idle [104–106].

2. **Rational User Cooperation in Multiple Channels**. In Chapter 2, we have analyzed the rational user cooperation in a single channel under the SINR model. It is interesting to extend the model to a multi-channel setting and analyze the interactions of the rational users. The idea of coalition formation game in coalitional game theory is an interesting direction for future work.

3. **Multi-hop Communication**. In Chapters 2, 3, and 4, we have assumed that the transmission between the transmitter and receiver of each user is only single-hop. It

is possible to consider the multi-hop setting by introducing binary routing variables and flow conservation constraints as in [57].

4. **Analysis of the Random Access Game**. In Chapter 3, we have analyzed the strategic declarations of the amplitude parameters $\hat{K}$ by the rational players in the non-cooperative random access game. It is possible to extend the analysis to consider the strategic declarations of the other utility parameters, i.e., $\hat{\alpha}$ and $\hat{p}^{\text{critical}}$.

5. **Joint AP Optimization Without Traffic Pattern Estimation**. In Chapter 5, we have considered the single-AP scenario with random vehicular traffic, and the multiple-AP scenario with deterministic vehicular traffic due to traffic pattern estimation. One possible direction for extension is to characterize the probability related to the number of vehicles in each AP in the multiple-AP scenario, and consider joint AP optimization with random vehicular traffic (i.e., without the need of traffic pattern estimation).

Bibliography

[1] 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*, Roppongi, Japan, May 2004.

[2] A. Leon-Garcia and I. Widjaja, *Communication Networks: Fundamental Concepts and Key Architectures*, 2nd ed. New York, NY: McGraw-Hill, 2004.

[3] FCC Spectrum Policy Task Force. (2002, Nov.) Report of the spectrum efficiency working group.

[4] S. Haykin, "Cognitive radio: Brain-empowered wireless communications," *IEEE J. Select. Areas Commun.*, vol. 23, no. 2, pp. 201–220, Feb. 2005.

[5] Q. Zhao and B. M. Sadler, "A survey of dynamic spectrum access," *IEEE Signal Processing Magazine*, vol. 24, no. 3, pp. 79–89, May 2007.

[6] J. M. Peha, "Approaches to spectrum sharing," *IEEE Communications Magazine*, vol. 43, no. 2, pp. 10–12, Feb. 2005.

[7] D. Niyato and E. Hossain, "Competitive pricing for spectrum sharing in cognitive radio networks: Dynamic game, inefficiency of Nash equilibrium, and collusion," *IEEE J. Select. Areas Commun.*, vol. 26, no. 1, pp. 192–202, Jan. 2008.

[8] L. Duan, J. Huang, and B. Shou, "Cognitive mobile virtual network operator: Investment and pricing with supply uncertainty," in *Proc. of IEEE INFOCOM*, San Diego, CA, Mar. 2010.

[9] G. Vazquez-Vilar, C. Mosquera, and S. K. Jayaweera, "Primary user enters the game: Performance of dynamic spectrum leasing in cognitive radio networks," *IEEE Trans. on Wireless Communications*, vol. 9, no. 12, pp. 3625–3629, Dec. 2010.

[10] W. Wang, B. Kasiri, J. Cai, and A. S. Alfa, "Distributed cooperative multi-channel spectrum sensing based on dynamic coalitional game," in *Proc. of IEEE Globecom*, Miami, FL, Dec. 2010.

[11] T. Yücek and H. Arslan, "A survey of spectrum sensing algorithms for cognitive radio applications," *IEEE Communications Surveys & Tutorials*, vol. 11, no. 1, pp. 116–130, first quarter 2009.

[12] S. Shenker, "Fundamental design issues for the future Internet," *IEEE J. Select. Areas Commun.*, vol. 13, no. 7, pp. 1176–1188, Sept. 1995.

[13] H. Hartenstein and K. P. Laberteaux, "A tutorial survey on vehicular ad hoc networks," *IEEE Communications Magazine*, vol. 46, no. 6, pp. 164–171, June 2008.

[14] Y. Toor, P. Mühlethaler, A. Laouiti, and A. de La Fortelle, "Vehicle ad hoc networks: Applications and related technical issues," *IEEE Communications Surveys & Tutorials*, vol. 10, no. 3, pp. 74–88, third quarter 2008.

[15] E. Hossain, G. Chow, V. C. M. Leung, R. D. McLeod, J. Misic, V. W. S. Wong, and O. Yang, "Vehicular telematics over heterogeneous wireless networks: A survey," *Elsevier Computer Communications*, vol. 33, no. 7, pp. 775–793, May 2010.

[16] *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification*, IEEE Std. 802.11p, 2010.

[17] R. A. Uzcategui and G. Acosta-Marum, "WAVE: A tutorial," *IEEE Communications Magazine*, vol. 47, no. 5, pp. 126–133, May 2009.

[18] Y. L. Morgan, "Notes on DSRC & WAVE standards suite: Its architecture, design, and characteristics," *IEEE Communications Surveys & Tutorials*, vol. 12, no. 4, pp. 504–518, fourth quarter 2010.

[19] D. P. Bertsekas, *Nonlinear Programming*, 2nd ed. Athena Scientific, 1999.

[20] M. Minoux, *Mathematical Programming: Theory and Algorithms*. New York, NY: Wiley, 1986.

[21] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, UK: Cambridge University Press, 2004.

[22] D. P. Bertsekas, A. Nedic, and A. E. Ozdaglar, *Convex Analysis and Optimization*. Belmont, MA: Athena Scientific, 2003.

[23] M. S. Bazarra, H. D. Sherali, and C. M. Shetty, *Nonlinear Programming: Theory and Algorithms*, 3rd ed. Hoboken, NJ: Wiley, 2006.

[24] M. J. Osborne and A. Rubinstein, *A Course in Game Theory*. MIT Press, 1994.

[25] Y. Shoham and K. Leyton-Brown, *Multi Agent Systems: Algorithmic, Game-Theoretic, and Logical Foundations*. Cambridge University Press, 2008.

[26] D. Fudenberg and J. Tirole, *Game Theory*. Cambridge, Massachusetts: MIT Press, 1991.

[27] A. Mas-Colell, M. D. Whinston, and J. R. Green, *Microeconomic Theory.* Oxford University Press, 1995.

[28] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming.* New York, NY: John Wiley and Sons, 2005.

[29] D. P. Bertsekas, *Dynamic Programming and Optimal Control: Volume 1*, 3rd ed. Athena Scientific, 2005.

[30] W. Saad, Z. Han, M. Debbah, A. Hjorungnes, and T. Basar, "Coalitional game theory for communication networks," *IEEE Signal Processing Magazine*, vol. 26, no. 5, pp. 77–97, Sept. 2009.

[31] K. R. Apt and A. Witzel, "A generic approach to coalition formation," *International Game Theory Review*, vol. 11, no. 3, pp. 347–367, 2009.

[32] W. Saad, Z. Han, T. Basar, M. Debbah, and A. Hjorungnes, "Hedonic coalition formation for distributed task allocation among wireless agents," *IEEE Trans. on Mobile Computing*, vol. 10, no. 9, pp. 1327–1344, Sept. 2011.

[33] A. Bogomolnaia and M. O. Jackson, "The stability of hedonic coalition structures," *Games and Economic Behavior*, vol. 38, pp. 201–230, Feb. 2002.

[34] D. C. Parkes, "Iterative combinatorial auctions: Achieving economic and computational efficiency," Ph.D. dissertation, University of Pennsylvania, May 2001.

[35] M. H. Cheung, V. W. S. Wong, and R. Schober, "SINR-based random access for cognitive radio: Distributed algorithm and coalitional game," *IEEE Trans. on Wireless Communications*, vol. 10, no. 11, pp. 3887–3897, Nov. 2011.

[36] M. H. Cheung, A. H. Mohsenian-Rad, V. W. S. Wong, and R. Schober, "Random access protocols for WLANs based on mechanism design," in *Proc. of IEEE ICC*, Dresden, Germany, June 2009.

[37] ——, "Random access for elastic and inelastic traffic in WLANs," *IEEE Trans. on Wireless Communications*, vol. 9, no. 6, pp. 1861–1866, June 2010.

[38] M. H. Cheung, F. Hou, V. W. S. Wong, and J. Huang, "Dynamic optimal random access for vehicle-to-roadside communications," in *Proc. of IEEE ICC*, Kyoto, Japan, June 2011.

[39] ——, "DORA: Dynamic optimal random access for vehicle-to-roadside communications," *accepted for publication in IEEE J. on Selected Areas in Commun.*, 2012.

[40] H. Su and X. Zhang, "Cross-layer based opportunisitc MAC protocols for QoS provisionings over cognitive radio wireless networks," *IEEE J. Select. Areas Commun.*, vol. 26, no. 1, pp. 118–129, Jan. 2008.

[41] M. Timmers, S. Pollin, A. Dejonghe, L. Van der Perre, and F. Catthoor, "A distributed multichannel MAC protocol for multihop cognitive radio networks," *IEEE Trans. on Vehicular Technology*, vol. 59, no. 1, pp. 446–459, Jan. 2010.

[42] P. Gupta and P. Kumar, "The capacity of wireless networks," *IEEE Trans. Inform. Theory*, vol. 46, no. 2, pp. 388–404, Mar. 2000.

[43] L. Fu, S. C. Liew, and J. Huang, "Effective carrier sensing in CSMA networks under cumulative interference," in *Proc. of IEEE INFOCOM*, San Diego, CA, Mar. 2010.

[44] A. H. Mohsenian-Rad, V. W. S. Wong, and R. Schober, "Optimal SINR-based random access," in *Proc. of IEEE INFOCOM*, San Diego, CA, Mar. 2010.

[45] O. Goussevskaia, Y. A. Oswald, and R. Wattenhofer, "Complexity in geometric SINR," in *Proc. of ACM MobiHoc*, Montreal, Quebec, Canada, Sept. 2007.

[46] D. Qian, D. Zheng, J. Zhang, and N. Shroff, "CSMA-based distributed scheduling in multi-hop MIMO networks under SINR model," in *Proc. of IEEE INFOCOM*, San Diego, CA, Mar. 2010.

[47] A. H. Mohsenian-Rad and V. W. S. Wong, "Distributed multi-interface multi-channel random access using convex optimization," *IEEE Trans. on Mobile Computing*, vol. 10, no. 1, pp. 67–80, Jan. 2011.

[48] M. Felegyhazi, M. Cagalj, and J.-P. Hubaux, "Efficient MAC in cognitive radio systems: A game-theoretic approach," *IEEE Trans. on Wireless Communications*, vol. 8, no. 4, pp. 1984–1995, Apr. 2009.

[49] C. Zou and C. Chigan, "A game theoretic DSA-driven MAC framework for cognitive radio networks," in *Proc. of IEEE ICC*, Beijing, China, May 2008.

[50] S. Mathur, L. Sankar, and N. B. Mandayam, "Coalitions in cooperative wireless networks," *IEEE J. on Selected Areas in Commun.*, vol. 26, no. 7, pp. 1104–1115, Sept. 2008.

[51] C. Cordeiro and K. Challapali, "C-MAC: A cognitive MAC protocol for multi-channel wireless networks," in *Proc. of IEEE DySPAN*, Dublin, Ireland, Oct. 2007.

[52] J. Jia, Q. Zhang, and X. Shen, "HC-MAC: A hardware-constrained cognitive MAC for efficient spectrum management," *IEEE J. Select. Areas Commun.*, vol. 26, no. 1, pp. 106–117, Jan. 2008.

[53] J. Mo and J. Walrand, "Fair end-to-end window-based congestion control," *IEEE/ACM Trans. on Networking*, vol. 8, no. 5, pp. 556–567, Oct. 2000.

[54] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*. Athena Scientific, 1997.

[55] S. Pleisch, M. Balakrishnan, K. Birman, and R. van Renesse, "MISTRAL: Efficient flooding in mobile ad-hoc networks," in *Proc. of ACM MobiHoc*, Florence, Italy, May 2006.

[56] W. Lou and J. Wu, "Toward broadcast reliability in mobile ad hoc networks with double coverage," *IEEE Trans. on Mobile Computing*, vol. 6, no. 2, pp. 148–163, Feb. 2007.

[57] A. H. Mohsenian-Rad and V. 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, Dec. 2007.

[58] D. Niyato and E. Hossain, "Competitive pricing in heterogeneous wireless access networks: Issues and approaches," *IEEE Network*, vol. 22, no. 6, pp. 4–11, Nov.-Dec. 2008.

[59] F. Fu and M. van der Schaar, "Noncollaborative resource management for wireless multimedia applications using mechanism design," *IEEE Trans. on Multimedia*, vol. 9, no. 4, pp. 851–868, June 2007.

[60] F. Wu, S. Zhong, and J. Liu, "An optimal, strategy-proof scheme for multi-path traffic assignment in non-cooperative networks," *IEEE Trans. on Wireless Commun.*, vol. 9, no. 3, pp. 1012–1021, Mar. 2010.

[61] N. Nisan and A. Ronen, "Algorithmic mechanism design," *Games and Economic Behavior*, vol. 35, pp. 166–196, 2001.

[62] J. Lee, M. Chiang, and A. R. Calderbank, "Utility-optimal random-access control," *IEEE Trans. on Wireless Communications*, vol. 6, no. 7, pp. 2741–2751, Jul. 2007.

[63] 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*, vol. 8, no. 2, pp. 898–911, Feb. 2009.

[64] F. Kelly, "Charging and rate control for elastic traffic," *European Trans. on Telecommunication*, vol. 8, pp. 33–37, 1997.

[65] 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.

[66] H. Inaltekin and S. B. Wicker, "The analysis of Nash equilibria of the one-shot random-access game for wireless networks and the behavior of selfish nodes," *IEEE/ACM Trans. on Networking*, vol. 16, no. 5, pp. 1094–1107, Oct. 2008.

[67] L. Chen, S. H. Low, and J. C. Doyle, "Random access game and medium access control design," *IEEE/ACM Trans. on Networking*, vol. 18, no. 4, pp. 1303–1316, Aug. 2010.

[68] Z. Han, Z. Ji, and K. J. R. Liu, "A cartel maintenance framework to enforce cooperation in wireless networks with selfish users," *IEEE Trans. on Wireless Communications*, vol. 7, no. 5, pp. 1889–1899, May 2008.

[69] W. Wang, X.-Y. Li, and Y. Wang, "Truthful multicast routing in selfish wireless networks," in *Proc. of ACM MobiCom*, Philadelphia, PA, Sept. 2004.

[70] P. Nuggehalli, M. Sarkar, and R. Rao, "QoS and selfish users: A MAC layer perspective," in *Proc. IEEE GLOBECOM*, Washington, DC, Nov. 2007.

[71] J. Bae, E. Beigman, R. Berry, M. L. Honig, and R. Vohra, "A dynamic auction for spectrum sharing," in *Proc. of Allerton Conference*, Monticello, IL, Sept. 2007.

[72] J. Huang, Z. Han, M. Chiang, and H. Poor, "Auction-based resource allocation for cooperative communications," *IEEE J. Select. Areas Commun.*, vol. 26, no. 7, pp. 1226–1237, Sept. 2008.

[73] C. Ko and H. Wei, "On-demand resource sharing mechanism design in two-tier OFDMA femtocell networks," *IEEE Trans. Veh. Technol.*, vol. 60, no. 3, pp. 1059–1071, Mar. 2011.

[74] J. Lee, A. Tang, J. Huang, M. Chiang, and A. R. Calderbank, "Reverse-engineering MAC: A non-cooperative game model," *IEEE J. Select. Areas Commun.*, vol. 25, no. 6, pp. 1135–1147, Aug. 2007.

[75] A. H. Mohsenian-Rad, J. Huang, M. Chiang, and V. W. S. Wong, "Utility-optimal random access without message passing," *IEEE Trans. on Wireless Communications*, vol. 8, no. 3, pp. 1073–1079, Mar. 2009.

[76] N. Ben Salem, L. Buttyan, J.-P. Hubaux, and M. Jakobsson, "Node cooperation in hybrid ad hoc networks," *IEEE Trans. on Mobile Computing*, vol. 5, no. 4, pp. 365–376, Apr. 2006.

[77] S. G. Rao, *Discrete Mathematical Structures*, 2nd ed. New Age International, 2009.

[78] "IEEE 802.11," http://standards.ieee.org/getieee802/download/802.11-2007.pdf, 2007.

[79] J. Lee, R. R. Mazumdar, and N. B. Shroff, "Nonconvexity issues for Internet rate control with multiclass services: Stability and optimality," in *Proc. of IEEE INFOCOM*, Hong Kong, China, Mar. 2004.

[80] P. Hande, S. Zhang, and M. Chiang, "Distributed rate allocation for inelastic flows," *IEEE/ACM Trans. on Networking*, vol. 15, no. 6, pp. 1240–1253, Dec. 2007.

[81] J. Lee, R. R. Mazumdar, and N. B. Shroff, "Downlink power allocation for multi-class wireless systems," *IEEE/ACM Trans. on Networking*, vol. 13, no. 4, pp. 854–867, 2005.

[82] M. Xiao, N. B. Shroff, and E. K. P. Chong, "A utility-based power-control scheme in wireless cellular systems," *IEEE/ACM Trans. on Networking*, vol. 11, no. 2, pp. 210–221, Apr. 2003.

[83] W.-H. Kuo and W. Liao, "Utility-based radio resource allocation for QoS traffic in wireless networks," *IEEE Trans. on Wireless Commun.*, vol. 7, no. 7, pp. 2714–2722, July 2008.

[84] W. G. Bardsley and R. M. W. Wood, "Critical points and sigmoidicity of positive rational functions," *The American Mathematical Monthly*, vol. 92, no. 1, pp. 37–48, 1985.

[85] J. Ott and D. Kutscher, "Drive-thru Internet: IEEE 802.11b for automobile users," in *Proc. of IEEE INFOCOM*, Hong Kong, China, Mar. 2004.

[86] B. Roman, I. Wassell, and I. Chatzigeorgiou, "Scalable cross-layer wireless access control using multi-carrier burst contention," *IEEE J. Select. Areas Commun.*, vol. 29, no. 1, pp. 113–128, Jan. 2011.

[87] D. Hadaller, S. Keshav, and T. Brecht, "MV-MAX: Improving wireless infrastructure access for multi-vehicular communication," in *Proc. of ACM SIGCOMM Workshop*, Pisa, Italy, Sept. 2006.

[88] C. Chang, R. Cheng, H. Shih, and Y. Chen, "Maximum freedom last scheduling algorithm for downlinks of DSRC networks," *IEEE Trans. on Intelligent Transportation Systems*, vol. 8, no. 2, pp. 223–232, June 2007.

[89] Y. Zhang, J. Zhao, and G. Cao, "On scheduling vehicle-roadside data access," in *Proc. of ACM International Workshop on VANETs*, Montreal, QC, Canada, Sept. 2007.

[90] J. J. Alcaraz, J. Vales-Alonso, and J. Garcia-Haro, "Link-layer scheduling in vehicle to infrastructure networks: An optimal control approach," *IEEE J. Select. Areas Commun.*, vol. 29, no. 1, pp. 103–112, Jan. 2011.

[91] K. Yang, S. Ou, H.-H. Chen, and J. He, "A multihop peer-communication protocol with fairness guarantee for IEEE 802.16-based vehicular networks," *IEEE Trans. on Vehicular Technology*, vol. 56, no. 6, pp. 3358–3370, Nov. 2007.

[92] B. Shrestha, D. Niyato, Z. Han, and E. Hossain, "Wireless access in vehicular environments using BitTorrent and bargaining," in *Proc. of IEEE GLOBECOM*, New Orleans, LA, Nov. 2008.

[93] B. Jarupan and E. Ekici, "Location- and delay-aware cross-layer communication in V2I multihop vehicular networks," *IEEE Communications Magazine*, vol. 47, no. 11, pp. 112–118, Nov. 2009.

[94] D. Niyato and E. Hossain, "A unified framework for optimal wireless access for data streaming over vehicle-to-roadside communications," *IEEE Trans. Veh. Technol.*, vol. 59, no. 6, pp. 3025–3035, Jul. 2010.

[95] B. Sikdar, "Characterization and abatement of the reassociation overhead in vehicle to roadside networks," *IEEE Trans. on Communications*, vol. 58, no. 11, pp. 3296–3304, Nov. 2010.

[96] W. L. Tan, W. C. Lau, O. Yue, and T. H. Hui, "Analytical models and performance evaluation of drive-thru Internet systems," *IEEE J. Select. Areas Commun.*, vol. 29, no. 1, pp. 207–222, Jan. 2011.

[97] J. Choi and H. Lee, "Supporting handover in an IEEE 802.11p-based wireless access system," in *Proc. of ACM International Workshop on VANETs*, Chicago, IL, Sept. 2010.

[98] R. J. Cowan, "Useful headway models," *Transportation Research*, vol. 9, no. 6, pp. 371–375, Dec. 1975.

[99] J. D. Fricker and R. K. Whitford, *Fundamentals of Transportation Engineering: A Multimodal Systems Approach*. Prentice Hall, 2004.

[100] M. H. Ngo and V. Krishnamurthy, "Optimality of threshold policies for transmission scheduling in correlated fading channels," *IEEE Trans. on Communications*, vol. 57, no. 8, pp. 2474–2483, Aug. 2009.

[101] S. Ross, *Introduction to Stochastic Dynamic Programming*. San Diego, CA: Academic Press, 1983.

[102] T. N. Schoepflin and D. J. Dailey, "Dynamic camera calibration of roadside traffic management cameras for vehicle speed estimation," *IEEE Trans. on Intelligent Transportation Systems*, vol. 4, no. 2, pp. 90–98, June 2003.

[103] S. A. Vaqar and O. Basir, "Traffic pattern detection in a partially deployed vehicular ad hoc network of vehicles," *IEEE Wireless Commun.*, vol. 16, no. 6, pp. 40–46, Dec. 2009.

[104] M. Lotfinezhad and P. Marbach, "Throughput-optimal random access with order-optimal delay," in *Proc. of IEEE INFOCOM*, Shanghai, China, April 2011.

[105] L. Jiang and J. Walrand, "A distributed CSMA algorithm for throughput and utility maximization in wireless networks," in *Proc. of Allerton Conference*, Monticello, IL, Sept. 2008.

[106] B. Nardelli, J. Lee, K. Lee, Y. Yi, S. Chong, E. Knightly, and M. Chiang, "Experimental evaluation of optimal CSMA," in *Proc. of IEEE INFOCOM*, Shanghai, China, April 2011.