Cooperative Internet Access using Heterogeneous Wireless Networks

CHEN BINBIN
B.Sc. Peking University

A THESIS SUBMITTED

FOR THE DEGREE OF PH.D. IN COMPUTER SCIENCE

DEPARTMENT OF COMPUTER SCIENCE

NATIONAL UNIVERSITY OF SINGAPORE

2009

Acknowledgement

It is my great fortune to have pursued my Ph.D. under the guidance of my advisor, Associate Professor Chan Mun Choon. He introduced me to the subject area of this thesis, keeps inspiring me with his profound insights, and always gives me full support. The work would not have been possible without him. I express my deep gratitude to him.

I have benefited in many aspects from the thesis committee members, Associate Professor Ooi Wei Tsang, Dr. Vikram Srinivasan, and Professor Tay Yong Chiang. Prof. Ooi Wei Tsang has been in the committee from the beginning, and he has kept inspiring me with his deep insights and infectious personality ever since. I thank Dr. Vikram Srinivasan for his kind encouragements, deep insights, and for travelling all the way across the Indian Ocean to attend my defense. I thank Prof. Tay Yong Chiang for his constructive criticisms, good advice, and warm encouragements. Besides the many inspiring face-to-face discussions with them, the wonderful modules they offered in NUS greatly help me in building my knowledge foundation.

I am deeply grateful to Associate Professor Akkihebbal L. Ananda for his insightful guidance and warm support. Since working as a teaching assistant for him, I have always kept his teaching as a goal and reference for my own.

I thank Associate Professor Pascale Vicat-Blanc Primet for providing me the valuable internship opportunity to work in INRIA. She gave me insightful guidance and warm support in the unforgettable six-month period, and her care has never stopped ever since.

My special thanks go to Dr. Yu HaiFeng, Dr. Ben Leong, Professor Chua Kee Chaing, Associate Professor Cheng Ee-Chien, Dr. Rajeev Shorey, Associate Professor Lau Hoong Chuin, Associate Professor Gary Tan, Professor Larry Rudolph, Professor Robert Deng, and Associate Professor Pang Hwee Hwa. I thank them for their insightful guidance on my research, through both the classes they offered and the many intellectual conversations we have.

I wish to express my thanks to all present and former members of Communication and Internet Research Lab, as well as my friends who helped me at different periods of my work. In particular, I would like to thank Mr. Padmanabha Venkatagiri. S for setting up the NUS shuttlenet testbed together with me, as well as Mr. Zhang MingZe and Mr. Hao Shuai, for setting up the sensor testbed together. I want to thank Mr. Wu XiuChao,

Dr. Sridhar K. N. Rao, Mr. Shao Tao, Ms. Tan Hwee Xian, Mr Choo Fai Cheong, Mr. Aaron Tan, Mr. Henry Chia, Mr. Sebastien Sudan, and Mr. Hablot Ludovic, for their helps in many aspects of my work and my life.

My dear parents and my two elder sisters, Chen LingLing and Chen TingTing, have always given me warm love and support, for which I am so grateful and without which I would not have been able to finish this dissertation.

I thank my wife Boey Shu Whuen for her love and encouragements. Her support helped me concentrate on completing this dissertation. Her kind, cheerful and always-positive personality encouraged me to survive during the difficult times. I am grateful for having found a life partner as self-sacrificing and bright as her.

Contents

Abstract

The forthcoming generation of mobile communication systems is widely perceived as a convergence platform, which encompasses both multiple heterogeneous wireless access technologies and diverse cooperative networking paradigms. Great efforts have been devoted to build flexible architecture capable of managing them as a whole.

Meanwhile, wireless user devices become more intelligent. They not only participate in the resource allocation process by feeding back their channel states, but also can choose to contribute to the resource provision process by forwarding data for each other. Opportunities bring new challenges. As mobile devices become smarter, a rational user can adapt its behavior in order to benefit more from the network, even if doing so may affect other users and the system's overall performance.

Thus, the design of resource management schemes for this new era of mobile communication should explore the cooperation possibility among heterogeneous wireless networks and their users, while taking the selfish nature of users and their strategic interactions into consideration. This thesis studies the problem of how to deliver Internet access service cooperatively to (selfish) users using heterogeneous wireless networks, in an efficient, fair, and incentive-compatible manner.

Firstly, this thesis addresses the *coordinated radio resource allocation problem* for users that are simultaneously covered by multiple overlapping heterogeneous wireless networks. We propose the *coordinated proportional fairness (CPF)* criterion, based on which a globally fair and efficient allocation decision can be easily computed. As *CPF* decision depends on the input from users, a selfish user may manipulate its channel state report if doing so can increase its gain from the network. We prove that *CPF* allocation is incentive compatible, i.e., a user's dominant strategy is to report its channel state honestly. In practice, the single-association setting, where a mobile station is only associated with one base station, is often desirable. We show that the solution using the same fairness criterion in single-association setting is both computationally expensive and prone to user-manipulation. Alternatively, we propose the *Selfish Load Balancing (SLB)* allocation scheme, which always converges to a Nash equilibrium, and often achieves performance near to *CPF* allocation.

Next, the thesis studies the *cooperative resource provision problem* for highly mobile users in areas where high-bandwidth connection is only available intermittently. We show that user-contributed mobile forwarding can greatly enhance users' Internet access experience. We design *MobTorrent*, a cooperative, on-demand framework, which uses the ubiquitous low-bandwidth cellular network as a control channel while forwarding data through high-bandwidth contacts using a *Delay-Tolerant Networking (DTN)* approach.

MobTorrent makes use of the semi-deterministic knowledge about future contacts, so that the user-contributed mobile forwarding process can be efficiently orchestrated.

To foster cooperation among selfish participants in a DTN environment (e.g., as required by *MobTorrent*), we propose *MobiCent*, a credit-based incentive system designed using the *algorithmic mechanism design* approach. We prove that the proposed scheme is incentive compatible, in the sense that rational nodes will not strategically waste any transfer opportunity or cheat by creating non-existing contacts. *MobiCent* also provides different pricing mechanisms to cater to client that wants to minimize either payment or data delivery delay.

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Convergence of Heterogeneous Wireless Networks

Development in new wireless access technologies and increase in mobile users' demands for ubiquitous high-speed Internet access services are driving the deployment of a wide array of wireless networks, ranging from satellite networks to Wireless Personal Area Networks, with Wireless Wide Area (Cellular) Networks and Wireless Local Area (Wi-Fi) Networks being the two most important components in between.

The cellular network has undergone fast evolution in the last few decades [43, 58]. The first generation (1G) dated back to the late 1970's, such as AMPS (Advanced Mobile Phone Systems), was an analog system providing voice-only service. In the 1990's, the second generation (2G), such as GSM (Global System for Mobile communications), drove the global penetration of mobile telephony into people's daily life. The transition to a digital platform also enabled some primitive but very popular data services, such as SMS (Short Message Services). To meet the rapid growth of demands for data services, the 2.5G wireless packet switched systems such as GPRS (General Packet Radio Service) are introduced to offer better support for data applications. The third generation systems (3G), developed since the late 1990's, are designed for multimedia communication. With data rates as high as several Megabits per second (Mbps), person-to-person communi-

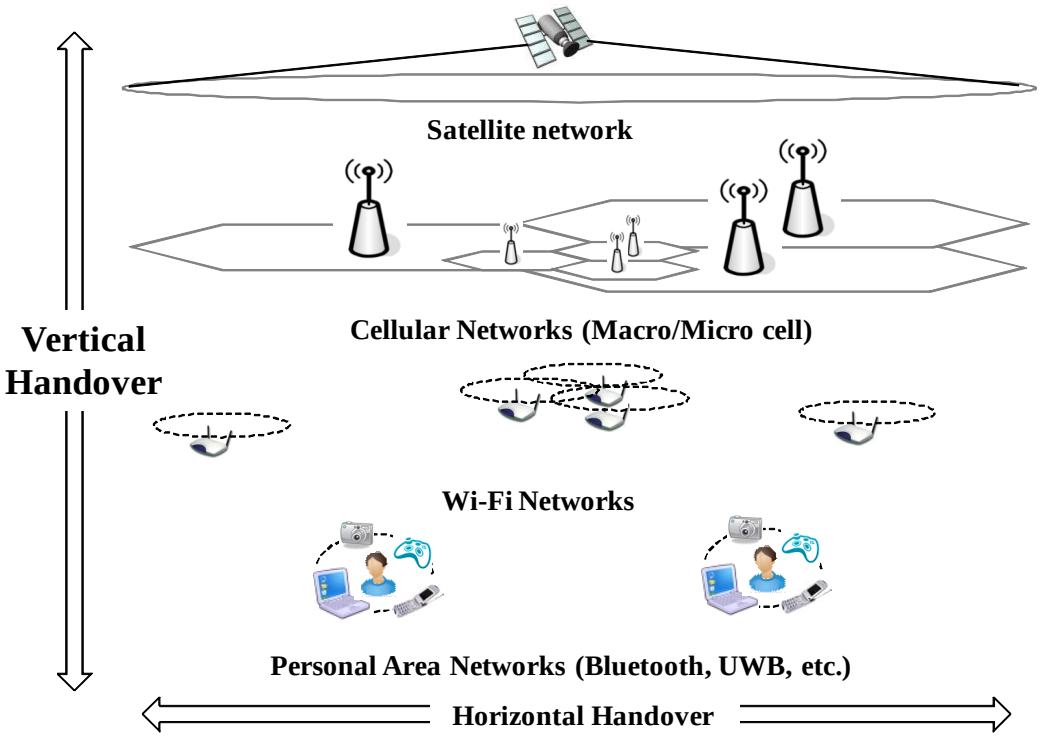


Figure 1.1: Layers of heterogeneous wireless networks

cation can be enhanced with high-quality images and videos, and fast access to information and services on Internet is also available. 3G standards have several variations. Among which, UMTS (Universal Mobile Telecommunication System)'s W-CDMA [43] and Qualcomm's CDMA2000 [11] are the most widely deployed variants. Both of them are evolving towards higher data rate, such as HSDPA (High Speed Data Packet Access) for UMTS, and 1xEV (Evolution) technology (also known as High Data Rate "HDR") system for CDMA2000. The *long term evolution* plans of both systems target to increase their network capacity further [96]. In contemporary cellular networks, macro-cells each covering a large area of multiple square kilometers are still the basis to ensure ubiquitous coverage, whereas micro-cells with much smaller footprints are often deployed in selected areas with high data access demand, to increase the spatial spectrum reuse, thus network capacity thereof.

Wireless users' high-speed access requirements that cannot be satisfied timely by cellular network evolution are effectively addressed by WLAN (Wireless Local Area Net-

work), which is the wireless counterpart of Ethernet. The dominating WLAN standard is IEEE 802.11 (Wi-Fi) [5], which operates on license-free ISM frequency bands and supports high data rate transfer. Wi-Fi networks are widely deployed all around the world. Service providers are offering hot spot access in airports, hotels and other public areas. Even residential users can operate as wireless service providers by themselves [37]. While cellular networks are carefully planned to ensure ubiquitous coverage and meet various traffic load of different areas, Wi-Fi networks are characterized by clustered and intermittent footprints. In addition, Wi-Fi's built-in support for ad-hoc mode, which allows wireless terminals to directly communicate with their peers, provides a more flexible networking solution compared to the traditional single-hop cellular network architecture, and it inspires new networking paradigms to be incorporated into the convergent wireless communication platform, which will be discussed later in Section 1.2.

Figure 1.1 illustrates the different layers of existing heterogeneous wireless networks. As each of these networks has complementary design tradeoffs in coverage, data rates and many other network parameters, it is widely agreed that they will coexist in the future and be integrated together to offer mobile users *"Always Best Connections"* [15, 39]. In addition to horizontal handover in the same layer of wireless network, a *multi-mode wireless terminal*, which is equipped with multiple radio interfaces or Software Defined Radio (SDR) [77], can also vertically handover to another layer when a more suitable access technology is available, or even simultaneously use multiple heterogeneous access technologies to achieve aggregate bandwidth [45].

From the system's point of view, the convergence of several heterogeneous networks into a single logical platform also promises the *best* of all components, including union of the network coverage and aggregation of the network capacity. An integrated platform brings the "trunking gain" to the system, by helping service providers manage the load better, such that the traffic demands varying with location and time can be largely smoothed. For example, if a Wi-Fi hotspot becomes overloaded, some mobile stations (MS) associating with the Wi-Fi access point (AP) can be directed to an overlapping 3G

base station (BS), and vice-versa.

To realize the envisioned benefits, a lot of research [1, 2, 17, 29] has been devoted to address a multitude of challenges, including: mobility management, AAA (Authentication, Authorization and Accounting) service, QoS (Quality of Service) guarantee, access network capacity provision, core network convergence, etc.

As the supporting network protocols are ready, and the various radio access networks begin to interwork with each other, the following resource management problem arises: *how to allocate the radio resources from the heterogeneous network components coordinately, such that users can be served in a fair and efficient way?*

Existing resource allocation schemes in wireless networks often exhibit a disconnection between the following two layers: the *inter-cell association control layer* that decides which BS[1] a MS should associate with, and the *intra-cell allocation layer* that determines how radio resource of a single BS should be shared among its associated MSs. On one hand, the inter-cell association control is often carried out using some simple heuristics, e.g., assigning a MS to the BS with the best signal strength, or to the BS with the least population. On the other hand, the intra-cell scheduling is executed only based on a local view. This disconnection often leads the system to a sub-optimal state from a global point of view.

In Chapter 2 of this thesis, we consider inter-cell association control and intra-cell allocation together, and propose schemes that allocate the resource fairly and efficiently in a network-wide context. *Fairness*, *efficiency*, and *load balancing* are incorporated in a succinct mathematical formulation of the proposed *coordinated radio resource allocation* schemes for such a multi-cell overlapping environment.

[1]Without ambiguity, we use BS as a general term to refer to both cellular base station and Wi-Fi access point.

1.2 User-contributed Mobile Forwarding

In addition to the coordination of heterogeneous radio access technologies as described above, convergence of heterogeneous wireless networks also encompasses the integration of a variety of novel cooperative networking paradigms. One prominent direction of innovation is the incorporation of multi-hop ad-hoc networking model with the traditional single-hop cellular network architecture. This general paradigm is often called multi-hop cellular networks (MCNs) [65]. A number of MCN-type frameworks have been studied. Some of these frameworks propose to deploy dedicated relaying entities for data forwarding, such as the proposal by Wu et al. [109], and the proposal by Fitzek et al. [36]. We refer to this type of relay stations as fixed relays. Alternatively, the mobile users themselves may forward data for each other, as suggested by Lin and Hsu [66], Wu et al. [111], Aggelou et al. [3], Hsieh et al. [44], Zadeh et al. [112], Luo et al. [69], Bhargava et al. [13], Hu and Zhang [47], and Lee et al. [59]. We refer to these forms of relay stations as *mobile relays*. We focus on the category of *user-contributed mobile forwarding* because of its greater flexibility and lower cost.

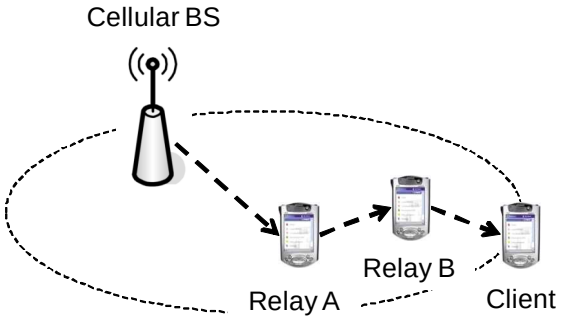


Figure 1.2: User-contributed forwarding using a multi-hop end-to-end path

The basic idea of MCN is illustrated in the example of Figure 1.2, where the client has both a 3G cellular link and a Wi-Fi based peer-to-peer link. As it situates in the fringe of the 3G cell, it experiences poor channel condition with the cellular BS. To make more efficient use of the spectrum, instead of sending data directly to the client in a single hop, the cellular BS forwards packets for it to a proxy client (Relay A) with better

channel quality. Relay A then uses an ad-hoc network probably composed of other users (Relay B in this example) and Wi-Fi links to forward the packets to the specified client. By leveraging a multi-hop path, the client can significantly improve its data throughput. Additionally, the enhanced transmission efficiency of these "resource-inefficient" clients results in less consumption of radio resources, thus can improve the performance of other clients in the same cell that are not even aware of the multi-hop forwarding. Furthermore, the relaying mechanism can effectively extend service coverage area, and can also help to achieve better load balance by dynamically diverting the traffic load from a hot cell (highly loaded cell) to a cool cell (lightly loaded cell) through relay nodes.

In frameworks proposed above, the peer-to-peer connection is often based on short-range radio transmission like Wi-Fi, and nodes can communicate with each other only when they are relatively close. As the locations of mobile users are essentially unplanned and largely unpredictable, a high-throughput end-to-end path may not exist in many realistic settings with sparse and highly mobile users, like vehicular networks or mobile human social networks. In particular, if the Internet access gateways are Wi-Fi APs, which themselves have short transmission range and provide only intermittent coverage, the probability of having contemporaneous multi-hop connectivity becomes extremely low.

While all existing MCN frameworks assume the existence of an end-to-end relaying path, the contemporaneous end-to-end connectivity is not a prerequisite to employ user-contributed mobile forwarding for delay-tolerant applications, like downloading a big file from Internet. For such applications, the *Delay-Tolerant Networking (DTN)* approach can be used to opportunistically exploit the available intermittent contacts for data delivery [25, 35, 49, 115]. The proposed DTN solution adopts the idea of *store*, *carry*, and *forward*, where a mobile node *stores* and *carries* the data until the client or another mobile relay moves into its vicinity, so that it can *forward* the data to the latter. The idea of DTN forwarding is illustrated in Figure 1.3, where Relay A retrieves the client's data from the Wi-Fi AP, carries the data, moves around, and forwards (or replicates) the data

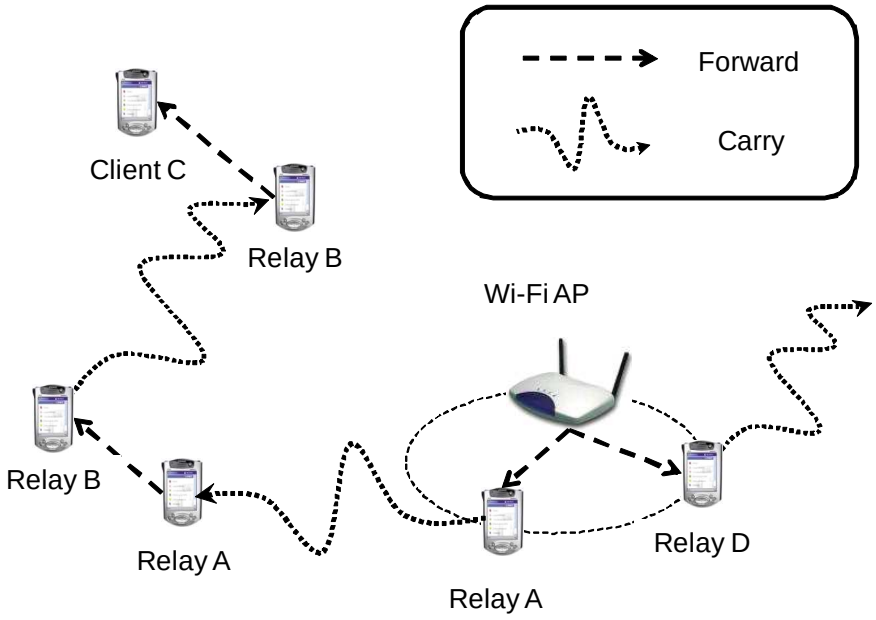


Figure 1.3: User-contributed forwarding using a DTN approach

when it meets another node Relay B. Relay B carries on with the data, until it meets the client to complete the data delivery. As contacts are often unpredictable, forwarding (or replication) of data among mobile relays happens in an opportunistic manner. To increase the delivery ratio and reduce the delivery delay, data are often propagated along multiple paths simultaneously (e.g., the AP also replicates the same data to Relay D as shown in Figure 1.3), in the hope that at least one of the relays can meet the client.

In Chapter 3 and Chapter 4 of this thesis, we study the resource provision problem for highly mobile users in areas where high-bandwidth connection is only available intermittently. Previously, the application of DTN routing approach is considered only in scenarios without infrastructure support, such as inter-planetary networks, wildlife tracking, disaster relief team networks, or information delivery for remote villages and nomadic people. We are the first to introduce the DTN-routing paradigm to enhance the performance of cellular network infrastructure. Our results show that, if the cooperation among participants can be efficiently orchestrated and properly fostered, user-contributed mobile forwarding can greatly enhance mobile users' Internet access experience.

1.3 Selfish User Behavior and Algorithmic Mechanism Design

With increased intelligence, the new generation of wireless terminals not only can facilitate the radio resource allocation process by feeding back the measured channel state, but also can contribute to the resource provision process by forwarding data for each other, as presented above. When users gain more control over their devices, an intelligent and selfish user can adapt its behavior in order to benefit more from the network, even if doing so may affect other users and the system's overall performance.

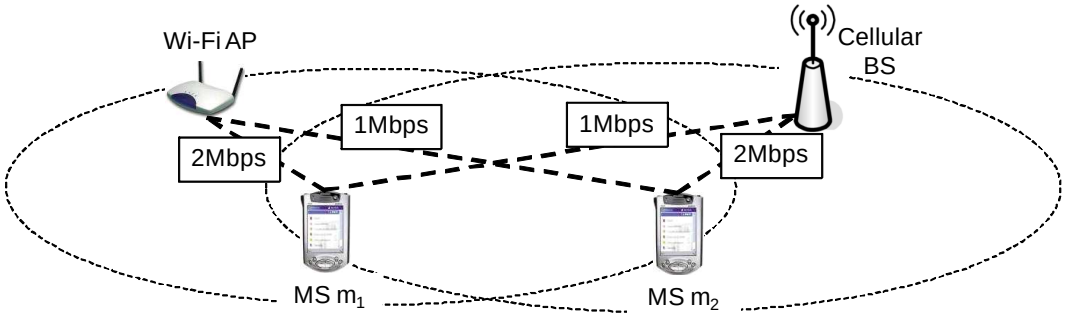
Thus, the resource allocation and provision schemes for future convergent wireless networks should take the selfish nature of participants and strategic interactions among them into consideration. *Game theory,* and *algorithmic mechanism design* in particular, provide a powerful tool to address these challenges [20, 81, 82, 83, 106].

Game theory aims to model situations in which multiple participants select strategies that have mutual consequences. Following the definitions used by Nisan et al. [82], a game consists of a set of n players, $1, 2, \ldots, n$. Each player i has its own set of possible strategies, say S_i. To play the game, each player i selects a strategy $s_i \in S_i$. We use $s = (s_1, \ldots s_n)$ to denote the vector of strategies selected by the players and $S = \times_i S_i$ to denote the set of all possible ways in which players can pick strategies. The vector of strategies $s \in S$ selected by the players determines the outcome for each player. If by using a unique strategy, a user always gets better outcome than using other strategies, independent of the strategies played by the other players, we say that the strategy is the user's *dominant strategy*. If players select strategies such that, no player can unilaterally change its strategy to gain more payoff, we say that the game reaches a *Nash equilibrium*. In another word, every player is playing the best response to others in a *Nash equilibrium*. As can be easily derived, if each user has a *dominant strategy*, the unique *Nash equilibrium* in the game is for each user to adopt its *dominant strategy*.

Game theory has been widely used in social sciences (most notably economics) and

other areas since it was formally introduced by J. von Neumann and O. Morgenstern in their 1944 monograph [106]. Computer networks researchers have used game theory to study Internet, since Internet emerged as a complex ecosystem without any central control decades ago [82]. However, its application in the research of wireless networks only began in recent years, as wireless terminals gain increased intelligence and mobile communication systems evolve towards an increasingly open platform [20].

To illustrate the strategic interactions among users in the forthcoming mobile communication era, we will introduce two games which naturally arise in the resource management problems that this thesis studies.



(a)

MS m_2 \ MS m_1	None	Wi-Fi Only	Cellular Only	Both
None	0 0	2 0	1 0	3 0
Wi-Fi Only	0 1	1 0.5	1 1	2 0.5
Cellular Only	0 2	2 2	0.5 1	2.5 1
Both	0 3	1 2.5	0.5 2	1.5 1.5

(b)

Figure 1.4: An association game example

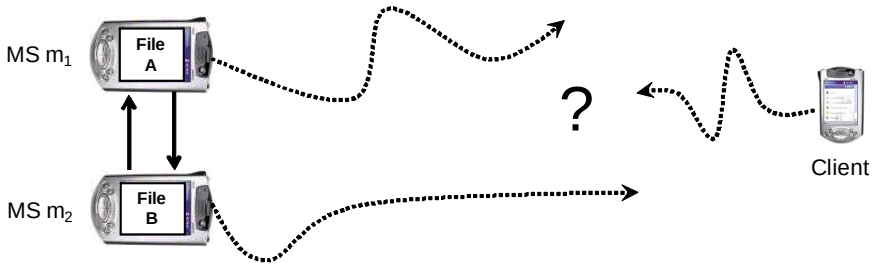
In the example of *association game* as illustrated in Figure 1.4 (a), there are two dual-radio mobile stations, MS m_1 and MS m_2, as players. Each of them is equipped with both a cellular interface and a Wi-Fi interface. Both mobile stations locate in the overlapping coverage area of a Wi-Fi AP and a cellular BS. However, their channel conditions to the

AP and the BS are different. MS m_1 can communicate with the AP at 2Mbps and with the BS at 1Mbps, while MS m_2 can communicate with the AP at 1Mbps and with the BS at 2Mbps. If the AP or the BS has only a single associated user, that user can monopolize all radio resource from the AP (or BS), and get a throughput value equal to its link data rate. Instead, if two users are simultaneously associated with the AP (or BS), the AP (or BS) implements some scheduling algorithm to divide its radio resource (e.g. transmission time slot) among them, so that each user only gets a fraction of its link data rate. Without loss of generality, we assume that both the AP and the BS adopt the popular time-based fair scheduling scheme [11, 101], such that the bandwidth allocated to each of the two users associated with the same AP (or BS) is half of its link data rate.

We assume that both users are running some bandwidth-greedy applications, so that each individual always prefers higher bandwidth allocation. For a player, its strategies include: (1) turn off both interfaces (*None*), (2) turn on the Wi-Fi interface only (*Wi-Fi Only*), (3) turn on the cellular interface only (*Cellular Only*), and (4) turn on both interfaces simultaneously to achieve aggregate throughput (*Both*). The reward matrix (in terms of the aggregate throughput value for each user) can be easily calculated as in Figure 1.4 (b) (the left entry for the row player MS m_2 and the right entry for the column player MS m_1). Clearly, there are sixteen total outcomes depending on the choice made by each of the two users.

The unique Nash equilibrium in this game is that both users turn on both of their interfaces; in each of the other fifteen cases, at least one of the players can switch to the *Both* strategy to improve its own payoff. On the other hand, a better outcome for both players happens when MS m_1 uses the Wi-Fi interface only, and MS m_2 uses the cellular interface only. However, this is not a Nash equilibrium, since each of the players would be tempted to turn on its silent interface and thereby increase its throughput.

A similar dilemma happens also in the user-contributed mobile forwarding scenario as depicted in Figure 1.5 (a). In this example of *mobile forwarding game*, we also assume that there are two mobile stations, MS m_1 and MS m_2, as players. Each of them has a



(a)

MS m_1 / MS m_2	Not replicate file A	Replicate file A
Not replicate file B	0.7 0.7	0.6825 0.9275
Replicate file B	0.9275 0.6825	0.91 0.91

(b)

Figure 1.5: A mobile forwarding game example

unique file, which is denoted as file A and file B respectively. A client is interested to get both files. Without loss of generality, we assume that the client is willing to pay 1 cent for each new file, and the reward will be shared equally among all relays on the DTN forwarding path with the minimum delay. We assume that both MS m_1 and MS m_2 have a probability of 0.7 to meet the client directly, and the two contact probabilities are identical and independent of each other. Suppose MS m_1 and MS m_2 meet each other before either of them meets the client. For each player, its strategies include: (1) not replicate its own file to the other player, and (2) replicate its own file to the other player.

If no replicate happens between the two nodes, each player can only forward its own file to the client, for which it monopolizes the reward of 1 cent. As each player's individual contact probability with the client is 0.7, each of them has an expected reward of 0.7 cent. Now let us look at the asymmetric setting when MS m_1 replicates file A to MS m_2, whereas MS m_2 does not replicate file B to MS m_1. File A can reach the client in two ways, either directly from MS m_1 in one hop, or via MS m_1 and MS m_2 in two hops. Because of the independence assumption, the probability that none of these two

possibilities happens is $(1-0.7)^2 = 0.09$. As the two possibilities happen with identical and independent chance, the probability for each of them to happen and happen first is $(1-0.09)/2 = 0.455$. On one hand, if file A is delivered first by MS m_1 in one hop, MS m_1 monopolizes the 1 cent reward. On the other hand, if file A is delivered first via the 2-hop path consisting of both MS m_1 and MS m_2, MS m_1 need to share the reward with MS m_2. As MS m_1 earns reward only from the delivery of file A, its expected gain is $0.455 \times 1 + 0.455 \times 0.5 = 0.6825$ cent. For MS m_2, in addition to the expected gain of 0.7 cent from delivering file B, it can also benefit from the half cent reward by forwarding file A, if it meets the client earlier than MS m_1. Thus, it has a total expected reward of $0.7 + 0.455 \times 0.5 = 0.9275$ cent. Similar analysis can be applied to find the reward for the situation when MS m_2 replicates file B to MS m_1, whereas MS m_1 does not replicate file A to MS m_2. Finally, when the two MSs carry out mutual replication, both files will be delivered if at least one MS meets the client. Thus, the delivery probabilities for both files are $1-(1-0.7)^2 = 0.91$. The expected total reward is $2 \times 0.91 = 1.82$ cent. Because of the symmetry assumption, the expected reward for each MS is $1.82/2 = 0.91$ cent.

The expected rewards for the two MSs in the four possible outcomes are summarized in Figure 1.5 (b). For each outcome, the left entry represents the reward for the row player MS m_2, and the right entry for the column player MS m_1. The unique Nash equilibrium in this game is that both users do not replicate to each other, despite the fact that mutual forwarding can increase the expected rewards of both players.

These two games clearly demonstrate that the strategic behavior of selfish users may lead to a sub-optimal state. In fact, both of them are instantiations of the famous Prisoners' dilemma [82] in their respective settings.

When we design resource management schemes for next generation mobile communication systems, the rules of how participants play a game and the outcome of the game under different combinations of users' strategies, can be taken into consideration, such that inefficiency could be potentially avoided or minimized by designing the game carefully.

Algorithmic mechanism design [81, 82] is a subarea of game theory that deals with the design of games. It studies optimization problems where the underlying data, e.g., the channel states experienced by MS in the *association game*, or the replication opportunities in the *mobile forwarding game*, are *a priori unknown* to the algorithm designer, and must be implicitly or explicitly elicited from selfish participants. The high-level goal is to design a protocol, or "mechanism", that interacts with participants so that *selfish behavior yields a desirable outcome*. More specifically, a mechanism is *incentive compatible*, or *strategy-proof*, if the dominant strategy of each participant under the designed mechanism is to reveal its state truthfully. We adopt the *algorithmic mechanism design* approach when designing and analyzing the resource management schemes for the forthcoming generation of mobile communication systems.

1.4 Thesis Contributions

In the era of convergent wireless networks, we need to design new resource management schemes to explore the cooperation possibility among heterogeneous wireless networks and their participants, while taking the selfish behavior of users and their strategic interactions into consideration. In this thesis, we investigate the problem of how to deliver Internet access service cooperatively to (selfish) users using heterogeneous wireless networks in an efficient, fair, and incentive-compatible manner.

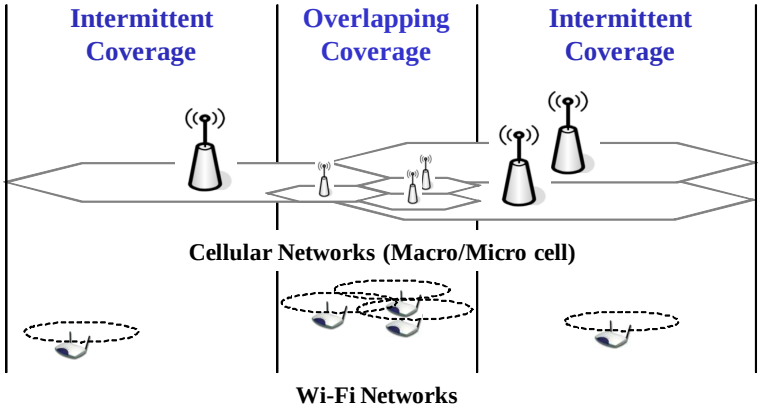


Figure 1.6: Heterogeneity in coverage

While cellular networks are carefully planned to ensure ubiquitous coverage and meet various traffic load of different areas, Wi-Fi networks are characterized by clustered and intermittent footprints. As shown in Figure 1.6, the heterogeneous geographic distribution of network coverage and capacity results in two dramatically different scenarios. On one hand, in "hot" areas where a large number of user demands are expected, such as shopping malls, hotels, and airports, densely deployed Wi-Fi and cellular networks often provide overlapping coverage. In these areas, a multi-mode wireless terminal can potentially be associated with one or multiple overlapping BSs. Note that, in such regions, cellular networks are often deployed as micro-cells (or femtocells), thus provide comparable capacity and coverage as Wi-Fi networks. On the other hand, in the rest of regions, such as residential areas, natural parks, and highways, high-bandwidth Wi-Fi connection is available only intermittently, and cellular networks are often deployed as macro-cells, thus only provide low-speed connection.

<table>
<tr><td>Coverage
Perspective</td><td>Overlapping</td><td>Intermittent</td></tr>
<tr><td>System performance</td><td rowspan="2">Chapter 2</td><td>Chapter 3</td></tr>
<tr><td>Incentive compatibility</td><td>Chapter 4</td></tr>
</table>

Figure 1.7: Thesis road map

To realize the vision of next generation mobile communication systems, which promises the *always best connection* for mobile users *anytime, anywhere, anyhow*, resource management schemes for both *overlapping-coverage* and *intermittent-coverage* scenarios should be designed carefully. This thesis studies both scenarios. For each scenario, we address the system design problem from two perspectives, as illustrated in Figure 1.7. Firstly, we consider the problem of how to achieve efficient system performance, given that users are fully cooperative. Secondly, we study the incentive compatibility problem, and provide rigorous analysis to show that cooperation can be fostered in the proposed resource management schemes. This thesis makes the following contributions:

- We study the *coordinated radio resource allocation problem* for users being simultaneously covered by multiple overlapping heterogeneous wireless networks. We propose the *coordinated proportional fairness (CPF)* allocation scheme, which makes globally fair and efficient allocation decision among networks. The proposed allocation decision can be calculated efficiently, and our simulations demonstrate that the proposed algorithms outperform popular heuristic approaches, by striking a good balance between efficiency and fairness, while achieving load balancing among network components.

- We formulate the resource allocation process as the *multi-cell resource allocation game*. The formulated game is associated with a resource allocation rule, which calculates the bandwidth allocation outcome based on the input from the MS players. A MS can manipulate its channel state report to game the system.

- Using the proposed game theory framework, we analyze the incentive compatibility of the *multi-cell resource allocation game* with *CPF* allocation scheme as its associated rule. We show that a *multi-cell resource allocation game* with *CPF* allocation is incentive compatible. However, the positive result does not hold for its variant in the single-association setting, where a MS is associated with a single BS. For the single-association setting, we propose the *Selfish Load Balancing (SLB)* allocation scheme, which always converges to a Nash equilibrium, and often provides performance near to *CPF* allocation.

- To address the challenges of allowing highly mobile users to transfer large amounts of data in areas with only intermittent but high-bandwidth connections, we propose *MobTorrent*, a cooperative, on-demand framework, which uses the ubiquitous low-bandwidth cellular network as a control channel to exploit the high-bandwidth intermittent Wi-Fi contacts for data delivery in a *Delay-Tolerant Networking (DTN)* approach.

- The scheduling algorithm in *MobTorrent* makes use of the semi-deterministic knowl-

edge about future contacts, so that the user-contributed mobile forwarding process can be efficiently orchestrated. We derive the achievable performance bound, and show that *MobTorrent* provides near optimal data delivery performance, in terms of both the delivery ratio and the delivery delay.

- We consider the incentive design for a DTN environment to foster cooperation among selfish participants (e.g., as required by *MobTorrent*). We identify *edge insertion attacks* and *edge hiding attacks* as the two major forms of attacks in a DTN environment. Both of them are difficult to detect, and can seriously degrade the performance of DTN routing. We formulate these two attacks in the *path revelation game*, and show that existing incentive schemes are not incentive compatible.

- We design *MobiCent*, a credit-based incentive system for DTN. We prove that the proposed scheme is incentive compatible under the two attacks, in the sense that a MS cannot increase its reward by launching *edge insertion attacks* and *edge hiding attacks*. *MobiCent* also provides different pricing mechanisms to cater to client that wants to minimize either payment or data delivery delay.

1.5 Thesis Organization

The rest of the thesis is organized as follows.

Chapter 2 studies the *coordinated radio resource allocation problem* for users that are simultaneously covered by multiple overlapping heterogeneous wireless networks. We formulate the *coordinated proportional fairness (CPF)* resource allocation criterion, based on which a globally fair and efficient allocation decision can be easily computed. A *multi-cell resource allocation game* is formulated to capture the selfish behavior of users. Based on which, we prove that *CPF* allocation is incentive compatible. We also formulate the integral version of the *CPF* problem (*Int-CPF*) for the practically desirable single-association setting, and show that it is both computationally expensive and prone to

user-manipulation. Alternatively, we propose the *Selfish Load Balancing (SLB)* scheme, which always leads to a Nash equilibrium, and often achieves performance near to *CPF* allocation.

Chapter 3 and Chapter 4 address the challenges in the intermittent-coverage scenario. Chapter 3 presents *MobTorrent*, a cooperative, on-demand framework to provide Internet access for vehicles. *MobTorrent* uses the ubiquitous low-bandwidth cellular network as a control channel, while forwarding data through high-bandwidth contacts in a DTN paradigm. We study the problem of how to schedule the transmission over intermittent contacts, such that the amount of data delivered is maximized and the delay is minimized.

After *MobTorrent*, we present in Chapter 4 the design of *MobiCent*, a credit-based incentive system for DTN. *MobiCent* is largely motivated by, and directly designed upon *MobTorrent*. In this chapter, we formulate the *path revelation game* with both edge insertion attacks and edge hiding attacks. We characterize the necessary conditions for a payment scheme to be incentive compatible under edge insertion attacks. Two different pricing mechanisms are designed to cater to client that wants to minimize either payment or data delivery delay. We prove that both of the proposed schemes are incentive compatible. As the two attacks are fundamental to the nature of DTN, we expect *MobiCent*'s credit-based solution can be extended to foster cooperation in other forms of DTN systems different from *MobTorrent*.

Finally, conclusion and possible future works are presented in Chapter 5.

Chapter 2

Coordinated Proportional Fairness for Overlapping Cells

2.1 Introduction

Overlapping coverage of wireless base stations (BS[1]) is a common phenomenon in mobile communication systems. For a particular radio access network, neighboring cells or sectors overlap with each other. In addition, deployment and inter-operation of a wide array of wireless access networks, ranging from 3G network to Wi-Fi hotspots, open the opportunity of overlapping coverage from BSs using heterogeneous radio access technologies. In such an environment, a multi-mode (e.g., Wi-Fi and 3G capable) MS can flexibly associate with either a Wi-Fi AP or a 3G BS or simultaneously with both (Wi-Fi and 3G) BSs.

As the various radio access networks begin to interwork with each other, the following resource management problem arises: *how to allocate the radio resources from the heterogeneous network components coordinately, such that users can be served in a fair and efficient way?*

As discussed in Chapter 1, new models and techniques should be developed to address

[1]Same as in Chapter 1, we use BS as a general term to refer to both cellular base station and Wi-Fi access point.

the resource allocation problem in this new environment for the following reasons.

Firstly, existing resource allocation schemes in wireless networks often exhibit a disconnection between the following two layers: the *inter-cell association control layer* that decides which BS a MS should associate with, and the *intra-cell scheduling layer* that determines how radio resource of a single BS should be assigned among its associated MSs. On one hand, the inter-cell association control is often carried out using some simple heuristics, e.g., assigning a MS to the BS with the best signal strength, or to the BS with the least population. On the other hand, the intra-cell scheduling is executed only based on a local view. When the association decision is made by selfish MSs, a system without coordination among BSs often operates in a state far from the optimal, as clearly indicated by the association game example presented in Section 1.3 of Chapter 1.

Secondly, despite the fact that research for wired networks does consider routing (the wired counterpart of inter-cell association control) and scheduling (the wired counterpart of intra-cell allocation) together, existing models for wired networks fail to capture some important characteristics that are unique to wireless networks. In this thesis, we focus on the aspect that a single MS may experience significantly different channel conditions with different BSs, and a single BS may experience different channel conditions with different MSs as well. In addition, the wireless networks often rely on individual MS to measure and report its current channel states with neighboring BSs, in order to make informed decisions. This allows an intelligent and selfish MS to game the system by manipulating its channel report, as to be shown in Section 2.4.3.

In this chapter, we consider the inter-cell association control and intra-cell allocation together, such that the resource is allocated fairly and efficiently in a network-wide context. The content of this chapter is organized as follows. In Section 2.2, we describe the system model. In Section 2.3, we review the existing fairness definitions, with an emphasis on *proportional fairness*. In Section 2.4, we present our *Coordinated Proportional Fairness (CPF)* formulation [24], and show that it can be easily solved as a convex programming problem. Considering the strategic behaviors of users, we formulate

the *multi-cell resource allocation game*, and show that the *CPF* mechanism is incentive compatible. For the practically attractive single-association scenario, where each MS is associated with a single BS, Section 2.5 formulates the integral variant of the *CPF* problem (*Int-CPF*) and shows that it is NP-hard. Furthermore, the *Int-CPF* allocation scheme is not incentive compatible. Alternatively, we present a *Selfish Load Balancing (SLB)* scheme, and analyze its convergence. In Section 2.6, we evaluate the performance of the various schemes proposed, and compare them to some popular heuristics. Section 2.7 presents the related work. We conclude in Section 2.8.

2.2 System Model

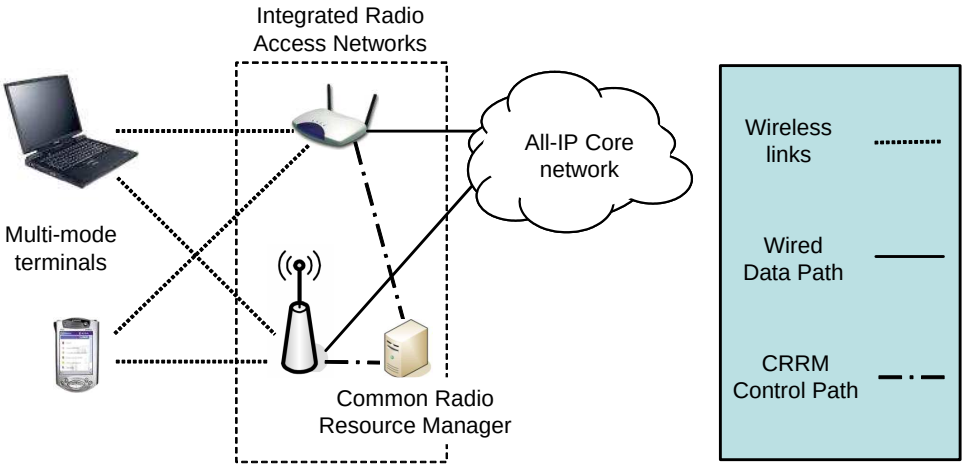


Figure 2.1: A convergent mobile communication system

Our discussion is based on a convergent system of heterogeneous wireless networks as shown in Figure 2.1. The main components of the considered architecture are: multi-mode terminals, all-IP core network, and the integrated radio access networks (RANs) sitting between them. We briefly describe each of them as follows.

- *Multi-mode terminals*. Ongoing silicon development enables chipmakers to integrate multiple forms of radio access technologies in a single chipset. For example, Qualcomm's Snapdragon chipset for mini-notebooks includes Wi-Fi alongside 3G,

Bluetooth, broadcast TV and GPS (Global Positioning System) capabilities [88]. Shipments of Wi-Fi chips in multi-mode mobile handsets are reported to grow by more than 50 percent in 2008 and reach 56 million units. The Apple iPhone, which was introduced in 2007 in the U.S. and expanded to more than 70 countries in 2008, helps drive that growth with shipments of more than 10 million units. It also helps set the tone for the industry, making Wi-Fi capability a standard feature on smartphones. This trend is expected to be further boosted by the recent development of SDR (Software Defined Radio) technologies [77, 102].

- *All-IP core network.* Wireless core networks are quickly evolving to packet switched IP-based mechanisms [96]. IP layer shields the applications from the underlying network technologies, thus enabling much richer set of common services to be provided independent of the access networks. The open specifications and platforms also greatly facilitate the creation of new service, and enable the use of cheaper, faster, and better core equipments.

- *Integrated Radio Access Networks*. As a bridge between the two components above, flexible architecture capable of managing a large variety of coexisting radio access networks is being standardized [1, 2, 33]. The proposed Common Radio Resource Management (CRRM) functions [67, 103] consider the pool of resources in all radio access technologies (RATs) as a whole, aiming at a better overall performance than that can be achieved by the stand-alone networks. As shown in the figure, the common radio resource manager can be interpreted as a logical entity which gathers input from different RATs (such as Wi-Fi networks and 3G networks), and coordinates resource allocation decisions among them. Both the input and output controls are carried out using the CRRM functions.

Consider a set of BSs using heterogeneous radio access technologies controlled by a single common radio resource manager, we assume that each BS has a fixed amount of radio resource (e.g. channel or transmit power) and operates orthogonally with each

other. A common example of such a scenario is a 3G BS and an overlapping Wi-Fi AP.

Note that this model is general and is applicable to cases where the BSs use the same radio technology as long as the channels are orthogonal. For example, this simplified model also roughly captures the current operation mode for both Wi-Fi networks and high data rate cellular networks. For Wi-Fi networks, 802.11b and 802.11g use the 2.4 GHz ISM band, which is divided into 13 channels each of width 22 MHz but spaced only 5 MHz apart, thus offers 3 non-overlapping channels. 802.11a uses the 5 GHz U-NII band, which offers 12 non-overlapping channels (in FCC and North America standard). Given the separation between two non-overlapping channels, the signal on one channel is sufficiently attenuated to minimally interfere with a transmitter on another channel. In today's typical deployment, each Wi-Fi AP operates in a *single channel* that is selected to be orthogonal to its neighboring APs, if possible. Ideally, there should be no co-channel APs in the same contention domain. Channel selection for neighboring Wi-Fi APs has been discussed by Kauffmann et al. [53], and their results demonstrate that interference among neighboring Wi-Fi APs can be effectively mitigated using the proposed frequency selection scheme. For cellular networks, we take the widely deployed High Data Rate (HDR) networks [11] as an example. Using a dedicated RF carrier, the HDR downlink for each BS is time multiplexed and transmitted at the *full power* available. To date, the BS location, antenna down-tilt and transmit power are determined at the time of deployment and hence are not dynamic.

Though in our model we focus on the case that the radio capacities of BSs are fixed and orthogonal, they can potentially be adapted to improve the network-wise performance. On one hand, Wi-Fi channel bonding is used in "Super G" technology, which bonds two channels of classic 802.11g to double the PHY data rate. On the other hand, in HDR networks, transmit power control can be applied to mitigate inter-cell interference. For future research, we would like to incorporate the BS capacity adaptation into the consideration of the network-wide radio resource allocation.

We say there is a link $l = (m, b)$ between a MS m and a BS b if they are able to

communicate with each other. We call such a pair an *adjacent MS-BS pair*. The input for CRRM is the channel states for all adjacent MS-BS pairs. We focus on the downlink from BS to MS. In wireless networks, the link data rate is determined by the channel condition between the transceiver and the receiver. For example, in HDR, MSs monitor the pilot bursts in the downlink channel to estimate the channel conditions in terms of Signal to Noise Ratio (SNR). This SNR is then mapped into a supported data rate, and fed back in every time slot to the BS through the data-rate-request channel in the reverse link.

We focus on *elastic traffic*, which can adapt to the bandwidth allocated by the system. To simplify the discussion, we assume that a user will consume all the bandwidth allocated and the queues are backlogged. The allocated bandwidth for a MS on a link is the product of the link data rate and the fraction of the radio resource allocated by the corresponding BS. Thus, the bandwidth equals to the link data rate only if the MS monopolizes the radio resource of the BS. Otherwise, the bandwidth of a MS is a fraction of its link data rate. In both Wi-Fi networks and HDR networks, time multiplexing is used to share the resource of BS among its associated MSs, i.e., data transfers to different users are scheduled at different time slots. Thus, the resource consumptions by different links at the same BS are orthogonal, and can be linearly summed up. In addition, we assume that there is no constraint in the number of MSs that can be associated to a BS[2].

Because of the lossy nature of wireless communication and the scarcity of spectrum resource, the wireless links are likely to be the bottleneck of the system described in Figure 2.1. Thus, a radio resource management scheme, which allocates the combined radio resource in a fair, efficient, and load-balancing way, is the key to meet mobile customers' requirements. Fairness, efficiency, and load balancing need to be considered together when designing radio resource allocation schemes for such a multi-cell environment. On one hand, a scheme which maximizes only the aggregate system throughput, or equivalently, the *arithmetic mean of per-user throughput values*, results in the starvation of

[2]There are 60 Walsh codes for orthogonal transmission in HDR. This puts an upper bound of 60 active users per BS at any given time. However, the limit of 60 users is rarely reached in practice.

resource-inefficient users, because it allocates all system resources to the users with the best link data rate. On the other hand, a scheme which makes users' allocation data rates as equal as possible, or equivalently, maximizes the *minimum per-user throughput value*, regardless of their link data rate, often results in poor overall system performance in wireless networks, as shown in Section 2.3.1. In addition, a scheme considering only each individual cell can easily lead to unfairness among users located in different areas.

2.3 Fairness Definition

Before we formulate the *coordinated proportional fairness (CPF)* resource allocation criterion, we first briefly review several important fairness definitions in computer networks literature.

2.3.1 Max-min Fairness

The most common understanding of fairness in computer networks is probably the *max-min fairness*, as defined by Bertsekas and Gallager [12]: rates are made as equal as possible subject only to the constraints imposed by link capacities. Formally, consider a bandwidth allocation $\mathbb{R} = (R_m, m \in M)$, where M is the set of users, and R_m is the bandwidth allocated to user $m \in M$, we define the sorted bandwidth allocation $\overline{\mathbb{R}} = (\overline{R_m})$ as the users' allocated bandwidths sorted in non-decreasing order.

Definition 2.1 ***Max-min Fairness*** *[10]: A feasible bandwidth allocation scheme S^* is called max-min fair if and only if, for any other feasible bandwidth allocation S, it satisfies: $\overline{\mathbb{R}^{(S^*)}}$ has the same or higher lexicographical value than $\overline{\mathbb{R}^{(S)}}$, where $\overline{\mathbb{R}^{(S)}}$ and $\overline{\mathbb{R}^{(S^*)}}$ are the sorted bandwidth allocation vectors under the two considered schemes S and S^* respectively.*

Although the *max-min fairness* is Pareto optimal (i.e., any change to make any MS better off is impossible without making some other MS worse off), it has been criticized

for favoring too much of resource-inefficient requests, thus it does not make efficient use of resource. In addition, there appears to be no clear economic reason why max-min sharing should be preferred over some other bandwidth allocation schemes.

In particular, *max-min fairness* is not efficient for elastic traffic in a multi-rate wireless communication system as considered in this thesis, because when some MSs use a lower bitrate than the others, the performance of all MSs sharing the same BS is considerably degraded to the same level as the worst one, as shown by Heusse et al. [42]. For example, 802.11b products degrade the bitrate from 11 Mbps to 5.5, 2, or 1 Mbps when repeated unsuccessful frame transmissions are detected. In such a case, a host transmitting at 1 Mbps reduces the throughput of all other hosts transmitting at higher data rates to a value below 1 Mbps. The basic CSMA/CA channel access method is at the root of this anomaly: it guarantees an equal long-term channel access probability to all hosts. Once a host gets the access opportunity, it starts sending a rate-independent length of frame using its available bitrate. A host captures the channel for a longer time if its bitrate is lower, thus it penalizes other hosts that use the higher rates.

2.3.2 Proportional Fairness

Compared to *max-min fairness, proportional fairness* as proposed by Kelly [55, 56] strikes a better balance between efficiency and fairness.

Definition 2.2 ***Proportional Fairness*** *[55]: A feasible bandwidth allocation scheme S^* is called proportionally fair if and only if, for any other feasible bandwidth allocation S, it satisfies:*

$$\sum_{m \in M} \frac{R_m^{(S)} - R_m^{(S^*)}}{R_m^{(S^*)}} \leq 0 \tag{2.1}$$

where $R_m^{(S)}$ and $R_m^{(S^*)}$ are the rates allocated to user m by the two considered schemes S and S^* respectively, and M is the set of users.

The rationale behind *proportional fairness* criterion can be interpreted from multiple angles as follows.

Engineering Viewpoint

Max-min fairness does not allow any increase of a large sharing if the increase is at the cost of some smaller sharing being decreased, even if significant increase for the large sharing can be achieved with only minor decrease of the small sharing. *Proportional fairness* relaxes this restriction by allowing large sharing to increase further with small sharing decreased, if changes of the assigned bandwidth vectors result in the sum of the proportional changes to be non-negative, as shown in Equation 2.1. By doing so, *proportional fairness* favors resource-efficient requests more than *max-min fairness*, thus helps improve system efficiency. On the other hand, although the requirement of non-negative proportional change is less strict than *max-min fairness*, *proportional fairness* still helps prevent resource-efficient connections from starving resource-inefficient connections totally. It is shown that both *max-min fairness* and *proportional fairness* can be viewed as special cases in a family of fairness definitions striking different tradeoffs between efficiency and fairness [56].

Utility Maximization Viewpoint

When *proportional fairness* is proposed [55], it is associated with the optimization of an objective function representing the overall utility of the flows in progress. The utility function chosen is logarithmic function of the allocated bandwidth, where the value of a flow increases with its allocated bandwidth R in proportional to $logR$. It is shown that the *"proportional fairness"* solution as defined in Equation 2.1 maximizes the logarithmic sum of the user throughput values, which can be formally written as

$$S^* = argmax_S \sum_{m \in M} logR_m^{(S)} \tag{2.2}$$

It's easy to see that the optimization of the logarithmic sum of the throughput values is equivalent to the optimization of their product form.

$$S^* = argmax_S \prod_{m \in M} R_m^{(S)} \tag{2.3}$$

Thus, the objective function of proportional fairness is also equivalent to the optimization of the *geometric mean of per-user throughput values*, which is the n^{th} root of the product of all MSs' throughput values, where n is the number of MSs.

Game Theory Viewpoint

The utility function approach used by Kelly [55] suffers from the disadvantages that user utilities or preferences are only known in some qualitative sense. Thus, although reasonable assumptions can be made on the behavior of utility functions, such an approach by itself still cannot put fairness definition on the foundation of a solid and precise mathematical framework. Another approach taken by Mazumdar et al. [73] is to consider measurable performance characteristics rather than abstract utility functions. In the context of elastic traffics, such a key metric is the allocated rate. They propose a game theoretic framework based on choosing this direct metric. Using the *Nash bargaining framework* from *cooperative game theory* [79], they show that *proportional fairness* is in fact a *Nash Bargaining Solution (NBS)* out of all Pareto Optimal points. *NBS* is the only equilibrium satisfying all four *axioms* as defined by Nash [79], namely: (1) invariance to affine transformations, (2) Pareto optimality, (3) independence of irrelevant alternatives, and (4) symmetry.

To summarize, *proportional fairness* criterion strikes a good balance between fairness and system efficiency, maximizes a reasonable overall utility function for elastic traffic, and satisfies the cooperative game theory axioms abstracted by Nash.

In a single-cell environment for both Wi-Fi networks [101] and cellular networks [11], the proportional fairness is implemented by allocating (asymptotically) the radio resource

(rather than bandwidth) of a BS equally among associated MSs, regardless of their different efficiency in using the resource, i.e., their various link data rates. If timely channel feedback is available, channel-aware opportunistic scheduling algorithms [11] are often employed to exploit the "multi-user diversity", as in the case of HDR network. In this work, we consider the time-averaged channel state as input, and assume that the underlying scheduling algorithm of each BS (which can be channel-aware) supports the resource allocation decision.

2.3.3 Minimum Potential Delay Fairness

Proportional fairness assumes the utility of a flow is a logarithmic utility function where the value of a flow increases with its allocated bandwidth R in proportion to $logR$. An alternative utility function with decreasing gradient is $-\frac{1}{R}$ as suggested by Massoulié and Roberts [71]. It leads to the bandwidth-sharing objective of minimizing the sum of the reciprocal of rates. This objective may alternatively be interpreted as minimizing the overall potential delay of the transfers in progress. Formally, *minimum potential delay fairness* can be written as:

Definition 2.3 ***Minimum Potential Delay Fairness*** *[71]: A feasible bandwidth allocation scheme S^* is called minimum potential delay fair if and only if:*

$$S^* = argmin_S \sum_{m \in M} \frac{1}{R_m^{(S)}} \tag{2.4}$$

where $R_m^{(S)}$ is the rate allocated to user m by scheme S, and M is the set of users.

In the example studied by Massoulié and Roberts [71], they show that this criterion is intermediate between the *max-min fairness* and *proportional fairness,* in that it penalizes more (less) severely resource-inefficient MSs than max-min (proportional) fairness, resulting in a larger (smaller) overall throughput. Our evaluations in Section 2.6.2 confirm this property.

Among all fairness definitions described above, our proposal is based on *proportional fairness*, because it is widely adopted in single-cell environment for both high data rate 3G network [11] and Wi-Fi network [101]. As discussed above, *proportional fairness* strikes a good balance between fairness and system efficiency. In addition, its cooperative game theory interpretation [73] puts it on the foundation of a solid and precise mathematical framework. We compare *proportional fairness* scheme with *max-min fairness* scheme and *minimum potential delay fairness* scheme in Section 2.6.2.

2.4 Coordinated Proportional Fairness

Fair scheduling in wireless networks is often considered in a single-cell context, while the joint routing-scheduling fairness formulation in wired networks cannot be directly applied to multi-cell wireless networks. In this section, we adopt *proportional fairness* as a resource allocation criterion suitable for elastic traffic in multi-rate wireless communication systems, and extend it to the general setting of overlapping cells from heterogeneous wireless networks, by defining the *coordinated proportional fairness (CPF)* allocation problem.

2.4.1 Formulation

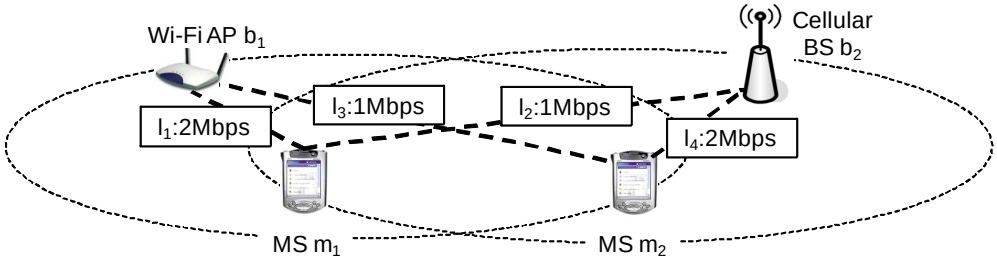
Consider a network with a set B of BSs and a set M of MSs. We let C_b be the finite radio resource capacity of BS b, for $b \in B$. Based on our system model as described in Section 2.2, C_b is fixed, and is independent of each other. We assume that each MS is equipped with sufficient number of radios, thus it can simultaneously associate with multiple neighboring BSs to achieve aggregate throughput. We will relax this assumption in Section 2.5.

Recall that a link $l = (m, b)$ represents an adjacent pair of MS and BS that are able to communicate with each other. Given a link l, we use $b(l)$ to denote the corresponding BS, and $m(l)$ to denote the corresponding MS. We write L for the set of all links. If $b = b(l)$,

we set A_{bl} to be the required radio resource in BS b to support per unit flow through link l. If the channel condition between $m(l)$ and $b(l)$ is poor, it can only support a low data rate, thus more radio resource is required to transfer a unit of flow, which implies a higher resource consumption rate, i.e., A_{bl} is larger. On the other hand, if a MS-BS link is under good channel condition, less resource is required to transfer the same amount of data, i.e., A_{bl} is smaller. As wireless channel state keeps changing with time, the value of A_{bl} used in our problem formulation is a time-averaged link state that is relatively stable for a decision period. For $b \neq b(l)$, we set $A_{bl} = 0$, because sending flow over link l does not consume any resource of BS b. This defines a matrix $A = (A_{bl}, b \in B, l \in L)$.

For a given MS m, its several links through different BSs may substitute for one another. Formally, suppose that a MS m has a subset of L. We write $H_{ml} = 1$ if $m = m(l)$, so that link l serves the MS m, and set $H_{ml} = 0$ otherwise. This defines a 0-1 matrix $H = (H_{ml}, m \in M, l \in L)$.

A flow pattern $y = (y_l, l \in L)$ supports the rates $x = (x_m, m \in M)$ if $Hy = x$, so that the flows over all links serving the MS m sum to the rate x_m.



(a)

$$\begin{array}{l} \mathrm{M} = \{\mathrm{m}_1, \mathrm{m}_2\}, \\ \mathrm{B} = \{\mathrm{b}_1, \mathrm{b}_2\}, \\ \mathrm{L} = \{\mathrm{l}_1, \mathrm{l}_2, \mathrm{l}_3, \mathrm{l}_4\}, \end{array} \quad A = \begin{bmatrix} \frac{1}{2} & 0 & 1 & 0 \\ 0 & 1 & 0 & \frac{1}{2} \end{bmatrix}, \quad H = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}, \quad C = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \Rightarrow \quad x = \begin{bmatrix} 2 \\ 2 \end{bmatrix}, \quad y = \begin{bmatrix} 2 \\ 0 \\ 0 \\ 2 \end{bmatrix}$$

(b)

Figure 2.2: CPF allocation example I

To illustrate the notations, we look at Figure 2.2 (a), which depicts the same setting as in the association game example in Section 1.3 of Chapter 1. We assume that the

capacities of both the Wi-Fi AP b_1 and the cellular BS b_2 are 1, thus $C = \begin{bmatrix} 1 & 1 \end{bmatrix}^T$. Each of MS m_1 and MS m_2 is equipped with both a cellular interface and a Wi-Fi interface. Both MSs locate in the overlapping coverage area of a Wi-Fi AP b_1 and a cellular BS b_2. However, their channel conditions to the Wi-Fi AP and the cellular BS are different because of their different locations. MS m_1 can communicate with the Wi-Fi AP at a link data rate of 2Mbps and with the cellular BS at a link data rate of 1Mbps, while MS m_2 can communicate with the Wi-Fi AP at a link data rate of 1Mbps and with the cellular BS at a link data rate of 2Mbps. There are 4 links corresponding to the 4 adjacent MS-BS pairs. We denote them as $l_1 = (m_1, b_1)$, $l_2 = (m_1, b_2)$, $l_3 = (m_2, b_1)$, and $l_4 = (m_2, b_2)$ respectively. The input to *CPF* allocation problem is: MS set $M = \{m_1, m_2\}$, BS set $B = \{b_1, b_2\}$, link set $L = \{l_1, l_2, l_3, l_4\}$, BS capacities $C = \begin{bmatrix} 1 & 1 \end{bmatrix}^T$, matrix $A = \begin{bmatrix} \frac{1}{2} & 0 & 1 & 0 \\ 0 & 1 & 0 & \frac{1}{2} \end{bmatrix}$, and matrix $H = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$. Note that the allocated bandwidth for a MS on a link equals to its link data rate only if the MS monopolizes the radio resource of the corresponding BS. Otherwise, the bandwidth of a MS over a link is the product of the link data rate and the portion of resource allocated by the corresponding BS.

A flow pattern y is feasible if $y \geq 0$ and $Ay \leq C$, so that the resource consumed by wireless links through a BS b sum to not more than its capacity. Based on our system model as described in Section 2.2, we assume that wireless transmissions are "orthogonal", thus resource consumed by different links at the same BS can be linearly summed up.

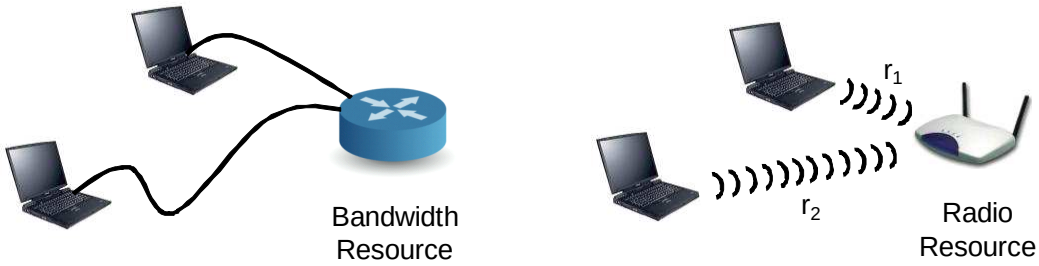
Formally, the **Coordinated Proportional Fairness (CPF) allocation** is the optimal solution for the following problem:

$$\begin{aligned} \textit{maximize} \quad & \sum_{m \in M} w_m log(x_m) \\ s.t. \quad & Hy = x, Ay \leq C \\ \textit{over} \quad & x, y \geq 0 \end{aligned} \tag{2.5}$$

where $w_m > 0$ is the weight assigned to different users representing their different priorities.

We consider only MSs with non-empty set of adjacent BSs, and BSs with non-empty set of adjacent MSs. Under this assumption, there are feasible allocations with the objective function bounded away from negative infinity, which implies that in the optimal solution, x_m for any MS m is bounded away from 0. Intuitively, not allocating *any* bandwidth to a connected user (despite its potentially poor channel condition) is considered unfair. Without affecting the calculation of the optimal solution, we can safely assume that each connected user m can get a minimum positive bandwidth allocation ε from the system, which translates to a lower bound $w_m log(\varepsilon)$ for m's utility function. We thus can define the utility function over the domain of $x \geq \varepsilon$ to ensure that the user's utility function is bounded from below. We can further add a constant value (e.g. $-w_m log(\varepsilon)$) to each user's utility function, such that its range is within the set of non-negative numbers. Note that, incorporation of additive constant values into the utility functions does not change the solution as defined in Equation 2.5.

The objective function is differentiable and strictly concave and the feasible region is compact. Thus, a maximizing value of (x,y) always exists and can be found by Lagrangian methods. There is a unique optimum for the rate vector x, since the objective function is a strictly concave function of x, but there may be many corresponding values of the flow rate y satisfying the constraints [75, 108].



(a) Wired network (b) Wireless network

Figure 2.3: Resource sharing in wired and wireless contexts

We briefly discuss the difference between our model and Kelly's original model for

wired networks [55].

As shown in Figure 2.3 (a), resources in wired networks are characterized directly in terms of bandwidth (such as a router's interface forwarding speed), and a resource can serve all routes passing through it with the same efficiency, so data rate and consumed resource can be treated equivalently. In contrast, to support per unit flow in wireless networks, different amount of radio resource (e.g., time slot, spectrum, power, or code words) is required due to location-dependent and time-varying channel condition, as shown in Figure 2.3 (b).

Compared to Kelly's original model for wired networks, our formulation changes the definition of A from a 0-1 matrix to a matrix with elements taking non-negative real values, to characterize the different link-dependent resource consumption rate in wireless networks.

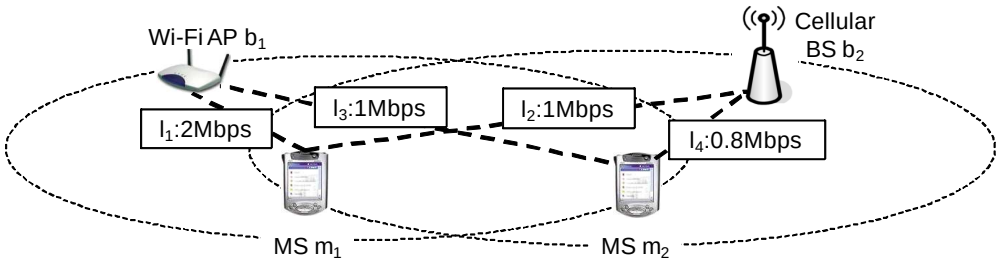
Note that, both *max-min fairness* and *minimum potential delay fairness* can be extended to multi-cell in a similar way [10, 53]. We call them *Coordinated Max-min Fairness* and *Coordinated Minimum Potential Delay Fairness*. We will compare the performance of these three coordinated fairness definitions in Section 2.6.2 and discuss the related work in Section 2.7.

2.4.2 Example

Let us look at the *CPF* allocation in the setting as shown in Figure 2.2.

Using Lagrangian method [108], the *CPF* solution for the given example is: $x = [2,2]^T$, $y = [2,0,0,2]^T$. The solution is Pareto-optimal. MS m_1 is served totally over link $l_1 = (m_1, b_1)$, and MS m_2 is served over link $l_4 = (m_2, b_2)$. Both m_1 and m_2 are assigned to their interface with more favorable channel condition, i.e., link with smaller resource consumption rate.

By considering fairness in a global sense (among all MSs), the *CPF* allocation solution automatically results in inter-cell load balance. For example, as shown in Figure 2.4, suppose the channel condition between MS m_2 and BS b_2 deteriorates, and supports



(a)

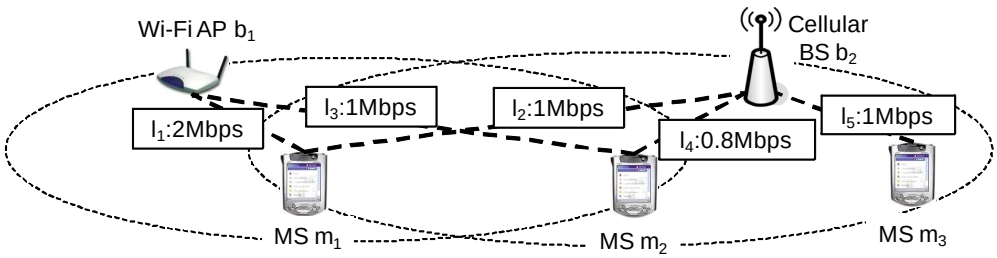
$$\mathrm{M}=\{\mathrm{m}_1,\mathrm{m}_2\},\quad \mathrm{B}=\{\mathrm{b}_1,\mathrm{b}_2\},\quad \mathrm{L}=\{\mathrm{l}_1,\mathrm{l}_2,\mathrm{l}_3,\mathrm{l}_4\},\quad \mathrm{A}=\begin{bmatrix}\frac{1}{2} & 0 & 1 & 0\\ 0 & 1 & 0 & \frac{1}{0.8}\end{bmatrix},\quad \mathrm{H}=\begin{bmatrix}1 & 1 & 0 & 0\\ 0 & 0 & 1 & 1\end{bmatrix},\quad C=\begin{bmatrix}1\\1\end{bmatrix}\ \Rightarrow\ x=\begin{bmatrix}1.8\\0.9\end{bmatrix},\quad y=\begin{bmatrix}1.8\\0\\0.1\\0.8\end{bmatrix}$$

(b)

Figure 2.4: CPF allocation example II

only a link data rate of $0.8Mbps$. BS b_2 becomes more congested than BS b_1, in the sense that BS b_2 requires extra capacity in order to support the original allocation as in Figure 2.2. The input for the *CPF* problem becomes $A = \begin{bmatrix} \frac{1}{2} & 0 & 1 & 0 \\ 0 & 1 & 0 & \frac{1}{0.8} \end{bmatrix}$, and matrix $H = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$ remains the same. The *CPF* solution becomes $x = [1.8, 0.9]^T$, and $y = [1.8, 0, 0.1, 0.8]^T$. The *CPF* allocation automatically shifts some load introduced by m_2 from b_2 to b_1. Note that, the resource-efficient MS m_1 has a higher throughput than the resource-inefficient MS m_2.

A third example is shown in Figure 2.5, where an additional active MS m_3 appears in the area covered only by BS b_2, thus making the traffic load even more asymmetric. We denote the new adjacent MS-BS pair as link $l_5 = (m_3, b_2)$. The matrix A becomes $\begin{bmatrix} \frac{1}{2} & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & \frac{1}{0.8} & 1 \end{bmatrix}$, and the matrix H becomes $\begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$. The *CPF* solution becomes $x = [1.2, 0.6, 0.75]^T$, and $y = [1.2, 0, 0.4, 0.2, 0.75]^T$. The *CPF* allocation shifts more load of m_2 from b_2 to b_1, to free more resource at b_2 to serve m_3.



(a)

$$\begin{array}{l} M=\{m_1, m_2, m_3\}, \\ B=\{b_1, b_2\}, \\ L=\{l_1, l_2, l_3, l_4, l_5\}, \end{array} \quad A=\begin{bmatrix} \frac{1}{2} & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & \frac{1}{0.8} & 1 \end{bmatrix}, \quad H=\begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad C=\begin{bmatrix} 1 \\ 1 \end{bmatrix} \Rightarrow x=\begin{bmatrix} 1.2 \\ 0.6 \\ 0.75 \end{bmatrix}, \quad y=\begin{bmatrix} 1.2 \\ 0 \\ 0.4 \\ 0.2 \\ 0.75 \end{bmatrix}$$

(b)

Figure 2.5: CPF allocation example III

Note that, under the *CPF* allocation, individual BS does not enforce time-based fair allocation among nodes associated with it. As shown in the second and third example, the *CPF* solution often requires a MS to be simultaneously assigned to multiple BSs. Further, the change of a single input parameter may change allocation decision for all MS-BS pairs. These factors need to be taken into consideration when implementing such a scheme in practice.

2.4.3 Incentive Compatibility

CPF allocation decision is based on the link state information of all adjacent MS-BS pairs. In practice, the link data rate is measured by individual MS, which periodically feeds it back to the common radio resource manager using CRRM functions for informed decision [1, 2, 33]. Thus, an intelligent and selfish MS can manipulate its reported link states, if it can gain more from the network by doing so.

Based on this observation, a *multi-cell resource allocation procedure* can be interpreted as a game, where each MS is a player. The strategy of a MS m can be described as a link data rate vector $R_m = (R_{mb}, b \in B)$, where R_{mb} gives the data rate supported be-

tween the MS m and a BS b. The resource allocation outcome is calculated according to the allocation scheme employed by the common radio resource manager, and individual BS enforces the decision. If the reported link data rate R_{mb} between the MS m and the BS b is not equal to the actual link data rate R^*_{mb}, the effective data rate will be less than R^*_{mb}. On one hand, if $R_{mb} < R^*_{mb}$, data are transferred by the BS using R_{mb}. On the other hand, if $R_{mb} > R^*_{mb}$, data are transferred by the BS at a rate higher than that can be fully decoded by the MS, the resulted effective data rate becomes lower than that can be achieved by the most appropriate rate R^*_{mb}. Note that, by collecting the link state vector R_m from each $m \in M$, the link vector L, matrix A and H required in calculating the *CPF* allocation can be derived accordingly. As over-report can be easily detected [117], we focus on the case where m may under-report its channel state, i.e. $R_{mb} \leq R^*_{mb}$.

Formally, a **multi-cell resource allocation game** is defined as $(M, R^*, \mathbb{R}, S, x)$, where

- M is the set of MS players.
- $R^* = (R^*_m, m \in M)$ consists of the actual link data rate vector R^*_m for each MS $m \in M$.
- $\mathbb{R} = \times_m \mathbb{R}_m, m \in M$, where $\mathbb{R}_m = \{R_m | R_m \leq R^*_m\}$ specifies the strategy space of MS m. m can choose any link data rate vector $R_m \in \mathbb{R}_m$ when playing the game.
- S is an allocation scheme that determines the allocation vector based on the specified channel state input $R \in \mathbb{R}$.
- $x = (x_m, m \in M)$ gives the allocated data rate vector.

Theorem 2.1 proves the positive result that in the *multi-cell resource allocation game* with *CPF* as the allocation scheme S, the dominant strategy for any MS is to report its channel state truthfully. We adopt the *algorithmic mechanism design* approach to analyze the game. As described in Section 1.3 of Chapter 1, *algorithmic mechanism design* [81, 82] studies optimization problems where the underlying data (the link data rates with neighboring BSs as measured by individual MS in our *multi-cell resource allocation*

game), is *a priori unknown* to the algorithm designer (the common radio resource manager in our game), and must be implicitly or explicitly elicited from selfish participants (through the periodic feedbacks of MSs using CRRM function in our game). The high-level goal is to design a mechanism (the allocation scheme in our game), that interacts with participants so that *selfish behavior yields a desirable outcome* (a fair and efficient resource allocation in our game). Recall that, a mechanism is said to be *incentive compatible*, or *strategy-proof*, if the dominant strategy of each participant under the designed mechanism is to truthfully reveal its state (each MS reports honestly its channel state in our game). In contrast, if a game is not incentive compatible, a MS can gain by cheating about its state, thus making the system operate under an inefficient state. Even worse, MSs may keep varying their behavior as response to others' strategies, which can lead to instability problem.

Theorem 2.1 *A multi-cell resource allocation game with CPF allocation scheme is incentive compatible.*

Proof: We prove this property by contradiction. We assume that there is a user m^* which can increase its aggregate bandwidth allocation by not using the truthful strategy. We denote the allocation decision for the original setting, where m^* does not cheat, as $D' = (x', y')$, and the allocation decision for the new setting, where m^* cheats, as $D'' = (x'', y'')$.

Given a MS m, we denote the subset of its adjacent BSs that allocate strictly more radio resource to it in D'' than in D' as $B^+(m)$, i.e., $\forall b \in B^+(m), \frac{y''_{(mb)}}{R''_{mb}} > \frac{y'_{(mb)}}{R'_{mb}}$.

Given a BS b, we denote the subset of its adjacent MSs that get strictly lower radio resource allocation from it in D'' than in D' as $M^-(b)$, i.e., $\forall m \in M^-(b), \frac{y''_{(mb)}}{R''_{mb}} < \frac{y'_{(mb)}}{R'_{mb}}$.

Denote the initial BS set as $B_0 = B^+(m^*)$. Based on our assumption, we have $x''_{m^*} > x'_{m^*}$. Thus, there must be some BSs which allocate more resource to m^* in D'' than in D'. More specifically, $B_0 \neq \emptyset$.

Denote the initial MS set as $M_0 = \cup_{b \in B_0} M^-(b)$. As a BS $b \in B_0$ allocates more

resource to m^* in D'', and in both solutions D' and D'' it allocates all of its resources, it must reduce allocation to some other MS in D''. Thus, $M_0 \neq \emptyset$.

Consider the Lagrangian form of the *CPF* problem:

$$\begin{aligned} & \mathbb{L}(x, y; \lambda, \mu) \\ = & \sum_{m \in M} w_m log(x_m) - \lambda^T (x - Hy) + \mu^T (C - Ay) \\ = & \sum_{m \in M} (w_m log(x_m) - \lambda_m x_m) + \\ & \sum_{l \in L} y_l (\lambda_{m(l)} - \mu_{b(l)} A_{b(l)l}) + \sum_{b \in B} \mu_b C_b \end{aligned} \tag{2.6}$$

where $\lambda = (\lambda_m, m \in M)$, $\mu = (\mu_b, b \in B)$ are vectors of Lagrange multipliers.

$$\frac{\partial \mathbb{L}}{\partial x_m} = (w_m log(x_m))' - \lambda_m \tag{2.7}$$

$$\frac{\partial \mathbb{L}}{\partial y_l} = \lambda_{m(l)} - \mu_{b(l)} A_{b(l)l} \tag{2.8}$$

Hence, at a maximum of $\mathbb{L}$, the following conditions hold:

$$\frac{w_m}{x_m} = \lambda_m \tag{2.9}$$

$$\begin{aligned} \lambda_{m(l)} &= \mu_{b(l)} A_{b(l)l} \text{ if } y_l > 0 \\ &\leq \mu_{b(l)} A_{b(l)l} \text{ if } y_l = 0 \end{aligned} \tag{2.10}$$

The Lagrange multipliers λ and μ have simple interpretations. We may view μ_b as the implied cost of using unit radio resource of BS b, or alternatively the shadow price of adding additional radio resource at BS b. λ_m can be viewed as the weighted charge of unit flow for MS m.

As $x''_{m^*} > x'_{m^*}$, because of Equation 2.9, $\lambda''_{m^*} < \lambda'_{m^*}$. Thus, for any $b \in B_0$, because of Equation 2.10, $\mu''_b < \mu'_b$. Based on Equation 2.10 again, for any $m \in M_0$, $\lambda''_m < \lambda'_m$, thus $x''_m > x'_m$.

We repeatedly carry out the following set expansion step:

$$B_{n+1} = \cup_{m \in M_n} B^+(m) \cup B_n \tag{2.11}$$

$$M_{n+1} = \cup_{b \in B_{n+1}} M^-(b) \tag{2.12}$$

As B is a finite set, the process always terminates at some $n = n^*$ where $B_{n^*+1} = B_{n^*}$. For each expansion step, the argument about the change of Lagrange multipliers as in the initial step can still be applied, thus: $x''_m > x'_m, \forall m \in M_{n^*}$.

Consider B_{n^*} and M_{n^*}. For any MS $m \in M_{n^*}$, its allocated data rate strictly increases. For any MS $m \notin M_{n^*}$, its radio resource allocation from any BS $b \in B_{n^*}$ is not reduced according to the definitions above. Thus, BSs in B_{n^*} jointly allocate higher data rate in D'' to all MS $m \in M_{n^*}$ without affecting their allocation to any MS outside M_{n^*}. Combining the resource allocation decision of D'' for BSs in B_{n^*} and the allocation decision of D' for BSs not in B_{n^*}, we have a feasible allocation solution $\tilde{x}$ for the original setting where m^* is honest. For MS m^*, $\tilde{x}_{m^*}$ is the aggregate rate of m^* using the actual link data rate, thus, we have $\tilde{x}_{m^*} \geq x''_{m^*} > x'_{m^*}$. For $m \in M_{n^*}$ and $m \neq m^*$, their reported data link rates are the same for the two settings, thus $\tilde{x}_m \geq x''_m > x'_m$. Similarly, for $m \notin M_{n^*}$, $\tilde{x} \geq x'$. As the vector $\tilde{x}$ is strictly larger than x', this contradicts with the fact that x' is Pareto optimal under the original setting where m^* is honest. □

2.5 Integral Coordinated Proportional Fairness

The optimal solution for the *CPF* allocation often requires MSs to be simultaneously assigned to multiple BSs, which may not be desirable in practice, due to the following reasons:

- It requires a node to be equipped with multiple simultaneous active radios. On one hand, a software defined radio that can dynamically switch to different radio access technologies may not satisfy such requirement, as it cannot simultaneously present

in multiple overlapping cells. On the other hand, turning on multiple radios can significantly increase the power consumption.

- When a single parameter changes in the network, the allocation decision may be adjusted globally. This may result in both system instability and excessive signaling overhead.

- Transport protocol at client may have difficulty to efficiently aggregate bandwidths from multiple interfaces, especially when the allocated bandwidth of each interface varies with time [45].

Thus, in this section, we study resource allocation schemes in a *single-association setting*, which associates each MS with a single BS.

2.5.1 Formulation and Complexity

The formulation for the *CPF* allocation can be modified to reflect the additional constraint in single-association setting.

Formally, the **Integral Coordinated Proportional Fairness (Int-CPF) allocation** is the optimal solution for the following problem:

$$\begin{aligned} \textit{maximize} \quad & \sum_{m \in M} w_m log(x_m) \\ s.t. \quad & Hy = x, Ay \leq C \\ & \forall m \in M, \exists l_m \in L, \forall l \neq l_m, H_{ml} y_l = 0 \\ \textit{over} \quad & x > 0, y \geq 0 \end{aligned} \tag{2.13}$$

If we decouple the solution for *Int-CPF allocation* scheme into the *inter-cell association control layer* and the *intra-cell scheduling layer*, we observe that, given its allocated MSs, the strategy for a BS in the second layer to optimize the defined objective function, is independent of the association control decision in the first layer and the second layer

strategy of each other. This is because one MS is served by a single BS, thus, every single BS should maximize the weighted logarithmic sum of data rates of the MSs assigned to it, and it can achieve this by employing individual proportional fairness scheduling. As the second level scheduling is clear, the remaining problem is to decide for each MS which BS it should associate to.

We show that, for *Int-CPF allocation* scheme, there does not exist an algorithm that can find the optimal solution in polynomial time unless $P = NP$, i.e., the problem is NP-hard. Similar to Lenstra et al. [61] and Bu et al. [16], our reduction is via *3-dimensional matching* problem that is known to be NP-complete. The *3-dimensional matching* problem is stated as follows.

Definition 2.4 *Let $X = \{x_1, \ldots, x_n\}$, $Y = \{y_1, \ldots, y_n\}$, $Z = \{z_1, \ldots, z_n\}$ be three disjoint sets with identical size n, and T is a subset of $X \times Y \times Z$. That is, T consists of triples (x, y, z) such that $x \in X$, $y \in Y$, and $z \in Z$. A $T' \subseteq T$ is a 3-dimensional matching if $|T'| = n$ and $\cup_{t_i \in T'} t_i = B \cup C \cup D$. The problem is to find whether such a T' exists.*

Theorem 2.2 *Int-CPF allocation problem is NP-hard.*

Proof: Consider a *3-dimensional matching* problem where T consists of k triples ($k > n$) and $\cup_{t_i \in T} t_i = B \cup C \cup D$, otherwise the problem becomes trivial. We construct a corresponding *Int-CPF allocation* problem as follows. For each tripe $t_i \in T$, we create a corresponding BS t_i with capacity 1. We create two types of MSs: normal MS and privileged MS. For each element $m \in X \cup Y \cup Z$, we create a corresponding normal MS m. There are totally $3n$ normal MSs. A normal MS m is covered by a BS t_i if and only if $m \in t_i$. In addition, we create $k - n$ privileged MSs, which are covered by all BSs. We assume that the link data rates of all adjacent MS-BS pairs are equal to a constant R. A normal MS has weight 1, and a privileged MS has weight $w_p > 2$. The weight of privileged MS is selected such that, if possible, packing the $3n$ normal MSs into n BSs, while assigning each of the $k - n$ privileged MSs into each of the rest of $k - n$ BSs, gives the highest value of $U_{max} = \sum_{m \in M} w_m log(x_m) = 3nlog(\frac{R}{3}) + (k-n)w_p log(R)$.

Thus, it is easy to verify that if there is a 3-dimensional matching solution T', the *Int-CPF allocation* problem achieves the optimal solution U_{max}. Conversely, if the *Int-CPF allocation* problem achieves the optimal solution U_{max}, there is a *3-dimensional matching* solution for the original problem. □

Suppose $w_m = 1, \forall m \in M$. If we know the congestion vector $(N_b, b \in B)$ for the optimal solution of *Int-CPF* problem, where N_b denotes the number of MSs assigned to a BS b, we can reduce the problem of finding the optimal solution for *Int-CPF allocation* to finding the maximum weight perfect k-matching in a bipartite graph as follows. Consider the bipartite graph $G(M, B, L)$ where M denotes the MSs, B denotes the BSs, and L is the set of adjacent edges. The requirement (k-value) of each MS $m \in M$ is 1. For a BS $b \in B$, its requirement $k(b) = N_b$. The weight on each edge (m, b) is set to $w(m, b) = log(\frac{R_{mb}}{N_b})$, where R_{mb} is the link data rate between MS m and BS b. The optimal *Int-CPF allocation* corresponds to the maximum weight perfect k-matching, as each MS is associated with one BS, each BS gets the number of MSs as specified by the optimal congestion vector, and the logarithmic sum of allocated data rates for all MSs is maximized. Note that the number of possible congestion vectors is polynomial in the number of MSs $|M|$ and can be enumerated if the number of BSs $|B|$ is a constant. In our evaluation, we use this approach to calculate the solution of *Int-CPF allocation* for a constant number of BSs.

2.5.2 Incentive Compatibility

In contrast to the multi-cell allocation game with *CPF allocation* scheme, the multi-cell allocation game with *Int-CPF allocation* scheme is not incentive compatible.

Theorem 2.3 *A multi-cell resource allocation game with Int-CPF allocation scheme is not incentive compatible.*

Proof: The theorem can be easily proved by providing counter examples.

In the example of Figure 2.6 (a), there is a MS m_1 covered by both a Wi-Fi AP b_1 and a cellular BS b_2, with link data rate of $0.9Mbps$ and $2Mbps$ respectively. In addition,

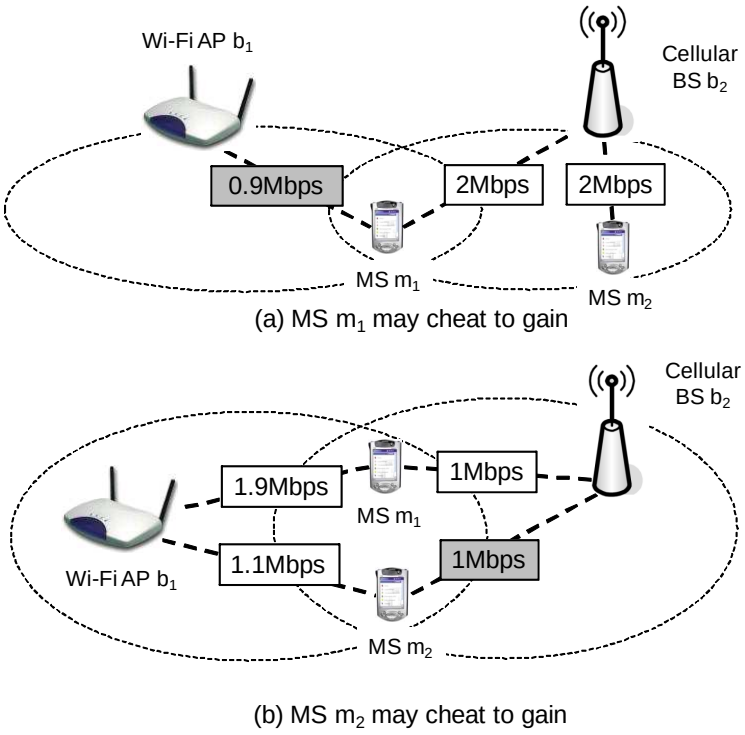


Figure 2.6: Cheating under Int-CPF allocation

there is a MS m_2 which is covered only by BS b_2 with link data rate of $2Mbps$. Recall that each BS's capacity is fixed, and the bandwidth allocated to a MS on a link is the product of the link data rate and the fraction of the radio resource allocated by the corresponding BS. The *Int-CPF allocation* is to assign MS m_1 to b_1, and MS m_2 to b_2, thus m_1 gets an allocation of $0.9Mbps$, and m_2 gets an allocation of $2Mbps$. However, if m_1 cheats by hiding its association with b_1, i.e., set the data rate of link (m_1, b_1) to 0, the *Int-CPF allocation* is to assign both m_1 and m_2 to b_2, and allocate half the resource to each of them. In this case, m_1 gets a higher throughput of $1Mbps$, whereas m_2's throughput is reduced to $1Mbps$.

This example shows that a MS with multiple adjacent BSs can manipulate its adjacent BS set, so as to be allocated to its favored BS.

On the other hand, the example of Figure 2.6 (b) shows that, a MS can also manipulate its reported data rate to increase its benefit by changing other MS's association. In the given setting, both MS m_1 and MS m_2 are covered by both the Wi-Fi AP b_1 and the

cellular BS b_2. Their link data rates to the AP and the BS are shown in the figure. It is easy to verify that, the *Int-CPF allocation* is to assign m_1 to b_1 and m_2 to b_2, such that the throughput of m_1 is $1.9Mbps$ and the throughput of m_2 is $1Mbps$. If m_2 cheats by hiding its adjacency with b_2, i.e., set the data rate of link (m_2, b_2) to 0, the *Int-CPF allocation* will swap the assignment, with m_2 associated to b_1 and m_1 to b_2. Thus, the throughput of m_2 increases to $1.1Mbps$. □

The example of Figure 2.6 (b) also shows that both the optimal *Integral Coordinated Max-min Fairness* [10] and the optimal *Integral Coordinated Minimum Potential Delay Fairness* [71] are not incentive compatible.

As we can scale the data rate such that the aggregate utility in the optimal solution is strictly greater than 0, while the aggregate utility in a *Nash equilibrium* is 0, the *price of anarchy*, which is defined as the ratio between the optimal social utility and the utility of the worst Nash equilibrium point of the game is unbounded. For example, in Figure 2.6 (a), the optimal social utility is $log(0.9) + log(2) > 0$, whereas the social utility under the Nash equilibrium point is $log(1) + log(1) = 0$.

2.5.3 Selfish Load Balancing: Congestion Game

As *Int-CPF allocation* decision is computationally expensive to solve, and does not define an incentive-compatible game, a natural alternative is to let selfish users decide for themselves which BS to associate with.

When each MS can make individual association decision directly, instead of the *multi-cell resource allocation game* as defined in Section 2.4.3, we have a *single-association game*.

Formally, a **single-association game** is defined as (M, S, x), where

- M is the set of MS players.
- $S = \times_m S_m$ denotes the set of all possible ways in which players can pick strategies. For each player $m \in M$, S_m denotes its own set of possible strategies, which corre-

sponds to the subset of BSs with which it can associate. In particular, one strategy $b_m \in S_m$ corresponds to the association of MS m with BS b_m. A *strategy profile* $s \in S$ consists of the vector of each player's selected strategy, i.e. $s = (b_m, m \in M)$.

- Under the assumption that each BS implements individual *proportional fairness* scheduling, and that all users have the same weight, the throughput x_m received by a MS m under a strategy profile $s = (b_m, m \in M)$ can be simply expressed as:

$$x_m(s) = \frac{R_{mb_m}}{N_{b_m}(s)} \tag{2.14}$$

where R_{mb_m} is the link data rate between MS m and its selected BS b_m, and $N_{b_m}(s)$ is the congestion level (number of associated users including MS m) of BS b_m under strategy profile s. Given the freedom to decide its own association, a player m has no incentive to cheat about its link data rate, as doing so only decreases its actual throughput.

For each player in the *single-association game*, its reward (in terms of allocated bandwidth) of employing a certain strategy is affected only by the number of other players who employ the same strategy (choosing the same BS to associate with), rather than who they are. Thus, this game falls into the class of *congestion games* which is first introduced by Rosenthal [92].

Rosenthal shows that if the cost (or reward) function is the same for all players choosing the same strategy, then these games possess a rich structure, in particular they always have a Nash equilibrium in pure strategies. The term of "pure strategy" means each player *deterministically* plays a single chosen strategy, instead of randomly picking among multiple strategies. This result follows from the existence of a *potential function*, which is a real-valued function defined over the set of strategy profiles having the property that the gain (or loss) of a player shifting to a new strategy is equal to the corresponding change of the potential function.

The existence of an exact potential function implies the *finite improvement property*

(FIP): Any sequence of strategy-tuples in which each strategy-tuple differs from the preceding one in only one coordinate (such a sequence is called a *path*), and the unique deviator in each step strictly increases its payoff (an *improvement path*), is finite. The first strategy-tuple of a path is called the *initial point*; the last one is called the *terminal point*. Obviously, any *maximal improvement path*, an improvement path that cannot be extended, is terminated by a Nash equilibrium.

Milchtaich [74] extends the definition of *congestion game* to allow player-specific cost (or reward) functions, i.e. different players have different costs (or rewards) by choosing the same strategy, and shows that even these games have a pure Nash equilibrium.

In our setting, different MSs have different wireless link data rate with the same BS, thus the reward function is player-specific. However, the simple structure of the player-specific reward function as defined in Equation 2.14 allows us to prove a stronger result than Milchtaich.

More specifically, Theorem 2.4 shows that the *single-association game* possesses the *finite improvement property (FIP)*. To prove FIP, we define for every strategy profile $s = (b_m, m \in M)$ the following potential function:

$$\Phi(s) = \sum_{m \in M} log(R_{mb_m}) - \sum_{b \in B} log(N_b(s)!) \tag{2.15}$$

where R_{mb_m} is the link data rate between MS m and its selected BS b_m, and $N_b(s)$ gives the number of MSs allocated to a BS b under the strategy profile s. The potential function is constructed such that the gain (or loss) of a player shifting to a new strategy is equal to the corresponding change of the potential function, as shown in the proof below. Similar construction has also been used by Gairing et al. [41] to analyze a delay minimization congestion game with user-specific cost function.

Theorem 2.4 *Single-association game possesses the finite improvement property.*

Proof: Consider a selfish step $s \rightarrow s'$ where a player $m \in M$ switches from BS b to

BS b'.

$$\begin{aligned}\Phi(s') - \Phi(s) &= (log(R_{mb'}) - log(N_{b'}(s)+1)) - (log(R_{mb}) - log(N_b(s))) \\ &= log(x_m(s')) - log(x_m(s)) \end{aligned} \tag{2.16}$$

□

Based on the result, we can formally describe the *Selfish Load Balancing (SLB)* scheme as follows. Under *SLB* scheme, the common radio resource manager starts from a feasible allocation decision, and greedily switches a MS to a BS that can improve its throughput. Only one MS is switched at a time, thus when there are multiple MSs that can improve by unilaterally switching association, *SLB* scheme selects one of them. The iteration ends until a *Nash equilibrium* is reached, i.e., no user can unilaterally change its association to achieve a higher throughput.

To make the presentation more concrete, we choose the following strategies when implementing *SLB* scheme: (1) We use the popular heuristic of *Strongest-Signal First (SSF)* allocation scheme as the initial allocation vector. In *SSF* allocation scheme, a MS is associated with the BS that provides the strongest signal strength. The decision is made regardless of the BS's load. (2) Given a selected MS to switch, if there are multiple BSs that can improve its allocated rate, we assign the selected MS to a BS that can increase its allocated bandwidth by the largest percentage. Draws are settled randomly. (3) If there are multiple MSs that can gain by unilaterally switching association, we select one of them randomly. Note that, the result of Theorem 2.4 does not rely on the choices made by our implementation. The convergence property holds for any strategy fitted into the general framework.

Note that, there can be multiple Nash equilibria in the *single-association game*, and *SLB* scheme can converge to any of them. For example, in Figure 2.6 (b), there are two Nash equilibria. In the first equilibrium, MS m_1 is associated to BS b_1, and MS m_2 is associated to BS b_2. In the second equilibrium, the associations are swapped. Individual

MS can have significantly different bandwidth allocation under different Nash equilibria. Note that although *SLB* scheme converges, it is not incentive compatible. For example, MS m_2 in Figure 2.6 (b) can hide its association with BS b_2, so as to make the system converge to the second equilibrium instead of the first one.

Despite the fact that it is not incentive compatible, *SLB* scheme is still a valuable solution, as no MS can gain by unilaterally changing its association. In addition, our evaluation in Section 2.6 shows that *SLB* scheme converges quickly and performs close to *Int-CPF* scheme. It remains an interesting research problem to design incentive-compatible resource allocation schemes for single-association setting, such that MSs cannot gain by cheating, while system can operate in a fair and efficient state.

2.6 Evaluation

2.6.1 Methodology

Our evaluation is based on a customized flow level simulator. The two metrics we consider are *arithmetic mean of per-user throughput values* and *geometric mean of per-user throughput values*. On one hand, *arithmetic mean of per-user throughput values* is the sum of all MSs' throughput divided by the number of MSs. It reflects the overall performance of the system. Note that although a higher *arithmetic mean* implies higher aggregated throughput for all MSs, resource sharing can be very unfair among them. Thus, we look at the *geometric mean of per-user throughput values*, i.e., the n^{th} root of the product of all MSs' throughput, where n is the number of MSs. Measure using *geometric mean* presents a better trade-off between efficiency and fairness, as a single starved MS makes the *geometric mean* equal to 0.

In Section 2.6.2, we first compare the performance of the three coordinated fairness definitions, i.e., *Coordinated Proportional Fairness*, *Coordinated Max-min Fairness*, and *Coordinated Minimum Potential Delay Fairness*. Section 2.6.3 compares the performance of the following six schemes that are based on *Proportional Fairness*. For allocation

schemes that can split a user's flow among multiple interfaces, we consider:

- *Coordinated Proportional Fairness (CPF)* scheme, which is the optimal solution for the convex programming problem as formulated in Equation 2.5. *CPF* scheme gives the upper bound of the geometric mean of per-user throughput values.

- *Uncoordinated Proportional Fairness (UPF)* scheme, where a MS associates to all neighboring BSs by simultaneously turning on multiple radio interfaces. This is the Nash equilibrium for the association game as discussed in Chapter 1, and represents a non-cooperative scenario where all users are selfish and the system is uncoordinated.

For allocation schemes under single-association constraint that enforces each MS to associate with only a single BS, we consider:

- *Int-CPF* scheme, which is the optimal solution for the integral optimization problem as formulated in Equation 2.13. The problem is proved to be NP hard. However, for a relatively small number of BSs and constant weight, we are able to find the optimal solution by iterating through all feasible congestion vector combinations to find the optimal value among all resulted maximum weighted perfect k-matching solutions. Note that, we assume that all MSs honestly report their channel states and association information.

- *Selfish Load Balancing (SLB)* scheme, which starts from *SSF* allocation, and allows user greedily switch BS to improve its own throughput. User switches in a random order, and the switching user selects a BS allocating the highest rate. The iteration ends until a Nash equilibrium is reached, i.e., no user can unilaterally change its association to achieve a higher throughput. Theorem 2.4 establishes the convergence of such a process.

- *Strongest-Signal-First (SSF)* scheme, which always associates a MS to the BS with the strongest received signal strength, regardless of its load. *SSF* scheme is the

default association method for multiple radio access technologies, including Wi-Fi networks.

- *Least-Population-First (LPF)* scheme, which always associates a MS to the BS with the least number of associated MSs, regardless of the channel condition. *LPF* scheme is a classical method for load balancing, especially in single-rate cellular networks.

For fair comparison, we assume that each BS implements individual proportional fairness scheduling for all schemes above, except for the case of *CPF* scheme, which decides for each BS its allocation vector, thus does not necessarily follow the individual proportional fairness scheduling.

As illustrated in Figure 2.7, our simulation is based on a $600m \times 600m$ *torus* topology where 9 BSs are placed on a 3 by 3 grid, with the distance between two adjacent BSs set to 200 meters. All BSs have identical transmission power and operate on non-interfering channels. The maximum transmission range of a BS is set to 150 meters. The set $B(m)$ of BSs covering a MS m are determined from MS m's location by examining whether its distance to a BS is within 150 meters. We have conducted evaluations for two user distributions:

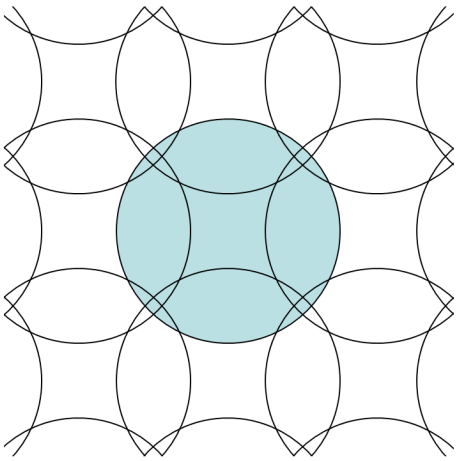


Figure 2.7: A torus BS topology

- In *uniform setting*, users are distributed within the torus uniformly at random;

- In *hot spot setting*, out of all MSs generated, $\kappa = 90\%$ are randomly positioned in a circle-shape hot spot with the radius of 150 meters around the center of a selected *hot BS*, as indicated by the shadow area in Figure 2.7.

The percentage of MSs covered by different number of overlapping BSs is shown in Table 2.1 for the two settings. Both the average and 90% confidence interval (CI) are shown.

	Uniform		Hot spot	
	Mean	90% CI	Mean	90% CI
$P(\|B(m)\|=1)$	29.1%	17.8% - 40.4%	17.9%	8.9% - 26.7%
$P(\|B(m)\|=2)$	65.9%	53.3% - 77.8%	73.7%	64.4% - 84.4%
$P(\|B(m)\|=3)$	4.3%	0% - 8.9%	6.9%	0.2% - 13.3%
$P(\|B(m)\|=4)$	0.7%	0% - 2.2%	1.5%	0% - 4.4%
$E[\|B(m)\|]$	1.77	1.64 - 1.9	1.92	1.77 - 2.07

Table 2.1: Overlapping coverage statistics

The arrival of MSs follows a Poisson process, and the sojourn time of a MS in the system follows an exponential distribution, both of which are assumed to be independent of MSs' allocated throughput for simplicity[3]. $\rho = \frac{E[|M|]}{|B|}$ is defined as the average number of active MSs in the system divided by the number of BSs, with default value set to 5. We use the log-normal shadowing propagation model to calculate the received signal strength at MS from each of its adjacent BSs. Given the distance $d < 150m$ between a MS m and a BS b, the received signal power $P_{dB}(d)$ from b at m is calculated as:

$$P_{dB}(d) = P_{dB}(d_0) - 10\beta log_{10}\frac{d}{d_0} + X_\sigma \tag{2.17}$$

where $d_0 = 10m$ is the reference distance, $\beta = 3$ is the path loss exponent, and X_σ is a Normal random variable in dB having a standard deviation of $\sigma = 12dB$ and zero mean. The parameters are set to model the typical loss in an urban environment [89]. We

[3]We note that it takes longer time for a MS with lower bandwidth to download some given amount of information, however, a MS with lower bandwidth also tends to download less amount of content. We leave the study of MS behavior to future research.

set the Signal Noise Ratio (dB) within reference distance to $P_{dB}(d_0) - P_{dB}(N_0) = 35dB$, and use a threshold-based mapping as shown in Table 2.2 to determine the link data rate accordingly. The selected values are commonly used in 802.11b network [5, 87, 110]. Table 2.3 shows the statistic of the link data rates among adjacent BS-MS pairs. Both the average and 90% confidence interval (CI) of the probability are presented.

	$SNR < 3dB$	$3dB \leq SNR < 8dB$	$8dB \leq SNR < 15dB$	$15dB \leq SNR$
Rate	1Mbps	2Mbps	5.5Mbps	11Mbps

Table 2.2: Mapping between Signal Noise Ratio and link data rate

	Uniform		Hot spot	
	Mean	90% CI	Mean	90% CI
$P(R=1Mbps)$	41.5%	32.5% -50.5%	44.3%	35.0% - 53.6%
$P(R=2Mbps)$	14.9%	7.9% - 21.9%	15.3%	8.5% - 22.1%
$P(R=5.5Mbps)$	18.6%	11.8% - 25.4%	18.4%	11.1% - 25.7%
$P(R=11Mbps)$	25.0%	16.3% - 33.7%	22.0%	14.0% - 30.0%
$E[R]$ (Mbps)	4.48	3.67 - 5.29	4.18	3.48 - 4.88

Table 2.3: Link data rate statistics

2.6.2 Comparison of Various Coordinated Fairness Definitions

This section compares the performance of the three fairness definitions as discussed in Section 2.3 when they are applied in a multi-cell environment, namely *Coordinated Proportional Fairness (CPF)*, *Coordinated Max-min Fairness*, and *Coordinated Minimum Potential Delay Fairness*.

As shown in Table 2.4, for uniform setting, under the *Coordinated Max-min Fairness*, the arithmetic mean of per-user throughput values is less than 30% of *CPF*, and the geometric mean of per-user throughput values is around 40% of *CPF*. The per-user throughput performance of the *Coordinated Minimum Potential Delay Fairness* allocation scheme is intermediate between the *Coordinated Max-min Fairness* scheme and the *Coordinated Proportional Fairness* scheme, in terms of both arithmetic mean and geometric mean. Similar phenomenon is observed in hot spot setting as well.

		Uniform		Hot spot	
		Arith. Mean	Geo. Mean	Arith. Mean	Geo. Mean
Proportional	average	1.26	0.89	0.93	0.57
	90% CI	1.06 - 1.46	0.67 - 1.11	0.64 - 1.22	0.45 - 0.69
Max-min	average	0.36	0.36	0.23	0.23
	90% CI	0.26 - 0.46	0.26 - 0.46	0.17 - 0.29	0.17 - 0.29
Minimum Potential Delay	average	0.89	0.79	0.68	0.51
	90% CI	0.69 - 1.09	0.62 - 0.96	0.47 - 0.89	0.39 - 0.63

Table 2.4: Throughput (Mbps) comparison of different coordinated fairness definitions

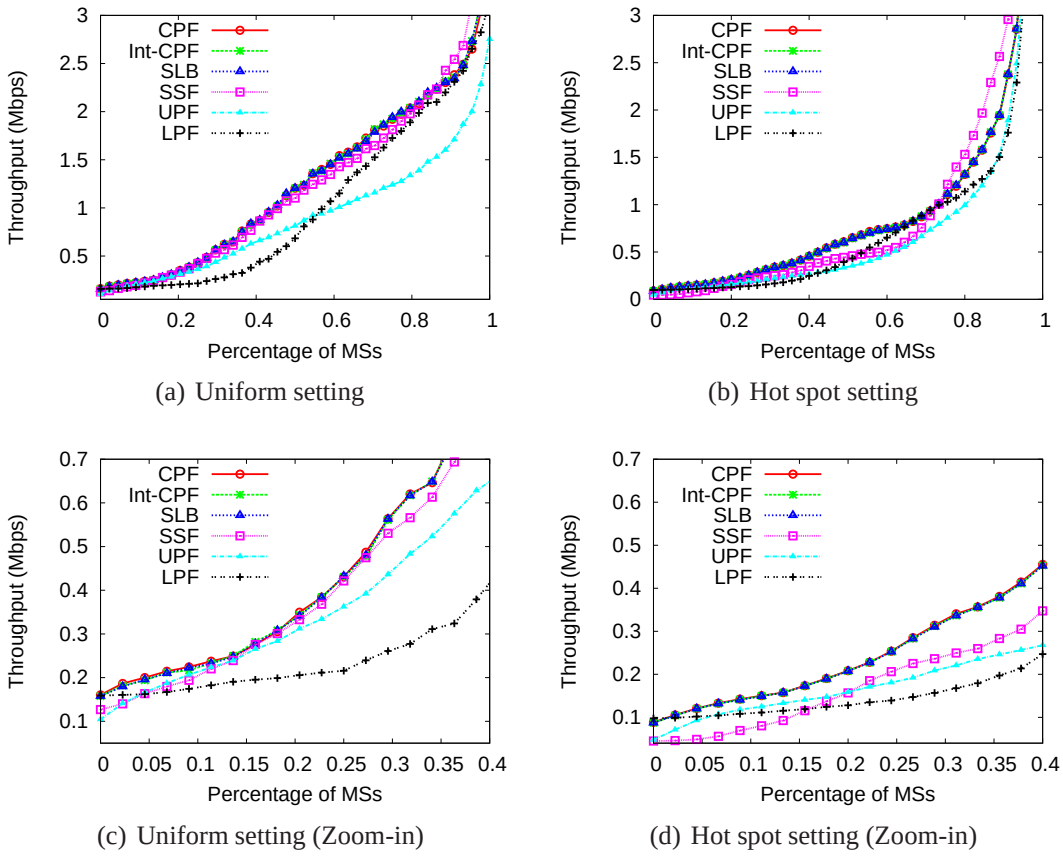
2.6.3 Performance of Various Schemes

This section compares the performance of the six schemes adopting Proportional Fairness, namely *CPF*, *UPF*, *Int-CPF*, *SLB*, *SSF*, and *LPF*.

Figure 2.8 (a) plots the per-user throughput values sorted in non-decreasing order under the uniform setting, and Figure 2.8 (b) plots the result under the hot spot setting. Figure 2.8 (c) and (d) provide a zoom-in view of MSs with low bandwidth allocation for the two settings. Table 2.5 summarizes the *arithmetic* and *geometric* mean (and the 90% confidence interval) of per-user throughput values under different schemes for the two settings respectively.

		Uniform		Hot spot	
		Arith. Mean	Geo. Mean	Arith. Mean	Geo. Mean
CPF	average	1.26	0.89	0.93	0.57
	90% CI	1.06 - 1.46	0.67 - 1.11	0.64 - 1.22	0.45 - 0.69
UPF	average	0.89	0.67	0.73	0.39
	90% CI	0.72 - 1.06	0.52 - 0.82	0.49 - 0.97	0.30 - 0.48
Int-CPF	average	1.26	0.89	0.92	0.56
	90% CI	1.06 - 1.46	0.67 - 1.11	0.63 - 1.21	0.45 - 0.67
SLB	average	1.26	0.89	0.92	0.56
	90% CI	1.06 - 1.46	0.67 - 1.11	0.63 - 1.21	0.45 - 0.67
SSF	average	1.29	0.86	0.98	0.45
	90% CI	1.06 - 1.52	0.64 - 1.08	0.63 - 1.33	0.31 - 0.59
LPF	average	1.02	0.63	0.75	0.42
	90% CI	0.78 - 1.26	0.48 - 0.78	0.49 - 1.01	0.31 - 0.53

Table 2.5: Arithmetic and geometric mean of per-user throughput values (Mbps)



(a) Uniform setting (b) Hot spot setting

(c) Uniform setting (Zoom-in) (d) Hot spot setting (Zoom-in)

Figure 2.8: Per-user throughput values sorted in non-decreasing order

Coordinated Proportional Fairness (CPF) scheme produces the optimal *geometric mean of per-user throughput values*. This is as expected, because the objective function of the optimization problem defined in Equation 2.5 can be transformed to the geometric mean of per-user throughput without affecting the solution. Note that, despite that *Strongest-Signal-First (SSF)* scheme often allocates higher throughput than *CPF* scheme to MSs with high bandwidth allocation (as shown in the right region of Figure 2.8 (a) and (b)), it provides lower throughput for MSs with low bandwidth allocation, especially for hot spot setting as shown in Figure 2.8 (d). Because of this unfairness, its geometric mean of per-user throughput is lower than *CPF* scheme.

In contrast to *CPF* scheme, *Uncoordinated Proportional Fairness (UPF)* scheme performs much worse, providing arithmetic/geometric mean of per-user throughput not only lower than *CPF* scheme, but also inferior to all other schemes except for *LPF* scheme in

some cases. This observation holds for both uniform and hot spot settings. The significant performance gap between *CPF* and *UPF* strongly advocates the adoption of a coordinated resource allocation approach for an integrated environment.

Among all allocation schemes for the single-association setting, *Int-CPF* scheme has the optimal geometric mean of per-user throughput, which agrees with the optimization problem defined in Equation 2.13. Its performance is close to *CPF* scheme under both uniform setting and hot spot setting. For both settings, the coordinate-wise performance gap between the two schemes is never greater than 3.5%, and is less than 1% for more than 70% of MSs. Our result also shows that, *Selfish Load Balancing (SLB)* scheme often has very close performance to *Int-CPF* scheme, thus to *CPF* scheme as well. For around 65% of user distribution in uniform setting, *SLB* scheme and *Int-CPF* scheme make the identical association decision. For 99% of user distributions, the performance gap between the two schemes is less than 1%. Similar phenomenon is observed under hot spot setting as well. In fact, such an approximation among *SLB* scheme, *Int-CPF* scheme, and *CPF* scheme holds when we vary the traffic load and asymmetry, as demonstrated later in Figure 2.9 and Figure 2.10. The only case that we observe obvious difference between *CPF* scheme and the two single-association schemes is when the average number of MSs per BS is very small (e.g. ≤ 3).

Based on this, we make the following observation: *By using an (appropriate) single radio per user, the system can largely achieve the performance when simultaneously using multiple radios per user.*

Among all six schemes, *Strongest-Signal-First (SSF)* scheme achieves the highest *arithmetic mean of per-user throughput*. This is because *SSF* scheme greedily assigns each MS to the BS providing the best channel condition. However, *SSF* scheme's geometric mean of per-user throughput is lower than *CPF* scheme, *Int-CPF* scheme, and *SLB* scheme, because a MS often associates to an overloaded BS that can only allocate a small portion of its overall radio resource to serve the MS, thus provide low throughput despite of the high link data rate between them. This situation is particularly common

in hot spot setting. As shown in Figure 2.8 (d), for hot spot setting, nearly 15% of MSs have throughput lower than 100 Kilobit per second (Kbps) under *SSF* scheme, compared to less than 1% under *CPF* scheme. For the 15% of MSs with lowest bandwidth allocation, their throughput under *SSF* scheme is less than 60% of their throughput under *CPF* scheme. Thus, *SSF* scheme can be unfair to a significant portion of users.

Least-Population-First (LPF) scheme often performs worst in terms of both arithmetic and geometric mean of per-user throughput, implying that traditional load balancing technique is not applicable to multi-rate wireless data networks.

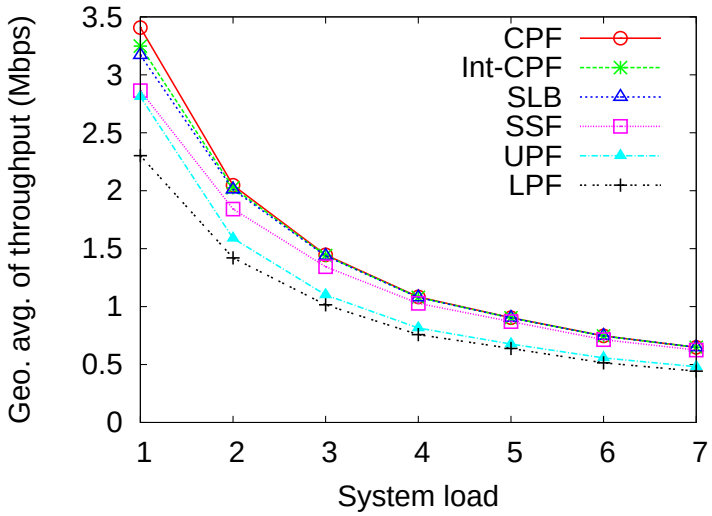


Figure 2.9: Geometric mean of throughput (Mbps) over varying load

Figure 2.9 demonstrates different schemes' performance under varying load in uniform setting. The performance of *Int-CPF* scheme and the performance of *SLB* scheme are close to each other in all range of load. Further, the performance gap between they and the *CPF* scheme reduces with increased traffic intensity. This is because both *Int-CPF* scheme and *SLB* scheme allocate resource on a per MS basis. Hence, the larger the traffic load, the finer the relative granularity of them. In fact, the only case that we observe obvious difference between *CPF* scheme and the two single-association schemes is when the average number of MSs per BS is very small (e.g. ≤ 3)

Figure 2.10 demonstrates the impact of asymmetric traffic distribution. The figure shows that the performance of both *Int-CPF* scheme and *SLB* scheme is close to *CPF*

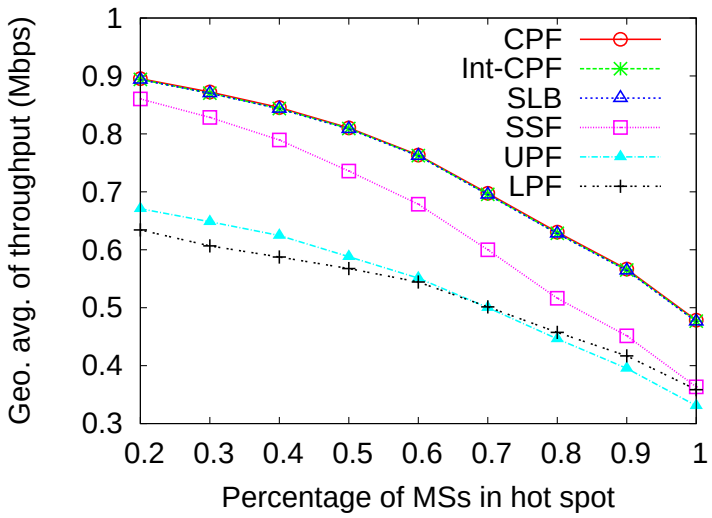


Figure 2.10: Geometric mean of throughput (Mbps) over varying traffic distribution asymmetry

scheme even under highly asymmetric traffic distribution. Such robustness is largely because both schemes take into account both network load and link data rate. In comparison, performance of *SSF* scheme deteriorates faster than all other schemes with increasing traffic asymmetry, and *LPF* scheme performs better than *UPF* scheme under high traffic asymmetry.

2.6.4 Strategic Interactions under SLB and Int-CPF

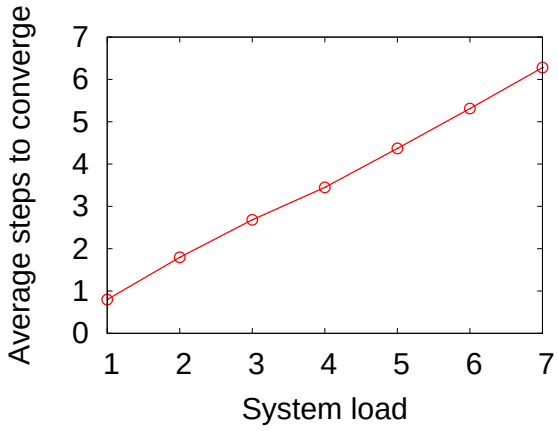


Figure 2.11: Convergence speed of SLB over varying load

Figure 2.11 shows the average number of steps required in the whole multi-cell sys-

tem for *SLB* scheme to converge to a Nash equilibrium, starting from the *SSF* allocation. As can be seen from the figure, *SLB* scheme converges quickly, and the number of steps required grows linearly with the system load.

While *SLB* scheme takes strategic interactions among MSs into consideration, *Int-CPF* scheme simply ignores them. Our evaluation shows that, up to 15% of decision made by *Int-CPF* scheme is not a Nash equilibrium in the single-association game. More specifically, there is at least one MS which can unilaterally change its association to gain higher throughput from the network under *Int-CPF* allocation. As illustrated in Figure 2.6 (a), the user can cheat by hiding all of its adjacent BSs except for its desired BS, so as to affect the *Int-CPF* resource allocation decision and increase its own bandwidth.

We observe that, in more than 30% of settings, there are multiple Nash equilibria in the induced *single-association game*, meaning that there is at least one MS that can cheat about its channel state to drive the system to the Nash equilibrium that it prefers.

2.7 Related Work

For cellular networks, schemes that dynamically balance loads among neighboring cells have been proposed, including directed retry (DR) and directed handoff (DH) [34, 40, 52]. The proposed schemes take advantage of the fact that some MSs may be able to obtain sufficient signal strength from multiple cells. With DR scheme, if a call finds its first-attempted cell has no free channel, it will try for a free channel in any other cell that can provide sufficient signal strength. The DH scheme takes this idea further, in that when a cell has all or almost all of its channels in use, it may, using DH scheme, direct some of its MSs to attempt handoff to an adjacent cell, with the goal to redistribute calls in heavily loaded cells to lighter loaded cells. Both schemes can improve system performance. The ratio of improvement depends on the percentage of MSs that can communicate with two or more cells simultaneously, which has been reported to be as high as 30-45 percent by Everitt [34]. However, these schemes are designed for voice calls, thus often assume that

each call consumes a fixed amount of radio resource.

Bianchi and Tinnirello [14] observe that in wireless communication systems, effective transmission rate depends on the channel quality, thus admitted calls weight unevenly in terms of effective resource consumption. They suggest using channel quality information to drive load balancing mechanisms and propose two metrics, “Gross Load” and “Packet Loss”, to quantify the information related to packet level retransmission load. Using the proposed metrics, they determine the best cell to attach to, during handover or new request arrival. Their simulation results show the superiority of their proposed schemes with respect to the *Least-Population-First (LPF)* load-balancing scheme. Sang et al. [95] propose a cross-layer framework to coordinate packet-level scheduling, flow-level cell selection and handoff, and system-level loading balancing based on the load, throughput, and channel measurements at different layers. In their proposed framework, an opportunistic scheduling algorithm, the *weighted Alpha-Rule*, exploits multiuser diversity gain in each cell independently, while providing minimum rate guarantees for MSs. Each MS adapts to its channel dynamics and the load fluctuations in neighboring cells, in accordance with MSs’ mobility and their arrivals or departures, by initiating load-aware handoff and cell selection. The central server adjusts the scheduling parameters of each cell to coordinate cells’ coverage, or cell breathing, by prompting distributed MS handoffs. Across the whole system, BSs and MSs constantly monitor their load, throughput, or channel quality in order to facilitate the overall system coordination. However, both works are designed for applications with stringent Quality of Service (QoS) requirement, such as voice calls, which demand a specified amount of bandwidth. Instead, we focus on elastic traffic, which can adapt to and make full use of various bandwidth allocation. While the major metric to be optimized by Bianchi and Tinnirello [14] and Sang et al. [95] is the blocking rate of MS (or more generally, the probability of not satisfying a user’s QoS requirement), we aim at a globally fair and efficient allocation decision for elastic traffics. Because of these differences, we cannot make direct comparison with them.

Bejerano et al. [10] consider the problem of achieving network-wide *max-min fair-*

ness using association control for Wi-Fi networks. The max-min time fairness problem they consider is intended for single-rate Wi-Fi networks only. Das et al. [27] consider scheduling schemes in which scheduling decisions are made jointly for a cluster of cells, thereby enhancing performance through both interference avoidance and dynamic load balancing. They consider algorithms for two scenarios. In the first scenario, they assume complete knowledge of the instantaneous channel quality information from each of the BSs to MSs at the centralized scheduler. In the second scenario, they propose a two tier scheduling strategy that assumes only the knowledge of the long-term channel conditions at the centralized scheduler. They demonstrate that significant throughput gains can be obtained in the case of asymmetric traffic distribution, whereas the gains in the symmetric case are modest. Since the load balancing is achieved through centralized scheduling, their scheme can adapt to time-varying traffic patterns dynamically. Both works adopt *max-min fairness* as the criterion for bandwidth allocation. *Max-min fairness* is not suitable for elastic traffic in multi-rate wireless networks, because it can severely affect the efficiency of the system as shown in Section 2.6.2.

In a parallel work with similar approach, Bu et al. [16] formulate the *generalized proportional fairness* problem in third generation (3G) wireless data networks, by considering proportional fairness in a network-wide context. However, their formulation is specific to HDR networks. In HDR networks, through a signaling channel, each user feeds back its channel condition continuously to the *proportional fairness* scheduler at the BS with which it associates. At each time slot, the scheduler at each BS schedules the user with the largest weight where the weight is the link data rate of the user at the current time slot divided by the average rate it has received so far. Instead, we consider general wireless networks, which may consist of heterogeneous radio access technologies (thus, opportunistic scheduling may not be feasible at all). Li et al. [63] consider the *generalized proportional fairness* problem in multi-rate Wi-Fi networks. Their technique is to intelligently associate users with APs to achieve optimal proportional fairness in a network of APs. They propose two approximation algorithms with a constant worst-

case guarantee for the NP-hard problem and demonstrate that the algorithms can obtain both higher aggregate throughput and better fairness than the *Strongest-Signal-First (SSF)* AP selection method in the 802.11b standard. The proposed schemes in both works are essentially approximations to our *Int-CPF* scheme. Their evaluations also show that (approximated) *Int-CPF* scheme outperforms common heuristics like *SSF* scheme and *LPF* scheme. However, both works do not consider the incentive compatibility issues.

Kauffmann et al. [53] also consider the fairness among MSs in a network-wide context. They propose the use of *minimum potential delay fairness* [71] as the optimization goal for user association control. Our simulation results in Section 2.6.2 show that this criterion is intermediate between the *max-min fairness* and *proportional fairness*, in that it results in a larger (smaller) overall throughput than max-min (proportional) fairness.

There are also works [57, 93] considering the use of multiple orthogonal channels in wireless mesh networks, where each router is equipped with multiple radios. They focus on the channel assignment algorithms to maximize throughput over multi-hop path. The basic theme is to mitigate interference among contending links in a multi-hop path by assigning them to different channels. In contrast, our work focuses on how to allocate the bandwidth of the single-hop downlink from a BS to a MS. We only consider multi-mode MS and assume that the capacity of BS is fixed. In our system model, the backhaul links of BSs are over provisioned with different technologies (e.g. using wired networks), and each BS operates orthogonally with each other. The techniques as proposed in multi-channel wireless mesh networks can potentially be used to extend our existing works, e.g., to model the situation that the backhual links of BSs use the same radio technology as the BS-MS link.

2.8 Summary

This chapter studies the *coordinated radio resource allocation problem* for users that are simultaneously covered by multiple overlapping cells using heterogeneous radio access

technologies. We formulate the *coordinated proportional fairness (CPF)* resource allocation criterion, based on which a globally fair and efficient allocation decision can be easily computed. As *CPF* decision depends on the input from users, a selfish user may manipulate its channel state report if doing so can increase its gain from the network. To capture this phenomenon, we formulate the resource allocation process as a *multi-cell resource allocation game*, which is associated with a rule to calculate bandwidth allocation outcome based on the input from the MS players. We prove that *CPF* allocation is incentive compatible, in the sense that a user's dominant strategy is to report its channel state honestly. In practice, the single-association setting, where each MS is associated with a single BS, is often desirable. We formulate the integral version of the *CPF* problem (*Int-CPF*) and show that it is both computationally expensive and prone to user-manipulation. Alternatively, we advocate the adoption of a *Selfish Load Balancing (SLB)* scheme, which always leads to a Nash equilibrium, and often achieves performance near to the *CPF* allocation. We use simulation to evaluate the performance of proposed schemes. The results show that the proposed algorithms outperform popular heuristic approaches, by striking a good balance between efficiency and fairness, while achieving load balancing among component BSs.

Chapter 3

MobTorrent: Cooperative Access for Delay-Tolerant Mobile Users

3.1 Introduction

For commuters and passengers on public buses, taxis or private vehicles, the most common and seamless way of getting Internet access is through the use of Wireless Wide Area (Cellular) Networks, e.g., GPRS, 3G or HSDPA. The cellular radio can be plugged into the end host (e.g. a laptop) or mounted on the vehicle from which shared network access is provided to all passengers in the vehicle using an on-board Wi-Fi network [1]. However, even though performance of cellular networks has improved significantly over the years, in particular with the deployment of HSDPA, the aggregate or per user data rate is still limited by the need to provide ubiquitous coverage to a large number of users. In a recent measurement, we observe around 300Kbps download speed from a vehicle (with a 1.5Mbps-limit subscription plan) using a local commercial HSDPA network.

Meanwhile, many cities around the world have witnessed large-scale deployment of open Wi-Fi hotspots. In Singapore, more than 7000 free Wi-Fi access points (APs) have

[1] A shared on-board network enables passengers without cellular subscription (and interface) to access Internet. In addition, a hand-held device can reduce its power consumption by using the short-range wireless communication instead of connecting to a remote BS directly. Further, the powerful and well-positioned vehicle antenna helps improve the wireless communication efficiency with the remote BS.

	Cellular networks	Wi-Fi networks
Bandwidth R	Low - Medium 56Kbps (GPRS) - 14Mbps (HSDPA)	Medium - High 1Mbps (802.11) - 600Mbps (802.11n)
Coverage P	Ubiquitous	Intermittent (e.g. 20%)
$R \times P$	56Kbps - 14Mbps	200Kbps - 120Mbps

Table 3.1: Complementary characteristics of cellular networks and Wi-Fi networks

been deployed in the last few years in public open areas, shopping malls and commercial buildings. On a smaller scale, in a measurement of our 150 hectares campus in Kent Ridge, we can observe more than 2000 APs installed in 90 buildings. Strong Wi-Fi signal can be received from about 25% of the 4km route traveled by the campus shuttle bus. Recent research works [21, 84] have also demonstrated the feasibility of providing network access via roadside APs.

On the other hand, Singapore being a city-state, has a dense deployment of public buses. The largest public transport provider has a fleet of more than 2000 buses. Currently, almost all buses are equipped with GPS and GPRS device. While the bandwidth provided by GPRS is sufficient for its main application, an Automatic Vehicle Location (AVL) system, it is too low to support Internet access service for passengers. Upgrading the whole system to HSDPA is costly. With such a large number of open Wi-Fi APs available already, providing network access to moving vehicles through roadside Wi-Fi APs offers an *alternative* and *complementary* solution that can significantly increase the bandwidth available to the vehicles.

Heterogeneous mobile broadband access architecture for commuters has been suggested previously [17, 91], where multiple network interfaces (e.g. 3G and Wi-Fi) are available and can be utilized simultaneously. The concept of Always Best Connection (ABC) is often adopted, where mobile nodes automatically start to use the Wi-Fi network as soon as an AP is in range. While Wi-Fi provides higher bandwidth at cheaper price, it is only usable when the vehicle is in range and the contact duration is often short. In comparison, although the speed of cellular link is lower, it has higher availability.

Table 3.1 gives some example figures for the coverage and bandwidth of both networks. As described earlier in Chapter 1, both Wi-Fi and cellular networks keep evolving to meet the increasing demands of mobile users. Meanwhile, other complementary / competitive technologies, such as WiMax, are continuously introduced to the market. Despite the fact that the various forms of technology advances (e.g. MIMO) can significantly increase the network capacity, spectrum efficiency, and data rate, the tradeoff between cell coverage (or communication distance) and factors such as spatial reuse, data rate, deployment cost is fundamental. Thus, we believe that there will be a long-term coexistence of two forms of networks, i.e., high-bandwidth networks with intermittent coverage, and lower-bandwidth networks with higher coverage, regardless of the actual technologies being used. For example, the high-bandwidth intermittent network can be in the form of femtocells using LTE (Long Term Evolution) technology instead of Wi-Fi hotspots as discussed here.

Our work focuses on delay-tolerant applications, such as downloading some large files (e.g. movie) from Internet. Thus, we are interested in the average throughput during a long time period, in the scale of dozens of minutes, which can be expressed by multiplying R, the bandwidth when in connection, with P, the probability of being connected. In terms of this criterion, Wi-Fi networks provide comparable performance as, or even higher performance than cellular networks. For example, a Wi-Fi contact lasting 10 seconds (a typical contact length in our measurement on a campus bus testbed) with an average data rate of $11 Mbps$ can transfer more than $13 MegaByte$ of data (the typical size of a 5-minute movie clip or four songs in mp3 format). In comparison, it takes a cellular network with $300 Kbps$ bandwidth more than 5 minutes to complete the same transfer. In addition, Wi-Fi networks are generally cheaper than cellular networks. The much larger number of Wi-Fi APs compared to cellular BSs makes Wi-Fi networks scale better with the number of users in terms of the network capacity.

In order to fully exploit the available high-bandwidth but intermittent contacts, we propose *MobTorrent* [25], an on-demand, user-driven framework designed to optimize

performance for vehicular network. The approach taken by *MobTorrent* is different from existing works in that we use the cellular network mainly as a control channel. We also assume that the mobility information can be predicted with high accuracy using AVL system and history. In this framework, a mobile client, instead of waiting for contact with the AP, uses the cellular radio (e.g. GPRS) to inform one (or multiple) selected AP(s) to prefetch the content. The prefetched data are then replicated on the mobile helpers, and further propagated by the latter in a store-carry-forward, i.e., Delay-Tolerant Network (DTN) routing fashion. As a result, instead of limiting high-speed data transfers to a few short contact periods with the selected APs, high-speed transfers among vehicles can be opportunistically exploited.

While *MobTorrent* exploits prefetching and replication, the key component is the scheduling algorithm, which replicates the prefetched data by taking into account locations of the mobile nodes and existing level of data replication. The objective is to maximize the total amount of data transferred and the average transfer rate to the mobile clients.

In this work, we first characterize the performance limits of opportunistic mobile forwarding through a simple scenario using only one AP. The insight gained is then used to design the scheduling scheme for inter-vehicle transmission. In the evaluation, we use testbed measurement to verify the benefit of prefetching and use trace-driven simulation to evaluate performance of scheduling. Our results show that *MobTorrent* provides substantial improvement over existing architecture and often performs close to what can be achieved by an off-line optimal scheduler. In case of multiple APs, our evaluation results show that *MobTorrent* is robust in a variety of settings.

The rest of this chapter is organized as follows. In Section 3.2, we describe the architecture of *MobTorrent*. In Section 3.3, we discuss scheduling issues and analyze the performance gain. In Section 3.4, we evaluate the performance of *MobTorrent*. In Section 3.5, we present related work. We conclude in Section 3.6.

3.2 System Model

3.2.1 Components

In order to deploy *MobTorrent*, we require wide adoption of GPS devices on vehicles (e.g., Japan's vehicle navigation system installation rate is estimated to be as high as 59%, while Europe and the United States are around 25% [90]). In addition, each vehicle must be equipped with both Wi-Fi interface and cellular interface. We use the cellular network mainly as a control channel, so the existing low-bandwidth GPRS suffices. Vehicles are expected to have an estimation of their travel route. This can be obtained from historical values or based on locations and digital street maps. We believe that all of these are reasonable requirements, in particular, for public buses and vehicles that travel on regular routes.

The components of *MobTorrent* are shown in Figure 3.1.

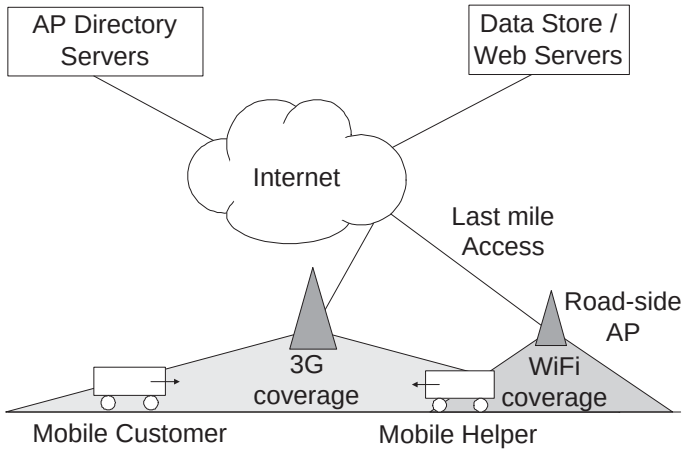


Figure 3.1: MobTorrent framework

- **Mobile Clients** are vehicles that require help to download data from data store / web servers through Internet.
- **Road-side APs** are static Wi-Fi access points reachable from the road. They have Internet backhaul, and offer their services to mobile clients. They can be residential gateways in apartments or installed as part of a vehicle network infrastructure

that is placed along the road, say at bus stops, taxi stands, or traffic lights. Note that the backhaul downlink bandwidth to these APs can be lower than the wireless bandwidth available on the 802.11 link. For example, in a residential home, the downlink speed could be a few Mbps or less, whereas the average Wi-Fi bandwidth can be over 20Mbps for 802.11g and much higher for 802.11n.

- **AP Directory Servers** provide location information on available roadside APs. There are a number of open Wi-Fi AP locators available on the web already, including *http://www.openwifispots.com*, *http://www.fon.com*, and *http://www.whisher.com*. The locations of participating APs need to be in the form of coordinates given in longitude and latitude, which can be easily found even without GPS by using digital street maps. Depending on the system requirements, these servers can also maintain information related to the AP's reputation and performance. For scalability purpose, it is likely that the servers are clustered into different geographical regions.

- **Mobile Helpers** are idle vehicles willing to offer their bandwidth to help peer vehicles with downloading demand.

3.2.2 Control and Data Flow

In this section, we describe a typical operation in *MobTorrent* data transfer, as illustrated in Figure 3.2.

Initially, a mobile client wants to download a (sufficiently large) file. Note that small requests are assigned to the always-on cellular link to minimize their delay. By downloading large files via Wi-Fi, the cellular link can be less congested, which benefits the small requests too. The mobile, with its location known through a GPS device, acquires the list of APs that are along its travel path. Based on a number of parameters (file size, location of AP, estimated travel time to the AP), the node selects a set of APs and contacts them through its cellular interface. For example, as shown in Figure 3.2, 2 data blocks are needed, while AP $A1$ and $A2$ are selected.

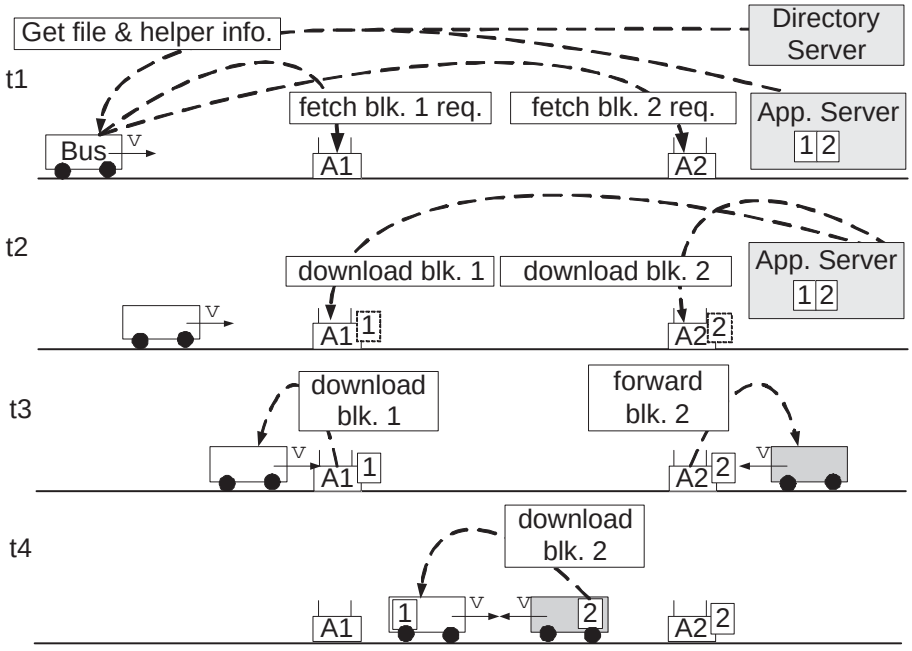


Figure 3.2: MobTorrent data downloading process

At time t_1, the mobile client requests $A1$ to prefetch block 1 and $A2$ to prefetch block 2, and the two APs begin to download the respective blocks.

At time t_2, the blocks are downloaded to the corresponding APs and cached locally.

At time t_3, the mobile client travels within the range of $A1$ and downloads block 1 from $A1$. At the same time, the mobile helper, a second bus moving towards the mobile node, enters the coverage of $A2$. $A2$ sends block 2 to the mobile helper.

At time t_4, the mobile client and the mobile helper meet at some point between $A1$ and $A2$. The mobile helper transfers block 2 to the mobile client, thus completing the transfer even before the mobile client reaches $A2$.

In order to efficiently orchestrate the whole downloading process, two questions need to be answered: (1) How much data should an AP prefetch? (2) How to relay the data to a client via mobile helpers, so that the amount is maximized and the delay is minimized? We answer these questions in the next section.

3.3 Scheduling in MobTorrent

3.3.1 Roles and Functions of Different Mobile Helpers

Before presenting the scheduling algorithm in *MobTorrent*, we first draw insight from how opportunistic relay should work in a simple mobility model on a 2-way street. This model abstracts the major properties for some typical settings, such as commuters on highway, and public buses with fixed routes within a city.

We consider a relatively sparse vehicle network, where the probability of forming a contemporaneous multi-hop path is negligible, so single hop forward in each contact opportunity is the main form of data transfer. Vehicles are assumed to move on a long, 2-way street without divergence in the path. A vehicle moves in one of the two directions (LEFT or RIGHT) on the road and it never changes its direction. In addition, there is no overtaking among the vehicles moving in the same direction. We focus on the case of a single AP in the model.

We define an *opportunistic contact* as the time period that two peers get in communication range, and can exchange data with each other directly. To fully describe a contact between two nodes, we need to record the contact start time and how long it lasts, as well as the varying link data rate available at each time point during this interval. To keep our discussion succinct, we define the notation of **contact capacity**, which is the amount of data that can be exchanged in a contact, i.e. the product of the average data rate and the length of the contact. Note that the two directions of transfer processes compete for the same contact duration, so the contact capacity limits the sum of the volumes that can flow in the two directions. We denote the start time of the contact period as the **contact time**. As we assume that the network is relatively sparse, and a single node's different contacts do not overlap with each other, the start time of the contact is sufficient for the purpose of ordering the contacts for a single node according to the sequence they happen.

Given the time and the client's location when the request is generated, we can categorize all mobile helpers into the following classes based on their moving directions and

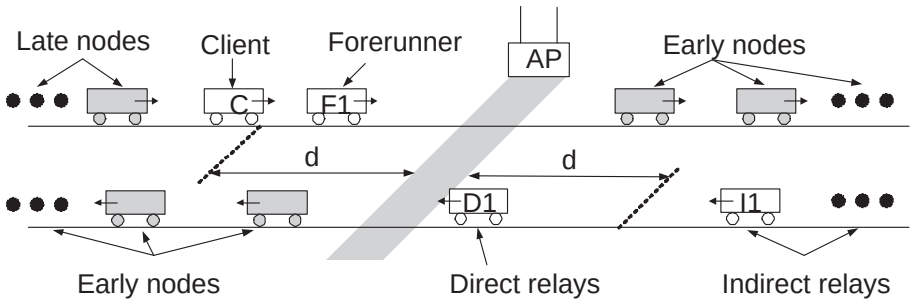


Figure 3.3: Classes of helpers

positions (relative to the client and the AP) as shown in Figure 3.3:

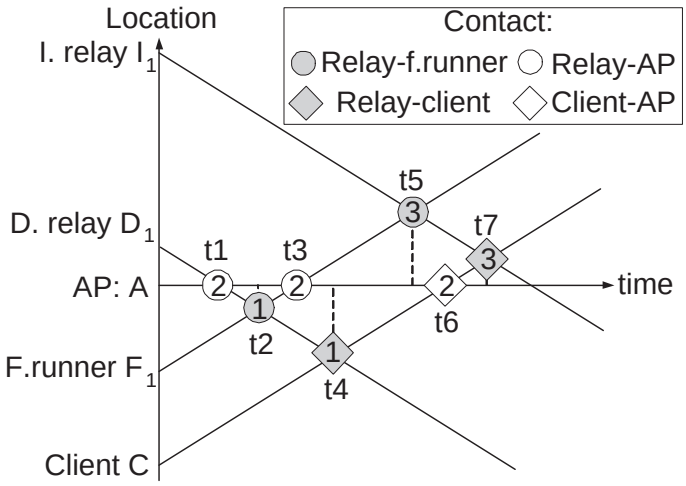
- **Direct relay:** mobile nodes that move towards the client and meet the AP after the request is generated but before they meet the client. As the name suggests, a direct relay can get data from the AP, carry it, and *directly* send to the client.

- **Forerunner:** mobile nodes that move in the same direction as the client and meet the AP after the request is generated but before the client meets the AP.

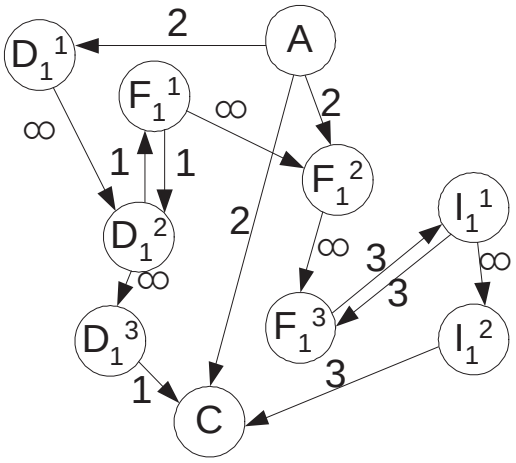
- **Indirect relay:** mobile nodes that move towards the client and meet the client before they meet the AP. If every node moves at the same velocity, the condition to distinguish between direct relay and indirect relay is to compare its distance to AP, denoted as d_r, with the distance d between client and AP at the same time. As shown in Figure 3.3, if $d_r \leq d$, the relay is a direct relay, otherwise, it is an indirect relay. Note that when an indirect relay meets the AP, there is no need for the AP to send data to it, as it will not meet the client or any node that will meet the client from then on.

The client attaches its mobility trajectory in the request, thus each node can determine its role for the given request according to its local information about its own mobility trajectory. They also learn about the mobility trajectory and roles of other helpers through inter-vehicle contacts. Further details are provided later in Section 3.3.4.

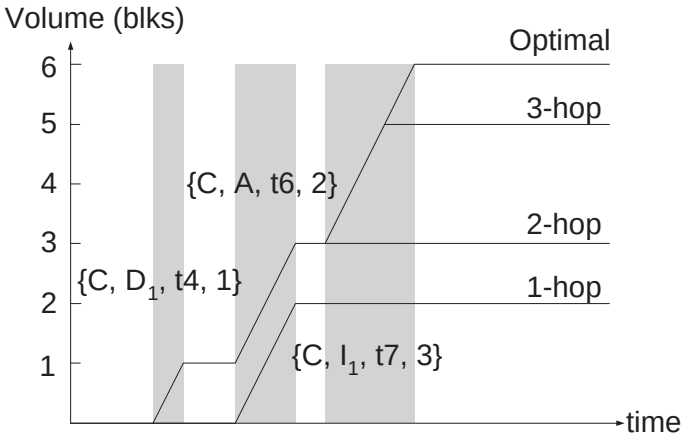
The set of direct relays, forerunners, and indirect relays are denoted as D, F, I respectively. No other mobile nodes can help the transfer for the following reasons:



(a) Time series trace



(b) Network flow formulation



(c) Performance of schemes

Figure 3.4: A simple two-way street example

- **Early node:** mobile nodes that have moved past the AP before the request is generated. They cannot help the data delivery because they cannot receive data directly from the AP, or from any vehicle carrying the desired data.

- **Late node:** mobile nodes that are always behind the client. They cannot help the data delivery because they cannot send data directly to the client, or to any mobile node that will meet the client after meeting them.

An example is given in Figure 3.4 (a), which shows the traces of vehicles and their contacts along the time axis. As explained above, we succinctly denote a contact using 4 parameters, namely $\{$*node1, node2, contact time, contact capacity*$\}$. As shown in Figure 3.4 (a), at time $t1$, AP (A) meets the first direct relay (D_1) (as their trace intersects), with contact capacity 2, which is the number marked at the point of intersection. This contact can be represented as: $\{A, D_1, t1, 2\}$. Similarly, we can write down the rest of contacts from the figure in their sequence as: $\{D_1, F_1, t2, 1\}$; $\{A, F_1, t3, 2\}$; $\{D_1, C, t4, 1\}$; $\{F_1, I_1, t5, 3\}$; $\{C, A, t6, 2\}$; $\{I_1, C, t7, 3\}$.

3.3.2 Performance Limits

In this section, we derive the performance limits by examining an off-line scheduler, which is assumed to have the information about the complete contact trace through some oracle.

The two performance metrics of interest are (1) the maximum amount of data sent by the AP that reaches the client eventually, and (2) the minimum delay to deliver a given amount of data. We assume that there is sufficient buffer on all nodes to accommodate packets in transfer, and the only bottleneck is the contact capacity constraint between nodes.

Denote the contact time and contact capacity between node i and node j by t_j^i and c_j^i respectively. t_j^i and c_j^i are subject to the random fluctuations like traffic jams and network congestion. In practice, such information is only revealed on-line. Thus, the performance

of an off-line optimal scheduler serves as an upper bound for what can be achieved by an on-line scheduling scheme. We consider the uncertainties in t_j^i and $\mathcal{C}_j^i$ as resolved in the following discussions of off-line scheduling performance.

Maximum Data Transfer from AP to Client

The maximum amount of data $\mathcal{C}$ that the AP (denoted as node *A*) can push to the network and stand a chance to reach client (denoted as node *C*) is:

$$\mathcal{C} = \mathcal{C}_C^A + \sum_{i \in D} \mathcal{C}_i^A + \sum_{j \in F} \mathcal{C}_j^A \tag{3.1}$$

$\mathcal{C}$ is the sum of three parts: $\mathcal{C}_C^A$ is the amount that can be transferred directly to the client by the AP, $\sum_{i \in D} \mathcal{C}_i^A$ is the amount that can be transferred to all direct relays (thus stand a chance to reach the client) by the AP, and $\sum_{j \in F} \mathcal{C}_j^A$ is the amount that can be transferred to all forerunners by the AP. Sending data to the rest of nodes (the indirect relay and late nodes) is useless. Note that, under our mobility model, all data stored in forerunner will eventually reach client, as we assume that there is an infinite flow of relays from the opposite direction. However, if a direct relay cannot replicate all of its data to the client or to some forerunner in time, the unfinished data will become lost permanently when the direct relay travels past the client. Thus, the capacity estimated in Equation (3.1) often cannot be achieved. To minimize such loss, the direct relay replicates its data to forerunner as soon as possible, so that even if it cannot send the data to the client by itself, the data can still be forwarded later via forerunner and other relays. Based on this observation, a tighter upper bound for the capacity achievable by the AP is:

$$\mathcal{C}_C^A + \sum_{i \in D} min(\mathcal{C}_i^A, \mathcal{C}_C^i + \sum_{t_i^j > t_i^A, j \in F} \mathcal{C}_j^i) + \sum_{j \in F} \mathcal{C}_j^A \tag{3.2}$$

The second part in Equation 3.2, i.e., $\sum_{i \in D} min(\mathcal{C}_i^A, \mathcal{C}_C^i + \sum_{t_i^j > t_i^A, j \in F} \mathcal{C}_j^i)$ gives the amount of data that direct relays can get from the AP and replicate to other nodes (including the client and forerunners). As to be shown later, a 4-hop scheme can achieve the

capacity in Equation 3.2. Thus, the bound is tight.

Minimum Delay from AP to Client

In terms of the minimum delay to deliver a set of n blocks, a lower bound can be obtained by assuming that all contacts between the client and a direct or indirect relay are fully utilized. However, this bound is also loose because it is possible that some contact capacity between the client and a relay cannot be fully utilized if the relay does not carry enough new data. For example, a direct relay may only get $5MB$ of data from the AP, then immediately meet the client. If its contact capacity with the client is $10MB$, half of the contact capacity between the direct relay and the client is wasted, as there is no new data to be transferred.

Given a contact trace, in order to obtain a tight bound for the delay, we observe that it is possible to find the maximum amount of data that can reach client, by modelling it as a maximum network flow problem. Hence, we can perform an off-line computation to characterize the performance. Given a sequence of contacts $\{c_1, c_2, ...c_n\}$ between the different nodes starting from the request generation time, the graph $G = (V, E)$ for the network flow problem is constructed in the following way.

- **Vertices** There is one vertex A representing the AP, and another vertex C representing the client. They are the source and destination of the network flow problem. For each of the non-client vehicle v, if it has n contacts in the trace, we split it into n vertices, v^1 to v^n. These constitute all vertices in graph G. Note that, given a contact trace, each node's contacts, which are assumed non-overlapping with each other, can be ordered according to their *contact times*. We say a contact is the i^{th} contact of a node, if it represents the contact with the i^{th} node encountered by the considered node in the given contact trace.

- **Edges** For each contact, we use one (or two) directed edge(s) to represent it in the graph. For the contact between the AP and the client, we add a directed edge from A

to C. For the contact between the AP and a mobile helper v, if this contact is the i^{th} contact of the helper, we add a directed edge from A to v^i. For the contact between the client and a mobile helper v, if this contact is the i^{th} contact of the helper, we add a directed edge from v^i to C. A single directed edge suffices because only the specified direction is useful to maximize the network flow from A to C. During the contact, the data flow should always follow the direction of the edge. For each edge added, the edge capacity is set to the corresponding contact capacity.

For a contact between two mobile helpers u and v, if this contact is the i^{th} contact for vehicle u, and j^{th} contact for vehicle v, we add a pair of directed edges between u^i and v^j, as both of the two directions may help to maximize the network flow. These two edges should share the same contact capacity (denoted as setting S1). However, as detailed below, we can set the capacity of both (instead of the sum of both) to the contact capacity (denoted as setting S2) without affecting the value of maximum flow. Finally, we add directed edge with unlimited capacity from u^i to u^{i+1}, for every node u and valid i. These directed edges represent the fact that a vehicle can carry the data it received from a previous contact to the next contact. A finite capacity for this type of edge can be used to model buffer limit if required.

We can show that S1 and S2 have identical maximum flow solution in the following way. First, the optimal solution of S2 is no worse than S1, as every feasible solution of S1 is also a feasible solution of S2. Second, given an optimal solution in S2, if there are flows over a pair of edges, it can be reduced to a solution with the same maximum flow using at most one of the edges, by offsetting the flows in the opposite directions with each other until one of them becomes 0. By repeating the above process for all pairs of edges, we can get a feasible solution in S1. Thus, S2's solution is no better than S1.

For example, the contact trace as depicted in Figure 3.4 (a) will result in a network flow graph as shown in Figure 3.4 (b). Given the above formulation, the minimum delay

for delivering a given amount of data can be calculated efficiently.

With the performance limits known, we next examine several typical schemes starting from the simplest.

3.3.3 Comparison of Scheduling Schemes

Figure 3.4 (c) shows the volume of data that can reach the client using different schemes, under Figure 3.4 (a)'s setting.

1-hop scheme

The AP only transfers directly to the client during their contact. Volume of data delivered is $\mathcal{C}_{1-hop} = \mathcal{C}_C^A$. In the example, the client can get 2 blocks of data from the AP directly.

2-hop scheme

In this scheme, the AP sends data to the client and direct relays. A direct relay keeps the data until it meets the client and sends the data to the client. The amount of data transferred by a direct relay (i) from the AP to the client is the minimum of the two contact capacities, $\mathcal{C}_i^A$ and $\mathcal{C}_C^i$. Therefore, $\mathcal{C}_{2-hop} = \mathcal{C}_C^A + \sum_{i \in D} min(\mathcal{C}_i^A, \mathcal{C}_C^i)$. In the example, the client can get 1 additional block of data from the direct relay D_1. Note that another block sent to D_1 by the AP is lost due to the low contact capacity between D_1 and the client.

3-hop scheme

In this scheme, AP sends data to the client, direct relays and forerunners. Although forerunners cannot send directly to the client, they can send their data via direct relay or indirect relay (thus 3 hops). Under our assumption, forerunners can meet enough relays to dump its data to the client, thus all data sent to forerunners can reach the client eventually. Therefore, $\mathcal{C}_{3-hop} = \mathcal{C}_C^A + \sum_{i \in D} min(\mathcal{C}_i^A, \mathcal{C}_C^i) + \sum_{j \in F} \mathcal{C}_j^A$. In the example, the client can get 2 additional data blocks from indirect relay I_1, which itself gets the data from F_1.

However, the 2^{nd} block carried by D_1 is still lost. Minimizing the loss of directed relay requires at least a 4-hop delivery.

4-hop scheme

In this scheme, a direct relay saves its data as soon (and as much) as possible to forerunners before meeting the client. This feature minimizes loss since forerunners can always transfer its data via indirect relay later. 4-hop delivery achieves the capacity as characterized in Equation 3.2. In the example, the missing block from 3-hop scheme reaches the client via four hops: $\{A, D_1\}$, $\{D_1, F_1\}$, $\{F_1, I_1\}$, $\{I_1, C\}$.

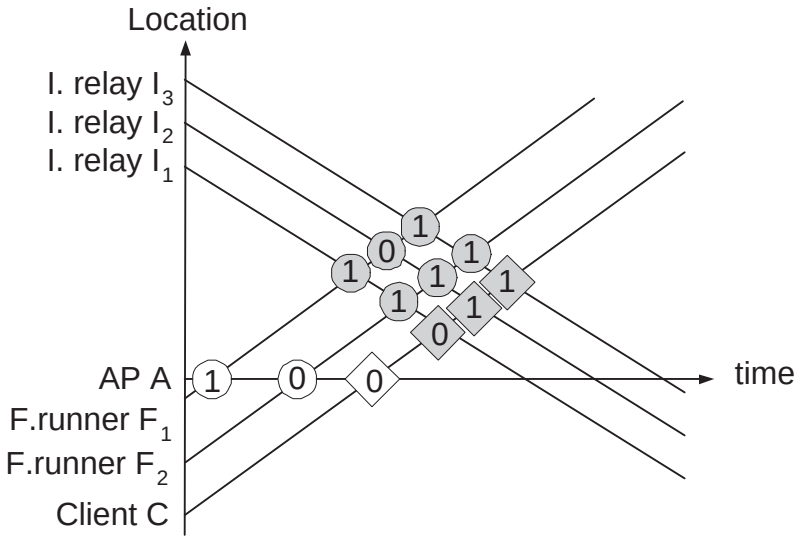


Figure 3.5: Scheduling to minimize delay

However, 4-hop is not yet optimal to minimize the delivery delay. Consider the situation as shown in Figure 3.5. The minimum delay to deliver the 1 unit of data is through 5 hops, i.e., the contact of $\{A, F_1\}$, $\{F_1, I_1\}$, $\{I_1, F_2\}$, $\{F_2, I_2\}$, $\{I_2, C\}$. Replication among forerunners (F_1 and F_2) is necessary to minimize the delay. Note that, the replication can only be carried among forerunners in one direction, i.e., from a forerunner to another forerunner moving behind it. For example, data can be replicated from forerunner F_1 to forerunner F_2, but not vice-versa.

3.3.4 MobTorrent Scheduling

Based on observations from the previous section, we design the scheduling algorithm of *MobTorrent* as follows:

- **Meta data:** *MobTorrent* keeps the following meta information with each data block k at each relevant node:

  1. Req_k: request id[2];
  2. b_k: block id;
  3. Ack_k: whether this block has reached the client;
  4. r_k: a (local) estimation of the *persistent* replication level of this block in the whole system, i.e. the number of mobile helpers that currently possess this block;
  5. ID_k: the ID of the forerunner that travels in the most front among all forerunners that possess this block.

- **First hop:** We assume that the AP always has new data to be forwarded to the client. When a data block k is sent to a forerunner F_j, the latter marks $r_k = 1$ and $ID_k = F_j$. When a data block k is sent to a direct relay, the latter marks $r_k = 0$, and $ID_k = 0$. Replication on direct relay is not counted as a persistent replication.

- **Role determination** Each mobile helper can determine its role (whether it is a direct relay, forerunner, indirect relay, or early/late node), based on the request information about the client's mobility trajectory and location of selected AP(s), as well as its own mobility trajectory. The request information is propagated to mobile helpers together with the propagation of data, whereas the node's own mobility trajectory is predicted locally. In addition, each forerunner appends its mobility

[2]For each request it currently serves, a helper records information including the client id and its mobility trajectory, the request start time and deadline, the information about selected AP(s) and mobile helpers for this request, etc..

trajectory to the request information before propagating it to the next hop, so that other nodes can calculate the relative locations of multiple forerunners locally.

- **Meta data reconciliation:** When two vehicles A and B meet, they will first exchange their meta data. Suppose the replication level estimation of block k is r_k^A at A, and r_k^B at B. After the exchange, both of them will set their estimation to $max(r_k^A, r_k^B)$. For all the following transmissions within this contact, the replication level of the transmitted block will be updated to the same new value at both sides. $Ack_k^A = Ack_k^B = Ack_k^A \vee Ack_k^B$.

- **Priority calculation:** Priority is calculated for each block to determine its order of transmission in the given contact. Suppose that the set of undelivered blocks (with $Ack = false$) at A is S_A, and the set of undelivered blocks at B is S_B. After the exchange, they calculate $S_A - S_B$ and $S_B - S_A$, which are the candidate set to be sent to each other. The set of undelivered blocks is updated as described later in the paragraph of *Last hop*. A node sorts its candidate blocks according to replication levels, *giving the highest priority to the block with the lowest level of replication*. The level of replication is calculated locally by the two nodes in contact according to the rules described later in the paragraph of *Minimize loss*, *Maximize rate*, and *Increase replication level*. Data block with higher priority is transferred first, thus has a higher chance to be replicated under the uncertainty of contact capacity. In case of ties, the blocks are sorted by ID_k. Therefore, given a block i replicated at forerunner F, another block j that is not replicated at any forerunner travelling in front of F (including F itself) has a higher priority than block i. This is because replication can happen only among forerunners in the direction reverse to their moving direction. Thus, in a long term, blocks from forerunner travelling in front have more opportunities to be replicated. Given a selected transfer direction among the contact nodes, data transmission is performed according to the priority calculated. Between two mobile helpers, the following three rules, i.e. *minimize loss*, *maximize*

rate, and *increase replication level*, take action in order to specify the scheduling of transmission directions.

- **Minimize loss:** If one party is a direct relay, it may carry blocks with $r_i = 0$, i.e., the data blocks that the relay received directly from the AP and have not yet been sent to any other node. Whenever such a block exists, transfer opportunity is given to the direct relay to minimize loss. After the transmission, both parties set the replication level of the block to 1, and set the *ID* to the ID of the current forerunner.

- **Maximize rate:** After the loss-minimizing step is done, this rule ensures that a direct or indirect relay has enough *new* data so that its contact capacity with the client can be fully used. Based on the contact capacity statistic between the client and relays, a threshold γ is selected to determine whether the amount of data carried by a relay is sufficient. We set γ as two times the expected contact capacity between vehicles. While this threshold is not reached yet, transfer opportunity is given to the forerunner. After the transmission, the replication level of the block is increased by a value of $\alpha < 1$, to capture the fact that it is replicated onto the relay. This value is less than one because unlike replication on the forerunner, the relay can go past the client without replicating this data block out and this particular copy is lost. As it is difficult to determine the "best" value for α, we simply set it to 0.5. Our simulation results show that performance does not change much when this value is varied.

- **Increase replication level:** Once the threshold for new data block is reached, data exchange happens in both directions. Remained candidate blocks from both sides are merged into a single priority queue sorted by replication level and *ID*. In case of ties, data stored on the direct or indirect relay is given higher priority, as transferring it will increase the replication level by 1, whereas the transfer at the other direction will only increase the replication level by α.

- **Last hop:** When the direct or indirect relay meets the client, blocks are transferred from the relay to the client according to their priorities. When the contact finishes,

the client uses its cellular interface as a control channel to update APs and all forerunners about new blocks that it has just received. Note that the client does not need to update direct and indirect relays, as they will be updated when they get contact with forerunners. Once past the client, a relay removes all of its data for the client (even those that have not been delivered yet), as there is no more opportunity to deliver them.

The intuition behind the scheduling scheme can be explained as follows. When few data blocks have been delivered, the relays often have sufficient "new" data to fully utilize the contact capacity with the client. However, as more blocks are delivered, it becomes harder for the relay to transfer sufficient undelivered data to the client if its contact capacity with forerunners does not match up well with the amount of undelivered data on them. As a result, data delivery rate to the client decreases as more blocks are delivered. *MobTorrent* scheduling scheme is designed such that as more blocks are delivered, the replication level of the undelivered data increases. In this way, data delivery rate can be maintained at a high level till all blocks are delivered.

Scheduling decisions in *MobTorrent* have some similarity in the spirit to the scheduling decisions made in existing DTN routing protocols, e.g., MaxProp [18] and RAPID [7]. However, in *MobTorrent*, the information of direction and relative position is fully exploited to optimize performance.

In the model presented, we have assumed that there is no overtaking and nodes do not leave the system. The *MobTorrent* system will work in the presence of overtaking and path divergence. The impact of these factors will be evaluated using simulation in the next section.

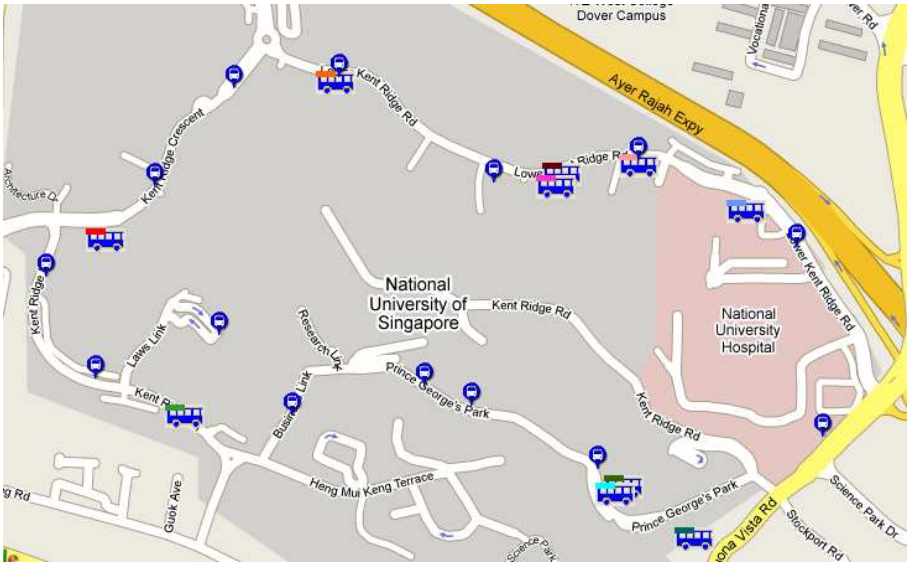


Figure 3.6: A snapshot of NUS bus monitoring system

3.4 Performance Evaluation

3.4.1 Testbed Configuration

We build a simple *MobTorrent* prototype to evaluate its performance in a real environment. We equipped 16 campus buses with an on-board LinkSys WRT54GL router as mobile clients. These clients run on the OpenWRT operating system. Each client is fixed at a bracket in front of the driver, and draws power from the bus. When the bus is moving, the client scans and attempts to associate to the campus Wi-Fi network. Once it successfully associates with an AP, the client uses a pre-stored map to figure out the valid IP address that it can use (the school APs along the route belong to several different IP subnets). We pre-load the mapping between AP and IP subnet on all clients to reduce the overhead of IP address acquisition. Similar optimization can be done on *MobTorrent*, as the AP and the client can exchange IP and authentication information via cellular network before they meet. Live bus tracking is available at http://mobtorrent.ddns.comp.nus.edu.sg/. Figure 3.6 gives a snapshot of the system. The campus shuttle buses run in a circle from both directions around the Kent Ridge campus. The average time for a bus to complete the 4km route is about 20 minutes. Over 120,000 contact statistics are collected from

File	Approach	Time	Speed	Ratio
A pdf file (80KB)	Without prefetching	3.2s	25KBps	68%
	With prefetching	0.33s	240KBps	98%
OpenWRT firmware (1513KB)	Without prefetching	12s	111KBps	45%
	With prefetching	1.77s	853KBps	78%
A 3 minute video clip (6.5MB)	Without prefetching	33s	201.5KBps	16%
	With prefetching	6.7s	993KBps	59%

Table 3.2: Download performance with and without prefetching

more than 1,300 driving hours over a 2 month period. The mean bus-AP contact duration is around 15s with mean contact capacity around 4.5MB, and the mean bus-bus contact duration is around 11s with the mean contact capacity around 3.2MB.

The evaluation has two parts. First, we evaluate the benefits of prefetching on the testbed. Next, we evaluate the benefits of using mobile helpers.

3.4.2 Benefits of Pre-fetching

When a client sends its request to an AP through cellular network before contact, the AP prefetches the data and stores it locally. In order to evaluate the potential gain, the client is programmed to download several selected files via the Wi-Fi network. In the measurement, there were 100 attempts to download each of these files over the Wi-Fi network when the clients were within range of a campus AP. Three of the selected files are shown in Table 3.2 together with the average downloading duration, downloading speed, and the downloading completion ratio. The completion ratio is computed as the number of times the files were completely transferred in a single AP-bus contact duration over the total number of attempts. Note that, as the file size increases, it becomes less likely that the file can be downloaded successfully in a single attempt. For the largest file of 6.5MB, complete downloading from Internet without prefetching is possible only 1 in 6 attempts.

For comparison, the files are stored on the APs in advance and downloaded to the mobile node on request. As shown in Table 3.2, there is a significant improvement of

Link	25%	50%	75%	95%
End-to-end Internet path	37.6	81.0	160.5	330.6
One hop Wi-Fi Link	2.347	2.742	4.668	26.129

Table 3.3: RTT measurement (ms)

downloading performance for files of all sizes. Note that such improvement is possible for all downloads made using advance requests, independent of the scheduling algorithm.

A closer look at the source of performance gain reveals that, for the first file with a small size (80KB), the downloading duration difference is mainly due to the decrease in RTT (round trip time). Table 3.3 shows the measured RTT distribution for end-to-end wired Internet Links and the local Wi-Fi link. The end-to-end wired Internet Link RTT is obtained from the Internet End-to-end performance Measurement (IEPM) with 413 different pairs of nodes across several continents. The local RTT is measured using ARP packets sent from the client box to the AP. ARP is used because many APs on campus do not response to ping but to ARP request. The measurement shows that the RTT of a local Wi-Fi link is about one magnitude shorter than a typical RTT over Internet. Shorter RTT allows TCP to increase its congestion window at a faster rate, which helps to shorten the downloading time for short files. For the larger files, the speed difference is also due to the avoidance of bottlenecks in the Internet. Though the download rate constraint from the server can be alleviated by using parallel downloads, the constraint in the local backhaul link (such as ADSL, cable, or wireless mesh) cannot be bypassed.

3.4.3 Benefits of Scheduling

We compare performance of the following schemes.

1. *1-hop:* The AP only sends data directly to the client. This serves as the baseline for performance comparison.

2. *Random:* A scheduling scheme that employs random replications between peers. When both sides have innovative data for each other (i.e., the data block that the

other side does not possess yet), one side is randomly selected to transfer a randomly selected innovative data block.

3. *Greedy:* A scheduling scheme that greedily replicates data in the order of their expected deductions in the delay to reach the client, following the design of RAPID [7]. The data delivery delay is determined according to the expected time the relay meets the client. For example, in a contact between a forerunner and a direct relay, transfer priority is always given to the direction from the forerunner to the direct relay.

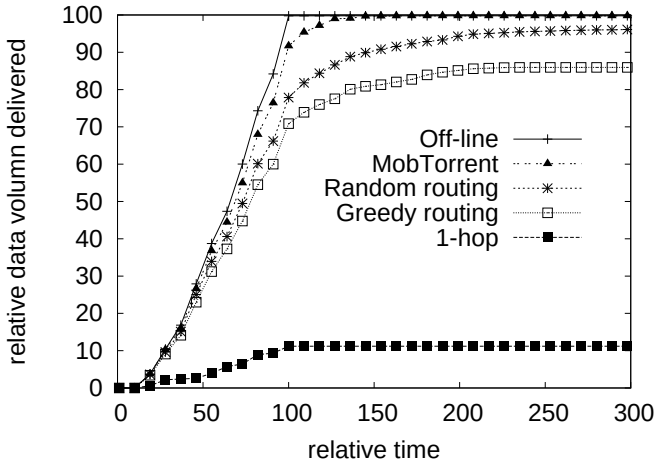
4. *MobTorrent:* As described in Section 3.3.4.

5. *Off-line:* A download mechanism that has off-line knowledge of contacts, and decides scheduling according to the solution of the network flow problem. This serves as the upper bound for achievable performance.

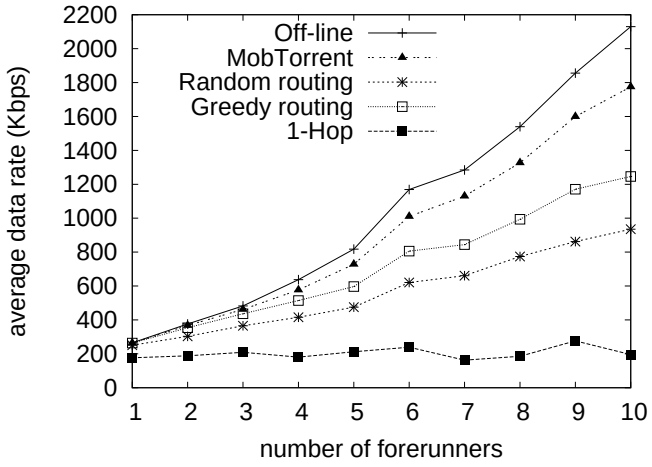
Note that, because of the significant difference in settings and assumptions, we cannot directly compare *MobTorrent* with existing related works, including PROPHET [68], Spray and Wait [99], MV routing [19], MaxProp [18], and RAPID [7]. For example, in our testbed, if two nodes just meet each other, the probability that they will meet again in recent future is almost 0. However, all existing schemes tend to assign a higher meeting probability to this pair of nodes. For a fair comparison, we implement the Random scheme and Greedy scheme such that, block is never replicated to a node which has no chance to deliver it to the client, while still keeping their original salient features. The Random scheme and Greedy scheme are selected to demonstrate that the heuristics incorporated in *MobTorrent* outperform the common practice used in existing DTN routing protocols.

Performance with Single AP and Single client

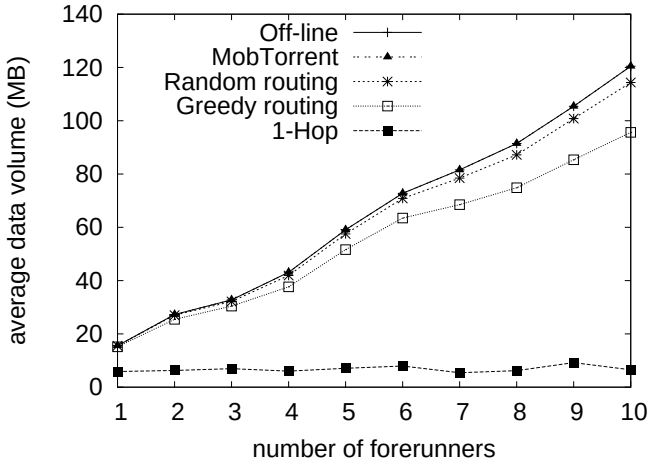
In this scenario, there is only one AP located on a 2-way street with an infinite flow of vehicles moving in both directions. The contact capacity is generated using the trace



(a) Downloading progress



(b) Average data rate ($\frac{\text{volume of data}}{\text{delay of last data}}$)



(c) Average volume

Figure 3.7: Performance under single-AP, single-client, ideal two-way street setting

collected from our testbed. 100 rounds of simulation are run, and the average is presented in Figure 3.7 (a). To make the averaging meaningful, for each run, the time and volume is normalized so that the off-line scheme reaches an optimal volume capacity of 100 at time clock 100. In the simulation, the average number of forerunners is 5.

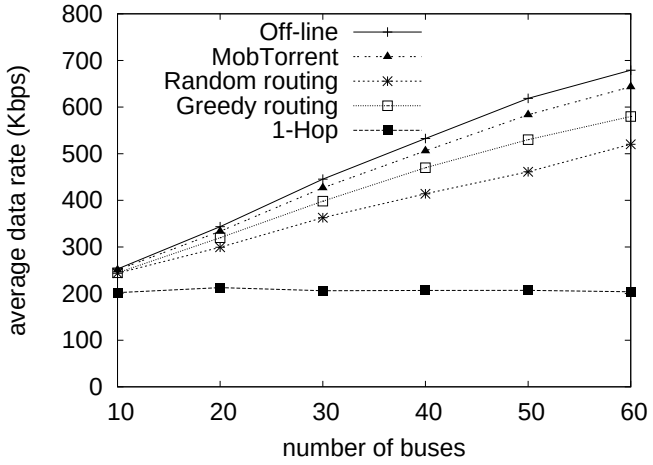
As shown in the figure, *MobTorrent* is close to off-line scheme in terms of both the volume of data delivered and the delay to deliver data. At time clock 100, *MobTorrent*, *random*, *greedy* and *1-hop* schemes deliver 91%, 78%, 71% and 11% of all data respectively.

The random scheme delivers most of the data eventually, but takes much longer than *MobTorrent*, due to the coupon collection phenomenon where new data are difficult to locate towards the end. *MobTorrent* alleviates such effect by giving priority to data blocks located only at forerunners travelling behind other forerunners. The greedy scheme does not reach the volume capacity (with a 20% gap) because the greedy transfer of data from a forerunner to a direct relay prevents the latter from replicating its data into the network. As expected, the 1-hop scheme only achieves a small fraction of the available capacity.

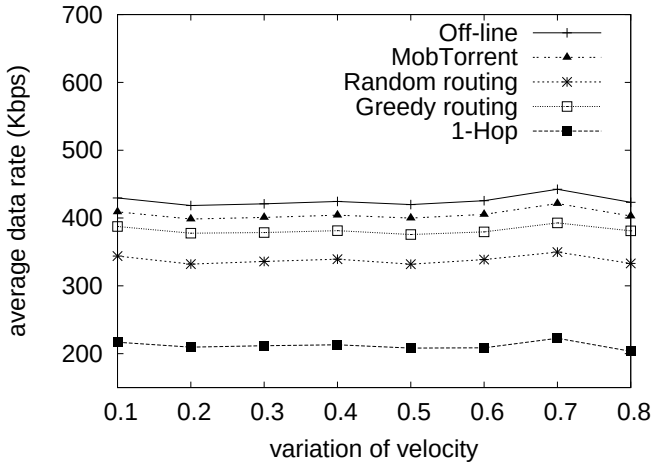
Next, we evaluate the impact of varying number of forerunners. The average number of direct relays is the same as the average number of forerunners. We define the *average data rate* as the ratio of the total data volume delivered and the time taken to deliver the last packet (lost packets are ignored). Figure 3.7 (b) and (c) show that when the number of forerunners (and hence helpers) increases, both the average data rate and the total volume of data delivered increase. In terms of the average data rate, the rate of increase for *MobTorrent* tracks the off-line scheme fairly well, whereas the random and greedy schemes improve at a slower rate. 1-hop scheme does not benefit from mobile helpers at all.

Performance with Multiple APs and Multiple clients

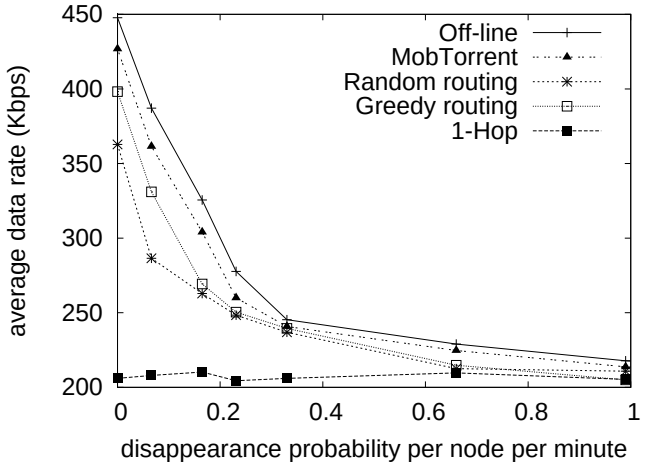
We use the mobility and contact trace collected from the testbed to drive the simulation. In the simulation, the request arrival to the whole system follows a Poisson process with



(a) Average data rate



(b) Impact of overtaking



(c) Impact of path divergence

Figure 3.8: Performance under multi-AP, multi-client, testbed trace setting

the average inter-arrival time of 20s. A running bus is randomly selected as the source of the request, whose size follows an exponential distribution with mean 5*MB*. We fix the number of APs in the network to 5, and vary the number of buses in the network. Note that the average load in the system does not change with the number of buses.

As shown in Figure 3.8 (a), schemes using mobile helpers improve the data download rate. As the number of buses grows, more mobile forwarding opportunities can be exploited.

We also investigate the impact of vehicle overtaking by varying the vehicle velocity. We achieve this by sampling the bus trace of both peak hour when buses tend to move slower and off-peak hour when buses move faster. As the variation of vehicle velocity increases, the overtaking probability increases. For example, when the variation increases to 0.6, the ratio of contacts due to overtaking is 25%. Figure 3.8 (b) shows that the average data rate remains fairly constant with respect to overtaking, and the *MobTorrent* scheduling scheme constantly outperforms other on-line schemes in all velocity variation settings. Since location information is not explicitly utilized by the other three on-line schemes, it is not surprising that performance of them remains fairly stable. For *MobTorrent*, when the relative node locations are not static any more, the performance is fairly robust for the following reasons. First, in *MobTorrent*, blocks only possessed by a forerunner that is nearer to the client are given higher priority for replication. This reduces the impact of overtaking, since the blocks on a forerunner being overtaken by the client may have already been delivered when overtaking occurs. Second, when a forerunner is overtaken by the client or another forerunner, a new opportunistic contact between the two nodes is created, which would not have occurred without overtaking. The overall impact of overtaking on *MobTorrent*'s performance is not significant.

Finally, we investigate the impact of path divergence by making vehicles disappear from the system suddenly. As shown in Figure 3.8 (c), the performance of all forwarding schemes degrades as the disappearance rate increases. While *MobTorrent*'s performance remains the best among all on-line schemes evaluated, overall performance is similar

among all schemes when path divergence occurs with a probability higher than 20% per node per minute.

3.5 Related Work

3.5.1 Multi-hop Cellular Networks

Due to the complementary characteristics of cellular networks and Wi-Fi networks, a number of research efforts have tried to combine them. In many of these approaches, only the cellular BSs are gateways to Internet and Wi-Fi networks are used to improve the performance of the cellular networks infrastructure, e.g., for coverage expansion [3], load balancing [109], and better channel utilization [69], as discussed in Section 1.2 of Chapter 1. Hsieh and Sivakumar give a comprehensive survey of these approaches [46]. In comparison, we use the cellular network mainly as a control channel for a vehicle to send out request and acknowledge the data it has received.

3.5.2 Vehicular Internet Access using Wi-Fi Networks

In the area of vehicular Internet access using Wi-Fi networks, Ott and Kutscher [84] propose a framework to support so-called drive-through Internet. The key component is a session protocol (PCMP) that offers *persistent* end-to-end communication even though the vehicles on the road only have intermittent contacts with roadside APs. In their work, for vehicles with velocity from 40km/h to 180km/h, a few Mega Bytes could be transferred to and from the mobile node using TCP and UDP. As part of the MIT Cartel project, Bychkovsky et al. [21] measure the upload bandwidth available to vehicles in the Boston metropolitan area using in-situ unplanned open APs. The result is also encouraging. The upload TCP bandwidth has a median of 30 KBps, and median transfer size per contact duration is 216 Kilo Bytes. Cabernet [32] further improves the performance by optimizing both the connection establishment procedure (QuickWiFi) and the transport protocol

(CTP). Another measurement of Wi-Fi connectivity from moving vehicle is described by Mahajan et al. [70]. Zhang et al. [114] investigate scheduling issues for vehicle uploading or downloading from a roadside unit. Balasubramanian et al. [8] propose ViFi, a protocol that opportunistically exploits base station diversity to minimize disruptions and support interactive applications for mobile clients. In comparison, we focus on the setting where roadside Wi-Fi AP only provides partial coverage (around 25% from our measurement or even lower), so that the main application of interest is delay-tolerant bulk file transfer.

3.5.3 Delay-Tolerant Network Routing

Another direction in the area of vehicular communication is from the angle of Delay-Tolerant Network (DTN). Vahdat and Becker [104] propose the "store-carry-forward" epidemic routing approach for intermittently connected networks. They use the hop count of messages to regulate the resource usage. Spyro et al. [99] show that binary splitting is optimal under certain assumptions for spreading a given number of replicas into the network. To improve over blind replication, Lendgren et al. [68] propose PROPHET, which is shown to work better than epidemic routing, based on the observation that real users tend to move in a predictable fashion with repeating behavioral patterns. UMASS's DieselNet project presents a study of vehicle (public buses) contact time [113], and proposes a series of routing protocols [7, 18, 19]. There are several major differences between *MobTorrent* and existing DTN routing works, as most of the latter are designed for the general case where the mobility pattern is largely structureless, and using historic meeting information is recognized as a good heuristic to estimate future contact probability. In above systems, the target application is communication among mobile nodes, the target message delivery delay is often in the scale of hours (e.g., the average delivery delay is 67.5 minutes using epidemic routing in the DieselNet trace [113]), and the target delivery rate is dozens of packets per hour. Instead, our interest is to use the capacity of intermittently connected networks to supplement the bandwidth of cellular networks, and we focus on the data transmission between roadside APs and mobile clients. The target de-

livery delay is in a few minutes, and the target delivery rate is hundreds of Kbps, which is comparable to that of HSDPA network. To achieve this, we make full use of the mobility information from in-situ AVL system.

Zhao et al. [116] and Li and Rus [64] propose to use Mobile ferry routing approach for data delivery in a sparsely connected network. The main idea is to introduce some non-randomness in node movement or actively change trajectories to help deliver data. However, in the scenario we are interested in, it is not likely that nodes will move just to accommodate communication.

Prefetching has been used extensively to speedup web download [85]. In a vehicular environment, Balasubramanian et al. propose using prefetching to speed up access to result of web queries [9]. We use prefetching in a different way, as the uncertainty comes from the varying contact opportunities instead of the file required.

Chakravorty et al. [23] propose the concept of treating the provision of wide-area wireless service for mobile users as a free market. Motani et al. [78] and Lee et al. [60] propose architecture to support a market place over mobile users.

3.6 Summary

In this chapter, we present *MobTorrent*, an on-demand, user-driven framework for vehicles to access Internet via roadside static APs and other mobile vehicles on the road.

MobTorrent has the following components. In order to improve network throughput performance, prefetching and caching are used to better exploit the short contact time between AP and client by having the data locally available for transfer. In addition, to address the issue of low coverage, data can be pushed to mobile helpers so that areas where Wi-Fi can be used for data transfer are not limited to coverage of static roadside APs but expand to include areas covered by mobile nodes. Our results based on real world experiments and trace-driven simulations show that *MobTorrent* provides substantial improvement over other existing frameworks.

Chapter 4

MobiCent: an Incentive-compatible Credit-based System for DTN

4.1 Introduction

Delay-Tolerant Networks are characterized by intermittent connectivity. Such networks are assumed to experience frequent, long-duration partitioning and often lack an end-to-end contemporaneous path [35]. As proposed in Chapter 3, in future mobile communication systems, the high-bandwidth but intermittent wireless connections among participants can be exploited using the Delay-Tolerant Networking (DTN) approach, so as to enhance the performance of traditional cellular networks. *MobTorrent* demonstrates the viability of the proposed approach in vehicular networks. In addition to that, DTN approach can also be potentially applied to mobile human social networks, where people carrying wireless mobile devices communicate through low-power high-bandwidth links, like Ultra WideBand (UWB). The MIT Reality Mining project [76] and the Pocket Switch Network [48] are examples of mobile human social networks.

In the targeted civilian and commercial environments, the mobile nodes are managed by autonomous and selfish parties, thus an incentive scheme should be employed to foster cooperation among participants. However, this opens the possibility that a selfish node

may game the system, by performing hidden actions that increase its own reward from the incentive scheme while degrading the overall system performance.

For example, if we assume that a fixed amount of reward is to be equally shared among all nodes on a forwarding path, a selfish node can create Sybil nodes [28] and forge phantom forwarding edges among its Sybil nodes to exaggerate its contribution. In this way (i.e. edge insertion attack), the node increases its share of the reward. However, such selfish behavior discourages other nodes from participating in the forwarding. As another way to maximize its own reward, a selfish node may also purposely not forward data to other relays, betting that it can deliver the data directly to the client and thus keeps the entire reward. Such an attack (i.e. edge hiding attack) is demonstrated earlier by the *mobile forwarding game* example in Section 1.3 of Chapter 1. These selfish actions reduce the network capacity, resulting in both lower delivery ratio and higher delay. In this work, we focus on *rational* nodes rather than *malicious* nodes. A rational node carries out an action only if doing so can increase its own payoff. In comparison, a malicious node is willing to take any action that degrades the system's performance, regardless of its own payoff.

There are two key challenges in designing the incentive scheme for DTN. First, disconnections among nodes are the norm rather than exception. As a result, selfish actions as described above are extremely difficult to detect. In sharp contrast, traditional end-to-end connected wireless networks can rely on the mutual control among the autonomous peers to detect any such deviation. Second, as contacts are often unpredictable in DTN, the delivery paths cannot be predetermined, but must be discovered along with the forwarding of data instead. Again, the routing behavior of traditional end-to-end connected wireless networks is fundamentally different, as the delivery path is often determined before the actual data pass through. Because of these two differences, existing incentive schemes for end-to-end connected wireless networks cannot be directly applied, as will be elaborated more later in Section 4.7.

In this chapter, we present *MobiCent* [26], a credit-based system for DTN. *MobiCent*

is largely motivated by, and directly designed upon *MobTorrent*. On one hand, *MobTorrent* shows that the application of DTN in a commercial environment can be useful. Thus, a natural follow-up question is how to motivate the nodes to cooperate. On the other hand, the existence of the highly available control channel in *MobTorrent* can facilitate multiple designs in the proposed *MobiCent* protocol. However, the attacks as identified and addressed by *MobiCent* are fundamental to the nature of DTN, thus, *MobiCent*'s credit-based solution can potentially be generalized to foster cooperation in other forms of DTN systems different from *MobTorrent*.

We make the following contributions in this chapter:

1) We identify *edge insertion attacks* and *edge hiding attacks* as the two major forms of attacks in a DTN environment. It is extremely difficult to detect them in DTN, and they can seriously degrade the performance of DTN routing.

2) We take the algorithmic mechanism design approach [82] to address the two forms of attacks, and identify the necessary conditions under *edge insertion attacks* for a payment scheme to be incentive compatible, i.e., truthful participation is adopted by selfish nodes.

3) We propose incentive-compatible payment mechanisms to cater to client that wants to minimize either payment or data delivery delay.

MobiCent does not require detection of selfish actions as it provides incentives for selfish nodes to behave honestly. In addition, *MobiCent* does not require pre-determined routing path. It works on top of existing DTN routing protocols to ensure that selfish actions do not result in larger rewards. To the best of our knowledge, *MobiCent* is the first incentive-compatible scheme proposed for replication-based DTN routing protocols.

The rest of this chapter is organized as follows. Section 4.2 presents the system model and formulates the attack model and the path revelation game. The message exchange protocol to support *MobiCent* is described in Section 4.3. We analyze the payment scheme required to thwart *edge insertion attacks* in Section 4.4, followed by the mechanisms designed to combat *edge hiding attacks* in Section 4.5. Evaluation is presented in Section

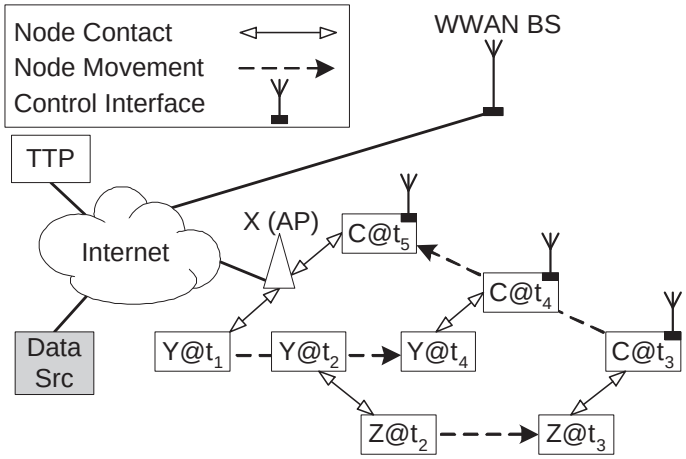


Figure 4.1: MobiCent Framework

4.6. In Section 4.7, we describe related incentive schemes. We conclude in Section 4.8.

4.2 System Model and Problem Formulation

4.2.1 System Model

As assumed in *MobTorrent*, *MobiCent* is based on a network model where the nodes can have access to two different networks. All nodes participate in a mostly disconnected network, where short-range high-bandwidth links are used for data transfer. At the same time, some of the nodes (in particular the source and destination nodes) have access to a mostly available network, where long-range low-bitrate links are used for control messages.

The network architecture assumed for *MobiCent* is shown in Figure 4.1. The components are:

- **Trusted Third Party (TTP)** stores key information for all nodes and provides verification and payment services.
- **Helpers** are mobile or static nodes (node X, Y, Z in the figure) that will help in data relaying using the high-bandwidth intermittent link. Helpers (except for the source node) do not need to have a highly available control channel.

- **Mobile Clients** are the destination nodes (node C in the figure) which initiate downloading. We assume that mobile clients have high-bandwidth but intermittent links for data transfer and highly available but low-bandwidth links for control messages.

A typical downloading process in *MobiCent* begins with the mobile client requesting data from a data source that can be another mobile node or a data store / web server in the Internet. In the former case, the mobile source node needs access to the control channel in order to initiate packet transfer. In the latter case (as studied in *MobTorrent*), the destination node obtains the data via some access points (APs). These APs are special helpers with Internet access, and they are the data sources within the wireless domain. In the example of Figure 4.1, data for a request initiated by the client C before time t_1 can be transferred from the AP X to the helper Y at time t_1, Y to Z at time t_2 and finally to C at time t_3. If data are replicated among the nodes, C can also receive data from Y at time t_4 and the AP X directly at time t_5. Different paths complement one another, as each of them is subject to uncertainty.

A detailed description of the system including the message exchange protocol is presented in Section 4.3. We will first present a brief overview here. We use standard cryptographic techniques and en-route onion encryption [72] to *prevent free riding*, *restrict strategy set of participants* and *handle dispute among relays and client*. More specifically, each relay encrypts the data payload with a one-time symmetric key before forwarding it. The key is also sent along with data in an encrypted form, such that only the TTP can recover the keys. Thus, after a client receives the encrypted data, the only way for the client to retrieve the decrypted data is to make payment to the TTP in exchange for the encryption key(s). Similarly, the only way for the relay to get payment is to be involved in the forwarding process. Note that the lightweight message exchange protocol handles a wide array of attacks, but it cannot prevent both client and relay from launching edge insertion attacks and edge hiding attacks, which will be described in detail in Section 4.2.3. To address these attacks, an incentive compatible payment scheme is needed.

4.2.2 MobiCent and DTN Routing

MobiCent runs on top of a given DTN routing module, and does not rely on any specific routing protocol. We first present a generic model of DTN routing. When two nodes meet, they exchange metadata on the packets they have in their respective buffers. Based on the information exchanged, each node decides which packets it wants the other node to transfer (replicate) to it. The order of the packet transfer depends on the priority a node associates with each packet. The amount of data that can be transferred in a single contact is dependent on the duration of the opportunistic contact.

Various DTN routing protocols differ mainly on how each packet's priority is determined. In the simplest version, all packets have the same priority. However, such simple stateless epidemic routing is not efficient, and researchers have proposed many improvements. For example, both direct and indirect contact histories are used in PROPHET [68]. In MaxProp [18], a combination of a few parameters, including contact history and packet hop count, are used to determine a packet's priority.

MobiCent works by setting the client's payment and the relays' rewards so that nodes will behave truthfully. Therefore, nodes will always forward packets without adding phantom links, and never waste contact opportunity unless the reward is inadequate or it is the decision of the underlying routing protocol. As a result, the (best) forwarding paths that should be discovered by the given routing protocol through replication will be discovered.

4.2.3 Path Revelation Game

Before formulating the problem as a *path revelation game*, we first define some terminologies.

Definition 4.1 *An **edge** e represents the opportunistic contact between two nodes, through which data can be forwarded between them. Formally, an edge e is defined by the two nodes $\{v_1, v_2\}$ in contact (referred to as the edge's vertices) and the contact time $t(e)$* [1].

[1] For easy presentation, we assume that contacts do not overlap and have enough capacities to exchange data. Thus, both the contact duration and its capacity are omitted in our formulation.

For example, Figure 4.2 plots the scenario depicted in Figure 4.1 as a contact graph over time axis. In the figure, X meets Y at time t_1, and the corresponding edge is denoted as $e = (\{X,Y\},t_1)$, where X and Y are e's vertices. Given a node v, the set of edges containing it as a vertex is denoted as $E(v)$.

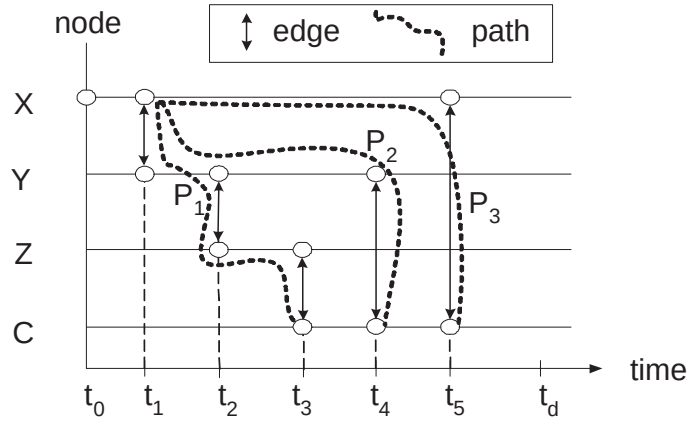


Figure 4.2: A contact graph plotted over time axis

Definition 4.2 *A **contact graph** is denoted by $G = (V,E)$, where V is the set of nodes in the system, and E is the set of edges among the nodes.*

In Figure 4.2, $V = \{X,Y,Z,C\}$, and $E = \{(\{X,Y\},t_1), (\{Y,Z\},t_2), (\{Z,C\},t_3), (\{Y,C\},t_4), (\{X,C\},t_5)\}$.

Definition 4.3 *A **forwarding path** is a sequence of nodes from the source to the destination, such that, from each of its nodes, there is an edge to the next node in the sequence, and edges appear in non-decreasing contact time.*

Given a path P, $Relay(P)$ is the set of relays on the path. Note that source is considered as a relay. The number of relays on path $|Relay(P)|$ is defined as the length of the path. A path P with length n is called a n-hop path. At the contact time of its last edge, a path P is *revealed* to the destination.

In Figure 4.2, there are three paths (P_1, P_2, and P_3) from the source node X to the destination node C. Path P_1 consists of three edges: $(\{X,Y\},t_1)$, $(\{Y,Z\},t_2)$, and $(\{Z,C\},t_3)$

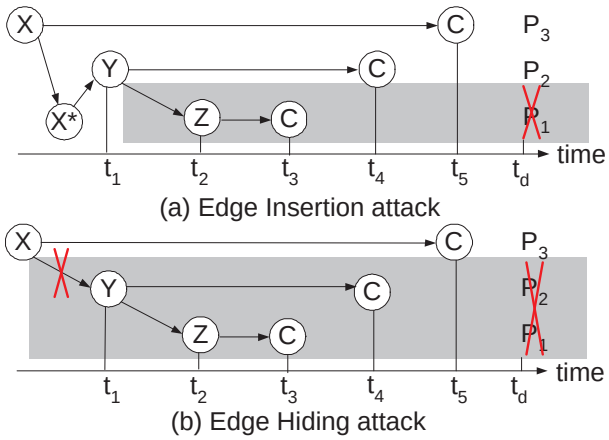


Figure 4.3: Attacks

in sequence; P_2 consists of two edges: $(\{X,Y\},t_1)$ and $(\{Y,C\},t_4)$ in sequence; and P_3 is a 1-hop path consisting of a single edge $(\{X,C\},t_5)$).

The charge to the client and the reward to the relays are determined by a *payment scheme* consisting of two algorithms, namely, a *payment set selection algorithm*, determining which relays should be paid, and a *payment calculation algorithm*, which determines how much credit should be paid to each selected relay, and how much should be charged to the client.

As stated in Section 4.2.1, *MobiCent* uses its message exchange protocol to constrain the strategy space of users, so that *edge insertion attacks* and *edge hiding attacks* are the two major forms of selfish actions that a node can take. We will illustrate how a selfish node gains from cheating under a natural payment scheme. The example is based on the contact graph in Figure 4.2. Without loss of generality, we assume the use of the *earliest-path fixed-amount* payment scheme. Under the scheme, a client pays for each received data block a fixed total amount of 3 cents, which is shared equally by all relays on the earliest delivery path. A helper participates if the payoff is more than 1 cent, thus the maximum path length is 3.

For illustration purpose, we redraw Figure 4.2 to highlight the edges that belong to different paths in Figure 4.3. Thus, some nodes (e.g., the client C), which are receivers in multiple edges, are plotted as multiple instances in the figure.

Figure 4.3 (a) shows an edge insertion attack. In the figure, when a selfish AP X gets the data, it estimates the delivery probability for all possible paths, denoted as $p(P_1)$, $p(P_2)$, and $p(P_3)$ respectively. Recall that the reward per node is $\frac{3}{n}$ cents where n is the length of the delivery path. Suppose $p(P_1) = 1$ and $p(P_2) = \frac{1}{2} + \varepsilon (> 0)$. By creating a Sybil node X^* and forging a phantom transfer from X to X^* before forwarding the data to Y, X can claim $\frac{2}{3}$ of the total payment if P_2 succeeds. However, due to this additional edge, Y will not be able to forward to Z, as the maximum length (3) is reached already. Thus, path P_1 is not revealed. By launching the edge insertion attack, the expected reward of X by forwarding via Y is $3 \times \frac{2}{3} \times p(P_2) = 1 + 2\varepsilon$. In comparison, the reward if X transfers honestly is only $3 \times \frac{1}{3} \times p(P_1) = 1$. As a result, the selfish behavior of node X increases its own payoff, but hurts the system performance by reducing the success delivery probability from 1 to as low as $\frac{1}{2} + \varepsilon$ (if $p(P_3) = 0$). The delivery time is delayed from t_3 to no earlier than t_4.

The client can also cheat by launching edge insertion attacks. For example, when it meets X directly through path P_3, it can pretend to be a relay instead, so that it can recover some of its payment as the Sybil relay.

Figure 4.3 (b) shows an edge hiding attack. Depending on the estimated delivery probabilities, node X may decide not to forward the packet to other relays at all. Suppose $p(P_3) = \frac{2}{3} + \varepsilon (> 0)$. In this case, in order to selfishly maximize its own reward, node X will not forward the data to Y, i.e., hiding the edge $(\{X,Y\}, t_1)$. This holds regardless of the value of $p(P_1)$ and $p(P_2)$, and even when X is allowed to launch edge insertion attacks (as described above). The selfish behavior of node X hurts the system performance, by reducing the success delivery probability from up to 1, to as low as $\frac{2}{3} + \varepsilon$, and delaying the delivery time to t_5.

Given $G = (V, E)$, the two attacks can be formalized as:

Definition 4.4 *A node v launches an* ***edge insertion attack*** *by creating a Sybil node v' such that G is modified to $G' = (V', E')$, where $V' = V \cup \{v'\}$, and $E' = E^{v \to (v,v')} \cup \{(v, v', t)\}$. $E^{v \to (v,v')}$ means for any edge e in $E(v)$, the vertex corresponding to node v*

can be set to either v or v'.

Definition 4.5 *A node v launches an **edge hiding attack** by modifying G to $G' = (V, E - e)$, where $e \in E(v)$.*

A selfish node can launch one or both attacks multiple times. Now we can define the path revelation game formally.

Definition 4.6 *A **path revelation game** is a distributed online game to reveal paths on a contact graph G.*

- *Each node (including both relay and client) is a player.*
- *As an edge e is formed, only its two vertex nodes together can reveal the existence of the edge.*
- *The possible strategies of a player are (1) acting honestly, or (2) launching edge insertion attacks and edge hiding attacks.*
- *The payment scheme calculates payoff for each player based on the revealed contact graph.*

The payment scheme determines the outcome of the game, and it should be designed to discover some desirable path(s) from source(s) to destination (e.g., the earliest path or the shortest path). More specifically, we design payment schemes to meet the following goals:

1) *Incentive compatible*: Truthful participation is adopted by both client and relay, despite of their selfish nature.

2) *Efficient and frugal*: If there is at least one path revealed before a given deadline, the client should be able to recover the data with minimum payment. If a client is willing to pay more (but still bounded amount) to recover its data as soon as possible, the client should be able to recover its data upon revelation of the earliest path.

Among auction games, our work is closest to the well-studied *path auction game*. In this game, there is a network $G = (V, E)$, in which each edge $e \in E$ is owned by an agent. The true cost of e is private information and known only to the owner. Given two vertices, source s and destination t, the customer's task is to buy a path from s to t. Path auction games have been extensively studied and much of the literature has focused on the Vickrey-Clarke-Groves (VCG) mechanism. In the VCG mechanism, the customer pays each agent on the winning path (i.e., the path with the minimum amount of total cost) an amount equal to the highest bid with which the agent would still be on the winning path. This mechanism is attractive as it is incentive compatible.

Existing works [30, 94] have shown that VCG is vulnerable to false-name manipulation, a form of the Sybil attack. Furthermore, it is well known that VCG is not frugal for the path auction game [6, 31, 51], i.e., a VCG-based incentive-compatible scheme may result in very large payment.

A key difference between our work and the work on the path auction game is that in our work the contact graph is the information to be elicited from the participants, whereas in the latter, the topology is static and known to all.

In the rest of this chapter, we first present the message exchange protocol to support *MobiCent* in Section 4.3. Following that, we analyze the payment algorithm required to combat edge insertion attacks in Section 4.4, then present the thwarting of edge hiding attacks in Section 4.5.

4.3 MobiCent Message Exchange Protocol

In *MobiCent*, we exploit the highly available low-bandwidth control channel at destination to allow a Trusted Third Party (TTP) to mediate the file transfer process. We will explain the message flow using file downloading from Internet as an example. The case of a source node initiating a file transfer to a destination node is similar. Message exchanges occur in three stages: (1) data request, (2) data forwarding, and (3) data recovery.

Suppose the TTP's public and private keys are P_T and S_T respectively, and a participating node (helper or client) R's public and private keys are P_R and S_R respectively. In addition, R shares with the TTP a symmetric key k_{TR}.

Each node only needs to know its own public and private keys, the shared secret key with the TTP and the TTP's public key. For the TTP, besides its own public and private keys, it has to know the shared secret keys and public keys of all nodes. A new participating node has to inform the TTP of its public key and choose the shared secret key with the TTP. Furthermore, the TTP encrypts the pair $\{$*node id, node's public key*$\}$ with its private key and this signature is stored on the participating node.

4.3.1 Data Request

To initiate the downloading process, a client C first sends the file download request $r =< C, f, p(), t_0, t_d, \alpha >$ to the TTP in a secure way. f is the file description including its name, size, and the approach to locate the file (e.g. URL address). $p()$ is the payment function, which will be discussed in detail in Section 4.4 and Section 4.5. t_0 and t_d are the start time and deadline of the request respectively. α indicates the valid geographical area/region for the request to propagate.

After receiving and successfully decoding/verifying the request, the TTP encrypts r with its private key and sends C the request signature $S_T(r)$.

Upon receiving the TTP's approval, C can then forward $< r, S_T(r) >$ to all APs within the specified area α. C may need to contact a directory server to find out the list of APs in the area.

When an AP gets the request, it first checks the validity of the signature from the TTP, as well as the file description, the time and area scope. It may also consider the amount of promised payment to decide whether to help or not. If the reward is sufficient, the AP begins to prefetch the file block by block, with a predetermined block size. These blocks are then replicated to the helpers using the DTN approach.

4.3.2 Data Forwarding

Each node R maintains a list of blocks $\mathbb{L}_1(R)$ that it currently holds, and a list of blocks $\mathbb{L}_2(R)$ that it has requested but not received yet. When two nodes A and B are near each other, they can communicate directly via the short-range high-bandwidth link. They will begin with an exchange of the metadata to reconcile their block lists $\mathbb{L}_1(A)$, $\mathbb{L}_1(B)$, $\mathbb{L}_2(A)$ and $\mathbb{L}_2(B)$, and agree on the subset of blocks to be exchanged and the sequence to exchange blocks. The exact block subsets that are exchanged depend on the routing algorithm [7, 18, 25].

For the i^{th} block of request r, the message being forwarded consists of three parts as shown in Figure 4.4 (a). The header H contains the basic information $< r, i, S_T(r) >$ which remains the same for all hops. The header is followed by the encrypted data and supplementary layers, which are being modified and appended to respectively at every hop.

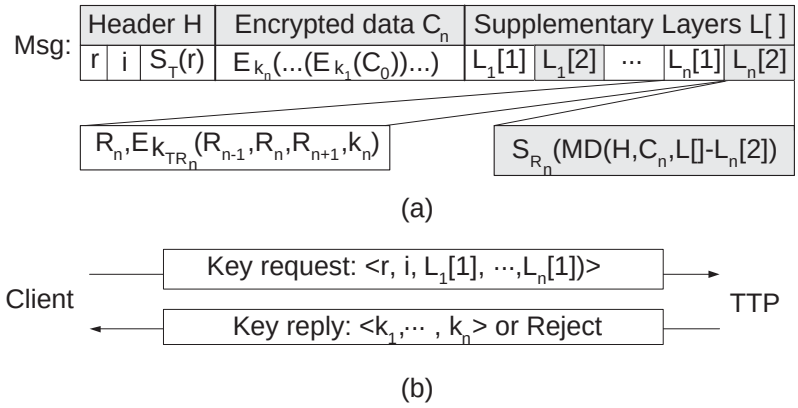


Figure 4.4: Message format

Denote C_0 as the requested content in clear text, and C_n as the encrypted content forwarded by the n^{th} hop node ($n = 1, 2, \ldots$). Let the n^{th} hop relay be denoted by R_n.

Before forwarding a received block with data payload C_{n-1} to the next hop, the relay R_n generates a unique symmetric key k_n for the block, and substitutes the data payload with $C_n = E_{k_n}(C_{n-1})$. Note that k_n is only used to encrypt the current block and a new key is generated for each block encryption.

In addition, it appends a new supplementary layer with 2 components, $L_n[1]$ and $L_n[2]$. The first component $L_n[1]$ contains the current relay's ID R_n, and an encrypted field of four subfields, namely the previous relay's id R_{n-1}, the current relay's id R_n, the next relay's id R_{n+1} and the secret key k_n used for data block encryption. The shared secret key of the TTP and R_n is used to encrypt this element. The array of $\{L_j[1]\}$ is the data that will be forwarded to the TTP later by the mobile client to recover the data. For the source node, a randomly generated value is used for R_{n-1}.

The second component $L_n[2]$ consists of just one fields, a cryptographic hash (e.g. using MD5 or SHA-1) of the whole block minus the currently computed hash values, encrypted using the current relay's private key (S_{R_N}). This component is required for verification and auditing purpose and is only sent to the TTP when there is a dispute.

The next relay R_{n+1} first verifies the header H to make sure that request r is valid. Then, the relay R_{n+1} stores the data block and the identity R_n which is needed to generate the next supplementary layer if it forwards the message further. Before forwarding, it also needs to verify $L_n[2]$ using R_n's public key. This key is verified using the TTP's signature for the pair $\{R_n, P_{R_n}\}$ obtained from R_n.

Note that a relay node does not need to contact the TTP during the process. This has two benefits: (1) reduce the load of the TTP, and (2) enable a mobile node without a highly available control channel to become a relay.

In the forwarding process, for each block, a sender R_n needs to perform 2 symmetric key encryption (over the data payload and $L_n[1]$), and signs 1 fields ($L_n[2]$) using its own private key. The receiver R_{n+1} needs to verify $L_n[2]$ using the sender's public key P_{R_n} for each block. The receiver also needs to verify the sender's public key (per neighbor overhead) and the TTP's signature for the request (per request overhead).

4.3.3 Data Recovery

Without loss of generality, suppose the block is delivered from source R_1 to the client C via two store-and-forward hops $R_1 \rightarrow R_2$ with one-time encryption key k_1, and $R_2 \rightarrow C$

with key k_2.

C sends to the TTP (in a secure way) the following key request $< r, i, L_1[1], L_2[1] >$ as shown in Figure 4.4 (b).

From this information, the TTP is able to recover the required secret keys k_1 and k_2. The TTP then sends $\{k_1, k_2\}$ to C.

With these keys, C is able to decrypt the data block using each key in the given list sequentially until all keys are used and the original text is recovered. At this point, we assume that C is able to validate clear text through checksum in the clear text or application level semantic. If data are valid, C sends confirmation to the TTP. Otherwise, C sends a dispute with the encrypted data it receives (C_n) and the list of elements in $\{L_n[2]\}$ to the TTP.

The TTP settles the credit transfer off-line. In addition, the TTP may broadcast the ACK for block r, i in the area α after the request is completed.

If a client does not submit any key request before the deadline, the TTP will assume that the process fails. All pending data blocks in the network automatically time out.

4.3.4 Protocol Properties

The message exchange protocol has the following properties. First, it prevents free-riding through the use of en-route onion encryption. More specifically, a client cannot get its desired content without payment, and a helper cannot get payment without helping with the forwarding process. Note that, there is no monetary barrier for a potential forwarder to participate. As the forwarder does not need to decrypt the data, it does not pay for the content.

Next, the protocol prevents a node from modifying an existing valid path segment since each relay encrypts the identities of the previous, current and next relays. Based on the information contained in the message, the protocol can deterministically detect nodes that modify the path. Thus a node's valid strategy space is to modify its own edges, i.e., by launching edge insertion attacks and edge hiding attacks.

Both communication overhead and computation load on the TTP are minimum. Relays do not need to contact the TTP during forwarding, and payment settlement is performed off-line. Finally, forwarding requires public-key cryptography that may be expensive. We discuss this issue further in Section 4.6.4.

4.4 Thwarting Edge Insertion Attacks

Suppose relays on a delivery path are selected for payment, we consider the design of payment calculation algorithm to thwart edge insertion attacks. The intuition behind our design is: (1) To deal with relay's cheating, we observe that introducing Sybil nodes allows a relay to claim a larger fraction of the total reward, while increasing the delivery path length. Although we cannot prevent a relay from stealing a larger fraction of rewards from the total reward (as we cannot distinguish between a real node and a Sybil node), we can make the total reward decreases as a function of the path length. As long as the total reward diminishes faster than the increase of a selfish node's relative share, the relay will only decrease its overall reward by introducing Sybil nodes. (2) Similar idea applies to thwarting edge insertion attacks from the client. More specifically, although we cannot prevent the client from reclaiming some of its payment back as a Sybil relay, we can increase its charge as a client according to the path length, such that there is no net gain for the client.

We consider a general payment scheme S. Given a n-hop path, we define the minimum payment to an individual relay in the path as $Reward_S^{min}(n)$, and define the charge to a client using a n-hop path as $Charge_S(n)$.

Lemma 4.1 *To prevent a relay from gaining by launching edge insertion attacks,* $2 \times Reward_S^{min}(n+1) \leq Reward_S^{min}(n)$.

Proof: Consider a relay R on a n-hop path. Suppose R gets the minimum reward $Reward_S^{min}(n)$. By inserting a Sybil node R', its reward is the sum of the payments to

two relays on a $(n+1)$-hop path, which is no less than $2 \times Reward_S^{min}(n+1)$. In order to prevent R from gaining by doing so, we must have $2 \times Reward_S^{min}(n+1) \leq Reward_S^{min}(n)$. □

Lemma 4.2 *To prevent a client from gaining by launching edge insertion attacks,*

$Charge_S(n+1) \geq Charge_S(n) + Reward_S^{min}(n+1)$.

Proof: By appending a phantom edge on a n-hop path, a client can gain reward as the Sybil node. Since the new path contains $n+1$ hops, the reward to the appended Sybil node is no less than $Reward_S^{min}(n+1)$. In order to prevent the client from gaining by doing so, $Reward_S^{min}(n+1) - Charge_S(n+1) \leq -Charge_S(n)$. □

Note that, our formulation is general, as it does not exclude the use of other factors to determine payment. For example, we allow the rewards for different relays on the same path to be different.

Lemma 4.1 states that the payment scheme should ensure that a relay's incremental gain by being paid as multiple Sybil nodes grows slower than the reduction of each individual's payment (due to the increase of the path length). Similarly, Lemma 4.2 states that the incremental increase of a client's payment for using a longer path is greater than the reward the client earns as the added Sybil node.

The two lemmas show that existing payment schemes, including the fixed-amount payment scheme we considered above, as well as others [50, 118] are not incentive compatible under edge insertion attacks.

To simplify the presentation without loss of generality, we assume that 1 cent is the minimum reward required to motivate a relay to participate in the forwarding process. Lemma 4.1 and Lemma 4.2 together lead to Theorem 4.1.

Theorem 4.1 *To enable incentive-compatible forwarding while ensuring deficit-free for the TTP*[2]*, the payment charged to a client for using a n-hop path is at least* $2^n - 1$.

[2]The deficit-free property means that the TTP charges no less credit from the client than the total amount it pays to the relays. If the deficit-free property is not ensured, malicious node can make profit from phantom transactions.

Proof: As $Reward_S^{min}(n) \geq 1$, from Lemma 4.1, we have $Reward_S^{min}(i) \geq 2^{n-i}$ for $1 \leq i \leq n$. Using Lemma 4.2, we have: $Charge_S(n) \geq \sum_{i=1}^{n} Reward_S^{min}(i) \geq \sum_{i=1}^{n} 2^{n-i} = 2^n - 1$

□

While the bound may seem large, we argue that it is feasible to be adopted in practice, because:

1) The client can specify the maximum hop N according to its requirement and utility function to control the maximum possible payment.

2) While the cost of using a small N (3 to 5) is low, it is sufficient in some typical DTN scenarios, as will be shown in Section 4.6.

3) Many practical DTN routing algorithms pose a limit on the hop count for better use of network resources. As an example, Spray and Wait [99] uses no more than 5 hops to spread 16 copies, a number sufficient in some typical DTN scenarios.

As existing schemes do not satisfy the required property, we introduce a new incentive-compatible payment algorithm that minimizes the client's payment.

Multiplicative Decreasing Reward (MDR)

Given the maximum path length N and an arbitrarily small positive ε, if a n-hop ($1 \leq n \leq N$) path is selected, each relay on the path gets the same reward of:

$$Reward_{MDR}(n) = (2+\varepsilon)^{N-n} cents \tag{4.1}$$

and the client is charged by

$$Charge_{MDR}(n) = (2+\varepsilon)^{N} - (2+\varepsilon)^{N-n} cents \tag{4.2}$$

Theorem 4.2 *Under the MDR payment algorithm, both client and relay have no incentive to launch edge insertion attacks.*

Proof: Under the MDR payment algorithm, if a client on a n-hop path launches edge insertion attacks, and inserts $k \geq 1$ extra edges, its net payoff is:

$$\begin{aligned} & k \times Reward_{MDR}(n+k) - Charge_{MDR}(n+k) \\ = \;& k \times (2+\varepsilon)^{N-n-k} - ((2+\varepsilon)^N - (2+\varepsilon)^{N-n-k}) \\ = \;& \frac{k+1}{(2+\varepsilon)^k}(2+\varepsilon)^{N-n} - (2+\varepsilon)^N \\ < \;& (2+\varepsilon)^{N-n} - (2+\varepsilon)^N \quad (since \quad \varepsilon > 0, k \geq 1) \\ = \;& -Charge_{MDR}(n) \end{aligned} \tag{4.3}$$

Hence, a client does not gain by inserting edge. Now let us consider the last relay R_n on a n-hop path. Regardless of the behavior of previous relays (whether some of them are Sybil nodes or not), if R_n launches edge insertion attacks and inserts k extra edges ($n < n+k \leq N$), its reward is:

$$\begin{aligned} & (k+1) \times Reward_{MDR}(n+k) \\ = \;& \frac{k+1}{(2+\varepsilon)^k}(2+\varepsilon)^{N-n} < Reward_{MDR}(n) \end{aligned} \tag{4.4}$$

Therefore, the dominant strategy for R_n is to act truthfully. Similar argument can be applied iteratively to the previous relays starting from the $(n-1)^{th}$ relay, assuming that later relays on the path are rational. Therefore, based on iterative elimination of dominated strategy, all relays adopt truth telling in the unique Nash equilibrium.

□

Note that truth telling is not dominant strategy for relays except for the last relay since the strategy of a relay appearing earlier on the path can be affected by an irrational relay appearing later on the path. However, the game is dominance solvable and all relays adopt truth telling in the unique Nash equilibrium. The small positive ε in MDR payment

algorithm is required in the proof to ensure the uniqueness of the Nash equilibrium. We omit ε in the following discussions for brevity.

Among all payment schemes that satisfy the necessary conditions for incentive compatibility, Theorem 4.1 and Theorem 4.2 together imply:

Corollary 4.1 *The MDR payment algorithm is the most frugal incentive compatible mechanism robust under edge insertion attacks.*

Under the MDR payment algorithm, each relay's individual reward and the sum of all relays' rewards decrease with the path length, whereas the client's payment increases with the path length. The maximum surplus or overpayment is reached when the longest path (N hops) is used, which is: $Charge_{MDR}(N) - N \times Reward_{MDR}(N) = (2)^N - (N+1)$ (with ε omitted).

This overpayment can be handled in the following ways. First, some of the overpayment can be considered as payment to the system provider. Second, the overpayment may be redistributed back to the mobile nodes if the redistribution is incentive compatible. Cavallo discusses an incentive-compatible redistribution mechanism [22].

MDR alone is sufficient to handle edge insertion attacks given a selected set of relays. However, edge hiding attacks may affect the set being selected. Thus, MDR algorithm need to be used together with some payment set selection algorithm, which will be considered in the next Section.

4.5 Thwarting Edge Hiding Attacks

The high-level idea to thwart edge hiding attacks is to determine an incentive-compatible relay set by examining a sufficient subset of the paths ever revealed before the deadline. Intuitively, our solution provides the following two properties: (1) Participating in more forwarding paths (by replicating to other relays) only increases a node's probability of being selected for payment. (2) If a relay participating in multiple paths is selected for

payment, it will always be paid according to the path that gives it the highest reward, regardless of the other paths that it participates. This should hold even if its most favorable path (e.g. direct contact with the client) is revealed later than other less-favorable paths. In another word, participating in more forwarding paths does not decrease a relay's reward amount if it is ever selected for payment. The combination of these two properties ensures that a rational relay does not launch edge hiding attacks.

In the following, we present selection algorithms for two types of clients, namely:

- *Cost-sensitive client*: The client's goal is to minimize payment under a given deadline constraint.
- *Delay-sensitive client*: The client's goal is to minimize delay under a given payment constraint.

While the algorithm for cost-sensitive clients is simpler than the algorithm for delay-sensitive clients, they share the same intuition as described above. By catering for the two types of clients, our schemes allow the trade-off between payment and delay. A client selects the desired scheme explicitly when issuing its request. The two types of clients can coexist in a single system.

4.5.1 Cost-sensitive Client

min-Cost Selection Algorithm Under this algorithm, the forwarding procedure is terminated only at the deadline of the request, or upon revelation of a 1-hop path, whichever is earlier. The client reports to the TTP the shortest path ever revealed when the terminating condition is met. Only relays on the reported path are paid. Payments by the client and to the relays are computed using the MDR algorithm.

Theorem 4.3 *Under the min-Cost selection algorithm, both client and relay have no incentive to launch edge insertion attacks or edge hiding attacks.*

Proof: We first consider the dominant strategy for the client. The client cannot arbitrarily fake the shortest path, as in that case it is not able to decode the correct data. Given that the client pays the least with the real shortest path it can reveal, it has no incentive to hide the shortest path it is able to get. Finally, Theorem 4.2 states that the client has no incentive to append any Sybil node on the reported path.

For a given relay, we consider the two attacks sequentially:

1) Edge insertion attack: For a relay on the selected shortest path, Theorem 4.2 states that inserting edge on the selected path does not benefit the relay. Inserting edge on any non-selected path only increases its length, and does not make it the shortest path, thus does not change the payment decision.

2) Edge hiding attack: for a relay on the selected shortest path P, hiding other paths does not have impact, and hiding the shortest path can result in two scenarios. First, another path that does not contain the relay is selected. Second, another path containing the relay but with length no shorter than P is selected. In both cases, the relay's payoff does not increase, hence there is no incentive for the relay to do so. For a relay not on the shortest path, hiding any path that containing it does not affect the shortest path being selected, thus its payoff remains zero. □

In Figure 4.5, all paths revealed to the client are shown at their revelation times. The maximum path length $N = 3$. Note that the client is not shown in the paths. Among all revealed paths, the client only accepts P_1, P_3, and P_6, as each of them is the single shortest path when they are revealed. The client reports the 1-hop path P_6 to the TTP at t_6, as there is no path shorter can be revealed. The client pays $Charge_{MDR}(1) = 2^3 - 2^{3-1} = 4$ cents, and relay U on the reported path is paid by $Reward_{MDR}(1) = 2^{3-1} = 4$ cents.

If the deadline t_d is between t_5 and t_6 instead, the client will report path P_3 at the new t_d. Relays Y and W on P_3 are paid, and each gets $Reward_{MDR}(2) = 2^{3-2} = 2$ cents, while the client is charged by $Charge_{MDR}(2) = 2^3 - 2^{3-2} = 6$ cents. The surplus is $6 - 2 \times 2 = 2$ cents. Note that, there are multiple sources (node U and node Y) in this example.

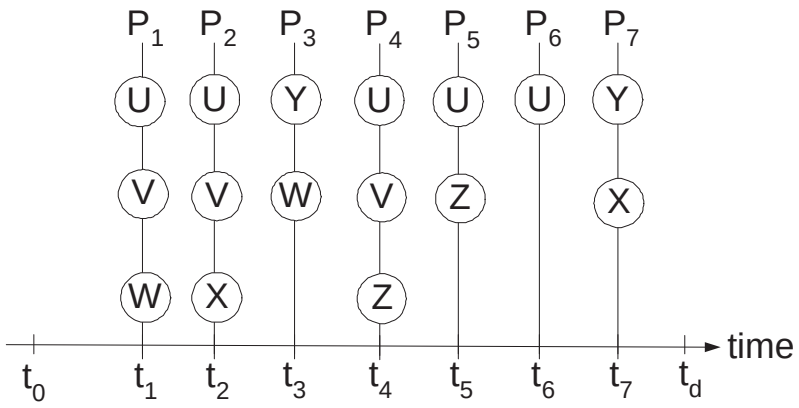


Figure 4.5: Paths revealed over time axis

4.5.2 Delay-sensitive Client

In this case, the decryption keys for data are given to the client by the TTP immediately when the earliest path is revealed. Designing incentive-compatible scheme for delay-sensitive clients is more complicated than for cost-sensitive clients because the payment decision can only be finalized after examining the rest of the paths. Therefore, a mechanism must be incorporated to motivate the client to continue to reveal paths to the TTP truthfully, even though it already has the decoded data.

Briefly, the **min-Delay Selection Algorithm** contains the following three steps:

1) Key revelation and initial payment by client: When the earliest path P_1 is revealed at t_1, the client immediately decrypts it through the TTP, and is charged $n \times 2^{N-1} + (2^n - 2)$ cents, where N is the maximum path length, and n is P_1's hop count.

2) Reimbursement to client for reporting eligible paths: The client continues to report eligible paths to the TTP, and the client is reimbursed 1 cent for every *eligible path* it reports to the TTP.

3) Payment set selection: Based on the eligible path sequence that the client reports, the TTP decides the set of relays $\mathbb{R}$ to be paid. Once $\mathbb{R}$ is determined, MDR payment algorithm is applied over $\mathbb{R}$ to calculate the payment to relays.

We discuss the steps in more detail as follows:

Initial payment: In this step, the first portion of the payment $n \times 2^{N-1}$ prevents the

client from gaining by inserting a Sybil node in the earliest path and claiming back the maximum reward 2^{N-1} with the inserted Sybil node. The second portion of the payment $2^n - 2$ is the provident fund to pay the client for reporting eligible paths (maximum $2^n - 2$ paths with 1 cent each) in the next step.

For example, in Figure 4.5, the earliest path P_1 is used for decoding the message and calculation of the client's initial payment. As $n = 3$, the client pays $n \times 2^{N-1} + (2^n - 2) = 3 \times 2^{3-1} + (2^3 - 2) = 18$ cents.

Eligible path: Ideally, information about all paths can be collected. However, the number of paths can be unbounded. Furthermore, if there is no eligibility constraint on the path, the client can fake any number of paths by appending its Sybil nodes on the earliest path or forging a path with only its Sybil nodes, to earn the reimbursement without receiving and reporting any real path. We define an *eligible path* in the following way.

Definition 4.7 *A path P is an* eligible path, *if and only if the intersection of its relays and the relays on the earliest path P_1 is a* unique non-empty *subset of $Relay(P_1)$.*

Uniqueness is defined in the following way. A path P is an eligible path if there is no other eligible path P' such that $Relay(P') \cap Relay(P_1) = Relay(P) \cap Relay(P_1)$.

The eligible path is defined to meet the following three conditions: (1) the size of the eligible path set must be bounded from above; (2) cheating from the client cannot increase the eligible path set; and finally (3) the TTP must be able to calculate an incentive compatible payment based on the eligible path set.

Condition (1) is clearly met since the number of non-empty subsets of $Relay(P_1)$ is $2^n - 1$. Each non-empty subset corresponds to at most one eligible path, thus the number of eligible paths (excluding P_1 itself) is bounded by $2^n - 2$, which corresponds to the amount we charged in the second portion of the initial payment by the client.

Condition (2) is met for the following two reasons. First, there is no Sybil node in P_1, and the size of eligible path set does not depend on any node outside $Relay(P_1)$. Thus, the client cannot construct phantom eligible paths. Second, reordering of eligible paths

does not change the size of the eligible path set. As the client does not gain by altering the order of the reported paths, we assume that it reports the eligible paths in the order they are revealed. The client can accumulate all paths, and report them to the TTP in one message.

Finally, condition (3) is met when the eligible path selection is used in conjunction with the relay payment set selection to be presented later.

We illustrate the determination of the eligible paths using Figure 4.5. Among all paths revealed after P_1, only path P_2, P_3, and P_5 are eligible. The total reimbursement to the client for these three eligible paths is 3 cents. Path P_4 and P_6 are not eligible paths due to the uniqueness constraint. Note that, the client can hide P_2 to make P_4 an eligible path. However, doing this does not increase the client's reimbursement. Finally, path P_7 is not an eligible path because its intersection with P_1 is empty.

Payment set selection: Denote the initial payment set as $\mathbb{R}_1 = Relay(P_1)$. The payment set is updated every time an eligible path is revealed. The update rule is as follows. Suppose before an eligible path P is revealed, the payment set is $\mathbb{R}_i$. If $\mathbb{R}_i \cap Relay(P) \neq \emptyset$, then the payment set is updated to $\mathbb{R}_{i+1} = \mathbb{R}_i \cap Relay(P)$. Relays in the final payment set $\mathbb{R}_k$ will be paid.

Let us look at the evolution of the payment set in the example given by Figure 4.5. The eligible paths are $\{P_1, P_2, P_3, P_5\}$, and the initial payment set $\mathbb{R}_1 = \{U, V, W\}$. P_2 updates the payment set to $\mathbb{R}_2 = Relay(P_2) \cap \mathbb{R}_1 = \{U, V\}$. As P_3's intersection with $\mathbb{R}_2$ is $\emptyset$, P_3 is not used. P_5 updates the payment set to $\mathbb{R}_3 = Relay(P_5) \cap \mathbb{R}_2 = \{U\}$, which is the final payment set. Thus, only relay U is paid, and the reward is $Reward_{MDR}(|\mathbb{R}_3|) = 2^{3-1} = 4$ cents.

Note that, the correct calculation of payment set using the above selection algorithm does not require the revelation of all eligible paths. However, reimbursing all eligible paths is important to prevent the client from manipulating the report. Otherwise, if the TTP reimburses the client only for the eligible paths used in the computation, the client may have the incentive to hide some eligible paths so as to increase the number of the

eligible paths needed. This will result in the incorrect (non incentive compatible) computation of the relay payment set.

We introduce a lemma before we present and prove the main theorem in this section.

Lemma 4.3 *Under the payment set selection algorithm specified above, suppose the payment set is $\mathbb{R}_i$ at time t, given a relay $R \in \mathbb{R}_i$, for every eligible path P revealed before t, $R \in Relay(P)$ implies $\mathbb{R}_i - \{R\} \subset Relay(P)$.*

Proof: We prove it by contradiction. Suppose P^* is the earliest eligible path that is revealed before t and satisfies both $R \in Relay(P^*)$ and $\exists R' \neq R$ such that, $R' \in \mathbb{R}_i \quad \& \quad R' \notin Relay(P^*)$. Suppose the payment set when P^* is revealed is $\mathbb{R}^*$. As P^* is revealed before $\mathbb{R}_i$. $\mathbb{R}_i \subseteq \mathbb{R}^*$, thus $R' \in \mathbb{R}^*$ as $R' \in \mathbb{R}_i$. We have $\emptyset \subset Relay(P^*) \cap \mathbb{R}_k$, as $R \in Relay(P^*) \cap \mathbb{R}_k$. We also have $Relay(P^*) \cap \mathbb{R}_k \subset \mathbb{R}_k$, as $R' \notin Relay(P^*) \cap \mathbb{R}_k$ but $R' \in \mathbb{R}_k$. P^* is the earliest path satisfying this condition, so it should be used to update the payment set to $Relay(P^*) \cap \mathbb{R}^*$, which results in the removal of R' from payment set, and causes contradiction. □

Theorem 4.4 *Under the min-Delay allocation algorithm, both client and relay have no incentive to launch edge insertion attacks and edge hiding attacks.*

Proof: First, we show that the client's dominant strategy is to act truthfully:

1) Edge insertion attack: By inserting a Sybil node into the earliest path (increasing its length from n to $n+1$), the client need to pay an extra $[(n+1)2^{N-1} + (2^{n+1} - 2)] - [n2^{N-1} + (2^n - 2)] = 2^{N-1} + 2^n$ cents. What it can earn through the Sybil node is at most 2^{N-1} (if the Sybil node is the single relay in the final payment set) plus 2^n cents (by reporting 2^n extra eligible paths). As the net payoff is non-positive, the client has no incentive to insert Sybil node into the earliest path. Inserting Sybil node into latter paths does not change the eligible path set, thus does not benefit the client either.

2) Edge hiding attack: Hiding the earliest path is against the client's goal to minimize the delay to recover data. Hiding latter eligible paths only reduces the client's payoff. Thus, the client has no incentive to hide path.

We now prove that the dominant strategy for any relay is to act truthfully too, by examining three types of relays in turn.

1) For a relay R in the final payment set $\mathbb{R}_k$: On one hand, creating Sybil node R' to launch an edge insertion attack does not help, because: if R' is not in the final payment set, it does not earn R any extra reward. If R' is in the final payment set, the total amount earned by R and R' is $2 \times 2^{N-(|\mathbb{R}_k|+1)} = 2^{N-|\mathbb{R}_k|}$, which is equal to the reward of having R alone. On the other hand, launching edge hiding attacks does not benefit as well. If R is the only relay in the final payment set, it gets its optimal payment already. If $\mathbb{R}_k - \{R\}$ is not empty, using Lemma 4.3, all paths containing R also contain $\mathbb{R}_k - \{R\}$. Unless R eliminates itself from the final payment set, it cannot exclude any node in $\mathbb{R}_k - \{R\}$ from final payment set either.

2) For a relay R not on the earliest path, inserting or hiding edge cannot affect the revelation of the earliest path, thus does not bring it any reward.

3) Now let us consider a relay R on the earliest path, but is excluded from the final payment set. Without loss of generality, suppose R is eliminated from payment set $\mathbb{R}_{i-1}$ by a path P^*, i.e., $R \in \mathbb{R}_{i-1}$ but $R \notin \mathbb{R}_i$. Thus, P^* satisfies $\mathbb{R}_i \subset Relay(P^*)$ and $R \notin Relay(P^*)$. In addition, using Lemma 4.3, for every path P containing R that is revealed before P^*, $\mathbb{R}_i \subset Relay(P)$. Thus, to make itself appear in payment set before the revelation of P^i, R must make $\mathbb{R}_i$ appear in payment set also. In this case, P^* is always an eligible path to filter R out of the payment set. Even if R can hide all paths before P^i, P^i becomes the new earliest path, and it defines a new initial payment set which does not contain R at all. In this case, R still gets zero reward. Creating Sybil node does not prevent R (or any of its Sybil nodes) from being eliminated by path P^* either.

Thus, the min-Delay algorithm is incentive compatible. □

From Theorem 4.4, we directly have:

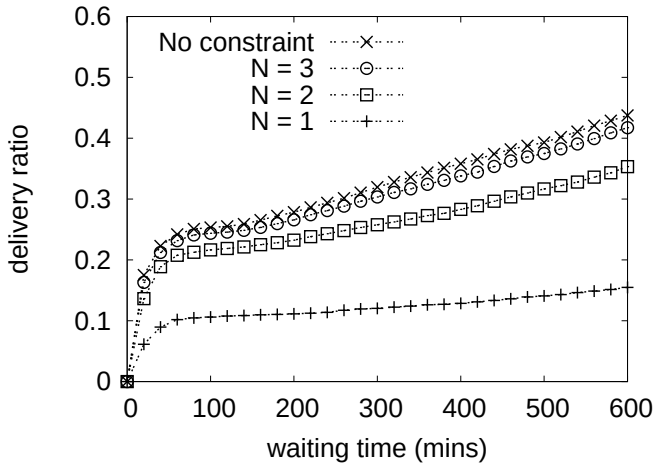
Corollary 4.2 *The min-Delay allocation algorithm reveals the earliest path, and the client's payment is bounded by $O(N \times 2^N)$, where N is the maximum path length allowed.*

4.6 Performance Evaluation

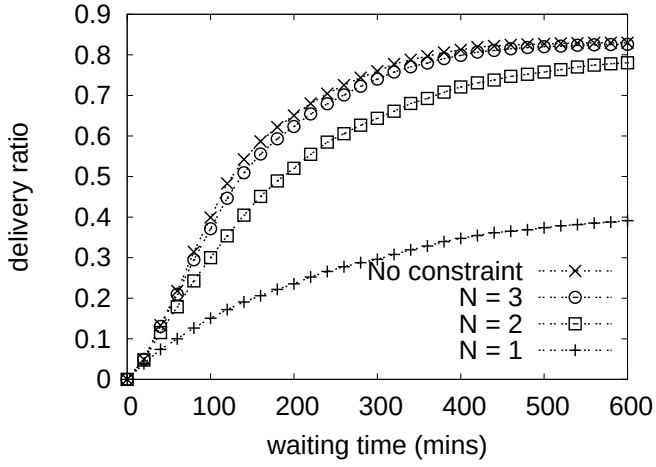
We evaluate *MobiCent* using the widely used traces from the Haggle project [48] and the DieselNet project [113], which represent human social networks and vehicular networks respectively. The Haggle trace [48] is collected in an experiment measuring forty-one humans' mobility at the Infocom 2005 conference. The device used to collect connection opportunity data and mobility statistics in the experiment is the Intel iMote. The iMotes were configured to perform a Bluetooth baseband layer "inquiry" discovering the MAC addresses of other Bluetooth nodes in range. The DieselNet trace [113] is taken from UMass DieselNet, a DTN consisting of Wi-Fi nodes attached to buses. As buses travel their routes, they encounter other buses and establish pair-wise bus-to-bus connections. The behavior of inter-contact times is important when considering the delay experienced by packets in a DTN. This is the time a node has to wait to get in contact with a specific node, counted from the moment from losing contact with that node. A closer look at the inter-contact distribution of the two traces shows that the inter-contact time in the Haggle trace tends to be longer than the inter-contact time in the DieselNet trace. For example, around 20% of inter-contact time in the Haggle trace is longer than 3 hour, whereas the value is only 10% in the DieselNet trace. This contributes to the difference in their delivery performance.

MobiCent treats the routing protocol as a black box and is independent of the specific algorithm used. Our evaluation uses epidemic routing, and assumes each contact has sufficient capacity to exchange data. Performance under other routing protocols and constrained contact capacity show similar trends, and are not presented here to save space. Each experiment below is carried out 500 times with different random seeds, and the average is presented.

We first evaluate the impact of hop count constraint on delivery performance. When all nodes are honest, we show that even if we set the maximum hop constraint N to a small value (3 to 5), the delivery performance already approximates the setting without



(a) Haggle trace



(b) DieselNet trace

Figure 4.6: Impact of hop count constraint

any constraint closely. Next, we evaluate the behavior of selfish nodes operating under the natural *earliest-path fixed-amount* payment scheme such that cheating may result in gains for some nodes. We show that cheating becomes the strategy of the majority of nodes, and overall delivery performance degrades significantly. Payment schemes described by Jakobsson et al. [50] and Zhong et al. [118] have the same vulnerability, as none of them satisfy the properties we identified for incentive-compatible payment scheme in Section 4.4. Lastly, we show the behavior of selfish nodes operating under *MobiCent*, and plot the resulted delivery performance as well as amount of payment by the client.

4.6.1 Hop Count Limit

To evaluate the impact of hop count limit, we plot the delivery ratio over time where the maximum hop count is limited to 1 (direct delivery), 2, and 3, against the setting where there is no hop count constraint. We assume all nodes act honestly.

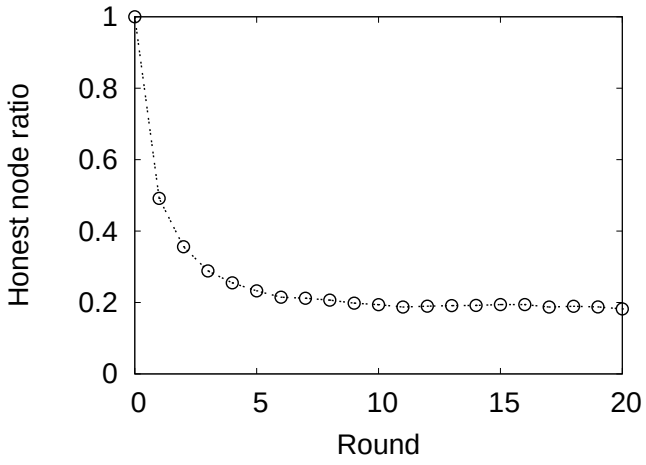
Figure 4.6 (a) plots the delivery ratio as a function of waiting time for the Haggle trace under various maximum hop constraints of forwarding path. As shown in the figure, for any given deadline, the delivery ratio increases with the maximum hop count allowed. Allowing 2-hop forwarding almost doubles the delivery performance of the 1-hop-only forwarding, while 3-hop forwarding achieves more than 95% of the delivery ratio at any given deadline compared to the case without hop count constraint. Though not shown in the figure, 5-hop forwarding achieves more than 99% of delivery performance. Similar result is shown in Figure 4.6 (b) for the DieselNet trace. As a small N such as N (≤ 5) suffices in most cases, the multiplicatively increasing payment of proposed schemes is practically affordable, as will be shown later.

4.6.2 Cheating under Earliest-path Fixed-amount Scheme

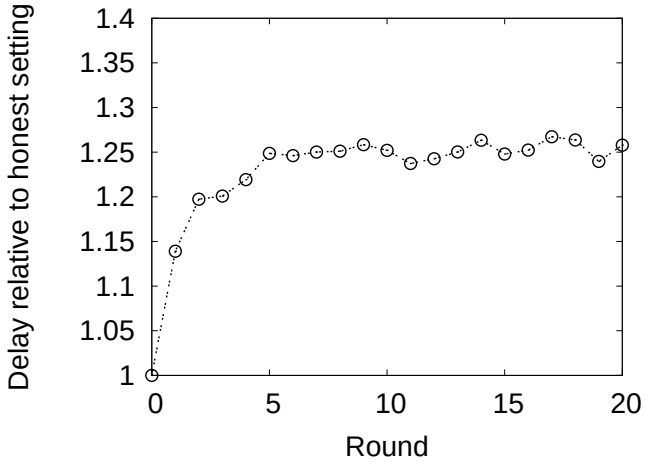
We study the user behavior under the earliest-path fixed-amount payment scheme, where a client pays a fixed amount (3 cents) to relays on the earliest path for each block delivered. The amount is shared equally by all relays on the earliest forwarding path.

Figure 4.7 illustrates the system behavior using the Haggle trace when relays can cheat by hiding edges or creating Sybil nodes to increase their own payoff. In each round, each user generates two requests on average. There are two possible strategies: acting truthfully or cheating. In the first round, all relays start truthfully. After each round, we assume that each relay has access to the revealed contact graph and varies its strategy in the next round if it has a higher expected payoff with the new strategy based on its own experience in the current round.

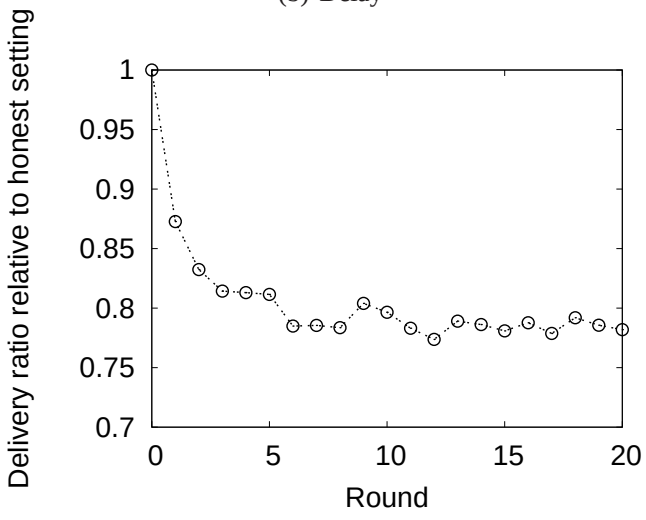
The nodes' behavior is shown in Figure 4.7 (a). Starting from a ratio of 100%, the



(a) Honest node ratio

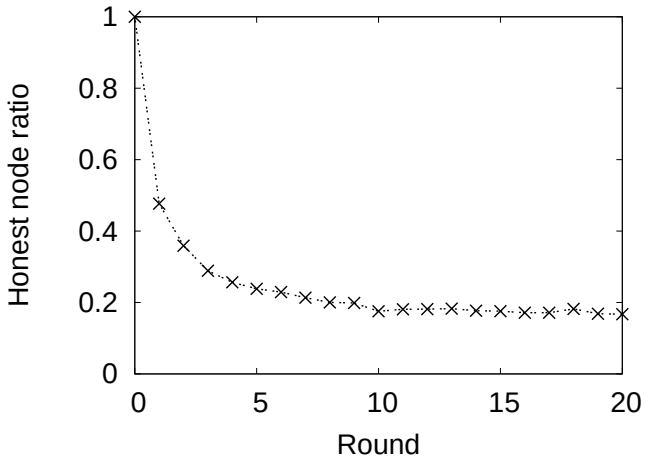


(b) Delay

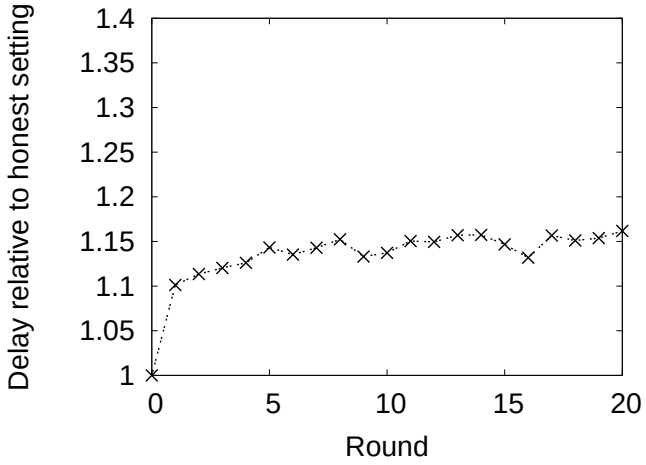


(c) Delivery ratio

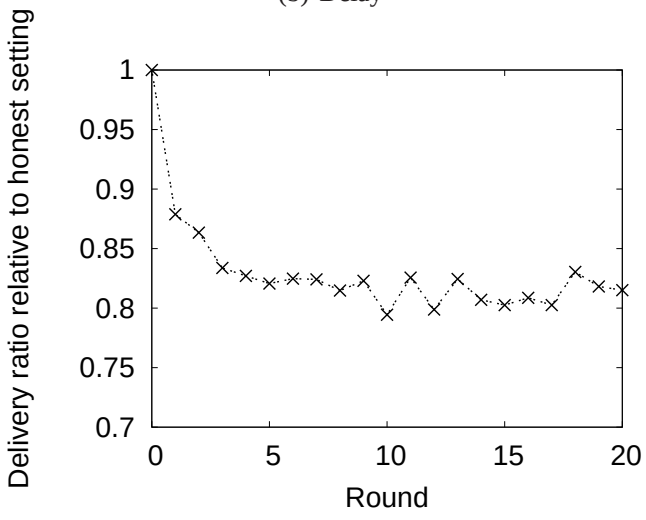
Figure 4.7: Evolution of user behavior and delivery performance under earliest-path fixed-amount payment scheme (Haggle trace)



(a) Honest node ratio



(b) Delay



(c) Delivery ratio

Figure 4.8: Evolution of user behavior and delivery performance under earliest-path fixed-amount payment scheme (DieselNet trace)

ratio of honest users keeps decreasing and after 10 rounds, the system converges to a sub-optimal state. Note that, cooperation may still be preferred by some users (20%), as forwarding to other relay (honestly) increases the chance the node is on the selected path, which compensates the loss in having to share the reward with others.

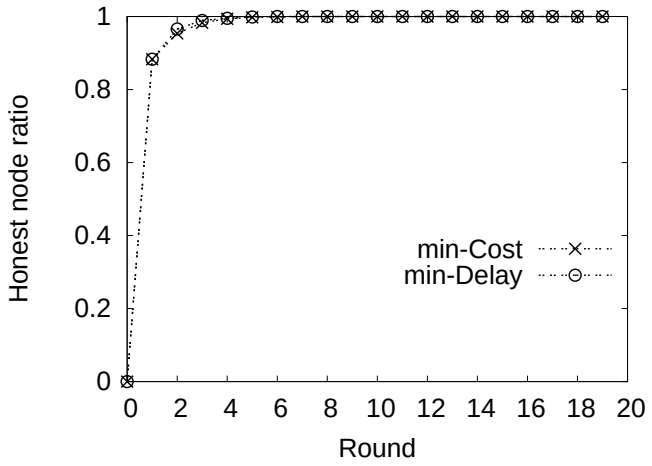
Figure 4.7 (b) shows that the delivery delay increases under attack. The average delay is increased by 25%. As shown in Figure 4.7 (c), delivery ratio decreases by around 20% under attack.

Figure 4.8 demonstrates similar trends for the DieselNet trace.

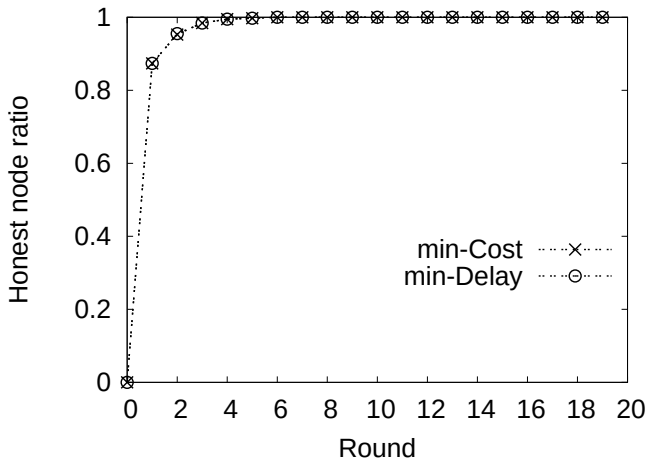
Another way to measure the impact of dishonest nodes is to consider the relative gain of dishonest nodes vs. the honest nodes. When the ratio of dishonest nodes is fixed at 20%, simulation result shows that they collect more than 33% of the reward for both the Haggle trace and the DieselNet trace. The average reward of honest participants is reduced by around 20%, and is only around half the reward earned by cheating participants. When the ratio is increased to 50%, they collect 65% of the reward in the Haggle trace and 75% of reward in the DieselNet trace. In the latter trace, honest node's reward is reduced by 50%, and is only $1/3$ of the rewards of dishonest nodes. This indicates that a large portion of dishonest nodes can significantly decrease the reward for honest nodes. This has the effect of discouraging honest nodes from joining the system, further reducing the overall performance.

4.6.3 MobiCent Performance

In order to evaluate how *MobiCent* fosters cooperation, we repeat the previous experiment but with all nodes initially cheating. As shown in Figure 4.9 (a), for both the min-Cost algorithm and the min-Delay algorithm under the Haggle trace, from a state where all players cheat and each player adapts its behavior based on its experience, all players converge to the truth-telling strategy very quickly, with 90% choosing to act truthfully after only 1 round. After 4 rounds, all nodes act truthfully and no node deviates from the truthful strategy any further. Such behavior applies also to the min-Cost algorithm and



(a) Haggle trace



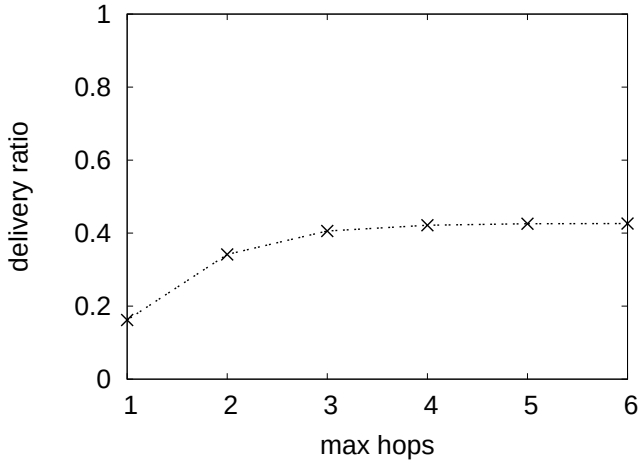
(b) DieselNet trace

Figure 4.9: Evolution of user behavior under MobiCent

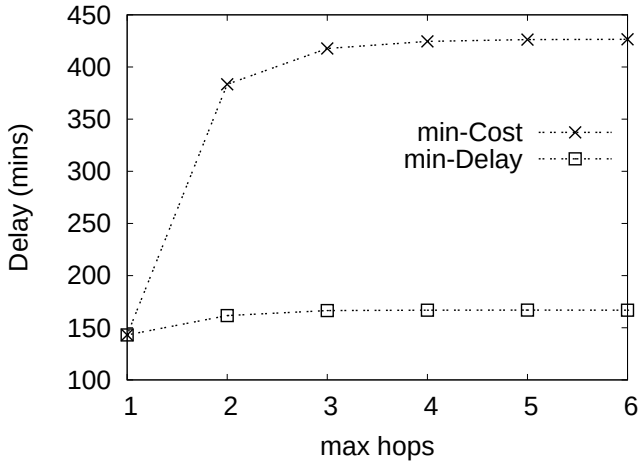
the min-Delay algorithm for the DieselNet trace, as shown in Figure 4.9 (b).

Figure 4.10 (a) shows the delivery ratios for the Haggle trace using both the min-Cost algorithm and the min-Delay algorithm. The delivery ratios of both algorithms are identical and equal to the case where all nodes act honestly. This is expected since both of these algorithms ensure that there is no edge insertion and hiding attacks.

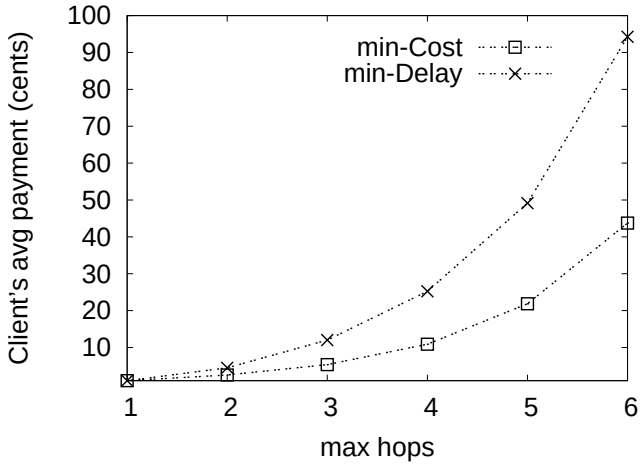
Figure 4.10 (b) plots the average delay for a client to recover data under both algorithms for the Haggle trace. The deadline is set to 600 minutes (10 hours). Since the first path received is reported in the min-Delay algorithm, the delay achieved is the same as the case where all nodes are honest. When $N = 1$, the earliest path is also a 1-hop path,



(a) Delivery ratio

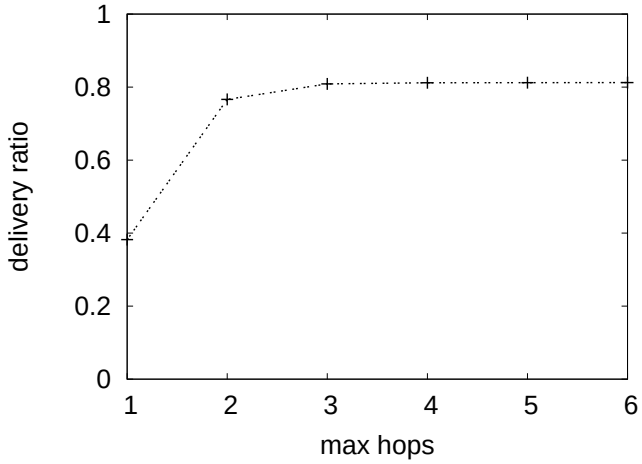


(b) Delivery delay

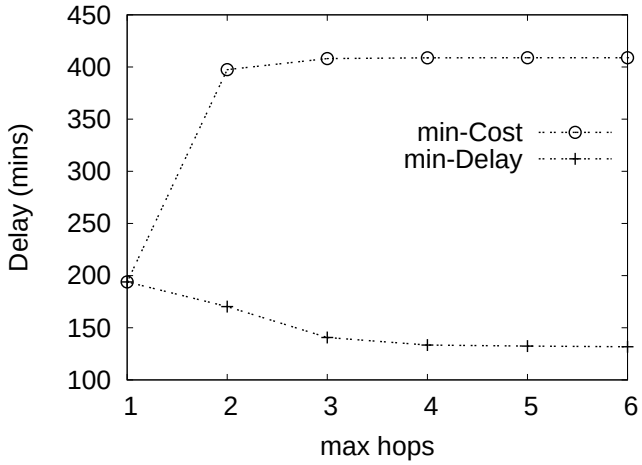


(c) Payment

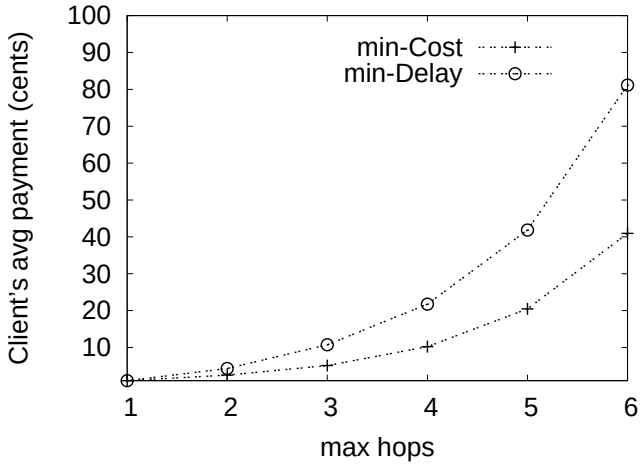
Figure 4.10: MobiCent performance under varying hop count constraint (Haggle trace)



(a) Delivery ratio



(b) Delivery delay



(c) Payment

Figure 4.11: MobiCent performance under varying hop count constraint (DieselNet trace)

thus the delay for both algorithms are identical. When $N > 1$, the min-Delay algorithm still recovers data in the earliest path, whereas the min-Cost algorithm needs to wait until the revelation of a 1-hop path or the deadline, whichever is earlier. As shown in the figure, the delay for the min-Cost algorithm is more than 100% more than the min-Delay algorithm. The client is compensated for this large increase in delay by having to pay a smaller amount of money to the TTP.

Figure 4.10 (c) plots the average payment by a client under both algorithms for the Haggle trace. Recall that, as the maximum hop count N grows, the maximum payment grows at $O(2^N)$ and $O(N \times 2^N)$ respectively for the min-Cost algorithm and the min-Delay algorithm. The figure shows that the average payment grows in an exponential rate. However, as the average length of the earliest path is around 2, the average payment by a client under the min-Delay algorithm is roughly two times of the average payment under the min-Cost algorithm. Also recall that, when $N = 3$, the performance obtained is close to the case without hop count constraint, in terms of both delivery ratio and delay. For $N = 3$, the average cost for the min-Cost algorithm is 5.36 cents, and the average cost for the min-Delay algorithm is 12.01 cents. Therefore, the payment is practically affordable based on the current traces used, despite the exponential growth.

Figure 4.11 demonstrates similar trend under the DieselNet trace.

4.6.4 Implementation Issues

We discuss two implementation issues, namely encryption key size and computation overhead.

There are two types of encryption keys. Public key encryption used is based on Elliptic Curve Cryptography (ECC) and 192-bit keys are used. The signature generated is 48 bytes. For symmetric key encryption, 128-bit AES algorithm is used. In order to reduce overhead, a 192-bit request identifier r_{id} can be selected with its signature computed by the TTP. These identifier and signature pairs can be used in the packet header instead of the original request string r. Assuming a 16KB data block and an average path hop

count of 2, the average overhead imposed by the header and supplementary layer is about 250 bytes, which is less than 2% of the 16KB data block. Note that since the reward for breaking the *MobiCent*'s encryption is relatively small, the one-time key size can be even smaller in practice.

In order to evaluate the computation overhead, we measure the encryption and verification time of ECC on the target implementation platform, a Soekris Net5501 box. Using the OpenSSL library, measurements show that the average signing time is 15ms and the average verification time is 20ms. The results show that these encryption schemes do not impose significant overhead. In fact, researchers have shown that it is viable to use public-key cryptography even on low power energy constraint platform using a 8-bit processor (Atml ATmegal128L), in particular, if ECC is used [107]. Finally, note that these encryption and verification tasks do not have to be performed in real-time and can be performed during the disconnected periods between contacts.

4.7 Related Work

In this section, we present related work of incentive scheme design in both Peer-to-Peer (P2P) network and wireless network.

4.7.1 Incentive Techniques in P2P Network to Avoid Free-riding

It is widely agreed that some form of incentive is needed for P2P network to overcome the free-riding problem, i.e., downloading files from the network without uploading any in return. The three main incentive mechanisms being studied in literature are reputation, barter (or Tit-for-Tat), and virtual currency.

In general, a P2P reputation scheme is coupled with a service differentiation scheme. Contributing peers possess good reputations and receive good service from other peers. For example, peers in the KaZaA file-sharing network [54] build up their reputation scores by uploading files to others, and are rewarded with higher priority when downloading files

from others.

Reputation-base approach is known to suffer from the Sybil attack [28] and the whitewashing attack [38]. Douceur [28] coins the name of Sybil attack. In a Sybil attack, a single malicious peer generates multiple identities that collude with one another. Multiple colluding peers may boost one another's reputation scores by giving false praise, or punish a target peer by giving false accusations. In a whitewashing attack, a peer defects in every P2P transaction, but repeatedly leaves and rejoins the P2P system using newly created identities, so that it will never suffer the negative consequences of a bad reputation. The availability of cheap pseudonyms in P2P systems makes reputation systems vulnerable to Sybil attacks and whitewashing attacks. Such attacks can also be easily launched in our target environment.

BitTorrent file-sharing system adopted an incentive mechanism based on barter (or Tit-for-Tat). By partitioning large files such as movies and software binaries into small chunks, file-sharing using the BitTorrent protocol necessitates repeat interactions among peers, allowing cooperation to flourish based on direct reciprocity rather than indirect reciprocity. Yet, analysis has demonstrated that the BitTorrent protocol can still be manipulated by selfish peers in their favor, and fixes are suggested [62].

Tit-for-Tat does not suit our target environment, because in our environment, one peer is likely to want more service from another peer than it could provide to that peer. In such a situation, a credit-based system can better support the asymmetric transactions needed.

The use of virtual currency for incentives has also been proposed in several P2P content distribution systems, e.g., KARMA [105] and Dandelion [98]. However, they are designed for connected networks and will not work in a multi-hop setting with frequent disconnections.

4.7.2 Security Protocol and Incentive Scheme in Wireless Networks

There are a number of incentive schemes for wireless networks. Incentive is needed for wireless networks with user-contributed forwarding (e.g. mobile ad hoc networks)

to overcome the free-riding problem, i.e., requesting others for forward its packets, but avoiding to transmit others' packets.

Incentive schemes based on micro-payment have been proposed in wireless networks. FON [37], the largest community-based Wi-Fi ISP, has officially used its Wi-Fi Money to encourage its member to cooperate. Every time a visitor (non-Fonero) uses a FON Wi-Fi network, the owner can earn some "dinero" according to the time it connected. However, they only need to motivate their next-hop neighbor, whereas in *MobiCent*, multiple helpers need to cooperate. Jakobsson et al. [50] discuss a micro-payment scheme to encourage collaboration in multi-hop cellular networks. Zhong et al. [118] propose Sprite, a cheat-proof, credit-based system for stimulating cooperation among selfish nodes in mobile ad hoc networks. Anderegg and Eidenbenz [4] and Zhong et al. [117] propose pricing schemes based on use of VCG mechanism.

These schemes are not suitable for DTNs due to the following reasons. First, a common assumption adopted in these schemes is that an end-to-end connection between the source and the destination is established before the data forwarding occurs. Second, the reported schemes are mainly designed for single path forwarding.

Recently, several works address the incentive problem in delay-tolerant network. Shevade et al. [97] propose the use of pair-wise Tit-for-Tat (TFT) as incentive mechanism for DTNs. They enhance their TFT mechanism with generosity and contrition to address the bootstrapping and link variation problem. However, their proposal is not suitable for DTN routing scenarios where the delivery path cannot be pre-determined. In addition, Tit-for-Tat is not suitable for our target environment, where there is a large population of participants and a peer is likely to want much more service from another peer than it could provide to that peer. Zhu et al. [119] propose a secure credit-based incentive scheme for DTNs, with an emphasis on generation and verification of secure bundle. They do not address the pricing issue. The link insertion (Sybil) attack is not considered in both works.

Newsome et al. [80] and Piro et al. [86] propose mechanisms to defend against the Sybil attack in wireless networks. The basic idea is to test the resource of a node. Based

on the observation that a given node only has limited resource (say, a single Wi-Fi radio), a testing node can assign its neighbors into different channels, and randomly probes for a neighbor in the channel specified. If a node mimics several Sybil nodes that are assigned to different channels, as it can only appear in one channel in any given time, the probability that one of its Sybil nodes is caught is high. Jakobsson et al. [50] use statistic techniques to detect the Sybil attack in multi-hop cellular networks over a long period of time. However, it is much more difficult to detect the Sybil attack in DTN, where disconnection is the norm rather than exception and high user population dynamic is expected. As a result, these techniques cannot be applied.

4.8 Summary

This chapter presents *MobiCent*, a credit-based incentive system for DTN and proves that it is incentive compatible. *MobiCent* uses a Multiplicative Decreasing Reward (MDR) algorithm to calculate payment and supports two types of clients, namely clients that want to minimize cost or minimize delay. Simulation results show that *MobiCent* can effectively foster cooperation among selfish nodes with bounded overhead.

Chapter 5

Conclusion and Future Works

Mobile communication system is experiencing a fast and exciting evolution, driven by both convergence of heterogeneous wireless networks and development of new cooperative networking approaches. Great efforts have been devoted to build flexible architecture capable of managing various network components as a whole, while new network approaches are being proposed to harvest the potential performance improvement of cooperation.

Users play a more central role in the stage. With increased intelligence, the new generation of wireless terminals not only can facilitate the radio resource allocation decision by feeding back the measured channel state, but also can contribute directly to the resource provision process by forwarding data for each other. As users gain more control over their devices, an intelligent and selfish user can adapt its behavior in order to benefit more from the network, even when doing so may affect other users and the system's overall utility.

The design of new cooperative resource allocation and provision schemes should explore the cooperation possibility among heterogeneous wireless network components and their users, while taking the selfish nature of users and their strategic interactions into consideration. This thesis systematically investigates several fundamental design problems of how to deliver Internet access service efficiently to (selfish) users using heterogeneous

wireless networks.

5.1 Research Summary

As stated in Chapter 1, this thesis studies both the overlapping-coverage scenario and intermittent-coverage scenario. For each scenario, we approach the problem from both the system performance perspective and the incentive compatibility perspective.

Chapter 2 focuses on the overlapping-coverage scenario. It studies the *coordinated radio resource allocation problem* for users that are simultaneously covered by multiple overlapping heterogeneous wireless networks. We formulate the *coordinated proportional fairness (CPF)* resource allocation criterion, based on which a globally fair and efficient allocation decision can be easily computed. As *CPF* decision depends on the input from users, a selfish user may manipulate its channel state report if doing so can increase its gain from the network. To capture this phenomenon, we formulate the resource allocation process as a *multi-cell resource allocation game*, which is associated with a rule to calculate bandwidth allocation outcome based on the input from the MS players. We prove that a multi-cell resource allocation game with *CPF* allocation is incentive compatible, which means a user's dominant strategy is to report its channel state honestly. In practice, the single-association setting, where a MS is only associated with one BS, is often desirable. We formulate the integral version of the *CPF* problem (*Int-CPF*) and show that it is both computationally expensive and prone to user-manipulation. Alternatively, we advocate the adoption of a *Selfish Load Balancing (SLB)* scheme, which always leads to a Nash equilibrium, and often achieves performance near to *CPF* allocation. We use simulation to evaluate the performance of proposed schemes. Our results show that the proposed algorithms outperform popular heuristic approaches, by striking a good balance between efficiency and fairness, while achieving load balancing among component BSs.

Chapter 3 and Chapter 4 focus on the intermittent-coverage scenario. Chapter 3 presents *MobTorrent*, a cooperative, on-demand framework, which uses the ubiquitous

low-bandwidth cellular network as a control channel while forwarding data through high-bandwidth contacts in a DTN paradigm. We design the architecture of *MobTorrent*, and analyze the problem of how to schedule the transmission over intermittent contacts, such that the amount of data delivered is maximized and the delay is minimized. We use both testbed and trace-driven simulation to evaluate the performance of *MobTorrent*.

Chapter 4 presents *MobiCent*, a credit-based incentive system for DTN. Following the algorithmic mechanism design approach, we formulate the path revelation game, and analyze the attack model. A message exchange protocol is carefully constructed to support the requirement of *MobiCent*, and two different algorithms are designed to cater to client that wants to minimize either payment or data delivery delay. We prove that both algorithms are incentive compatible, as rational nodes will not purposely waste any opportunistic transfer or cheat by creating non-existing contacts to increase their rewards.

To summarize, this thesis analyzes the opportunities and challenges that appear in the forthcoming generation of mobile communication systems. We develop novel models and techniques that can be used to exploit the new cooperative opportunities, and address the challenges to foster cooperation.

5.2 Future Work

There are several possible extensions to the research work presented in this thesis.

- In our system model of overlapping cells, we assume that each cell has a fixed amount of radio resource and they operate orthogonally. For future research, we would like to incorporate the BS capacity adaptation and interference mitigation into the consideration of the network-wide radio resource allocation.

- For the coordinated resource allocation problem in a convergent platform, we assume that the ownership of radio cards is known by the network and cannot be modified by users. Though it is a valid assumption for existing networks, the increase

of system openness will eventually enable users to game the system by manipulating their radio card ownership as well. A resource allocation scheme should be designed to address the arising challenges. In addition, it remains a research problem to design an efficient incentive-compatible scheme for the single-association setting.

- When studying the incentive compatibility of the radio resource allocation problem, we focus on preventing users from cheating. As future mobile communication system is an open environment where even the normal residential users can operate as service provider, it is important to investigate the design of incentive-compatible schemes that are robust to cheating of service providers as well.

- *MobTorrent* is designed for mobile users travelling with vehicles, and the performance is evaluated under such settings. We are looking towards the possibility of applying the idea of *MobTorrent* to human social networks. The mobility pattern of human is shown to be predictable by Srinivasan et al. [100]. However, the uncertainty tends to be greater, and the properties of the time-varying connectivity graph are significantly different. In addition, the power consumption constraint of hand-held devices is much more stringent. These factors raise new challenges that require systematic investigations.

- It is worth investigating the design of intelligent applications and transport protocols for mobile users, such that they can fully exploit the complementary characteristics of two types of networks, one is highly available but with low bandwidth, and the other is only available intermittently but provides high-bandwidth connections.

- As the delay-tolerant networking paradigm plays a more important role for mobile Internet service provision, we are looking towards evaluating *MobiCent*'s performance involving real users. Depending on the characteristics of applications and user behaviors, further extensions of *MobiCent* can be expected.

- The *MobiCent* pricing scheme provides a deterministic guarantee for incentive compatibility regardless of the mobility pattern of users. If we relax this requirement, and aim at providing a stochastic guarantee about the user behavior instead, better performance can possibly be achieved, in terms of both the frugality and the efficiency. Further optimization can be expected by customizing the scheme according to some specific characteristics of mobility patterns and routing protocols.

Bibliography

[1] 3GPP. Feasibility study on 3GPP system to Wireless Local Area Network (WLAN) interworking. *3GPP TR 22.934 V6.2.0*, September 2003.

[2] 3GPP. Requirements on 3GPP system to Wireless Local Area Network (WLAN) interworking (Release 6). *3GPP TS 22.234 V6.0.0*, March 2004.

[3] G. Aggelou and R. Tafazolli. On the relaying capacity of next-generation GSM cellular networks. *IEEE Personal Communications Magazine*, 8(1):40–47, February 2001.

[4] L. Anderegg and S. Eidenbenz. Ad hoc-VCG: a truthful and cost-efficient routing protocol for mobile ad hoc networks with selfish agents. In *MobiCom*, San Diego, CA, USA, September 2003.

[5] ANSI/IEEE. Standard 802.11b, part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications: Higher-speed physical layer extension in the 2.4 GHz band, 1999.

[6] A. Archer and E. Tardos. Frugal path mechanisms. In *SODA*, San Francisco, CA, USA, January 2002.

[7] A. Balasubramanian, B. Levine, and A. Venkataramani. DTN routing as a resource allocation problem. In *SIGCOMM*, Kyoto, Japan, August 2007.

[8] A. Balasubramanian, R. Mahajan, A. Venkataramani, B. Levine, and J. Zahorjan. Interactive WiFi connectivity for moving vehicles. In *SIGCOMM*, Seattle, WA, USA, August 2008.

[9] A. Balasubramanian, Y Zhou, W. Croft, B. Levine, and A. Venkataramani. Web search from a bus. In *CHANTS*, Montreal, QC, Canada, September 2007.

[10] Y. Bejerano, S. Han, and L. Li. Fairness and load balancing in wireless LANs using association control. In *MobiCom*, Philadelphia, PA, USA, September 2004.

[11] P. Bender, P. Black, M. Grob, R. Padovani, N. Sindhushyana, and S. Viterbi. CDMA/HDR: a bandwidth efficient high speed wireless data service for nomadic users. *IEEE Communications Magazine*, 38(7):70–77, July 2000.

[12] D. Bertsekas and R. Gallager. *Data Networks (Second edition)*. Prentice Hall, 1992.

[13] B. Bhargava, X. Wu, Y. Lu, and W. Wang. Integrating heterogeneous wireless technologies: a cellular aided mobile Ad Hoc network (CAMA) . *Mob. Netw. Appl.*, 9(4):393–408, 2004.

[14] G. Bianchi and I. Tinnirello. Channel-dependent load balancing in wireless packet networks. *Wireless Communications and Mobile Computing*, 4(1):43–53, February 2004.

[15] E. Brewer, R. Katz, E. Amir, H. Balakrishnan, Y. Chawathe, A. Fox, S. Gribble, T. Hodes, G. Nguyen, V. Padmanabhan, M. Stemm, S. Seshan, and T. Henderson. A network architecture for heterogeneous mobile computing. *IEEE Personal Communications Magazine*, 5(5):8–24, October 1998.

[16] T. Bu, L. Li, and R. Ramjee. Generalized proportional fair scheduling in third generation wireless data networks. In *IEEE INFOCOM*, Barcelona, Spain, April 2006.

[17] M. Buddhikot, G. Chandranmenon, S. Han, Y.W. Lee, S. Miller, and L. Salgarelli. Integration of 802.11 and third generation wireless data networks. In *INFOCOM*, San Francisco, CA, USA, March 2003.

[18] J. Burgess, B. Gallagher, D. Jensen, and B. Levine. MaxProp: Routing for vehicle-based Disruption-Tolerant Networks. In *INFOCOM*, Barcelona, Spain, April 2006.

[19] B. Burns, O. Brock, and B. Levine. MV routing and capacity building in Disruption Tolerant Networks. In *INFOCOM*, Miami, FL, USA, March 2005.

[20] L. Buttyan and J. Hubaux. *Security and cooperatation in wireless networks: Thwarting malicious and selfish behavior in the age of ubiquitous computing*. Cambridge University Press, 2007.

[21] V. Bychkovsky, B. Hull, A. Miu, H. Balalkrishnan, and S. Madden. A measurement study of vehicular Internet access using in situ Wi-Fi networks. In *MobiCom*, Los Angeles, CA, USA, September 2006.

[22] R. Cavallo. Optimal decision-making with minimal waste: Strategyproof redistribution of VCG payments. In *AAMAS*, Hakodate, Hokkaido, Japan, March 2006.

[23] R. Chakravorty, S. Agarwal, and S. Banerjee. MoB: A mobile bazaar for wide-area wireless services. In *MobiCom*, Cologne, Germany, August 2005.

[24] B. Chen and M. Chan. Proportional fairness for overlapping cells in wireless networks. In *VTC Fall*, Montreal, QC, Canada, Septempber 2006.

[25] B. Chen and M. Chan. MobTorrent: A framework for mobile Internet access from vehicles. In *INFOCOM*, Rio de Janeiro, Brazil, April 2009.

[26] B. Chen and M. Chan. MobiCent: a credit-based incentive system for disruption tolerant network. In *INFOCOM*, San Diego, CA, USA, March 2010.

[27] S. Das, H. Viswanathan, and G. Rittenhouse. Dynamic load balancing through coordinated scheduling in packet data systems. In *INFOCOM*, San Francisco, CA, USA, March 2003.

[28] J. Douceur. The sybil attack. In *IPTPS*, Cambridge, MA, USA, March 2002.

[29] A. Doufexi, E. Tameh, A. Nix, S. Armour, and A. Molina. Hotspot wireless LANs to enhance the performance of 3G and beyond cellular networks. *IEEE Communications Magazine*, 41(7):58–65, July 2003.

[30] Y. Du, R. Sami, and Y. Shi. Path auction games when an agent can own multiple edges. In *NetEcon*, Ann Arbor, MI, USA, June 2006.

[31] E. Elkind, A. Sahai, and K. Steiglitz. Frugality in path auctions. In *SODA*, New Orleans, LA, USA, January 2004.

[32] J. Eriksson, H. Balakrishnan, and S. Madden. Cabernet: Vehicular content delivery using WiFi. In *MobiCom*, San Francisco, CA, USA, September 2008.

[33] EVEREST. Deliverable 11: First report on the evaluation of RRM/CRRM algorithms, November 2004.

[34] D. Everitt. Traffic capacity of cellular mobile communications systems. *Computer Networks and ISDN Systems*, 20(1):447–454, 1990.

[35] K. Fall. A Delay-Tolerant Network architecture for challenged Internets. In *SIGCOMM*, Karlsruhe, Germany, August 2003.

[36] F. Fitzek, F. Bertocchi, and M. Zorzi. Complementing 3G cellular networks by multi hop capabilities of an enhanced MAC protocol for wireless LAN. In *VTC Spring*, Milan, Italy, May 2004.

[37] FON. wireless access provider. http://www.fon.com/en/info/makeMoney.

[38] E. Friedman and P. Resnick. The social cost of cheap pseudonyms. *Journal of Economics and Management Strategy*, 10(2):173–199, 1998.

[39] M. Frodigh, S. Parkvall, C. Roobol, P. Johansson, and P. Larsson. Future-generation wireless networks. *IEEE Personal Communications*, 8(5):10–17, October 2001.

[40] T. Fujii and S. Nishioka. Selective handover for traffic balance in mobile radio communications. In *ICC*, Chicago, IL, USA, June 1992.

[41] M. Gairing, B. Monien, and K. Tiemann. Routing (un-)splittable flow in games with player-specific linear latency functions. In *ICALP*, Venice, Italy, July 2006.

[42] M. Heusse, F. Rousseau, G. Sabbatel, and A. Duda. Performance anomaly of 802.11b. In *IEEE INFOCOM*, San Francisco, CA, USA, March 2003.

[43] H. Holma and A. Toskala (editors). *W-CDMA for UMTS*. John Wiley and Sons, 2000.

[44] H. Hsieh and R. Sivakumar. Performance comparison of cellular and multi-hop wireless networks: a quantitative study. In *SIGMETRICS*, Cambridge, MA, USA, June 2001.

[45] H. Hsieh and R. Sivakumar. A transport layer approach for achieving aggregate bandwidths on multi-homed mobile hosts. In *MobiCom*, Atlanta, GA, USA, 2002.

[46] H. Hsieh and R. Sivakumar. On using peer-to-peer communication in cellular wireless data networks. *IEEE Transactions on Mobile Computing*, 3(1):57–72, January 2004.

[47] M. Hu and J. Zhang. Opportunistic multi-access: multiuser diversity, relay-aided opportunistic scheduling, and traffic-aided smooth admission control. *Mob. Netw. Appl.*, 9(4):435–444, 2004.

[48] P. Hui, A. Chaintreau, J. Scott, R. Gass, J. Crowcroft, and C. Diot. Pocket switched networks and human mobility in conference environments. In *WDTN*, Philadelphia, PA, USA, August 2005.

[49] S. Jain, K. Fall, and R. Patra. Routing in a Delay Tolerant Network. In *SIGCOMM*, Portland, OR, USA, August 2004.

[50] M. Jakobsson, J. Hubaux, and L. Buttyan. A micro-payment scheme encouraging collaboration in multi-hop cellular networks. In *Financial Crypto*, Gosier, Guadeloupe, January 2003.

[51] A. Karlin. Beyond VCG: Frugality of truthful mechanisms. In *FOCS*, Pittsburgh, PA, USA, October 2005.

[52] I. Katzela and M. Naghshineh. Channel assignment schemes for cellular mobile telecommunication systems: A comprehensive survey. *IEEE Personal Communications*, 3(3):10–31, June 1996.

[53] B. Kauffmann, F. Baccelli, A. Chaintreau, V. Mhatre, K. Papagiannaki, and C. Diot. Measurement-based self organization of interfering 802.11 wireless access networks. In *INFOCOM*, Anchorage, AK, USA, May 2007.

[54] Kazaa. Participation level. http://www.kazaa.com/.

[55] F. Kelly. Charging and rate control for elastic traffic. *European Transactions on Telecommunications*, 8(3):33–37, 1997.

[56] F. Kelly, A. Maulloo, and D. Tan. Rate control in communication networks: shadow prices, proportional fairness and stability. *Journal of the Operational Research Society*, 49(3):237–252, March 1998.

[57] M. Kodialam and T. Nandagopal. Characterizing the capacity region in multi-radio multi-channel wireless mesh networks. In *MobiCom*, Cologne, Germany, August 2005.

[58] J. Korhonen. *Introduction to 3G Mobile Communications (Second Edition)*. Artech House, 2003.

[59] S. Lee, S. Banerjee, and B. Bhattacharjee. The case for a multi-hop wireless local area network. In *INFOCOM*, HongKong, China, March 2004.

[60] U. Lee, J. Park, E. Amir, and M. Gerla. FleaNet: A virtual market place on vehicular networks. In *International Conference on Mobile and Ubiquitous Systems: Networking and Services*, San Jose, CA, USA, July 2006.

[61] J. Lenstra, D. Shmoys, and E. Tardos. Approximation algorithms for scheduling unrelated parallel machines. *Mathematical Programming*, 46(1):259–271, January 1990.

[62] D. Levin, K. LaCurts, N. Spring, and B. Bhattacharjee. BitTorrent is an auction: Analyzing and improving bitTorrents incentives. In *SIGCOMM*, Seattle, WA, USA, August 2008.

[63] L. Li, M. Pal, and Y. Yang. Proportional fairness in multi-rate wireless LANs. In *INFOCOM*, Phoenix, AZ, USA, April 2008.

[64] Q. Li and D. Rus. Communication in disconnected ad-hoc networks using message relay. *Journal of Parallel and Distributed Computing*, 63(1):75–86, January 2003.

[65] X. Li, B. Seet, and P. Chong. Multihop cellular networks: Technology and economics. *Computer Networks*, 52(9):1825–1837, June 2008.

[66] Y. Lin and Y. Hsu. Multihop cellular: A new architecture for wireless communications. In *INFOCOM*, Tel-Aviv, Israel, March 2000.

[67] S. Lincke. Vertical handover policies for common radio resource management. *International Journal of Communication Systems*, 18(6):527–543, August 2005.

[68] A. Lindgren, A. Doria, and O. Schelen. Probabilistic routing in intermittently connected networks. In *SAPIR Workshop*, Fortaleza, Brazil, August 2004.

[69] H. Luo, R. Ramjee, P. Sinha, and S. Lu. UCAN: a unified cellular and ad-hoc network architecture. In *MobiCom*, San Diego, CA, USA, 2003.

[70] R. Mahajan, J. Zahorjan, and B. Zill. Understanding WiFi-based connectivity from moving vehicles. In *IMC*, San Diego, CA, USA, October 2007.

[71] L. Massoulié and J. Roberts. Bandwidth sharing: Objectives and algorithms. *Transactions on Networking*, 10(3):320–328, August 2002.

[72] N. Mathewson, P. Syverson, and R. Dingledine. Tor: the second-generation onion router. In *USENIX Security Symp.*, San Diego, CA, USA, August 2004.

[73] R. Mazumdar, H. Yaiche, and C. Rosenberg. A game theoretic framework for bandwidth allocation and pricing in broadband networks. *IEEE/ACM Trans. on Networking*, 8(5):667–677, 2000.

[74] I. Milchtaich. Congestion games with player-specific payoff functions. *Games and Economic Behavior*, 13(1):111–124, 1996.

[75] M. Minoux. *Mathematical Programming: Theory and Algorithms*. Wiley, Chichester, 1986.

[76] MIT. Reality mining project. http://reality.media.mit.edu.

[77] J. Mitola. *Software radio architecture : object-oriented approaches to wireless systems engineering*. John Wiley and Sons, 2000.

[78] M. Motani, V. Srinivasan, and P.S. Nuggehalli. PeopleNet: engineering a wireless virtual social network. In *MobiCom*, Cologne, Germany, August 2005.

[79] J. Nash. The bargaining problem. *Econometrica*, 18(2):155–162, April 1950.

[80] J. Newsome, E. Shi, D. Song, and A. Perrig. The sybil attack in sensor networks: Analysis & defenses. In *IPSN*, Berkeley, CA, USA, April 2004.

[81] N. Nisan and A. Ronen. Algorithmic mechanism design. *Games and Economic Behavior*, 35(1-2):166–196, 2001.

[82] N. Nisan, T. Roughgarden, E. Tardos, and V. Vazirani (editors). *Algorithmic game theory*. Cambridge University Press, 2007.

[83] M. Osborne. *An introduction to Game Theory*. Oxford University Press, New York, NY, 2004.

[84] J. Ott and D. Kutscher. A disconnection-tolerant transport for drive-thru Internet environments. In *INFOCOM*, Miami, FL, USA, March 2005.

[85] V. Padmanabhan and J. Mogul. Using predictive prefetching to improve World Wide Web latency. *Computer Comm. Rev.*, 26(3):22–36, July 1996.

[86] C. Piro, C. Shields, and B. Levine. Detecting the sybil attack in mobile ad hoc networks. In *SecureComm*, Baltimore, MD, USA, August 2006.

[87] J. Proakis. *Digital Communications (4th Edition)*. New York: McGraw-Hill, 2001.

[88] Qualcomm. Snapdragon chipset. http://www.qctconnect.com/products/snapdragon.html.

[89] T. Rappaport. *Wireless Communications: Principles and Practice (2nd Edition)*. Prentice Hall PTR, 2001.

[90] RNCOS. World GPS market forecast (2006-2008).

[91] P. Rodriguez, R. Chakravorty, J. Chesterfield, I. Pratt, and S. Banjeree. MAR: A commuter router infrastructure for the mobile Internet. In *MobiSys*, Boston, MA, USA, June 2004.

[92] R. Rosenthal. A class of games possessing pure-strategy Nash equilibria. *International Journal of Game Theory*, 2:65–67, 1973.

[93] S. Roy, A. Das, R. Vijayakumar, H. Alazemi, H. Ma, and E. Alotaibi. Capacity scaling with multiple radios and multiple channels in wireless mesh networks. In *WiMesh*, Santa Clara, CA, USA, September 2005.

[94] Y. Sakurai, M. Yokoo, and K. Kamei. An efficient approximate algorithm for winner determination in combinatorial auctions. In *EC*, Minneapolis, MN, USA, October 2000.

[95] A. Sang, X. Wang, M. Madihian, and R. Gitlin. Coordinated load balancing, handoff/cell-site selection, and scheduling in multi-cell packet data systems. In *MobiCom*, Philadelphia, PA, USA, September 2004.

[96] S. Sesia, I. Toufik, and M. Baker (editors). *LTE - The UMTS Long Term Evolution - From Theory to Practice*. John Wiley and Sons, 2009.

[97] U. Shevade, H. Song, L. Qiu, and Y. Zhang. Incentive-aware routing in DTNs. In *ICNP*, Orlando, FL, USA, October 2008.

[98] M. Sirivianos, J. Park. X. Yang, and S. Jarecki. Dandelion: Cooperative content distribution with robust incentives. In *USENIX*, Santa Clara, CA, USA, June 2007.

[99] T. Spyropoulos, K. Psounis, and C. S. Raghavendra. Spray and wait: An efficient routing scheme for intermittently connected mobile networks. In *WDTN*, Philadelphia, PA, USA, August 2005.

[100] V. Srinivasan, M. Motani, and W. Ooi. Analysis and implications of student contact patterns derived from campus schedules. In *MobiCom*, Los Angeles, CA, USA, September 2006.

[101] G. Tan and J. Guttag. Time-based fairness improves performance in multi-rate WLANs. In *USENIX*, Boston, MA, USA, June 2004.

[102] K. Tan, J. Zhang, J. Fang, H. Liu, Y. Ye, S. Wang, Y. Zhang, H. Wu, W. Wang, and G. Voelker. Sora: High performance software radio using general purpose multi-core processors. In *NSDI*, Boston, MA, USA, April 2009.

[103] A. Tolli, P. Hakalin, and H. Holma. Performance evaluation of common radio resource management (CRRM). In *ICC*, New York City, NY, USA, April 2002.

[104] A. Vahdat and D. Becker. Epidemic routing for partially-connected ad hoc networks. Technical report, CS-200006, Duke University, April 2000.

[105] V. Vishnumurthy, S. Chandrakumar, and E. Sirer. KARMA: A secure economic framework for p2p resource sharing. In *Workshop on the Economics of Peer-to-Peer Systems*, Berkeley, CA, USA, June 2003.

[106] J. von Neumann and O. Morgenstern. *Theory of games and economic behavior*. Princeton University Press, 1944.

[107] A. Wander, N. Gura, H. Eberle, V. Gupta, and S. Chang-Shantz. Energy analysis of public-key cryptography for wireless sensor networks. In *PERCOM*, Kauai Island, Hawaii, USA, March 2005.

[108] P. Whittle. *Optimization Under Constraints*. Wiley, Chichester, 1971.

[109] H. Wu, C. Qiao, S. De, and O. Tonguz. Integrated cellular and ad hoc relaying systems: iCAR. *IEEE Selected Areas in Communications*, 19(10):2105–2115, October 2001.

[110] Q. Wu and C. Williamson. Dynamic channel rate assignment for multi-radio WLANs. In *Wireless Networks and Emerging Technologies*, Banff, AB, Canada, July 2005.

[111] X. Wu, S. Chan, and B. Mukherjee. MADF: A novel approach to add an ad-hoc overlay on a fixed cellular infrastructure. In *WCNC*, Chicago, IL, USA, September 2000.

[112] A. Zadeh, B. Jabbari, R. Pickholtz, and B. Vojcic. Self-organizing packet radio ad hoc networks with overlay. *IEEE Communications Magazine*, 40(6):140–157, June 2002.

[113] X. Zhang, J. Kurose, B. Levine, D. Towsley, and H. Zhang. Study of a bus-based Disruption-Tolerant Network: mobility modeling and impact on routing. In *MobiCom*, Montreal, QC, Canada, September 2007.

[114] Y. Zhang, J. Zhao, and G. Cao. On scheduling vehicle-roadside data access. In *VANET*, Montreal, QC, Canada, September 2007.

[115] J. Zhao and G. Cao. VADD: Vehicle-assisted data delivery in vehicular ad hoc networks. In *INFOCOM*, Barcelona, Spain, April 2006.

[116] W. Zhao, M. Ammar, and E. Zegura. A message ferrying approach for data delivery in sparse mobile ad hoc networks. In *MobiHoc*, Roppongi, Japan, May 2004.

[117] S. Zhong, L. Li, Y. Liu, and Y. Yang. On designing incentive-compatible routing and forwarding protocols in wireless ad-hoc networks — an integrated approach using game theoretical and cryptographic techniques. In *MobiCom*, Cologne, Germany, August 2005.

[118] S. Zhong, Y. Yang, and J. Chen. Sprite: A simple, cheat-proof, credit-based system for mobile ad-hoc networks. In *INFOCOM*, San Francisco, CA, USA, March 2003.

[119] H. Zhu, X. Lin, R. Lu, and X. Shen. A secure incentive scheme for Delay Tolerant Networks. In *Chinacom*, HangZhou, China, August 2008.