

ABSTRACT

WON, HYUNG SUK. Multicast in Wireless Networks. (Under the direction of Dr. Injong Rhee).

Multicast is an efficient communication means of transmitting the same content to multiple receivers while minimizing network resource usage. However, wireless networks are very diverse and they have their own characteristics. Multicast has not been studied extensively for the networks different from traditional wired networks.

Our thesis is to prove that multicast is an effective means in wireless networks. By efficiently using network resources through schedulers, multicast can be an effective communication means. Therefore, we study multicast scheduler adaptation to the wireless networks where existing multicast schemes are not applied directly. Among the wireless networks, we consider multicast schedulers in cellular data networks and disruption tolerant networks.

First, we propose two proportionally fair multicast scheduling algorithms at the air interface in the downlink direction to adapt dynamic channel states in cellular data networks: Multicast Proportional Fairness (MPF) and Inter-Group Proportional Fairness (IPF) algorithms. Our algorithms take into account reported data rate requests from users and the average throughput of each user inside a cell and use this information to select an appropriate data rate for each group. We prove that MPF and IPF algorithms are proportionally fair among all users and among groups inside

a cell respectively. Through simulations, we demonstrate that these algorithms achieve good balance between throughput and fairness among users and groups.

Second, we study joint optimization of link scheduling, routing and replication for disruption-tolerant networks (DTNs). We define a new notion of optimality for DTNs called “snapshot optimality” which uses only contemporarily available knowledge. We then present a new efficient approximation algorithm called Distributed Max-Contribution (DMC) based only on locally and contemporarily available information. Through a simulation study based on real GPS traces, we show that DMC demonstrates near-optimal performance.

By proposing an efficient multicast schedulers to cellular data networks and DTNs, we prove that multicast is an effective means of communication for wireless networks.

Multicast in Wireless Networks

by
Hyung Suk Won

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Computer Science

Raleigh, North Carolina

2009

APPROVED BY:

Dr. Rudra Dutta

Dr. Mihail Sichitiu

Dr. Injong Rhee
Chair of Advisory Committee

Dr. George N. Rouskas

DEDICATION

Dedicated to my parents, for their love and unwavering faith and support
with my love and respect.

BIOGRAPHY

Hyungsuk Won was born and grew up in Pohang, Korea. He received his Bachelor degree in Computer Engineering from Kyungbuk National University, Korea in 1997 and Master degree in Computer Science from Postech (Pohang University of Science and Technology) in 1999. After completed his M.S. degree, he worked for Postech, Digital Gen, and ETRI about 4.5 years. Then he has been continuing to pursue his Ph.D. degree in North Carolina State University starting from August 2003. His research areas include MAC scheduling and Routing protocols in wireless network, Information security, and Information Retrieval.

ACKNOWLEDGMENTS

I would like first to thank my advisor, Dr. Injong Rhee, for his guidance and support during my entire graduate study. He inspired me to find research ideas and encouraged me to continue my study. This thesis would not have been accomplished without his devotion.

I also would like to thank my committee members: Dr. George N. Rouskas, Dr. Rudra Dutta, and Dr. Mihail Sichitiu. Dr. David Thuente processed many documents during my graduate study as a director of graduate program. I appreciate his help and time.

I had a chance to work with Dr. Do Young Eun in ECE Dept. of NCSU and Dr. Katherine Guo in Lucent Bell Lab. The experience of research collaboration was very useful in my study. I am thankful to them.

I would like to express my thanks to my lab members : Jeongki Min, Sangtae Ha, Seongik Hong, Sankararaman Janakiraman, Sungro Yoon. In particular, sincere thanks to Jeongki who started his study at the same year as mine and always helped me when I was in trouble. Dr. Seongjoon Kim had worked in the same lab as a postdoctoral. Through many discussions, he helped me to clarify many ideas and to understand experimental results. Dr. Ajit Warrier and Dr. Sangjoon Park are alumni of my lab. We spent much time before they graduated. When we studied

together, they helped me to understand many theories in papers. I would like to show my gratitude to them.

Many thanks go to many peoples who helped me in many ways. Dr. Minyeol Lim and his wife Younghee Park, who cooked foods for me many times. Eunyoung Ha and her husband held parties several times to cheer me up. Dr. Sangmin Kim and Young June Pyun always helped me in many ways. My friends sent email frequently to me, which cheered me up when I was depressed: many thanks to Dongmin Kim, Dongki Kwak, Yeongkyu Lim, Gyuho Jang and Gyubong Park. Also, I would like to show my gratitude to Dr. Jeongwon Cha, Dr. Byungchang Kim, and Dr. Seungwoo Lee. They always helped and advised me when I needed it. Above all, I would like to express my deepest gratitude to Gwangmyung Kim. He and his family have helped me since I arrived in Raleigh, NC. Without his encouragement, I could not finish my study.

I would like to thank my parents and my sister and brother for their constant love and support. Especially, my brother Hyungdon and his wife have taken care of many family businesses in place of me during the long hours of my study.

Finally, I would like to thank and honor all those upon whose shoulders I stand.

TABLE OF CONTENTS

LIST OF TABLES	viii
LIST OF FIGURES	ix
1 Introduction.....	1
1.1 Motivation	1
1.2 Thesis Statement	6
1.3 Contributions	7
1.4 Thesis Outline	8
2 Related Work	10
2.1 Multicast in Conventional Networks	10
2.2 Multicast in Cellular Data Networks	13
2.3 Multicast in Delay/Disruption Tolerant Networks	16
3 Multicast in Cellular Data Networks.....	23
3.1 Background	30
3.1.1 Proportional Fairness	30
3.1.2 Unicast Scheduler	31
3.2 Model Description	32
3.2.1 Notation	33
3.3 Inter-group PF Scheduler	34
3.3.1 IPF Scheduling Algorithm	36
3.3.2 Inter-group PF	37
3.3.3 General Inter-group PF Scheduler	42
3.4 Multicast PF Scheduler	43
3.4.1 MPF Scheduling Algorithm	44
3.4.2 Multicast PF	49
3.4.3 General MPF Scheduler	51
3.5 Simulation	53
3.5.1 Simulation Setup	53
3.5.2 Objective Functions	54
3.5.3 Distribution of Throughput	59
3.6 Summary	62

4	Multicast in DTNs	65
4.1	Optimal Resource Allocation in DTNs	70
4.1.1	Notations	70
4.1.2	System Model	71
4.1.3	Snapshot Optimality	74
4.2	Snapshot Optimality:	
	Hardness and Max-Contribution	77
4.2.1	Value and Contribution	77
4.2.2	S-OPT: A Snapshot Optimal Algorithm	78
4.2.3	Link/Copy Scheduling Decomposition: Max-Contribution	81
4.3	Distributed Max-Contribution	83
4.3.1	Distributed Max-Contribution (DMC)- Unicast	84
4.3.2	DMC- Multicast	86
4.4	Performance Evaluation	91
4.4.1	Node Delivery Probability from Shanghai Trace	91
4.4.2	Setup, Metric and Tested Algorithms	95
4.4.3	Simulation Results	97
4.5	Summary	101
5	Conclusion	103
	Bibliography	107
	Glossary	123
	Index	126

LIST OF TABLES

Table 3.1 A summary of multicast scheduling algorithms	54
Table 4.1 Tested Algorithms (★ corresponds to the items that we added for fair comparison)	96

LIST OF FIGURES

Figure 1.1 Types of Communications: (a) unicast: 1 to 1 communication; (b) broadcast: 1 to all communication; (c) multicast: 1 to many communication	2
Figure 1.2 Comparison of unicast and multicast; assume that S sends the same data to B , C , and D	4
Figure 3.1 Architecture of 3G Cellular Data Networks	25
Figure 3.2 Time slots for multicast schedulers.....	27
Figure 3.3 Multicast Example	27
Figure 3.4 The value of $\sum \log T_i$ for various multicast schedulers.....	56
Figure 3.5 The value of $\sum \log T_k^g$ for various multicast schedulers	57
Figure 3.6 The value of $\sum T_i$ for various multicast schedulers	58
Figure 3.7 The distribution of user throughput taken at each three-second interval for various schedulers	60
Figure 3.8 The distribution of group throughput at the end of simulation for various schedulers	63
Figure 4.1 An Example of Unicast Routings in DTNs. Assume that each node has delivery probability for destinations D_1 and D_2 . Also assume that marginal delivery probability difference per packet is used to evaluate node's quality.	67
Figure 4.2 An Example of Multicast Routings in DTNs. Assume that each node has delivery probability for destinations D_1 and D_2 . Also assume that sum of marginal delivery probability per packet is used to evaluate node's group quality.	69

- Figure 4.3 An Example of DMC-Multicast in DTNs. Assume that each node has delivery probability for destinations D_1 and D_2 . Also assume that sum of marginal delivery probability per packet is used to evaluate node's group quality. 87
- Figure 4.4 Snapshot distribution of taxis in Shanghai. Each dot indicates the location of a taxi during one hour at 11/28/2006. Circles indicate the candidate locations of sources and destinations. 91
- Figure 4.5 Inter-visit time (IVT) distribution and inter-contact time (ICT) distribution of a taxi to locations. They are fitted by maximum likelihood estimation (MLE) to exponential distributions. The maximum and minimum intensity of the best fitting exponential distributions are λ_{min} and λ_{max} , respectively. 93
- Figure 4.6 We plot the individual intensity values (λ^{IVT} and λ^{ICT}) of IVT and ICT exponential distributions from 100 taxis. IVT is plotted for 100 destination locations. A high intensity value of a particular location by a particular taxi in implies that a taxi has a high rate of visit to a particular location. Likewise, a high intensity value of a taxi with respect to another taxi meets they tend to meet very often. From the plots, we find that different taxis show different biases in the locations they visit and in the set of taxis they meet daily. 93
- Figure 4.7 The Delivery ratio and Efficiency of algorithms listed in Table 4.1 against the offered load to 32 S-D pairs. Transmission range is 300m. Each value shows 95% confidence interval. We do not show cost as it is implied in the efficiency. 97
- Figure 4.8 The delivery ratio of RAPID and DMC under different radio ranges. The input load is set to 700 packets. 97
- Figure 4.9 The Delivery ratio and Efficiency of algorithms listed in Table 4.1 against the offered load to 32 S-D pairs. Transmission range is 500m. 98
- Figure 4.10 The performance comparison of DMC-multicast and DMC-unicast. The number of destinations are varied from 1 to 32. Also the input load is set to 700 packets for each destination. 101
- Figure 4.11 The performance comparison of DMC-multicast and DMC-unicast. The number of destinations are varied from 1 to 32. Also the input load is

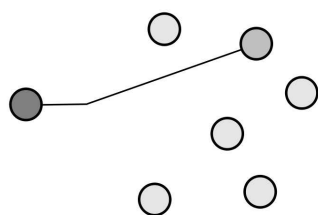
set to 1000 packets for each destination.	101
--	-----

Chapter 1

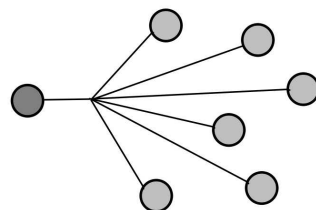
Introduction

1.1 Motivation

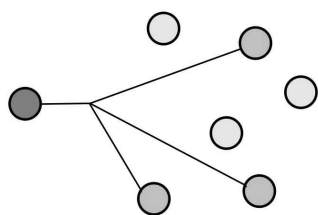
All communications in communication networks can be categorized into three types: unicast, broadcast, and multicast. Figures 1.1(a), 1.1(b), and 1.1(c) show each type of communication. Unicast is one-to-one communication wherein a node sends a packet to only one destination. Broadcast is one-to-all communication when a node sends a packet to all nodes in a network. Multicast is one-to-many communication when a node sends the packet to a group. A group of multicast consists of one or more nodes in a network. If a group consists of only one node, multicast is the same as unicast. Also, if a group consists of all nodes in a network, multicast can be regarded as broadcast. Consequently, multicast can range from unicast to broadcast.



(a) Unicast



(b) Broadcast



(c) Multicast

Figure 1.1: Types of Communications: (a) unicast: 1 to 1 communication; (b) broadcast: 1 to all communication; (c) multicast: 1 to many communication

Multicast is especially useful for applications where the same data is delivered to many users: for example, multi-player games, weather or news services, and multimedia streaming. If the same data is delivered from one source to multiple destinations by using unicast, duplicate links may happen on paths between the source and destinations. Multicast in wired networks transmits only one copy of data over a link. Consequently, as the number of duplicate links on paths increases, the number of transmissions decreases by using multicast.

In Figure 1.2, assume that S sends the same data to B , C , and D . If unicast is used to delivery the data, the paths $S - A - B$, $S - A - C$, and $S - A - D$ are used respectively. It requires 6 transmissions in a network. The same data is delivered three times over the link $S - A$. If multicast is used to do the same thing, the data is delivered once over the link $S - A$. Then the data is delivered over $A - B$, $A - C$, and $A - D$ respectively. It requires 4 transmissions. Consequently, by using multicast we save 2 transmissions.

Fewer transmissions means less network resource usage. Since through multicasting network resources are being used effectively, we can expect less delay and higher system-wide throughput.

In wireless networks, the same types of communications exist as in wired networks. However, all wireless communications have a nature of broadcasting. In wireless networks, unicast communications for one-to-one are processed as follows. Nodes that

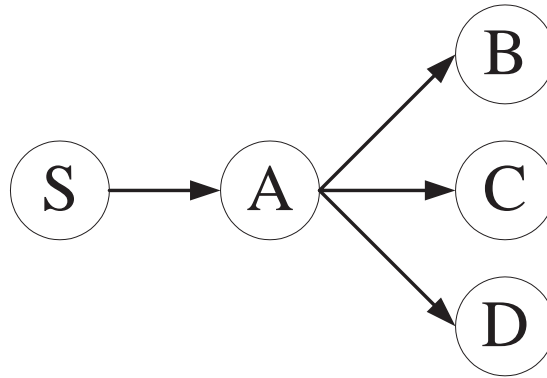


Figure 1.2: Comparison of unicast and multicast; assume that S sends the same data to B , C , and D .

are within a sender's transmission range hear the data the sender transmits. Only the node that is the destination of the data receives the data, while the other nodes simply drop it. Instead of simply dropping the data, we can expect more system-wide throughput through multicast that exploits the broadcast nature in wireless communications.

Therefore, communications over wireless networks efficiently support group applications that have multiple destinations by exploiting the broadcast nature of wireless communications. The applications of multicast in wireless networks include intelligent services such as location-based weather services, local news and traffic announcements, and interactive games as well as the multicast applications in wired networks. Therefore, multicast in wireless networks is an important and challenging goal.

The first target of multicast in wireless networks was mobile ad-hoc networks (MANETs) that consist of mobile nodes connected by wireless links. There has been

some research done on multicast in MANETs (e.g., [42, 92, 95, 97]). For multicast in MANETs, the source node constructs paths between the source and multiple destinations called “multicast tree” by using routing information.

However, mobile wireless networks are very diverse and they have their own characteristics. Multicast has not been studied extensively for the networks different from the mobile ad-hoc networks. Although cellular data networks are one of the most widely used wireless networks, multicast is still not widely used. A cellular data network consists of fixed base stations and mobile users. The “cell” is defined by the area covered by a base station. A cell consists of one base station and mobile users connecting to the base station. Because all mobile users in a cell connect to the base station directly, routing is not needed. However, the channel conditions of mobile users vary over time. Such channel conditions should be considered in the base station in order to get higher system throughput. The study of multicast in cellular data networks can be applied to the similar type of wireless networks such as wireless local area networks (WLAN).

Delay/Disruption tolerant networks (DTNs) are one of “challenged networks” [75] that have totally different characteristics than wired and mobile ad-hoc networks. DTNs are proposed to communicate in mobile ad-hoc wireless networks where disconnection is frequent. Under such conditions, network partitions happen frequently, and thus conventional wireless routing protocols cannot find routes from a source to a

destination over the partitioned networks. Consequently, existing multicast schemes in MANETs, which are based on routing information of routing protocols, do not work in DTNs. The study of multicast in DTNs can support multicast communications in other types of wireless networks, namely, wireless mesh networks, vehicular ad-hoc networks, and peer-to-peer (P2P) networks including MANETs.

Since Tassiulas’s study in [86], there has been research done on scheduling in wireless networks (e.g., [17, 19, 43, 51, 62, 69, 86]). The objective of optimal scheduling algorithms is the efficient usage of network resources. By efficiently using network resources, scheduling algorithms play an important role in enhancing system-wide throughput. If multicast schemes in wireless networks are supported by efficient multicast scheduling algorithms, multicast can be a more effective communication means in wireless networks.

1.2 Thesis Statement

The primary purpose of this thesis is to prove that multicast is an effective communication means in wireless networks. As we have discussed so far, existing multicast studies in wired networks and MANETs do not apply well to other types of wireless networks. As a result, new approaches are needed to study multicast in wireless networks. In order to make multicast feasible in wireless networks, we study “multicast scheduler” adaptation to the wireless networks where existing multicast schemes are

not applied directly.

To show the feasibility of multicast schedulers in wireless networks, we choose cellular data networks and delay/disruption tolerant networks (DTNs) as target networks. Cellular data networks are the most widely and practically used networks among wireless networks. DTNs [2] are one of the new recently introduced “challenged networks” that have different characteristics than conventional communication environments of existing wireless networks. In both networks, unicast has been studied intensively, but multicast still remains a challenging problem. In addition, we believe that this study can be also applied to many similar types of wireless networks such as WLAN and wireless mesh networks.

1.3 Contributions

Our contributions of this thesis are summarized as follows.

For cellular data networks, we

- propose Proportionally Fair (PF) Multicast schedulers:
Inter-Group Proportional Fairness (IPF) and Multicast Proportional Fairness (MPF).
- prove Proportional Fairness of Multicast Schedulers;

Multicast Proportional Fairness and Inter-Group Proportional Fairness algorithm are proportionally fair among all users and among groups inside a cell respectively.

For delay/disruption tolerant networks (DTNs), we

- first consider joint optimization of link scheduling, routing and replication.
- present a new efficient approximation algorithm, called Distributed Max-Contribution (DMC);

DMC uses only locally and contemporary available information.

- present DMC algorithm for multicast, called DMC-Multicast that includes a measurement scheme for multiple destinations.

1.4 Thesis Outline

The rest of the thesis is organized as follows. In Chapters 2, we discuss related works. In Chapter 3, we present multicast in cellular data networks, introduce our system model, present our multicast algorithms and provide proofs of the proportional fairness of our algorithms. We also describe our simulation setup and results. In Chapter 4, we demonstrate ways in which multicast is enabled in DTNs. After introducing a system model and notations, we go on to discuss the impracticality of global optimality as well as the practicality of snapshot optimality. To achieve

a practical, approximation algorithm from snapshot optimality, called DMC-unicast that uses only local and contemporary information is proposed; then DMC unicast scheme is extended to DMC multicast. After presenting the simulation results, we finally conclude the dissertation in Chapter 5.

Chapter 2

Related Work

In this Chapter, we discuss recent multicast-related works in cellular data networks and delay/disruption tolerant networks, starting from conventional multicast studies in ad-hoc wireless networks.

2.1 Multicast in Conventional Networks

Since Deering's proposal of IP multicast in 1990 [32], many multicast protocols have been suggested [4, 20, 33, 76, 91]. Most multicast protocols of conventional wired networks construct multicast trees for groups. Through such trees, the data from senders is delivered to receivers which are the members of groups. The construction of multicast trees is based on links/paths information which is supplied by traditional routing protocols or by topology discovery mechanisms inside multicast protocols.

In the Distance Vector Multicast Routing Protocol (DVMRP) [91], packets from a source are delivered to all nodes in a network. The nodes that receive unwanted packets send "prune" messages to the source. By using the information, the source maintains multicast trees. Because of the way DVMRP operates, it is called "broadcast and prune" scheme. However, the nature of broadcast limits the scalability of DVMRP. The Multicast Open Shortest Path First (MOSPF) protocol [4] is an extension to the Open Shortest Path First (OSPF) protocol to support multicast routing. MOSPF also uses similar techniques in DVMRP to manage multicast trees.

To solve the poor scaling problems of DVMRP and MOSPF, shared tree protocols are proposed [20] [33]. The Core-Based Trees (CBT) [20] constructs a tree starting from the core routers towards the routers directly adjacent to the multicast group members. The Protocol Independent Multicast (PIM) [33] is most widely used. PIM does not include its own topology discovery mechanism, but instead uses routing information from other traditional routing protocols. There are some variants of PIM: Sparse Mode (SM) [9], Dense Mode (DM) [8], Source Specific Mode (SSM) and Bidirectional Mode (Bidir, or Sparse-Dense Mode, SDM). Of these, PIM-SM is the most widely deployed, especially in IPTV.

The Internet Group Management Protocol (IGMP) is used in order to manage the membership of multicast groups [5]: leave and join.

In ad-hoc wireless networks, nodes do not have any information about the topology

of their networks. Consequently, nodes have to discover their neighbors by listening to the announcements from neighbors after announcing their own presence. In this way, each node manages its own routing table. Routing protocols of ad-hoc networks are categorized into two types: pro-active and reactive.

Pro-active protocols maintain fresh lists of destinations and their routes by periodically distributing routing tables throughout the network. To maintain such algorithms, significant data exchanges are required. Also, when link failures happen, the path reconstruction is slow. Examples of pro-active algorithms are DSDV (Highly Dynamic Destination-Sequenced Distance Vector routing protocol) [73] and OLSR (Optimized Link State Routing Protocol) [7].

Reactive protocols have to find a path "on demand" by flooding the network with "Route Request" packets before transmissions. Therefore, such algorithms take a long time in finding paths to the destination. Examples of reactive (on-demand) routing algorithms are AODV (Ad hoc On-Demand Distance Vector) [6] and DSR (Dynamic Source Routing) [50].

There are also routing protocols for multicast in mobile ad-hoc networks. Examples are Maximum Residual Multicast Protocol (MRMP) [45], Multicast AODV (MAODV), ODMRP [52], and DDM [49]. MAODV routing protocol [77] is intended for multicast based on the AODV protocol. While MAODV actively repairs link failures, the maintenance cost of multicast trees becomes very high. ODMRP [52]

considers alternative paths to adapt to frequent topology changes. In DDM [49], each source maintains its own multicast group. [49] assumes that the size of a multicast group is small. MRMP [45] considers residual energy in constructing multicast trees on demand. Such algorithms do not scale well with network sizes.

Multicast trees can be constructed only in traditional wireless networks where the movement of mobile nodes is not so fast that the construction of paths is possible from a source to destinations, and the constructed links are available for a comparatively long time. However, modern wireless networks are very diverse and they have their own characteristics, multicast has not been studied well for the networks different from traditional mobile ad-hoc wireless networks.

2.2 Multicast in Cellular Data Networks

We propose a suite of scheduling algorithms for multicast. While a large number of papers have proposed and evaluated unicast scheduling algorithms where multiple users share a time-varying wireless channel using Time Division Multiplexing (TDM) [13–16, 22–24, 48, 63, 64, 67, 70, 71, 88, 90], to the best of our knowledge, no paper has addressed multicast scheduler design for cellular data networks using TDM such as CDMA2000 1xEV-DO.

The proportional fair scheduling algorithm for unicast is proposed in [48, 88, 90] for CDMA2000 1xEV-DO systems to maximize the log utility function. The weighted

proportional fair schemes proposed in [14] demonstrate how one can choose a scheduler with an efficiency-fairness tradeoff between the two extreme cases, namely, a channel-unaware scheduler, and a channel-aware scheduler that serves the best mobile handset at any given time. The schemes proposed in [63] maximize linear utility functions, and they are based on stochastic approximation.

Proportional fairness is generalized into (p, α) -proportional fairness in [67] to unify the max-min fairness, proportional fairness and the worst case fairness, which is total-throughput maximization. A unified scheduler is proposed in [71] that achieves (p, α) -proportional fairness in terms of the asymptotic behavior of the long-term average throughput.

Properties of the rate region achievable by a general class of opportunistic scheduling algorithms are discussed in [24]. In addition, [24] also considers different fairness criteria and discusses optimal scheduling algorithms. The optimality of a general class of “gradient-like” opportunistic scheduling algorithms is proved in [16, 58, 84].

A scheduling algorithm that maintains mobile handset service rates in proportions to one another is proposed in [24]. The scheduling algorithm studied in [23] uses the assumption that each mobile handset has a finite amount of data to receive and it leaves the system once this data is received.

As the need to serve real-time traffic increases, quality of service guarantees have become an important aspect of the design of scheduling algorithms. Typically a

scheduler would either offer delay guarantees or throughput restrictions to individual flows. The EXP algorithm [79] controls the delay distribution of each flow for a mixture of real-time and non-real-time data in CDMA2000 1xEV-DO. The GMR algorithm [18] is a gradient algorithm with minimum and maximum rate constraints using a token counter mechanism.

Broadcast scheduling algorithms are used in database and content distribution systems, taking into consideration application level information such as the size and importance of data items to be broadcast. In the context of 802.16e like mobile networks, where mobile hosts can go into sleep mode to save energy, [30] proposes a number of scheduling algorithms letting each host assign its own merit to each broadcast data item. The mechanisms divide each broadcast super-frame into a number of logical channels, and let mobile hosts decide which channel to listen to at the beginning of each super-frame in order to maximize the normalized throughput of the system for each super-frame. In the calculation for normalized throughput, each data item is multiplied by the merit for each host. Our work is quite different in the sense that we study the scheduler design for base stations in TDM systems without any application information.

Scheduling is part of resource allocation, and resource allocation in the form of power and rate control for multicast over mobile wireless networks has recently received more research attention [34–36,98]. [34] employs superposition coding and se-

lective retransmission to solve the multicast throughput optimization problem, where the transmitted signal consists of several parallel sub-stream signals. The scheme proposed in [35] aims at achieving maximum system goodput by dynamically allocating power and rate jointly based on all multicast receivers' channel state information. Taking quality of service requirement into consideration, [98] proposes an efficient hybrid automatic repeat request-forward error correction erasure-correcting scheme for mobile multicast, whereas [36] proposes a cross-layer rate control scheme that maximizes the physical-layer throughput by using rate adaptation based on channel state information subject to upper layer loss rate QoS requirement. Our work is complementary since we focus on scheduling for multiple groups in the system. Instead of maximizing system throughput aimed by the above power and rate control schemes, our proposed algorithms are proportional fair.

2.3 Multicast in Delay/Disruption Tolerant Networks

The Delay Tolerant Network Research Group (DTNRG) [2] has proposed an architecture for DTNs that is wireless networks where delay is high and contacts are infrequent [39]. Most existing routing protocols for wireless networks works under the assumption that end-to-end paths exist. The traditional routing protocols for

wireless networks do not suit DTNs well due to sparseness of networks and frequent disconnections that are characteristics of DTNs. Therefore, the new routing concepts of "message bundle" and "store, carry, and forward" for DTNs are first introduced in [27]. After that, various routing schemes have been studied and proposed.

One of the simplest ways in routing schemes is that wherein the source delivers packets directly to the destination (Direct Transmission) [78]. Direct Transmission scheme requires only one transmission per packet. It takes much time for the source to meet the destination directly, thus delivery time is very long. Another simplest way is flooding that a node copies its all packets to encountered nodes (Epidemic Routing) [89] [74]. Therefore Epidemic Routing requires many transmissions per packet. As a result, there are many duplicate packets in a network. These consume much of network resources such as buffer, bandwidth, and energy. If there is no contention between nodes, the delivery of Epidemic Routing is very fast. However, in real situations, there is limitation to the usage of network resources. It significantly degrades the performance of Epidemic Routing. Therefore, two extreme routing schemes have trade-off between delivery time and resource usage.

The routing schemes of DTNs can be categorized into two types, depending on existence of copies of a packet in a network; single copy schemes and multi-copy schemes (or forward schemes and replication schemes). In single copy schemes, only one node including a source has the original copy of a packet, so there is no duplicate

copy of a packet in a network; the node "*forwards*" the packet to an encountered node [83]. In multi-copy schemes, a node "*copies or replicates*" its packet to an encountered node [82]. So the number of copies of a packet in a network is more than or equal to one at any time.

Many single copy (forwarding) schemes had been introduced at the first stage of DTNs routings [11, 44, 83], but the performance of the single copy schemes was very poor due to the nature of DTNs such as infrequent contacts and long delay: low delivery ratio and large delivery time. However, the intuitions and methodologies forwarding schemes used become the foundation of multi-copy schemes. There are utility based, social based, and hybrid schemes in forwarding schemes.

In [11, 44], utility based routings are proposed. In utility based forwarding, we assume that each node keeps a utility value. Node A forwards to another node B only if the utility value of B is larger than that of A . Utility function is a monotonically decreasing function such as the last encounter timer after meeting a destination. "Seek and Focus" [83] is a hybrid routing scheme that uses randomized forwarding and utility based forwarding. When the utility value of a node is lower higher than pre-defined threshold value, the node forwards packets randomly and the utility value of a node is higher than pre-defined threshold value, the node forwards packets based on utility values.

Social based forwarding schemes such as SimBet [31] and BUBBLE Rap [46]

introduce the concepts of social networks such as communities and betweenness. Nodes in the same communities which are the destinations of packets or nodes that move frequently between communities are selected as good relays.

To make DTNs routings practical and efficient, multi-copy schemes have been proposed [19, 21, 59, 82]. While multi-copy schemes show higher performance than single copy schemes, the increased copies in multi-copy schemes may waste network resources too much or cause high contentions. As a result, it causes performance degradation. So most multi-copy schemes have tried to sustain the performance by reducing the number of (useless) copies as much as possible [74], [38].

"Spray and wait" [81] consists of two phases: spray and wait. In a spray phase, a source copies pre-defined number of copies (L) to another node. In a wait phase, each spread L copy per packet uses a single forward scheme to deliver the packet to the destination such as "direct transmission" and "utility forwarding". "PRoPHET" (Probabilistic Routing Protocol using History of Encounters and Transitivity) [61] uses transitive delivery probability as a utility value. When two nodes meet, packets are copied to another node only if the transitive delivery probability for the destination of the packet is higher at the other node including.

Since the semantic model of DTNs multicast was first introduced in [55], multicast in DTNs has been also studied in [12, 40, 85, 94]. In [100], four possible routing schemes for multicast in DTNs are introduced: unicast-based, broadcast-(flooding) based,

tree-based, and group-based. The possible approaches are adapted from multicast routing schemes of conventional mobile ad-hoc networks (MANETs). In group-based routing, a forwarding group is selected per packet and the forwarding groups that consist of multiple nodes are responsible for delivering packets to receivers. To do that, flooding is used within a forwarding group. Finally, [100] showed that group-based routing shows the best performance while unicast-based routing is the worst, and that the performance of routing schemes depends on the degree of knowledge to routings. Tree-based approach is also studied for multicast in DTNs [94]. [40] focuses a relay selection scheme in order to satisfy small number of hops to deliver packets.

Although multicast in DTNs has been extensively studied, most studies have been based on the tree-based approaches of mobile ad-hoc wireless networks (MANETs). To adapt the results from MANETs, these studies assume that mobile nodes move so slowly that multicast trees can be constructed and managed without problems. Due to the long delay on paths in DTNs, it is unrealistic to manage multicast trees efficiently.

In [72, 99], data disseminations in DTNs are considered using "message ferries" and cars respectively. Only direct contacts between a node and a message ferry or between car and infrastructure are considered, and contacts between nodes, between message ferries and between cars are not considered.

There are some other works [53, 80] for DTNs. [53] considers the feasibility of

receiver side broadcast in DTNs, and [80] assumes that networks consist of rational entities that behave selfish, which causes significant damage. To prevent selfish users from dropping others' message and also from replicating excessive their own messages, an incentive mechanism that the selfish nodes comforts to pair-wise tit-for-tat (TFT) is proposed.

However, the routing algorithms of most work aforementioned have been depended not on the optimal rules but on the intuitions or heuristics. Our approach is based on optimal framework. Much work has been done for routing related optimization in (traditional) wireless networks [43, 51, 69, 86]. Under the single-traffic class and end-to-end transmission requirements, optimal transmission scheduling schemes avoiding interference among transmissions from neighbor links are provided [86]. In [51], the capacity region is studied when greedy maximal scheduling algorithm is used in multi-hop wireless networks. In [43, 69], delay is analyzed in single hop and multi-hop wireless networks respectively. The results of most work are based on the assumptions of infinite backlogging and infinite time. Consequently, the work shows that proposed routing schemes achieve optimal throughput if the routing is done by using the queue lengths difference.

Recent theoretical works in DTNs [17, 62] aim at maximizing the delivery ratio. Other researchers additionally consider the energy constraints which limit the total number of copies [38] and [19] also formulates the DTNs as a resource allocation

problem that provides the optimal replication rule for packets.

To simulate the optimal rule, RAPID [19] provides a heuristic packet scheduling based on per-packet marginal utility where utility is defined as the average delay or the ratio of packets which missed a deadline. [57] adds an optimal drop policy for a limited buffer which drops packets in consideration of the per-packet marginal utility. Recent work [26] considers the case where link bandwidth is limited when contacts happen and it also considers packet scheduling for packets in a buffer (i.e., copy schedules) to enhance performance. By far, no other work has considered the joint constraints of interference and link and copy schedules. To solve DTNs' optimization problems, dynamic programming is used. Dynamic programming solutions require the future events and global knowledge for a network. Therefore dynamic programming has limitations in producing on-line solutions because of its complexity and the above mentioned requirement. Therefore, to make forwarding or replication rules online and practical, approximation is need. Our approach also uses the approximations for an optimal rule through the new concepts "snapshot optimality" and "DMC" we call. Those terms are explained in the later Sections.

Chapter 3

Multicast in 3G Cellular Data Networks

As of July 2006, the number of CDMA2000 1xEV-DO subscribers has exceeded 38.5 million [1]. Third-generation (3G) wireless data networks support high data rates, e.g. 2.4 Mbps for CDMA Evolution Data-Only (EV-DO) [22] and up to 14.4 Mbps for UMTS High-Speed Downlink Packet Access (HSDPA) [10], enabling a broader range of bandwidth-intensive services. These include streaming media such as the MobiTV service [68] from Sprint, Cingular and Alltel, and the VCast service [66] from Verizon. More sophisticated services, ones which incorporate location information, e.g., live regional traffic reports, geographically targeted advertisements, are expected next. A key factor distinguishing these new applications is that they

are naturally amenable to multicast transmission from the base station in a cell.

Figure 3.1 shows the architecture of today's 3G cellular data networks. Mobiles need to go through the hierarchy of base station, base station controller and gateway in order to access the Internet. In this hierarchy, the air interface connecting the mobiles to the base station is the bottleneck because of limited bandwidth. Therefore, air interface scheduling is very important. Also, the MAC layer scheduler is located at the base station, and the scheduler makes independent decisions for users within each cell.

There has been much research on unicast scheduling in cellular networks (e.g., [13, 15, 22, 23, 48, 64, 67, 71, 90]). Typically, these systems employ Time Division Multiplexing (TDM) in the downlink direction; real time is divided into small fixed time slots. For example, CDMA2000 1xEV-DO downlinks use TDM with a time slot length of 1.67 ms. During different time slots, each user may experience a different signal-to-noise ratio (SNR), which determines the maximum rate at which this user can receive data reliably. For unicast, for each slot, each user sends the Data Rate Control (DRC) message specifying this maximum rate back to the base station. Since each user can specify a different DRC, the *unicast* scheduler at the base station needs to decide on *which user* to be served at each slot based on user DRC feedback. Once the base station selects a user, it transmits to the user using a modulation and error coding scheme suitable for its DRC rate. Note that if the base station sends at a rate

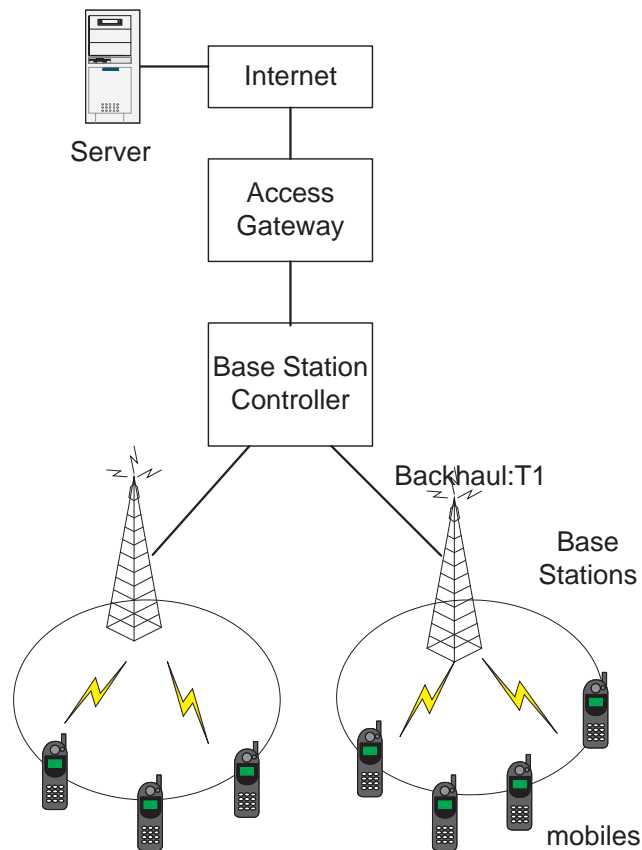


Figure 3.1: Architecture of 3G Cellular Data Networks. In this architecture, only the interface connecting the mobiles to the base station is the wireless links.

higher than the DRC rate of user, then the user *cannot* receive *any* data. State of the art unicast schedulers exploit channel states of users to increase overall throughput; usually by favoring transmissions to users with high DRC rates.

What makes the design of a multicast scheduler different from unicast? In multicast, at each time slot, the base station can transmit only to one multicast group at one rate. There are multiple multicast groups in a cell and each multicast group

may contain a different number of users which might be located at diverse locations within a cell. Note that a multicast may span over multiple cell, but since multicast scheduling applies to one base station, we restrict our consideration only to one cell. The primary difficulty in multicast scheduling for 3G cellular data networks stems from mismatch in data rates attainable by individual members within a multicast group. Recall that if the base station transmits data at a higher rate than the maximum rate that a user can handle, then the device is incapable of decoding *any* of the transmitted data. Since all users in a multicast group must be subject to the same transmission rate picked by the base station at each time slot, it is difficult to find one rate at which the base station sends multicast to a group. If it sends at the highest rate that users ask for, then there will be many receivers who may not get the transmission, and if the base station sends at the lowest rate requested by the users in a group, then other users with higher DRC rates (better channel condition) will be subject to lower rates than their DRC rates. Therefore the challenging job of the multicast base station is that at each time slot, the base station must decide *which group* to transmit to, and choose *what rate* at which it transmits to that group as shown in Figure 3.2. Figure 3.3 shows an example of multicast transmission. Assume that the base station transmits data to group 2 with rate 5. Because users that have lower DRC than the transmitted data rate cannot handle the transmitted data, only users 3 and 4 in group 2 can receive the data.

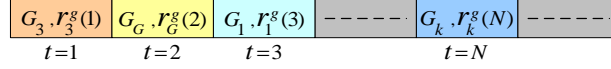


Figure 3.2: Time slots for multicast schedulers. At each time slot, based on the DRC feedbacks from all users, one group is selected, and one data rate is selected to transmit to the chosen group. G_i and $r_i^g(t)$ represent a selected group id i and transmission rate at time slot t respectively.

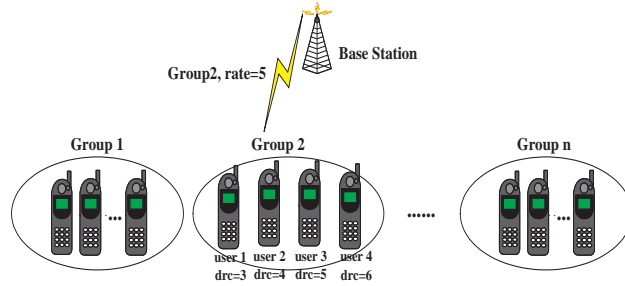


Figure 3.3: Multicast Example: if the base station transmits data to group 2 with transmission rate 5, then only user 3 and 4 within group 2 receive the data.

One simple way of multicast scheduling is to fix the transmission rate to a default value and do a round-robin among all the groups. The default rate is typically set to handle the possible DRC values of users located at the edge of a cell. This rate is the worst rate because it assumes that there is always a user at the edge of the cell whether such a user is actually present or not. The current CDMA2000 1xEV-DO [13] networks use this approach. However, this scheme does not take into consideration user DRC rates, so it significantly limits the throughput of users, especially for those close to the base station with good channel conditions. Also, this scheme does not necessarily maximize any form of user utility, so it is oblivious towards fairness be-

tween users and between groups.

Perhaps the natural solution to improve multicast data throughput is to partition users with similar channel conditions into the same multicast group. However, because of the channel condition dynamics and significant signaling overhead associated with group membership changes, such a solution is not practical. Our goal is to improve multicast throughput without imposing membership changes.

Instead of fixing multicast data rates to the lowest rate requested by users in a group, we propose two multicast scheduling schemes that leverage the unicast feedback of user DRC rates to select an appropriate multicast data rate and a group for transmission in each TDM slot. To improve the system throughput, our algorithms may select higher rates than the lowest rate requested by a group. This means that when the base station chooses a multicast group and its transmission rate, some users in that group may not be able to receive the transmitted data. Therefore choosing the group data rate is important as well as deciding the transmitted group. Because user channel conditions vary, different users will miss different packets in the multicast stream. We next describe how such a scheduler is useful in different application scenarios.

In this thesis, we consider two different types of multicast application scenarios and present two multicast scheduling algorithms that maximize two different utility functions. The first is applicable to delay tolerant cooperative data downloads while

the second applies to multimedia content distribution for typical 3G multicast data networks.

In the first scenario, the objective of this scheduler is to maximize the sum of $\log T_k^g$ for all groups. We assume that for group k , the utility of the entire group is $\log T_k^g$, where T_k^g is the group throughput for group k and the group throughput is defined as the sum of individual user throughput within a group. We call this scheduler the *Inter-group Proportional Fair* (IPF) scheduler.

IPF is likely to be useful in delay tolerant networks, possibly with nomadic users who have intermittent connectivity. IPF is useful when group members can cooperatively download data (which they share within the group), perhaps by forming an ad-hoc network. The original data could be source coded, e.g. using digital fountain type codes [29,41], and the downloaded data could be subsequently reconciled within a group [28].

In the second scenario, multiple groups of users stay in a cell and these users receive some (possibly different) multimedia content from the base station. We consider a utility function of the form, $\log T_i$, where T_i is the receiving throughput of user i . User i 's utility (or happiness) increases if more packets are received (user happiness increases fast at the beginning and slows down gradually). The scheduler's objective is to maximize the sum of $\log T_i$ for all user i 's in a cell; we call such a scheduler *Multicast Proportional Fair* (MPF) scheduler as it needs to achieve proportional fairness [54]

(because of the log utility function) within the constraint of multicast.

Recall that the MPF scheduler will sometimes select data rates that cause certain users (at the far edge of the cell) not to be able to receive data during certain time slots. This will certainly translate into lost packets and subsequently lost application frames for the users. Some forward error correction schemes can be applied (at a packet level) such that these poorly situated users can recover some portion of the application data. However, doing so defeats the purpose of varying the rates since the users who are capable of receiving at a higher rate will, instead, receive (redundant) error correcting packets. Ideally, while users with poor channel conditions receive a “base” level of service, users with better channel conditions receive a higher quality data service.

3.1 Background

3.1.1 Proportional Fairness

Maximizing the network throughput as a primary objective may lead to gross unfairness; in the worst case, some users may get zero throughput, which is considered unfair by these users. So we need to consider both efficiency and fairness. Informally, proportional fairness takes into consideration the usage of network resources and favors smaller rates less emphatically [25] [55].

A vector of rates x^* is proportionally fair if it is feasible and if for any other feasible vector x , the aggregate of proportional change is zero or negative [55]: i.e.

$$\sum_i \frac{x_i - x_i^*}{x_i^*} \leq 0$$

The Proportional Fair (PF) scheduling algorithm offers optimal channel utilization for the base station. At the same time, the algorithm is proportionally fair to users according to their channel conditions. For example, if the channel conditions are identical for all users, then all users have the same throughput. However, if channel conditions are different among the users, then the users with better channel conditions will have higher throughput than users with worse channel conditions.

3.1.2 Unicast Scheduler

On the forward link of unicast connections in CDMA2000 1xEV-DO, data is transmitted to different users in TDM fashion using the full transmission power of the base station. The rate transmitted to each user varies depending on each user's measured SNR. Based on the SNR, the user sends to the base station the highest data rate which the base station can use to send data [65].

In the unicast case, the scheduler determines the next user to be served based on the reported data rate request from the users and the amount of data that has already been transmitted to each user. At the time t , the scheduler transmits data to

the user i^* with the largest $\frac{DRC_i(t)}{T_i(t)}$ among all active users in a cell, where $T_i(t)$ is the average throughput and $DRC_i(t)$ is the current data request rate of user i at time t . Also the average throughput received by each user is updated as follows.

$$T_i(t+1) = \begin{cases} (1 - \frac{1}{t_c})T_i(t) + \frac{1}{t_c}DRC_i(t) & i = i^* \\ (1 - \frac{1}{t_c})T_i(n) & i \neq i^* \end{cases}$$

, where t_c is a latency time scale in number of time slots

3.2 Model Description

We consider a system with one base station (BS), and only multicast transmission is scheduled. Mobile devices are capable of maintaining unicast and multicast transmissions simultaneously and the base station uses the DRC feedback for the unicast connection to determine multicast data rate. Throughout this thesis, we use the terms user, terminal, and mobile device/access terminal (AT) interchangeably.

Recall that during a time slot, the unicast scheduler has to make a single decision, namely, which AT to transmit to. However, the multicast scheduler needs to make two decisions: (a) which group to transmit to and (b) the data rate at which to transmit.

3.2.1 Notation

We use the following notation:

- G : the number of multicast groups.
- $r_{ik}(t)$: DRC of Access Terminal (AT) i in group k at time t . We assume $0 < r_{min} \leq r_{ik}(t) \leq r_{max}$ where r_{min} and r_{max} represent the minimum and maximum possible DRC, respectively, and set $D = r_{max} - r_{min} + 1$.
- $r_k^g(t)$: feasible rate assigned to group k at time t .
- $T_{ik}(t)$: the (exponential/moving) average throughput of AT i in group k at time t .
- S_k : size of group k , i.e., total number of ATs in group k .
- $\vec{r}_i(t) = \{r_{1i}(t), \dots, r_{S_{ii}}(t)\}$: DRC of group i at time t .
- $X = \{\vec{r}_1(t), \vec{r}_2(t), \dots, \vec{r}_G(t)\} \in \Omega$ is a DRC vector of the system where Ω is a collection of all feasible DRC vectors. We say X is feasible, i.e., $X \in \Omega$ if each component of X lies between r_{min} and r_{max} . Let P be the number of all feasible DRC vectors, i.e., $\Omega = \{X_1, X_2, \dots, X_P\}$ where

$$X_i = \{x_{1,1}^i, x_{2,1}^i, \dots, x_{S_{1,1}}^i, x_{1,2}^i, \dots, x_{S_G,G}^i\}$$

We assume $x_{m,k}^i$ ($i=1, 2, \dots, P$; $k=1, 2, \dots, G$; $m=1, 2, \dots, S_k$) are all positive integers, as well as r_{min}, r_{max} (or else we can rescale them); and define

$$R_i = r_{min} + i - 1. \quad (i = 1, 2, \dots, D) \quad (3.1)$$

Let \mathcal{S} denote the scheduler under discussion and

$$1_k^{\mathcal{S}}(t) := \begin{cases} 1, & \mathcal{S} \text{ chooses to serve group } k \text{ at time } t \\ 0, & \text{otherwise.} \end{cases} \quad (3.2)$$

Then, the (exponential/moving) average throughput is updated at each time t by

$$T_{ik}(t+1) = (1 - \frac{1}{t_c})T_{ik}(t) + \frac{1}{t_c}r_k^g(t)1_k^s(t)1_{\{r_k^g(t) \leq r_{ik}(t)\}}$$

where t_c is latency time scale in number of time slots. In other words, AT i in group k will receive rate of $r_k^g(t)$ only if the group k is selected and its DRC value $r_{ik}(t)$ is no less than $r_k^g(t)$. Similarly, we define

$$I_i^{\mathcal{S}}(t) := \begin{cases} 1, & \mathcal{S} \text{ chooses to serve at rate } R_i \text{ at time } t \\ 0, & \text{otherwise.} \end{cases} \quad (3.3)$$

3.3 Inter-group PF Scheduler

In this section we propose a multicast scheduler that achieves inter-group proportional fairness (IPF). Denote the (aggregate) throughput of group k at time t

by

$$T_k^g(t) := \sum_{i=1}^{S_k} T_{ik}(t). \quad (3.4)$$

We define

$$\phi_{k,t}(y) = \sum_{n=1}^{S_k} y 1_{\{y \leq r_{nk}(t)\}} \quad (3.5)$$

as the aggregate rate of all ATs in group k at time t when the BS's transmission rate is y . Let $T_k^{\mathcal{S}}$ be the long-term (arithmetic) average throughput of group k under scheduler \mathcal{S} . Then, we have

$$T_k^{\mathcal{S}} = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{t=1}^N 1_k^{\mathcal{S}}(t) \phi_{k,t}(r_k^g(t)), \quad (3.6)$$

and similarly for $T_k^{\mathcal{S}^*}$ with $1_k^{\mathcal{S}}(t)$ replaced by $1_k^{\mathcal{S}^*}(t)$. We assume that the long-term average throughput always exists [71] for any multicast scheduler \mathcal{S} under our consideration. In this case, note that we can also write $T_k^{\mathcal{S}} = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{t=1}^N T_k^g(t)$, i.e., the arithmetic time average and exponential average become identical in the limit.

Definition 1 *A scheduler \mathcal{S}^* is IPF if, for any other scheduler \mathcal{S} ,*

$$\sum_{k=1}^G \frac{T_k^{\mathcal{S}} - T_k^{\mathcal{S}^*}}{T_k^{\mathcal{S}^*}} \leq 0, \quad (3.7)$$

where $T_k^{\mathcal{S}}$ and $T_k^{\mathcal{S}^*}$ are defined as in (3.6). In other words, the aggregate of proportional changes in the long-term average group throughput of \mathcal{S}^* caused by any other scheduler \mathcal{S} must be non-positive. \square

3.3.1 IPF Scheduling Algorithm

IPF scheduler \mathcal{S}^* :

The feasible rate assigned to group k at time t is

$$r_k^g(t) = \arg \max_y \phi_{k,t}(y). \quad (3.8)$$

The BS chooses group $k(t)$ to transmit at rate $r_k^g(t)$,

$$\text{where } k(t) = \arg \max_{1 \leq k \leq G} \frac{\phi_{k,t}(r_k^g(t))}{T_k^g(t)}. \quad (3.9)$$

Note that $r_k^g(t)$ represents the transmission rate at which the aggregate rate of group k is maximized. We observe that $r_k^g(t)$ is always equal to the DRC of some AT in that group. To see this, suppose otherwise that $r_k^g(t)$ falls between DRC values of two ATs in that group. Then, from (3.5), we can always increase $r_k^g(t)$ to the larger DRC (between these two values), which further increases $\phi_{k,t}$, and this leads to a contradiction. In consequence, we call $r_k^g(t)$ the *DRC of group k* , which plays the similar role in selecting a group to transmit, as the DRC of one node does in selecting a node to transmit in the unicast setting [48, 90, 93].

Once the *DRC of each group* is determined, the BS selects group $k(t)$ as indicated by (3.9). Note that this is similar to what unicast PF scheduler does, i.e., preferring to the group that receives smaller amount of service ($T_k^g(t)$) up till now compared to its capability ($\phi_{k,t}(r_k^g(t))$) [93].

Note that the definition of IPF in (3.7) is ill-defined when $T_h^{\mathcal{S}^*} = 0$ for some h . Under the proposed scheduler \mathcal{S}^* , however, we show that this never happens.

Lemma 1 *For any $1 \leq k \leq G$, $T_k^{\mathcal{S}^*} > 0$.* □

Proof: The outline of the proof is as follows.

First, we see that $T_{k'}^{\mathcal{S}^*} > 0$ for at least one k' . This is because $\sum_{k=1}^G T_k^{\mathcal{S}^*}$ is lower bounded by $r_{\min}/(S_1 + S_2 + \dots + S_G) > 0$ since at any time at least one AT in one group gets served at rate no less than $r_{\min} > 0$. Then it can be shown that if $T_{k'}^{\mathcal{S}^*} > 0$ for some k' , then there exists $\xi > 0$ (as a function of r_{\min}, r_{\max} and $S_{k'}$) such that $T_h^{\mathcal{S}^*}/T_{k'}^{\mathcal{S}^*} \geq \xi > 0$ for any $1 \leq h \leq G$. To see this, assume the reverse inequality, that is, there exists some h such that $T_h^{\mathcal{S}^*}/T_{k'}^{\mathcal{S}^*} < \xi > 0$. The inequality becomes $1/T_{k'}^{\mathcal{S}^*} < \xi/T_h^{\mathcal{S}^*}$. Then there exists $t_0 > 0$ such that group k' will never be served after t_0 since

$$\frac{\phi_{k',t}(r_{k'}^g(t))}{T_{k'}^g(t)} < \frac{\phi_{h,t}(r_h^g(t))}{T_h^g(t)}, \quad \text{for all } t > t_0,$$

which leads to $T_{k'}^{\mathcal{S}^*} = 0$, a desired contradiction. ■

3.3.2 Inter-group PF

We here prove that scheduler \mathcal{S}^* achieves IPF. To proceed, we define by $f_{k,h,i,j}^{\mathcal{S}^*,\mathcal{S}}(p)$ ($1 \leq i, j \leq D$; $1 \leq k, h \leq G$; $1 \leq p \leq P$) the empirical probability that the DRC vector is X_p , \mathcal{S}^* selects group k , transmission rate R_i as defined in (3.1), and \mathcal{S} selects group

h , transmission rate R_j . To be precise,

$$f_{k,h,i,j}^{\mathcal{S}^*,\mathcal{S}}(p) = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{t=1}^N 1_k^{\mathcal{S}^*}(t) 1_h^{\mathcal{S}}(t) I_i^{\mathcal{S}^*}(t) I_j^{\mathcal{S}}(t) \times 1_{\{r_{11}(t), \dots, r_{S_G G}(t)\} = X_p\}} \quad (3.10)$$

where $1_k^{\mathcal{S}^*}$ and $1_h^{\mathcal{S}}$ are defined in (3.2), whereas $I_i^{\mathcal{S}^*}$ and $I_j^{\mathcal{S}}$ are defined in (3.3). $f_{k,h,i,j}^{\mathcal{S}^*,\mathcal{S}}(p)$ can be looked as the joint probability mass function (pmf) indicating the probability that {group index, AT index} given by \mathcal{S}^* and \mathcal{S} are $\{k, i\}$ and $\{h, j\}$ respectively, when the DRC vector is X_p . From above arguments, we have the following:

Lemma 2 *For any k, h, i, j, p , we have*

$$\sum_{n=1}^{S_k} \frac{f_{k,h,i,j}^{\mathcal{S}^*,\mathcal{S}}(p) R_i 1_{\{x_{n,k}^p \geq R_i\}}}{T_k^{\mathcal{S}^*}} \geq \sum_{n=1}^{S_h} \frac{f_{k,h,i,j}^{\mathcal{S}^*,\mathcal{S}}(p) R_j 1_{\{x_{n,h}^p \geq R_j\}}}{T_h^{\mathcal{S}^*}}. \quad (3.11)$$

□

Proof: The result becomes trivial when $f_{k,h,i,j}^{\mathcal{S}^*,\mathcal{S}}(p) = 0$. So, we only need to consider the case of $f_{k,h,i,j}^{\mathcal{S}^*,\mathcal{S}}(p) > 0$. In this case, there exists t_m ($1 \leq m \leq T_f$), where T_f is the total number of time slots, such that

$$1_k^{\mathcal{S}^*}(t_m) 1_h^{\mathcal{S}}(t_m) I_i^{\mathcal{S}^*}(t_m) I_j^{\mathcal{S}}(t_m) 1_{\{r_{11}(t_m), \dots, r_{S_G G}(t_m)\} = X_p\}} = 1.$$

Hence, all terms on the LHS of the above equation are equal to 1, and thus $r_k^g(t_m) = R_i$ for all such m since $I_i^{\mathcal{S}^*}(t_m) = 1$, which then leads to $\phi_{k,t}(r_k^g(t_m)) = \phi_{k,t}(R_i)$. Also, group k is selected by our scheduler \mathcal{S}^* at every such t_m since $1_k^{\mathcal{S}^*}(t_m) = 1$. Consequently,

from (3.9), we have

$$\frac{\phi_{h,t_m}(r_h^g(t_m))}{T_h^{\mathcal{S}^*}} \leq \frac{\phi_{k,t_m}(r_k^g(t_m))}{T_k^{\mathcal{S}^*}}, \quad (1 \leq m \leq T_f). \quad (3.12)$$

When $h = k$, from (3.8), the result follows by noting that

$$\frac{\text{LHS of (3.11)}}{f_{k,h,i,j}^{\mathcal{S}^*,\mathcal{S}}(p)} = \frac{\phi_{k,t_m}(r_k^g(t_m))}{T_k^{\mathcal{S}^*}} \geq \frac{\phi_{k,t_m}(R_j)}{T_k^{\mathcal{S}^*}} = \frac{\text{RHS of (3.11)}}{f_{k,h,i,j}^{\mathcal{S}^*,\mathcal{S}}(p)}.$$

When $h \neq k$, we have

$$\frac{\text{LHS of (3.11)}}{f_{k,h,i,j}^{\mathcal{S}^*,\mathcal{S}}(p)} = \frac{\phi_{k,t_m}(r_k^g(t_m))}{T_k^{\mathcal{S}^*}} \geq \frac{\phi_{h,t_m}(r_h^g(t_m))}{T_h^{\mathcal{S}^*}} \quad (3.13)$$

$$\geq \frac{\phi_{h,t_m}(R_j)}{T_h^{\mathcal{S}^*}} = \frac{\text{RHS of (3.11)}}{f_{k,h,i,j}^{\mathcal{S}^*,\mathcal{S}}(p)}, \quad (3.14)$$

where the inequality in (3.13) is from (3.12) and (3.14) is from (3.8). This completes the proof. \blacksquare

We now show the following.

Proposition 1 *The multicast scheduler \mathcal{S}^* described in (3.8) and (3.9) is IPF.* \square

Proof: By expanding $T_k^{\mathcal{S}^*}$ and $T_h^{\mathcal{S}}$ in terms of their joint pmf with respect to the DRC vector X_p , we have

$$\begin{aligned} T_k^{\mathcal{S}^*} &= \sum_{m=1}^{S_k} \sum_{h=1}^G \sum_{p \in \mathbb{P}, i, j \in \mathbb{D}} f_{k,h,i,j}^{\mathcal{S}^*,\mathcal{S}}(p) R_i 1_{\{x_{m,k}^p \geq R_i\}} \\ T_h^{\mathcal{S}} &= \sum_{n=1}^{S_h} \sum_{k=1}^G \sum_{p \in \mathbb{P}, i, j \in \mathbb{D}} f_{k,h,i,j}^{\mathcal{S}^*,\mathcal{S}}(p) R_j 1_{\{x_{n,h}^p \geq R_j\}}, \end{aligned} \quad (3.15)$$

where $\mathbb{P} = \{1, \dots, P\}$ and $\mathbb{D} = \{1, \dots, D\}$.

For notational simplicity, we use $\sum_{p,i,j}$ instead of $\sum_{p \in \mathbb{P}} \sum_{i,j \in \mathbb{D}}$. Observe now that

$$\begin{aligned}
\sum_{h=1}^G \frac{T_h^{\mathcal{S}}}{T_h^{\mathcal{S}^*}} &= \sum_{h=1}^G \sum_{n=1}^{S_h} \sum_{k=1}^G \sum_{p,i,j} \frac{f_{k,h,i,j}^{\mathcal{S}^*,\mathcal{S}}(p) R_j 1_{\{x_{n,h}^p \geq R_j\}}}{T_h^{\mathcal{S}^*}} \\
&= \sum_{h=1}^G \sum_{k=1}^G \sum_{p,i,j} \sum_{n=1}^{S_h} \frac{f_{k,h,i,j}^{\mathcal{S}^*,\mathcal{S}}(p) R_j 1_{\{x_{n,h}^p \geq R_j\}}}{T_h^{\mathcal{S}^*}} \\
&\leq \sum_{h=1}^G \sum_{k=1}^G \sum_{p,i,j} \sum_{n=1}^{S_k} \frac{f_{k,h,i,j}^{\mathcal{S}^*,\mathcal{S}}(p) R_i 1_{\{x_{n,k}^p \geq R_i\}}}{T_k^{\mathcal{S}^*}} \\
&= \sum_{k=1}^G \sum_{n=1}^{S_k} \sum_{h=1}^G \sum_{p,i,j} \frac{f_{k,h,i,j}^{\mathcal{S}^*,\mathcal{S}}(p) R_i 1_{\{x_{n,k}^p \geq R_i\}}}{T_k^{\mathcal{S}^*}} \\
&= \sum_{k=1}^G \frac{T_k^{\mathcal{S}^*}}{T_k^{\mathcal{S}^*}} = \sum_{h=1}^G \frac{T_h^{\mathcal{S}^*}}{T_h^{\mathcal{S}^*}},
\end{aligned}$$

where both the first and the fourth equalities are from (3.15) and the inequality follows from Lemma 2. This shows (3.7) and we are done. \blacksquare

In our IPF algorithm, each group has to decide its “group DRC” r^g as a representative value ¹ (See (3.8)). Depending on the DRC values of users in that group (r_i , $i = 1, 2, \dots, S$), some of them will get that rate and some of them will not (zero rate). Note that the BS does not care why each group asks that rate; it only performs the usual PF scheduling based on these group DRCs. We already saw that under IPF, the group rate r^g maximizes the total aggregate rate to that group if the BS chooses that group for service at rate r^g .

As briefly mentioned in the introduction, we can interpret the IPF algorithm as a fair and reliable transfer in delay-tolerant, lossy multicast networks. Consider a mul-

¹For simple exposition, we here suppress the group index k and time index t from the original definitions.

unicast network with some delay-tolerant applications where each packet transmission is subject to high error/loss due to severe wireless link characteristics or interference, etc. Let q be the probability that each packet transmission (from the BS to each user in a chosen group) is successful. We assume q is very small (unreliable network) and that each packet is subject to an error independently of everything else. Note that under IPF scheduler \mathcal{S}^* , we have

$$r^g = \arg \max_y \phi(y), \quad (3.16)$$

where $\phi(y) = \sum_{i=1}^S y 1_{\{y \leq r_i\}}$.

Suppose the BS chooses rate y for this group (y packets per unit time slot). Then, the number of users $M = M(y)$ that receive rate of y (non-zero rate) in that group becomes $M = \sum_{i=1}^S 1_{\{y \leq r_i\}}$. Note that each packet transmission will be successful with probability q , and that there are M such duplicate packets. Thus, the probability P that at least one of these M packets will get through (delivered to at least one user in that group) becomes

$$P = 1 - (1 - q)^M \approx qM \quad \text{for small } q.$$

Then, the average number of successful packets $\mathbb{E}\{N_s\}$ delivered to that group becomes

$$\mathbb{E}\{N_s\} = yP \approx qyM = q \sum_{i=1}^S y 1_{\{y \leq r_i\}} = q\phi(y). \quad (3.17)$$

So $\mathbb{E}\{N_s\}$ is maximized when y is the maximizer of the RHS of (3.17). Note that, this is exactly the group rate selection algorithm for IPF as in (3.16). In other words, under IPF, the group DRC corresponds to the “most reliable” rate for that group.

3.3.3 General Inter-group PF Scheduler

In this section, we extend proportional fairness to be more general. We use the notion of (\vec{p}, α) PF [67] to develop more candidate schedulers which have different weights in tradeoff between throughput and fairness. While the (\vec{p}, α) PF ($\vec{p} = (p_1, \dots, p_m)$ where $p_i > 0$) can be achieved in unicast setting [71], in multicast, it is only possible when $p_1 = p_2 = \dots, p_m$. In other words, we can design multicast schedulers that is (\vec{e}, α) PF.

Definition 2 A scheduler $\mathcal{S}^*(\alpha)$ is (\vec{e}, α) IPF if, for any other scheduler \mathcal{S} ,

$$\sum_{h=1}^G \frac{T_h^{\mathcal{S}} - T_h^{\mathcal{S}^*(\alpha)}}{(T_h^{\mathcal{S}^*(\alpha)})^\alpha} \leq 0, \quad (3.18)$$

where $\vec{e} = \{1, 1, \dots, 1\}$ and $0 \leq \alpha \leq 1$. □

We propose the following set of schedulers:

(\vec{e}, α) **IPF scheduler** $\mathcal{S}^*(\alpha)$:

The feasible rate assigned to group k is $r_k^g(t)$ defined in (3.8) (same as scheduler \mathcal{S}^*).

The BS chooses group $k(t)$ to transmit at rate $r_k^g(t)$ where

$$k(t) = \arg \max_{1 \leq k \leq G} \frac{\phi_{k,t}(r_k^g(t))}{(T_k^g(t))^\alpha}. \quad (3.19)$$

Here, $T_k^g(t)$, $\phi_{k,t}(y)$ are defined in (3.4), (3.5), respectively.

Note that when $\alpha = 0$, the scheduler becomes the *MAX throughput scheduler*; when $\alpha = 1$, the scheduler becomes the *IPF scheduler* \mathcal{S}^* . Similarly as in the proof of Proposition 1, we can show that $\mathcal{S}^*(\alpha)$ is (\vec{e}, α) IPF.

3.4 Multicast PF Scheduler

In this section we propose a multicast scheduler that achieves the proportional fairness among all the users (ATs) in the system. To distinguish it from PF in the unicast setting, we call it Multicast PF (MPF).

Let $T_{m,k}^{\mathcal{S}}$ be the long-term (arithmetic) average throughput of AT m in group k under scheduler \mathcal{S} . Then, we have

$$T_{m,k}^{\mathcal{S}} = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{t=1}^N 1_k^{\mathcal{S}}(t) r_k^g(t) 1_{\{r_k^g(t) \leq r_{mk}(t)\}}. \quad (3.20)$$

As in Section 3.3, we assume that the long-term average throughput exists for any multicast scheduler \mathcal{S} under consideration.

Definition 3 A scheduler \mathcal{S}^\diamond is MPF if, for any other scheduler \mathcal{S} ,

$$\sum_{h=1}^G \sum_{n=1}^{S_h} \frac{T_{n,h}^{\mathcal{S}} - T_{n,h}^{\mathcal{S}^\diamond}}{T_{n,h}^{\mathcal{S}^\diamond}} \leq 0, \quad (3.21)$$

where $T_{n,h}^{\mathcal{S}}$ and $T_{n,h}^{\mathcal{S}^\diamond}$ are defined as in (3.20). In other words, the aggregate of proportional changes in the long-term average user throughput of \mathcal{S}^\diamond caused by any other scheduler \mathcal{S} must be non-positive. \square

3.4.1 MPF Scheduling Algorithm

MPF scheduler \mathcal{S}^\diamond :

The feasible rate assigned to group k at time t is

$$r_k^g(t) = \arg \max_y \varphi_{k,t}(y), \quad (3.22)$$

$$\text{where } \varphi_{k,t}(y) = \sum_{n=1}^{S_k} \frac{y}{T_{nk}(t)} 1_{\{y \leq r_{nk}(t)\}}. \quad (3.23)$$

The BS chooses group $k(t)$ to transmit at rate $r_k^g(t)$

$$\text{where } k(t) = \arg \max_{1 \leq k \leq G} \varphi_{k,t}(r_k^g(t)). \quad (3.24)$$

Note that $r_k^g(t)$ is the transmission rate at which ‘weighted’ aggregate rate of group k is maximized, where the weight of each AT is $1/T_{nk}(t)$, and $T_{nk}(t)$ is the (exponential/moving) average throughput of AT n in group k at time t . Thus, if the AT has received less throughput so far (smaller T_{nk}), then that AT will have more

weight (importance) in determining the feasible rate $r_k^g(t)$ in (3.22) as well as the group selection in (3.24), since smaller T_{nk} will make $\varphi_{k,t}(r_k^g(t))$ also larger. As a special case, when there is only one user in each group, i.e., the size of all groups is 1, note that MPF scheduler \mathcal{S}^\diamond becomes the usual unicast PF scheduler [48, 90, 93].

Similarly as in Lemma 1, we can also show the following for scheduler \mathcal{S}^\diamond .

Lemma 3 *For any $1 \leq k \leq G$ and $1 \leq m \leq S_k$, $T_{m,k}^{\mathcal{S}^\diamond} > 0$.* \square

Proof: For simpler notation, use $T_{m,k}$ instead of $T_{m,k}^{\mathcal{S}^\diamond}$ to denote the long-term average throughput of AT m in group k under the scheduler \mathcal{S}^\diamond defined in (3.22) and (3.24). Moreover, assume $T_{m,k}$ is ordered in each group k , i.e., $T_{1,k} \geq T_{2,k} \geq \dots T_{S_k,k}$. Note that here we only assume the ordered relationship for long-term throughput, i.e., for their value at arbitrary time t , this order relationship may not be true. In the following, three corollaries are provided to complete the proof. \blacksquare

Corollary 1 *There exists at least one k ($1 \leq k \leq G$) such that $T_{S_k,k} > 0$.* \square

Proof: Assume to any group k , $T_{S_k,k} = 0$. In other words, the minimum throughput of every group is 0. Then to each group k , define

$$j_k := \min\{j : T_{j,k} = 0, 1 \leq j \leq S_k\} \quad (3.25)$$

In other words, if $j_k > 1$, $T_{j_k,k} = 0$, $T_{j_k-1,k} > 0$. Note that $j_k = 1$ means that the long-term averaged throughput of all ATs in group k are 0.

Since $T_{j_k-1,k}(t) \rightarrow T_{j_k-1,k}$, there exist t_1^k such that $T_{i,k}(t) \in (0.5T_{i,k}, 2T_{i,k})$ to all $t > t_1^k$ where $1 \leq i \leq j_k - 1$. Similarly, since $T_{j_k,k}(t) \rightarrow 0$, there exist t_2^k such that $T_{j_k,k}(t) \in [0, \xi_k)$ to all $t > t_2^k$ where

$$\xi_k = \frac{r_{min}}{r_{max}} \frac{T_{j_k-1,k}}{2} \frac{1}{S_k}.$$

Hence, to all $t > t_0^k = \max\{t_1^k, t_2^k\}$, let $j_k^*(t) = \arg \min_{1 \leq i \leq j_k-1} T_{i,k}(t)$, then exists $\xi > 0$ such that

$$\frac{T_{j_k,k}(t)}{T_{j_k^*(t),k}(t)} < \xi$$

In other words, to all $t > t_0^k$

$$\frac{1}{T_{j_k,k}(t)} > \frac{1}{T_{j_k^*(t),k}(t)}$$

which means once group k is selected after some time t , the transmission rate of the BS is always no larger than the DRC of AT j_k . It means at least one of the ATs with throughput 0 until that time is served at that rate.

To be more clear, after t_0^k , once group k is selected, at least one of the AT with long-term averaged throughput 0 will be served at a rate no less than r_{min} . Since this is true to any group k , let $t_0 = \max_{1 \leq k \leq G} t_0^k$, then after t_0 , at least one of the AT with long-term averaged throughput 0 will be served at each time slot. Define

$$T_0(t) := \sum_{k=1}^G \sum_{i=j_k}^{S_k} T_{i,k}(t)$$

From (3.25), $T_0(t) \rightarrow 0$. However, from the above argument after t_0 , $T_0(t)$ is increased at least by $r_{min} > 0$ at each time slot. This introduces contradiction and completes

our proof. ■

Corollary 2 *Suppose there exists k such that $T_{S_k,k} > 0$, then to any h ($1 \leq h \leq G$),*

$$\frac{T_{1,h}}{T_{S_k,k}} > 0 \quad (3.26)$$

□

Proof: Note that $T_{i,k} \geq T_{S_k,k} > 0$. When $h = k$, (3.26) is obvious. When $h \neq k$.

Assume (3.26) is not true, then to all $t > t_0$,

$$\frac{1}{T_{1,h}(t)} > \frac{1}{T_{j_k^*(t),k}(t)} > \frac{1}{T_{S_k,k}(t)} \quad (3.27)$$

which means that the group k will never be selected after t_0 .

This contradicts with our assumption and completes the proof. ■

Corollary 3 *In any group k ($1 \leq k \leq G$),*

$$T_{i,k} \geq \left[\frac{r_{min}}{4S_k r_{max}} \right]^{i-1} T_{1,k} > 0 \quad (i = 1, 2, \dots, S_k).$$

□

Proof: The second equality is direct from Corollary 1 and 2. Follow the same line as in Corollary 1 and 2, we can show that

$$\frac{T_{2,k}}{T_{1,k}} \geq \frac{r_{min}}{4S_k r_{max}}. \quad (3.28)$$

Or else, there exists t_0 such that to all $t > t_0$,

$$\frac{r_{min}}{T_{2,k}(t)} > \frac{S_k r_{max}}{T_{1,k}(t)},$$

i.e., after t_0 , once group k is selected, the transmission rate of the BS is not larger than the DRC of AT 2. Consequently, the ratio $T_{2,k}/T_{1,k}$ will increase to 1 (note that $T_{1,k} > 0$ by assumption), which causes contradiction and completes the proof for (3.28).

As for other ATs $j = 3, 4, \dots, S_k$, follow the same way to prove

$$\frac{T_{j,k}}{T_{j-1,k}} \geq \frac{r_{min}}{4S_k r_{max}}, \quad (3.29)$$

then by induction, we have (3). ■

Corollary 1, 2 and 3 show that all ATs' long-term throughput is bounded away from 0, as a result, complete the proof. ■

Intuitively, if we view $1/T_{nk}(t)$ as the *degree of dissatisfaction* (the node is more unhappy with less mean throughput), then $\varphi_{k,t}(y)$ in (3.23) represents the *degree of dissatisfaction* in group k can be eliminated by the BS's service at time t when the transmission rate is y . As a result, the BS selects to serve the group $k(t)$ such that the maximum *degree of dissatisfaction* among all the groups is eliminated.

3.4.2 Multicast PF

Define $f_{k,h,i,j}^{\mathcal{S}^\diamond, \mathcal{S}}(p)$ in the same way as in (3.10) with \mathcal{S}^* replaced by \mathcal{S}^\diamond . The following will then be used to prove MPF:

Lemma 4 *For any k, h, i, j, p , we have*

$$\sum_{n=1}^{S_k} \frac{f_{k,h,i,j}^{\mathcal{S}^\diamond, \mathcal{S}}(p) R_i 1_{\{x_{n,k}^p \geq R_i\}}}{T_{n,k}^{\mathcal{S}^\diamond}} \geq \sum_{n=1}^{S_h} \frac{f_{k,h,i,j}^{\mathcal{S}^\diamond, \mathcal{S}}(p) R_j 1_{\{x_{n,h}^p \geq R_j\}}}{T_{n,h}^{\mathcal{S}^\diamond}}. \quad (3.30)$$

□

Proof: The proof follows the same line as in Lemma 2 by using (3.22) – (3.24) instead of (3.8) and (3.9).

The result becomes trivial when $f_{k,h,i,j}^{\mathcal{S}^\diamond, \mathcal{S}}(p) = 0$. So, we only need to consider the case of $f_{k,h,i,j}^{\mathcal{S}^\diamond, \mathcal{S}}(p) > 0$. In this case, there exists t_m ($1 \leq m \leq T_f$) such that

$$1_k^{\mathcal{S}^\diamond}(t_m) 1_h^{\mathcal{S}}(t_m) I_i^{\mathcal{S}^\diamond}(t_m) I_j^{\mathcal{S}}(t_m) 1_{\{r_{11}(t_m), \dots, r_{S_G G}(t_m)\} = X^p\}} = 1.$$

Hence, all terms on the LHS of the above equation are equal to 1, and thus $r_k^g(t_m) = R_i$ for all such m since $I_i^{\mathcal{S}^\diamond}(t_m) = 1$, which then leads to $\varphi_{k,t}(r_k^g(t_m)) = \varphi_{k,t}(R_i)$. Also, group k is selected by our scheduler \mathcal{S}^\diamond at every such t_m since $1_k^{\mathcal{S}^\diamond}(t_m) = 1$. Consequently, from (3.24), we have

$$\varphi_{h,t_m}(r_h^g(t_m)) \leq \varphi_{k,t_m}(r_k^g(t_m)), \quad (1 \leq m \leq T_f). \quad (3.31)$$

When $h = k$, from (3.22) and (3.23), the result follows by noting that

$$\frac{\text{LHS of (3.30)}}{f_{k,h,i,j}^{\mathcal{S}^\diamond, \mathcal{S}}(p)} = \varphi_{k,t_m}(r_k^g(t_m)) \geq \varphi_{k,t_m}(R_j) = \frac{\text{RHS of (3.30)}}{f_{k,h,i,j}^{\mathcal{S}^\diamond, \mathcal{S}}(p)}.$$

When $h \neq k$, we have

$$\frac{\text{LHS of (3.30)}}{f_{k,h,i,j}^{\mathcal{S},\mathcal{S}}(p)} = \varphi_{k,t_m}(r_k^g(t_m)) \geq \varphi_{h,t_m}(r_h^g(t_m)) \quad (3.32)$$

$$\geq \varphi_{h,t_m}(R_j) = \frac{\text{RHS of (3.30)}}{f_{k,h,i,j}^{\mathcal{S},\mathcal{S}}(p)}, \quad (3.33)$$

where the inequality in (3.32) is from (3.31) and (3.33) is from (3.22). This completes the proof. \blacksquare

In the following, we will show:

Proposition 2 *The multicast scheduler \mathcal{S}^\diamond described in (3.22)–(3.24) is MPF.* \square

Proof: Expand $T_{m,k}^{\mathcal{S}^\diamond}, T_{n,h}^{\mathcal{S}}$ in terms of their joint pmf with respect to the DRC vector X_p , i.e.,

$$\begin{aligned} T_{m,k}^{\mathcal{S}^\diamond} &= \sum_{h=1}^G \sum_{p \in \mathbb{P}, i, j \in \mathbb{D}} f_{k,h,i,j}^{\mathcal{S}^\diamond, \mathcal{S}}(p) R_i 1_{\{x_{m,k}^p \geq R_i\}} \\ T_{n,h}^{\mathcal{S}} &= \sum_{k=1}^G \sum_{p \in \mathbb{P}, i, j \in \mathbb{D}} f_{k,h,i,j}^{\mathcal{S}^\diamond, \mathcal{S}}(p) R_j 1_{\{x_{n,h}^p \geq R_j\}}, \end{aligned} \quad (3.34)$$

where $\mathbb{P} = \{1, \dots, P\}$ and $\mathbb{D} = \{1, \dots, D\}$.

Then the proof becomes similar to that of Proposition 1, by using (3.34) and Lemma 4.

For notational simplicity, we use $\sum_{p,i,j}$ instead of $\sum_{p \in \mathbb{P}} \sum_{i,j \in \mathbb{D}}$. Observe now that

$$\begin{aligned}
\sum_{h=1}^G \sum_{n=1}^{S_h} \frac{T_{n,h}^{\mathcal{S}}}{T_{n,h}^{S^\diamond}} &= \sum_{h=1}^G \sum_{n=1}^{S_h} \sum_{k=1}^G \sum_{p,i,j} \frac{f_{k,h,i,j}^{S^\diamond, \mathcal{S}}(p) R_j 1_{\{x_{n,h}^p \geq R_j\}}}{T_{n,h}^{S^\diamond}} \\
&= \sum_{h=1}^G \sum_{k=1}^G \sum_{p,i,j} \sum_{n=1}^{S_h} \frac{f_{k,h,i,j}^{S^\diamond, \mathcal{S}}(p) R_j 1_{\{x_{n,h}^p \geq R_j\}}}{T_{n,h}^{S^\diamond}} \\
&\leq \sum_{h=1}^G \sum_{k=1}^G \sum_{p,i,j} \sum_{n=1}^{S_k} \frac{f_{k,h,i,j}^{S^\diamond, \mathcal{S}}(p) R_i 1_{\{x_{n,k}^p \geq R_i\}}}{T_{n,k}^{S^\diamond}} \\
&= \sum_{h=1}^G \sum_{n=1}^{S_k} \sum_{k=1}^G \sum_{p,i,j} \frac{f_{k,h,i,j}^{S^\diamond, \mathcal{S}}(p) R_i 1_{\{x_{n,k}^p \geq R_i\}}}{T_{n,k}^{S^\diamond}} \\
&= \sum_{k=1}^G \sum_{n=1}^{S_k} \frac{T_{n,k}^{S^\diamond}}{T_{n,k}^{S^\diamond}} = \sum_{h=1}^G \sum_{n=1}^{S_h} \frac{T_{n,h}^{S^\diamond}}{T_{n,h}^{S^\diamond}}
\end{aligned}$$

where both the first and the fourth equalities are from (3.34) and the inequality follows from Lemma 4. This shows (3.21) and we are done. ■

3.4.3 General MPF Scheduler

In Section 3.3.3, we extend the notion of IPF to the general notion of (\vec{e}, α) IPF and develop the general (\vec{e}, α) IPF scheduler $\mathcal{S}^*(\alpha)$. Similarly, in this section, we extend the notion of MPF to the general notion of (\vec{e}, α) MPF.

Definition 4 A scheduler $\mathcal{S}^\diamond(\alpha)$ is (\vec{e}, α) MPF if, for any other scheduler \mathcal{S} ,

$$\sum_{h=1}^G \sum_{n=1}^{S_h} \frac{T_{n,h}^{\mathcal{S}} - T_{n,h}^{\mathcal{S}^\diamond(\alpha)}}{(T_{n,h}^{\mathcal{S}^\diamond(\alpha)})^\alpha} \leq 0, \tag{3.35}$$

where $\vec{e} = \{1, 1, \dots, 1\}$, $0 \leq \alpha \leq 1$.

We propose the following set of schedulers:

(\vec{e}, α) **MPF scheduler** $\mathcal{S}^\diamond(\alpha)$:

The feasible rate assigned to group k at time t is

$$r_k^g(t) = \arg \max_y \varphi_{k,t}^\alpha(y), \quad (3.36)$$

where

$$\varphi_{k,t}^\alpha(y) = \sum_{n=1}^{S_k} \frac{y}{(T_{nk}(t))^\alpha} 1_{\{y \leq r_{nk}(t)\}}. \quad (3.37)$$

The BS chooses group $k(t)$ to transmit at rate $r_{k(t)}^g(t)$, where

$$k(t) = \arg \max_{1 \leq k \leq G} \varphi_{k,t}(r_k^g(t)). \quad (3.38)$$

Note that when $\alpha = 0$, the scheduler becomes the *MAX throughput scheduler*; when $\alpha = 1$, it becomes the *MPF scheduler* \mathcal{S}^\diamond . Following the same line as in Proposition 2, we can show that $\mathcal{S}^\diamond(\alpha)$ is (\vec{e}, α) MPF.

3.5 Simulation

3.5.1 Simulation Setup

To evaluate the performance of our algorithms, we conduct packet level simulation using ns-2 [3]. We consider one base station serving multiple mobile users with random locations. We generate a DRC trace for each user as follows: at each time slot t , the DRC value for user i is predicted based on its position and a simulated channel fading process considering both slow fading and fast fading. Channel is modeled with slow fading as a function of the client's distance from the base station, and fast fading using Rayleigh fading. The combined effect is then mapped to a list of supported DRC values (in kbps) of $\{0, 38.4, 76.8, 153.6, 204.8, 307.2, 409.6, 614.4, 921.6, 1228.8, 1843.2, 2457.6\}$ according to CDMA2000 1xEV-DO specification. We assume the input buffer at the base station is constantly backlogged. We run the simulation for 30 seconds, and evaluate the performance of different algorithms listed in Table 3.1 with different group formation using the same DRC traces. In order to avoid transient effects, we discard results from the initial 3 seconds.

Table 3.1 shows the four scheduling algorithms we test in our simulation studies. Among them, IPF and MPF have been discussed in Sections 3.3 and 3.4 respectively. Recall that we provide IPF and MPF in multicast systems for tradeoff between throughput and fairness. The best way to test their performance is to compare them

Table 3.1: A summary of scheduling algorithms. The definitions for $\phi_{k,t}$, $\varphi_{k,t}$, and $T_k^g(t)$ are defined in (3.5), (3.23) and (3.4) respectively. n can be any number in $\{1, 2, \dots, S_k\}$.

Algorithm	Rate selection scheme	Group selection scheme
IPF	$r_k^g(t) = \arg \max_y \phi_{k,t}(y)$	$k(t) = \arg \max_{1 \leq k \leq G} \frac{\phi_{k,t}(r_k^g(t))}{T_k^g(t)}$
MPF	$r_k^g(t) = \arg \max_y \varphi_{k,t}(y)$	$k(t) = \arg \max_{1 \leq k \leq G} \varphi_{k,t}(r_k^g(t))$
MAX	$r_k^g(t) = \arg \max_y \phi_{k,t}(y)$	$k(t) = \arg \max_{1 \leq k \leq G} \phi_{k,t}(r_k^g(t))$
MIN	$r_k^g(t) = \min_n r_{nk}(t)$	$k(t) = \arg \max_{1 \leq k \leq G} \frac{r_k^g(t) S_k}{T_k^g(t)}$

with other algorithms that achieve the maximum throughput or optimal fairness. This is why MAX and MIN schedulers are provided. MAX aims at the maximum aggregate throughput, and MIN gives every user in the same group the absolutely fair share.

3.5.2 Objective Functions

As shown in Section 3.3 and 3.4, different algorithms maximize different objective functions. Given that T_i is user i 's throughput and T_k^g is group k 's throughput, MAX maximizes the aggregate throughput of all users, $\sum_i T_i$, MPF maximizes the sum of log utility function of individual user throughput, $\sum_i \log T_i$, which is a measure of total happiness of individual users, and IPF maximizes the sum of log utility function

of group throughput $\sum_k \log T_k^g$ and group throughput T_k^g is measured by the aggregate of user throughput in a group k . In the case of IPF, the objective is to achieve the highest group happiness rather than individual happiness.

We conduct two sets of experiments to verify that these algorithms do indeed maximize their respective objective functions given various group configurations and to see the performance difference among various algorithms in terms of the given objective functions. In the first set of experiments, we fix the number of users served by the base station to 32, and divide them into equal-sized groups. Each setup can be represented by a tuple (n, g) , where n is the number of groups and g is the group size. We conduct simulation runs for the following 6 values of (n, g) : $(1, 32)$, $(2, 16)$, $(4, 8)$, $(8, 4)$, $(16, 2)$, $(32, 1)$. When there is only one group, MAX and IPF behave the same way as noted earlier and when the number of group is 32, this case degenerates to a unicast scenario. As a result, IPF, MPF, and MIN show exactly same result as in the unicast case. This setup allows us to examine the performance of schedulers under various network scenarios.

Figures 3.4, 3.5, and 3.6 plot the values of the three objective functions for the above experiment. Figure 3.4 shows the value of $\sum \log T_i$. As expected, MPF shows the highest value as it is designed to maximize that objective function. When there is only one group, since MAX and IPF behave the same way, their values are the same. The objective value decreases as the number of groups increases. This is because as

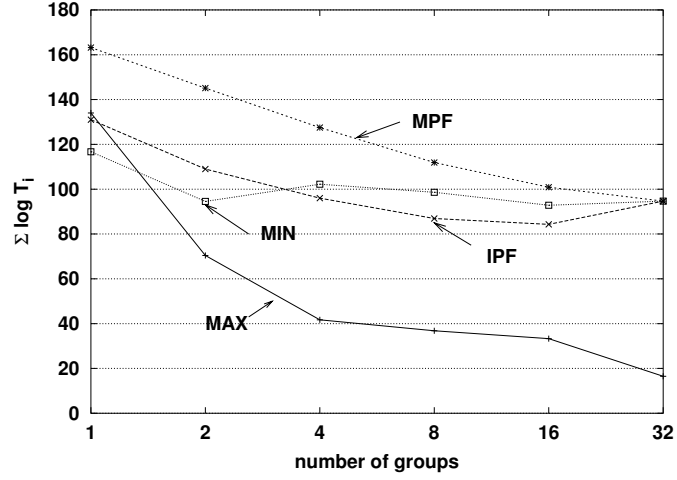


Figure 3.4: The value of $\sum \log T_i$ for various multicast schedulers as we increases the number of groups from 1 to 32 while fixing the total number of users in a cell to 32. T_i is the throughput of user i . MPF is optimized for this metric.

the number of groups increases while fixing the number of total users, the system degenerates into a unicast scenario. Thus, the benefit of multicast diminishes, so the total throughput decreases (as shown in Figure 3.6). Under the unicast case, we see that IPF, MPF, and MIN behave the same way. As IPF is designed to achieve fairness among groups, it does not maximize this objective value and thus, shows smaller values than MPF. MIN tends to perform fairly well in terms of fairness as the utility value is pretty high since all group members are receiving data all the time, albeit, at the minimum rate, but as we see in Figure 3.6, its total throughput is very low.

Figure 3.5 shows the values of $\sum \log T_k^g$. This objective function measures the total log utility of group throughput that is the sum of member throughput in each

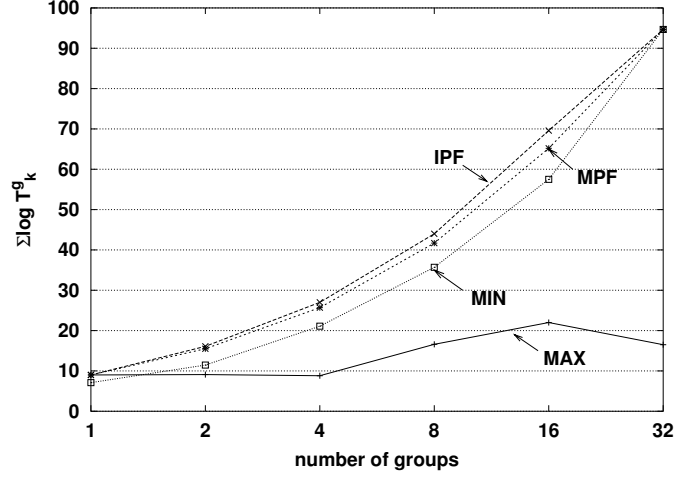


Figure 3.5: The value of $\sum \log T_k^g$ for various multicast schedulers as we increases the number of groups from 1 to 32 while fixing the total number of users in a cell to 32. T_k^g is the total throughput of group k . IPF is optimized for this metric.

group, and in some sense, it makes the optimal tradeoff between group throughput and fairness among groups. Since we are taking log on the group aggregate throughput, the actual magnitude of values is small and the difference between two values is also very small. But in actual values, we observe the difference is very large in Figure 3.6. According to this metric, we find that IPF performs the best. Under the unicast scenario, we can find that the values become the maximum. Moreover, note that $\sum \log T_k^g$ increases as the number of groups increase. To see this, consider this objective function, $\sum \log T_k^g = \log (T_1^g \times T_2^g \times \dots T_G^g)$ where G is the number of groups. On one hand, the increasing of G increases the number of factors. On the other hand, as G increases, the throughput of each group, T_k^g ($1 \leq k \leq G$), tends to decrease because of decrease of the number of users in each group. When $T_k^g \gg 1$

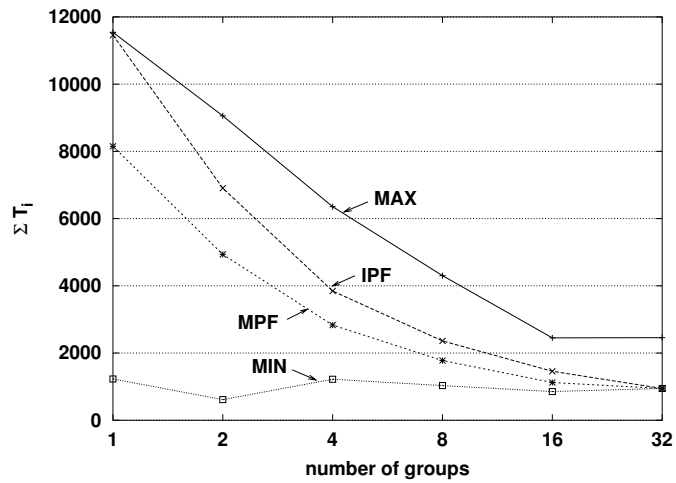


Figure 3.6: The value of $\sum T_i$ for various multicast schedulers as we increases the number of groups from 1 to 32 while fixing the total number of users in a cell to 32. T_i is the throughput of user i . MAX is optimized for this metric.

regardless the number of users in a group, which is the case of our situation, the effect of increasing number of factors will dominate and as a result the product increases, so as $\sum \log T_k^g$. This coincides with our simulation result.

Figure 3.6 shows the total sum of user throughput. In this metric, MAX performs the best as expected. We also find that MAX performs the worst in Figures 3.4 and 3.5. This indicates that MAX is a very greedy scheme that although it achieves high total throughput in the system, it is unfair among users (as seen in Figure 3.4) and among groups (as seen in Figure 3.5). We observe that IPF and MPF give pretty good throughput. This result, along with the other results above, indicates that MPF and IPF strike a good balance between throughput and fairness.

3.5.3 Distribution of Throughput

In the above experiment, we have seen the average performance of the system after the system converges. In this section, we are interested in the distribution of throughput among users and groups under various schedulers. This sheds more light into the tradeoff between fairness and throughput that each scheme makes. As mentioned before, MIN gives each user in the same group the equal share. However, recall that the default scheduler FIXED used in CDMA2000 1xEV-DO systems gives equal share even to users not in the same group with the rate of 204.8 kbps [13]. Hence, in this experiment, we also take FIXED into consideration.

We conduct the following experiment. We divide 100 users randomly placed in a cell into 10 groups, each with 10 users. Then we run various schedulers and plot the average throughput of each user measured for each three-second interval during the entire simulation run. Figure 3.7 shows the results of the experiment. We find that FIXED serves all users with a fixed rate, but penalizes both distant users and close users from the base station as throughput for distant users becomes zero while close users are subject to low throughput even if it can receive data at a higher rate. MIN maintains low throughput for all users, but no users will starve. MAX tends to favor only users close to the base station, so while close users have very high throughput, distant users receive nothing. IPF achieves higher throughput for close users than MPF because its rate selection algorithm always finds the rate that maximizes the

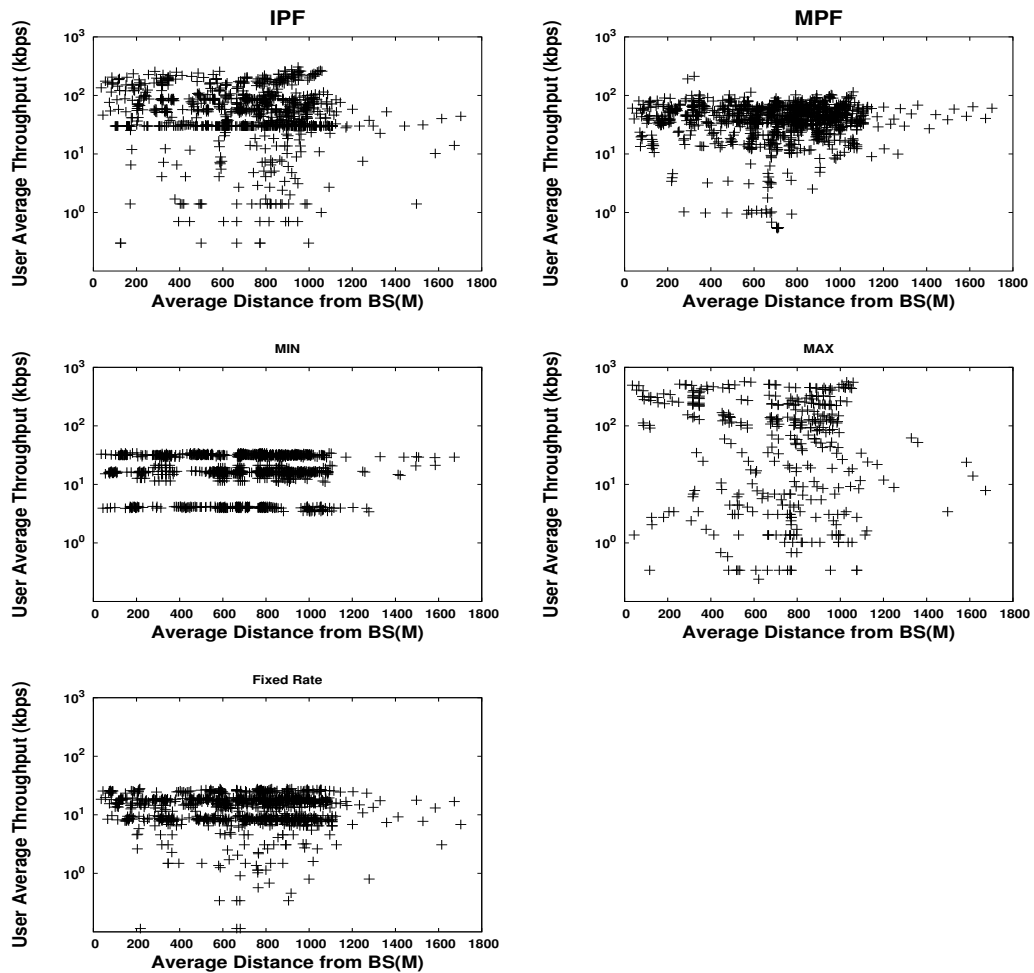


Figure 3.7: The distribution of user throughput taken at each three-second interval for various schedulers. Users are sorted over X-axis by their distance to the base station. The key point in these figures is that the distribution of user throughput tends to follow the characteristics of tradeoff that each scheduler makes regarding the throughput and fairness of users.

aggregate group rate. But IPF tends to have larger variance in throughput than MPF and have a lower minimum than MPF. This is because MPF tries to provide fairness to distant users with poor channel states by giving them more chances to communicate than IPF does. In order for MPF to give more chances to distant users, it has to take some time slots from close users, which makes its maximum throughput lower than IPF.

From the same experiment run, we now plot the group throughput measured by the aggregate user throughput of each group. We plot them for all 10 groups in Figure 3.8. In the figure, we can find that groups 1, 4, 5, 6 and 10 have members very close to the base station (so we call them *close groups*) while the other groups tend to have more distant members (we call these groups *distant groups*). In every experiment we conduct, we choose user locations completely randomly. MAX shows the biggest skew of throughput distribution among groups as we find that some groups do not get any throughput while the close groups tend to get all the bandwidth. FIXED gives almost a constant bit rate to all groups (with rate variance due to user locations) and MIN has more varying group rates because it tries to achieve fairness while honoring only the worst user in each group. The difference between IPF and MPF is clearly shown in this figure. IPF tries to maximize the group rate while keeping the minimum group rate high. Thus, it tends to achieve very good balance between group throughput and fairness. MPF does not care much about the group throughput

and focuses on equalizing user happiness. As the utility of users are equalized across groups, we tend to see less variance in the group rates, but it may not be able to achieve as high group throughput as IPF.

3.6 Summary

In this Chapter, we propose two sets of unified multicast proportional fair scheduling algorithms, the (\vec{e}, α) Inter-Group Proportional Fairness(IPF) schedulers and the (\vec{e}, α) Multicast Proportional Fairness(MPF) schedulers. Our algorithms take into account the reported data rate request from users and the average throughput of each user inside the cell and use this information to select an appropriate data rate for each group and a group among groups to transmit data. We prove that Multicast Proportional Fairness and Inter-Group Proportional Fairness algorithm are proportionally fair among all users and among groups inside a cell respectively. Each of these schedulers supports a different utility function. In particular, the IPF scheduler supports the utility function of log of aggregate group throughput that is computed by summing up all the user throughputs in a group, and the MPF scheduler supports the utility of log of individual user throughput. These multicast scheduling algorithms can be applied to different scenarios depending on the application and business model of ISPs. We compare the performance of these schedulers with several multicast scheduling algorithms and show through simulation that they achieve

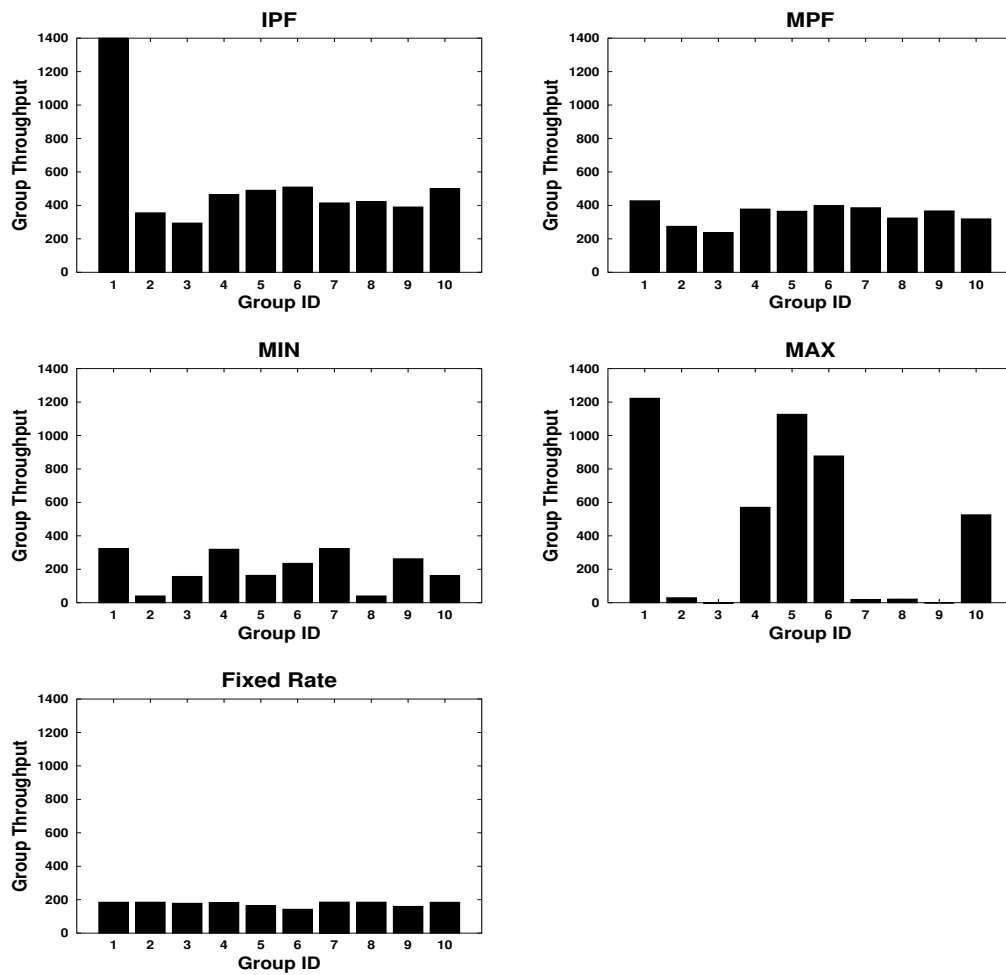


Figure 3.8: The distribution of group throughput at the end of simulation for various schedulers. The key point in these figures is that the distribution of group throughput tends to follow the characteristics of tradeoff that each scheduler makes regarding the throughput and fairness of groups.

good balance between fairness and throughput of groups or users.

Chapter 4

Multicast in DTNs

The Delay or Disruption Tolerant Networks (DTNs) are proposed to communicate in mobile wireless networks where delay may be very large and unpredictable: for example, terrestrial mobile networks, military ad-hoc networks, sensor and actuator networks [39]. Due to node's mobility, limited power, or limited bandwidth and environment factors, DTNs are characterized by very long delay on paths, infrequent contacts between nodes, and partitioned networks or no end-to-end paths most of the time. Most traditional wireless schemes are based on end-to-end path to transfer data. Therefore, the existing wireless routing schemes do not suit well to/for DTNs.

To transfer data in DTNs, various routing schemes have been studied [47]. Depending on the number of copies of a packet in a network, routing schemes in DTNs are categorized into single-copy and multi-copy schemes.

In single copy schemes, only one node including a source has the original copy of a packet; the node “*forwards*” the packets to an encountered node [83]. In multi-copy schemes, a node “*copies or replicates*” its packets to an encountered node [82]. So the number of copies of a packet in a network is more than or equal to one at any time.

The performance of single copy schemes is very poor due to the nature of DTNs such as infrequent contacts and long delay. It is well known that multi-copy schemes are more practical routing schemes in DTNs. However, the increased copies in multi-copy schemes may waste limited network resources too much such as nodes’ energy and storage. As a result, it causes performance degradation of multi-copy schemes. Consequently, most multi-copy schemes have tried to sustain the performance by reducing the number of (useless) copies as much as possible.

Multicast is an efficient way to transmit “same data” to multiple nodes. While a unicast packet has one destination, the destination of a multicast packet is a group that consists of more than one node. So we can expect significant savings in the number of copies, especially when the same data such as advertisements and public announcements should be spread to multiple locations. By reducing the duplicated copies, we can expect higher performance through multicast.

Most multi-copy routing schemes in DTNs work as follows. Assume that node A holds a packet and another node B does not have the packet. When A encounters B ,

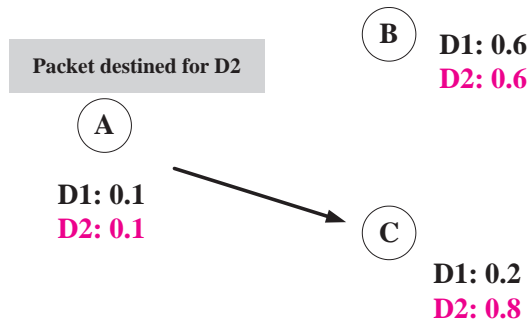


Figure 4.1: An Example of Unicast Routings in DTNs. Assume that each node has delivery probability for destinations D_1 and D_2 . Also assume that marginal delivery probability difference per packet is used to evaluate node's quality.

A evaluates the “*quality*” of node B for the destination of the packet. If the quality of B is better than that of A , A copies the packet to B . Therefore, the evaluation of a node's quality for the destinations of packets is very important in routing schemes. As mentioned above, a multicast packet has multiple destinations (group). Therefore, an efficient “*group quality*” measurement scheme should be considered for multicast in DTNs, in order to evaluate the qualities of nodes for multiple destinations.

Figure 4.1 shows an example of unicast routing schemes in DTNs. Let's assume that each node has delivery probability for each destination. As shown in the example above, node A , B and C have delivery probability for destinations D_1 and D_2 . Also assume that A has a packet destined for D_2 and B and C do not have the packet, and A meets B and C at the same time. To assess the quality of encountered nodes, routing protocols use their own routing metrics such as expected delivery time to the destination and delivery probability to the destination. Routing protocols may use

absolute or relative value of the metric values between encountered nodes in order to evaluate “node quality”.

In this example, we simply use the difference of delivery probabilities between two nodes as the metric. The difference of delivery probabilities between A and B for destination D_2 is $(0.6-0.1)=0.5$ and the difference of delivery probabilities between A and C for destination D_2 is $(0.8-0.1)=0.7$. The node qualities of B and C are 0.5 and 0.7 respectively. Node A chooses a node that has the highest node quality. Therefore, A chooses C to copy the packet destined for D_2 .

Figure 4.2 shows an example of multicast routing schemes in DTNs. We use same assumptions as in Figure 4.1, except that one multicast packet has two different destinations D_1 and D_2 . To evaluate node’s group quality for multicast packets, node A considers differences of delivery probabilities for each destination D_1 and D_2 . First, A considers difference of delivery probabilities for destinations D_1 and D_2 between A and B ; $(0.6-0.1)=0.5$ and $(0.6-0.1)=0.5$. The group quality of a node is sum of the difference values. So the group quality of B is 1.0. Second, A considers difference of delivery probabilities for destinations D_1 and D_2 of C ; $(0.2-0.1)=0.1$ and $(0.8-0.1)=0.7$. The group quality of C is 0.8. To copy the multicast packet, A chooses a node that has the highest group quality. B is selected. The “group quality” metric of a multicast packet is natural extension of “node quality” in unicast.

We presents a new efficient approximation algorithm for optimal unicast in DTNs,

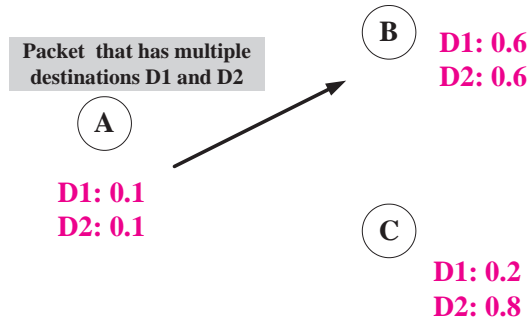


Figure 4.2: An Example of Multicast Routings in DTNs. Assume that each node has delivery probability for destinations D_1 and D_2 . Also assume that sum of marginal delivery probability per packet is used to evaluate node's group quality.

called Distributed Max-Contribution (DMC), which performs greedy scheduling, routing and replication based only on locally and contemporarily available information. We then extend DMC unicast to multicast in DTNs. The key to the extension is how to evaluate “group quality” of nodes for multiple destinations in multicast packets. Our “group quality” metric is defined by the sum of qualities for each destination of multicast packets as shown in the example of Figure 4.2.

Our simulation studies are based on detailed GPS (Global Positioning System) traces of tracking the movements of over 4000 taxis, each equipped with GPS in Shanghai [87] for about 30 days, by far the largest traces of vehicle-based networks. Through simulations, we demonstrate that DMC multicast greatly outperforms existing unicast algorithms for DTNs.

To make DTNs more practical, many applications should also be practical on DTNs. However, making applications practical on DTNs has been a challenging

issue until now. To the best of our knowledge, this is the first thesis considering “group quality” of nodes for multiple destinations, and applying multicast to data dissemination applications in DTNs.

4.1 Optimal Resource Allocation in DTNs

We consider a system where a source sends the same data to multiple destinations (group). We assume that a source and destination nodes are fixed and the other nodes move. Fixed source and destinations are regarded as mobile nodes that have a long pause during simulations. Therefore mobile nodes are responsible for relaying packets from a source and destinations. Recall that the destination of multicast packets is a group consisting of multiple destinations.

4.1.1 Notations

Throughout the thesis, we use following notations.

- N : a set of nodes
- f : a flow
- F : a set of flows
- $S(t)$: feasible link schedule at time t

- L : a set of links
- l : link
- g : packets
- $N_f(t_1, t_2)$: the number of delivered packets in flow f to its destination over an interval $[t_1, t_2]$
- $\pi(t)$: feasible copy schedule at time t . it represents a packet that can be copied when link l is active at time t
- $v_{m,d}$: the node quality for a packet m destined for destination d
- $\Delta v_{m,d}^l$: the quality of a packet m destined for destination d over a link l .

4.1.2 System Model

Network and traffic model.

We consider a network consisting of a set \mathcal{N} of n nodes that move and meet intermittently. Two nodes v and w are said to *meet* if v is within the communication range of w , and vice versa. Every node is equipped with an infinite-size queue to store packets. A node v can copy packets from its queue to the node that v meets¹. There is a set \mathcal{F} of f sessions (flows) that are identified by a pair of source-destination nodes.

¹We also use the word ‘packet’ to refer to the copies of the original packet, unless explicitly specified otherwise.

Associated with each session f , a file consists of a set \mathcal{G}_f of equal-sized packets. We use the *packet-company* m to refer to the original packet m and its copies together. The source of a session f is responsible for transferring the packets in \mathcal{G}_f to its destination with some QoS constraints.

Resource model.

Time is assumed to be slotted, indexed by $t = 0, 1, \dots$. The length of a time-slot is suitably chosen to schedule one packet and nodes are stationary. Then, network resources are represented by a finite set $\mathcal{S}(t) \subset \{0, 1\}^L$ of feasible link schedules, where L is the number of all possible links. A *feasible link schedule*, $S = (S_l \in \{0, 1\} : l = 1, \dots, L)$ is a vector representing a set of schedulable links without interference where $S_l = 1$ if the link l is scheduled, and 0 otherwise. We also use notation $l \in S$ when $S_l = 1$. Denote by $\Pi(t) \subset \mathcal{G}^L$, a set of feasible copy schedules where $\mathcal{G} = \cup_{f \in \mathcal{F}} \mathcal{G}_f$. A *feasible copy schedule* is a vector whose l -th element represents a packet that can be potentially copied if link l is scheduled. Note that a packet m can be copied from $v \in \mathcal{N}$ to $w \in \mathcal{N}$ when v holds m but w does not hold packet m . Note that in a feasible copy schedule, two different packets belonging to a single packet-company can be scheduled over different links.

Interference and resource allocation.

A set $\mathcal{S}(t)$ depends on interference patterns among links. We generally model interference by a $L \times L$ symmetric matrix $I = [I_{ij}]$, where $I_{ij} = 1$ means that links i

and j interfere with each other. The matrix I is able to model various wireless systems, ranging from FH-CDMA (one-hop interference) to 802.11 (two-hop interference²). For ease of presentation, we assume that when a link is established when two nodes meet, the link is configured to have a unit capacity, but it can be readily extended to more general cases. Resource allocation at each slot t consists of two parts: (i) *link scheduling* and (ii) *copy scheduling* where a copy schedule $\pi \in \Pi(t)$ and a link schedule $S \in \mathcal{S}(t)$ are selected. Then, the element-wise multiplication of two vectors, $\pi \times S$, represents which packets are served and copied over the links.

Objectives.

The primary objectives of resource allocation is delivery ratio maximization or delay minimization. Denoted by the random variable, $N_f(t_1, t_2)$, the total aggregate number of delivered packets in flow f to its destination over an interval $[t_1, t_2]$, where t_2 is a given deadline (we henceforth omit t_2 and just use $N_f(t)$ in all notations unless confusion arises). When t_2 is infinity, the objective function is the same as optimal maximization of total delivered packets. Similarly, we also denote by $D_f(t)$ the total aggregate remaining time in flow f from t to the delivery.

The following four objectives are meaningful to the realistic systems and considered in existing studies.

²In the K-hop interference model, two links that are within K -hops interfere with each other.

$$\begin{aligned}
\mathbf{R1. Max-Delivery} & \quad \max \sum_{f \in \mathcal{F}} \mathbb{E}[N_f(t)] \\
\mathbf{R2. Fair-Delivery} & \quad \max \min_{f \in \mathcal{F}} \mathbb{E}[N_f(t)] \\
\mathbf{R3. Min-Delay} & \quad \min \sum_{f \in \mathcal{F}} \mathbb{E}[D_f(t)] \\
\mathbf{R4. Fair-Delay} & \quad \min \max_{f \in \mathcal{F}} \mathbb{E}[D_f(t)].
\end{aligned}$$

4.1.3 Snapshot Optimality

Hardness of full optimality. Solving the optimization problems in **R1**, **R2**, **D1**, and **D2** via practical, on-line, decentralized algorithms is hard. It can be formulated by a dynamic programming (DP), often requiring a large dimensional search (i.e., curse of dimensionality) and knowledge of the future. There are studies that use DP to develop optimal solutions. However, they have been done in much simpler models and less complex assumptions, e.g., a model without consideration of link scheduling [17, 62]. Our main interest lies in practical, on-line algorithms. To that end, rather than pursuing the “full”-optimality based on DP, we adopt a *temporal approximation* where implementable algorithms may be temporally restricted in terms of available information. In other words, we only look at system-states available contemporarily and try to optimize a certain objective naturally interpreted as a *snapshot-optimal approximation* to the original problem. It is possible to do so simply by temporally stretching the original optimization problems over the entire slots, and look at what needs to be optimized just using the information available at time t .

Objective functions. We now elaborate the snapshot-optimal problems for various objectives introduced in the subsection 4.1.2.

(a) Max-delivery. We stretch the objective function over the entire time-interval $[0, t_{\text{dl}}]$. Then we have

$$\max_{f \in \mathcal{F}} \sum \mathbb{E}[N_f(t)] = \max_{f \in \mathcal{F}} \mathbb{E} \left[\sum [N_f(0) + \sum_{i=0}^{t-1} \Delta N_f(i)] \right], \quad (4.1)$$

where $\Delta N_f(t) = N_f(t+1) - N_f(t)$ is a marginal increase of N_f over a time interval $[t, t+1]$. From (4.1), what we can do, given the available information at slot t , is to maximize $\mathbb{E}[\sum_f \Delta N_f(t)]$ i.e., maximize the average increase in the total number of delivered packets over $[t, t+1]$ across all sessions.

(b) Fair-delivery. Similarly to the above, we get

$$\max_{f \in \mathcal{F}} \min_{f \in \mathcal{F}} \mathbb{E}[N_f(t)] = \max_{f \in \mathcal{F}} \min_{f \in \mathcal{F}} \left(\mathbb{E}[N_f(0)] + \sum_{i=0}^{t-1} \mathbb{E}[\Delta N_f(i)] \right). \quad (4.2)$$

In contrast to max-delivery, we give higher priority to the flows with the less average number of delivered packets. Again, since only $(\mathbb{E}[N_f(t)], f \in \mathcal{F})$ is available to resource allocation at slot t , we first choose a session f^* such that $f^* = f^*(t) = \arg \min_{f \in \mathcal{F}} \mathbb{E}[N_f(t)]$, and allocate resource to maximize $\Delta N_{f^*}(t)$.

(c) Min-delay. The structure of minimizing delay is similar to maximizing that of the delivery ratio. Similarly to $\Delta N_f(t)$, we define $\Delta D_f(t) \triangleq D_f(t) - D_f(t+1)$ to be a marginal decrease in delay of flow f over interval $[t, t+1]$. Note that this delay

decrease is possible by copying the packet in question to other nodes.

$$\begin{aligned} \min \sum_{f \in \mathcal{F}} \mathbb{E}[D_f(t)] &= \min \mathbb{E} \left[\sum_{f \in \mathcal{F}} \left(D_f(0) - \sum_{i=0}^{t-1} \Delta D_f(i) \right) \right] \\ &= \min \left(\mathbb{E} \left[\sum_{f \in \mathcal{F}} D_f(0) \right] - \mathbb{E} \left[\sum_{f \in \mathcal{F}} \sum_{i=0}^{t-1} \Delta D_f(i) \right] \right). \end{aligned} \quad (4.3)$$

At slot t , the first step to approximate the above using the snapshot information, is to maximize $\Delta D_f(t)$. Recall that $\Delta D_f(t)$ is random in terms of random mobility. It means that the maximization of $\Delta D_f(t)$ is feasible (in the sample-path sense) only if the full information about mobility (even including future) were given to nodes, which is impossible due to limited knowledge of mobility in the future. Thus, an alternative approach is to take the expectation of $\Delta D_f(t)$, i.e., $\mathbb{E}[\Delta D_f(t)]$, which we maximize at the snapshot. Thus, our snapshot optimization problem is $\max \sum_{f \in \mathcal{F}} \mathbb{E}[\Delta D_f(t)]$.

(d) Fair-delay. Similarly to fair-delivery, we have:

$$\min \max_{f \in \mathcal{F}} \mathbb{E}[D_f(t)] = \min \max_{f \in \mathcal{F}} \mathbb{E} \left[\left(D_f(0) - \sum_{i=0}^{t-1} \Delta D_f(i) \right) \right]. \quad (4.4)$$

However, the issues of approximating sample-paths with the expectation exist, which we handle similarly to min-delay. Thus, our snapshot objective is to maximize $\Delta D_{f^*}(t)$ where $f^* \triangleq f^*(t) = \arg \max_{f \in \mathcal{F}} \mathbb{E}[D_f(t)]$.

We are interested in getting maximum throughput, so we focus on objective **R1**.

4.2 Snapshot Optimality:

Hardness and Max-Contribution

In this thesis, we take a systematic multi-step approach towards practical and distributed algorithms. First, we develop an algorithm, called S-OPT, that is a snapshot optimal. We show that S-OPT requires centralized, intractable computations and the contemporary global knowledge of network state. Next, we develop a centralized approximation algorithm to S-OPT, called *Max-Contribution* (MC). MC provides an insight to the development of a distributed approximation algorithm to S-OPT, called *Distributed Max-Contribution* (DMC). DMC algorithm is presented in Section 4.3.

4.2.1 Value and Contribution

We first introduce a notion of *value*. Value $v_{m,d}$ is associated with each packet-company m destined for d . In unicast, packets have one destination, so we can write $v_{m,d} = v_m$. A packet value quantifies a per-packet metric defined according to the target objective. For a given objective, the value of a packet-company at a time slot is time-varying over slots and it depends on the mobility patterns of the nodes holding the copies of the packet at that time slot. For max-delivery (**R1**) objective, the value of a packet-company m is defined as the delivery probability of any packet

in m to be delivered to its destination. Since all packets in the same company share the same value, we interchangeably use the value of a packet and the value of a packet company that the packet belongs to. As a measure of the improvement in the value incurred by packet forwarding or replication, we introduce the notion of *contribution* of a packet of m , Δv_m to be the increased amount of the value v_m when new copies of m are forwarded or copied to the network. Note that when multiple packets in the packet-company are copied at the same time in the network, the contribution is the sum of all the contributions that each copy makes.

4.2.2 S-OPT: A Snapshot Optimal Algorithm

We now describe the generic algorithm, S-OPT, that is snapshot-optimal for the four objectives, when value v_m is properly defined. The key idea of S-OPT is to make link/copy scheduling decisions (over slots) that maximize the expectation of the total increase in the packet *values* over the entire network.

S(napshot)-OPT

At each time slot t , copy packets according to (π^*, S^*) , which is the optimal solution of

$$\max_{\pi \in \Pi(t), S \in \mathcal{S}(t)} \sum_{m \in \mathcal{G}(\pi, S)} \Delta v_m(t) \quad (4.5)$$

, where $\mathcal{G}(\pi, S)$ is the set of all packet-companies scheduled by a pair of copy and link schedule (π, S) .

Note that $\mathcal{G}(\pi, S)$ is a set. Thus, even in the case when the packets in the same company m are scheduled over different links, only the company index m is in $\mathcal{G}(\pi, S)$. As an example, we now explain that S-OPT with $v_m = p_m$ is snapshot-optimal for the max-delivery objective, **R1**, where p_m is the probability that at least one packet in the packet-company m is delivered to the destination. Recall that the snapshot objective for **R1** is to $\max_{\pi, S} \sum_f \mathbb{E}[\Delta N_f(t)]$.

Example 4.2.1 (R1. Max-Delivery) *First, denote by $I_m(t)$ is an indicator random variable recording whether at least one packet in company m is delivered over $[t, t_{dl}]$ or not. Let $\Delta p_m(t) = p_m(t) - p_m(t-1)$. Then, remarking that $N_f(t) = \sum_{m \in \mathcal{G}_f} I_m(t)$, we get*

$$\begin{aligned}
& \max_{\pi, S} \sum_f \mathbb{E}[\Delta N_f(t)] \\
&= \max_{\pi, S} \sum_f \mathbb{E} \left[\sum_{m \in \mathcal{G}_f} I_m(t) \right] \\
&= \max_{\pi, S} \sum_{m \in \cup_f \mathcal{G}_f} p_m(t) \\
&= \max_{\pi, S} \sum_{m \in \cup_f \mathcal{G}_f} \left(p_m(t) - p_m(t-1) + p_m(t-1) \right) \\
&= \max_{\pi, S} \left(\sum_{m \in \mathcal{G}(\pi, S)} \Delta p_m(t) + \sum_{m \in \cup_f \mathcal{G}_f \setminus \mathcal{G}(\pi, S)} \Delta p_m(t) \right) + \sum_{m \in \cup_f \mathcal{G}_f} p_m(t-1) \quad (4.6)
\end{aligned}$$

$$= \max_{\pi \in \Pi(t), S \in \mathcal{S}(t)} \sum_{m \in \mathcal{G}(\pi, S)} \Delta p_m(t) + K_1(t) + K_2(t), \quad (4.7)$$

where in (4.6) we divide the packet-companies into ones that are scheduled and not by (π, S) . $K_1(t)$ and $K_2(t)$ correspond to the second and third term in 4.6. For a fixed t , $K_1(t)$ is a constant as the packet-companies that are not scheduled do not depend on (π, S) . $K_2(t)$ is also a constant at time $t - 1$. Finally, from $\Delta v_m(t) = \Delta p_m(t)$ by definition, the result follows.

The S-OPT algorithm is impractical for the following reasons:

1) *Coupling between copy and link scheduling.*

v_m jointly depends on both copy and link schedules. For **R1**, when two different packets in the *same* packet-company m are scheduled over different links, the contribution of m should jointly consider the two copies because its delivery probability p_m is determined by any copy in m .

2) *Global knowledge of qualities.*

All nodes holding any packet in a packet-company m should have the same value v_m , which is hard to achieve in a realistic distributed environment. A vanilla method is to flood the value change event, requiring heavy message passing, thereby wasting network resources.

3) *Computational intractability.*

The S-OPT algorithm requires the exhaustive search to find a solution in the large-scale search space. Formally, the problem can generally be formulated by an integer programming with an exponential size of search space. In fact, for a fixed

π , the inner maximization of Eq. (4.5) over all feasible schedules is a variant of an NP-hard wireless scheduling problem (see [96] for details) that can be reduced to the NP-hard WMIS (Weight Maximum Independent Set) problem.

4.2.3 Link/Copy Scheduling Decomposition: Max-Contribution

A complex coupling between copy and link scheduling happens when multiple copies of the same packet are scheduled over different links simultaneously. In our approximation, *Max-Contribution*, S-OPT is solved with the set $\Pi'(t)$ of copy schedules, where

$$\Pi'(t) = \{\pi \in \Pi(t) \mid \pi_i \neq \pi_j, \forall i, j\}.$$

Since $\Pi'(t) \subset \Pi(t)$ for all t , it is clear that the contribution computed from S-OPT is no less than that from MC. We transform the original optimization problem into one over a *reduced* constraint set. Then, as we have discussed, the optimal algorithm becomes much simpler, which we in turn use to develop practical, on-line, distributed algorithms later in Section 4.3.

From the use of $\Pi'(t)$ instead of $\Pi(t)$, the contributions do not depend on the entire schedule, but only on the corresponding link l (more precisely, its receiver node, $rx(l)$), because only node, say v , changes the contribution of a packet that it holds. This approximation enables us to decompose copy scheduling from link scheduling, and first solve the outer-maximization by, for each link l , selecting the

packet-company m_l^* that has the maximum contribution. For clarity, we now use a notation Δv_m^l to refer to the contribution of a packet in packet company m when it is copied over link l .

Note that $|\Pi'(t)|$ gets closer to $|\Pi(t)|$ as $|\cup_f \mathcal{G}_f|/|(\mathcal{S}(t))|$ gets larger. Thus, MC is near-optimal when the offered load in the network is high compared to the number of schedulable links.

Max-Contribution (MC)

At each time slot t ,

Step 1. *Contribution computation.*

Each node computes the contributions of the packets (or copies) in its buffer over its connected links.

Step 2. *Copy scheduling.*

On each link $l \in \mathcal{S}(t)$, set the weight $W_l(t)$ of the link l to be $\max_m \Delta v_m^l(t)$,

and let

$$m_l^* = \arg \max_m \Delta v_m^l(t)$$

Step 3. *Link scheduling.*

Select the schedule $S^*(t)$ that satisfies

$$S^*(t) = \arg \max_{S \in \mathcal{S}(t)} \sum_{l \in S} W_l(t), \quad (4.8)$$

Step 4. *Packet copying.*

Replicate the packet (or the copy) m_l^* over the link l , for all $l \in S^*(t)$.

Unfortunately, Max-Contribution is still expensive to implement even with decoupling between link and copy scheduling. The need to have global knowledge of v_m remains, and the link scheduling problem maximizing the sum weights of links is NP-hard, which, again, can be reduced to the WMIS problem³.

4.3 Distributed Max-Contribution

In the previous Section, MC algorithm is presented to achieve snapshot optimality. MC algorithm requires contemporary network-wide global knowledge; all nodes holding a copy of a packet company m have to have the same value of v_m . To maintain concurrent information, additional control channel or flooding is proposed. However, the overhead becomes significant as the number of packets or the number of nodes in a network increases. To mitigate the main problem of MC algorithm, we present Distributed Max Contribution (DMC) algorithms that approximate MC algorithm

³Under one-hop interference model, the link scheduling problem is reduced to Weighted Maximum Matching (WMM) whose complexity is $O(L^3)$.

and use only local information. Then we extend DMC-unicast to DMC-multicast by introducing “group quality” metric.

4.3.1 Distributed Max-Contribution (DMC)- Unicast

Copy scheduling

DMC approximates MC algorithm through a technique called *fusion* which is used to maintain the set of nodes that currently own a copy of a packet m . Each node i keeps track of a set of other nodes, $\mathcal{N}_{m,i}$, that have a copy of each packet m that node i currently holds. $\mathcal{N}_{m,i}$ is called a *node set* of i for m . Along with a node set for m , node i maintains the delivery probability of each member in the set. It is initially empty and adds another node j when node i replicates a copy of m to j . After the replication happens, node j sets $\mathcal{N}_{m,j} = \mathcal{N}_{m,i}$. When node i meets a node k with the same copy m , then nodes i and k synchronize their node sets for m by taking union of $\mathcal{N}_{m,i}$ and $\mathcal{N}_{m,k}$. Whenever $\mathcal{N}_{m,i}$ is updated either by replicating the copy or by meeting another node with the same copy, node i recomputes v_m . If the global performance objective is **R1**, v_m is equal to the probability, p_m that any node holding any copy of m meets the destination of m and delivers m . v_m is recomputed in the following manner. Denote the value of packet m at node i by $v_{m,i}$ and the delivery probability (i.e., meeting probability) of i with the destination of m by $q_{m,i}$. Then

$$v_{m,i}(t) = p_{m,i}(t) = 1 - \prod_{k \in \mathcal{N}_{m,i}} (1 - q_{m,k}(t)). \quad (4.9)$$

For making a copy schedule at time t , DMC performs the following operations. When a node i with a packet m meets other nodes, they first exchange the IDs of packets whose copy they currently hold and then perform fusion by synchronizing their node sets and corresponding value information (i.e., delivery probabilities) and recomputing packet values. After this process, a node performs copy scheduling. For each packet m , node i computes the marginal increase of packet value of m when i is copied to each neighbor j . If j is already holding m , then the marginal increase is zero. If it is not, then the marginal increase is the difference between the current value of m and the new value of m if m is copied to j (i.e., recomputed value after adding j to $\mathcal{N}_{m,i}$). Node i picks the packet with the biggest marginal value increase for scheduling. Denote such a packet by $m_{i,j}^*$ where m is scheduled for copy for a link between nodes i and j . We call $m_{i,j}^*(t)$ the *candidate copy* of node i at time t .

Link scheduling

The scheduling algorithm that solves Eq. (4.8), referred to as Max-Weight scheduling, has been extensively studied to provide provable throughput guarantee. Recent efforts on distributed scheduling can provide us with an array of candidate, low-cost algorithms to Max-Weight. Examples include greedy, locally-greedy, random pick-and-compare algorithms (see [96] the references therein for the detailed algorithm

description). Such algorithms provide (partial) throughput performance guarantee, where throughput is defined by the achieved stability region. We can also adopt one of them in our framework as a distributed heuristic.

For our simulation, we use a locally-greedy algorithm which schedules, at each time t , the transmission of a packet whose marginal value increase is the biggest among all candidate copies of nodes that are in an interference region at time t .

4.3.2 DMC- Multicast

In this Section, we extend DMC-unicast to DMC multicast. To explain DMC-multicast algorithm, we first extend the parameters in notations that are used in DMC-unicast. The assumption that each node i maintains delivery probability for each destination d is also applicable to unicast. In DMC-unicast, $v_{m,i}$ represents the probability of any packet of packet company m meets the destination and $q_{m,i}$ is the delivery probability of node i for the destination of m . In multicast, the destination of multicast packets is a group that consists of one or more destinations. To consider multiple destinations in notations, destination parameters are added to existing notations. Denote the delivery probability of node i for each destination d_j of m by $q_{m,d_j,i}$. In case of unicast, each m has only one destination d_j . So d_j has no meaning in unicast. $v_{m,i}$ also considers destination parameters, $v_{m,d_j,i}$. The value is

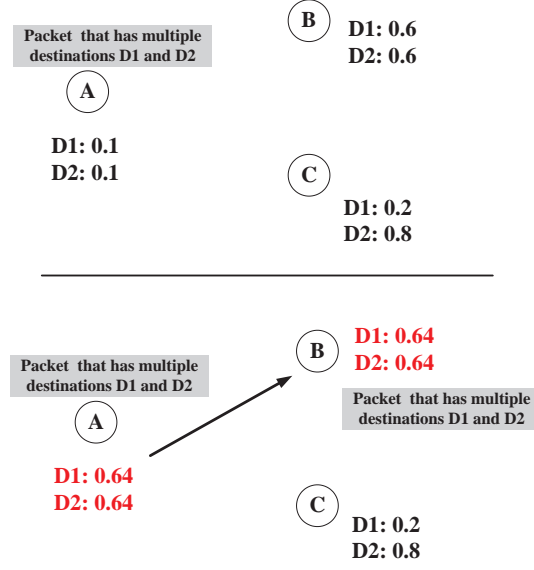


Figure 4.3: An Example of DMC-Multicast in DTNs. Assume that each node has delivery probability for destinations D_1 and D_2 . Also assume that sum of marginal delivery probability per packet is used to evaluate node's group quality.

computed for each destination d_j of a multicast packet m as follows.

$$v_{m,d_j,i}(t) = p_{m,d_j,i}(t) = 1 - \prod_{k \in \mathcal{N}_{m,i}} (1 - q_{m,d_j,k}(t)). \quad (4.10)$$

Then, the marginal delivery probability of packet company m over the link l , $\Delta v_m^l(t)$, is defined by sum of marginal delivery probabilities for all destinations of packet company m as follows:

$$\Delta v_{m,i}(t) = \sum_{d_j \in G} v_{m,d_j,i}(t) - p_{m,d_j,i}(t). \quad (4.11)$$

, where G represents a group, which is the destinations of packet company m

Figure 4.3 shows the example of DMC-multicast to select representative group quality. Each node maintains delivery probability for each destination D_1 , D_2 . Also

the multicast packet has two destinations D_1 and D_2 . Node A holds the packet and B and C don't have the packet. When node A encounters nodes B and C , marginal delivery probability, $\Delta v_{m,d}$, comparison between nodes happens as in unicast but to the number of destinations for the multicast packet m . Following presented notations, $p_{m,D_1,A} = q_{m,D_1,A} = 0.1$, $p_{m,D_2,A} = q_{m,D_2,A} = 0.1$, $q_{m,D_1,B} = 0.6$, $q_{m,D_2,B} = 0.6$, $q_{m,D_1,C} = 0.2$, and $q_{m,D_2,C} = 0.8$.

Calculate marginal probability of packet company m for nodes B and C respectively:

$$v_{m,D_1,A} = 1 - (1 - q_{m,D_1,A})(1 - q_{m,D_1,B}) = 1 - (1 - 0.6) * (1 - 0.1) = 1 - 0.36 = 0.64.$$

$$v_{m,D_2,A} = 1 - (1 - q_{m,D_2,A})(1 - q_{m,D_2,B}) = 1 - (1 - 0.6) * (1 - 0.1) = 1 - 0.36 = 0.64.$$

$$\sum \Delta v_{m,A} = (v_{m,D_1,A} - p_{m,D_1,A}) + (v_{m,D_2,A} - p_{m,D_1,A}) = 1.08$$

$$v_{m,D_1,A} = 1 - (1 - q_{m,D_1,A})(1 - q_{m,D_1,C}) = 1 - (1 - 0.2) * (1 - 0.1) = 1 - 0.72 = 0.28.$$

$$v_{m,D_2,A} = 1 - (1 - q_{m,D_2,A})(1 - q_{m,D_1,C}) = 1 - (1 - 0.8) * (1 - 0.1) = 1 - 0.18 = 0.82.$$

$$\sum \Delta v_{m,A} = (v_{m,D_1,A} - p_{m,D_1,A}) + (v_{m,D_2,A} - p_{m,D_1,A}) = 0.90$$

Among $\{1.08, 0.90\}$, the link with the biggest sum of marginal probabilities, 1.08 for node B is selected. The values for $v_{m,D_1,A}$ and $v_{m,D_1,B}$, and $v_{m,D_2,A}$ and $v_{m,D_2,B}$, are updated as 0.64 and 0.64 respectively following Eq. 4.11.

Algorithm 1 shows the DMC-multicast algorithm. DMC-multicast is different from DMC-unicast in that it considers a group that is the destination of the multicast packets.

The “quality” of a node for a packet is defined by the delivery probability of the packet to the destination through the node in the thesis. The “node quality” refers to the quality of a node for unicast packets and “group quality” refers to the quality of a node for multicast packets. In DMC-multicast, each destination in a group is considered as independent unicast packet. If a group consists of only one destination, DMC-multicast degenerates into DMC-unicast.

Algorithm 1 DMC-MULTICAST

At each time slot t ,

Step 1. *Contribution computation.*

Each node computes the contributions of the multicast packets (or copies) in its buffer over its connected links. When computing the contributions of the packets, all destinations in a group that is the destination of the multicast packet are considered independently.

Step 2. *Copy scheduling.*

On each link $l \in \mathcal{S}(t)$, set the weight $W_l(t)$ on the link l to be the $\max_m \sum_{d \in G} \Delta v_{m,d}^l(t)$, and let:

$$m_l^* = \arg \max_m \sum_{d \in G} \Delta v_{m,d}^l(t)$$

, where G represents a group that is a set of destinations of a multicast packet company m .

Step 3. *Link scheduling.*

Select the schedule $S^*(t)$ that satisfies

$$S^*(t) = \arg \max_{S \in \mathcal{S}(t)} \sum_{l \in S} W_l(t), \quad (4.12)$$

Step 4. *Packet copying.*

Replicate the packet (or the copy) m_l^* over the link l , for all $l \in S^*(t)$.

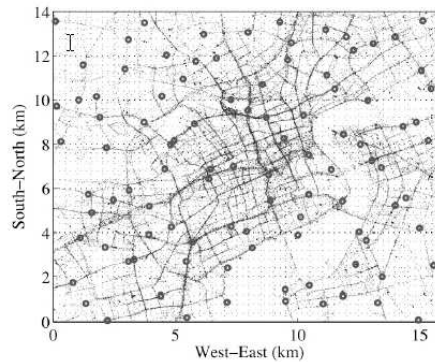


Figure 4.4: Snapshot distribution of taxis in Shanghai. Each dot indicates the location of a taxi during one hour at 11/28/2006. Circles indicate the candidate locations of sources and destinations.

4.4 Performance Evaluation

4.4.1 Node Delivery Probability from Shanghai Trace

For performance evaluation, we use GPS traces of over 4000 Shanghai taxis [87], by far, the largest vehicular GPS traces publicly available. The location information of each taxi is recorded at every 40 seconds within a city-wide area for 28 days (4 weeks). We consider a DTN application where many infostations are randomly scattered around the city in a uniform manner. Data is moved from one infostation (i.e., source) to another infostation (i.e., destination) by using a mobile wireless network of taxis equipped with WiFi. The infostations do not have access to infrastructure and they simply upload data in units of packets to passing-by taxis. These infostations are like public bulletin boards or street advertisement boards. Daily updates from one location are delivered to a set of destination infostations for display or announcement.

We consider unicast scenarios first, and show that DMC-unicast is near optimal in the performance through the simulations. We then proceed to show that DMC-multicast outperforms DMC-unicast in multicast scenarios.

People do not move randomly. Any mobile wireless networks whose constituent members are humans or vehicles driven by them cannot be described as random movement and there exists some regularity or periodicity in their mobility [56, 60]. Examination of taxi traces also points to some regularity in (1) in the patterns of locations each taxi visits daily and (2) in the patterns of meetings among taxis.

These regularities are essential in extracting information required to run DMC. To illustrate this, we plot the CCDF (complementary CDF) of the inter-contact times (ICT) and inter-visit times (IVT) of taxis in the traces as shown in Figure 4.5. The distributions are best fitted with exponential distributions. This is quite different from the human mobility pattern which shows power-law inter-contact time distributions. Figure 4.6 plots the individual intensity values (λ^{IVT} and λ^{ICT}) of IVT and ICT exponential distributions from 100 taxis. IVT is plotted for 100 destination locations. From the plots, we find that different taxis show different biases in the locations they visit and in the set of taxis they meet daily.

These characteristics of the Shanghai taxi network allow us to extract routing metadata. In particular, from the exponential distributions we fitted to each individual taxi's IVT and ICT, we can derive the *node delivery probability*, $q_{m,i}$ of a node i

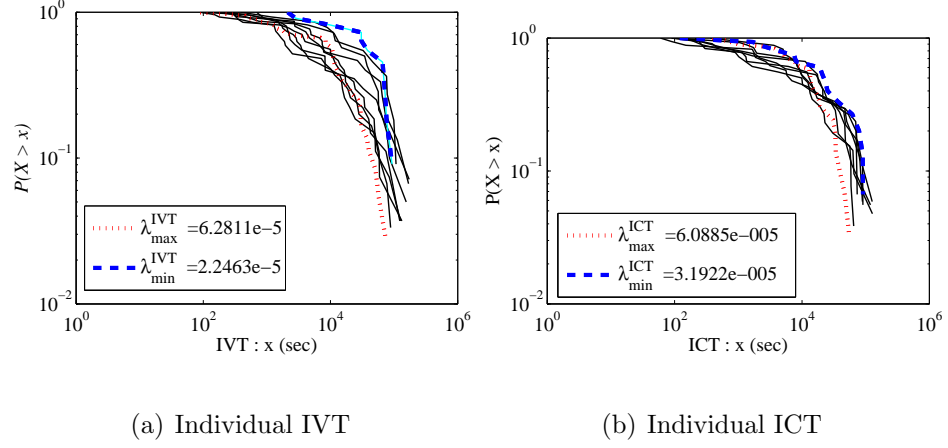


Figure 4.5: Inter-visit time (IVT) distribution and inter-contact time (ICT) distribution of a taxi to locations. They are fitted by maximum likelihood estimation (MLE) to exponential distributions. The maximum and minimum intensity of the best fitting exponential distributions are λ_{min} and λ_{max} , respectively.

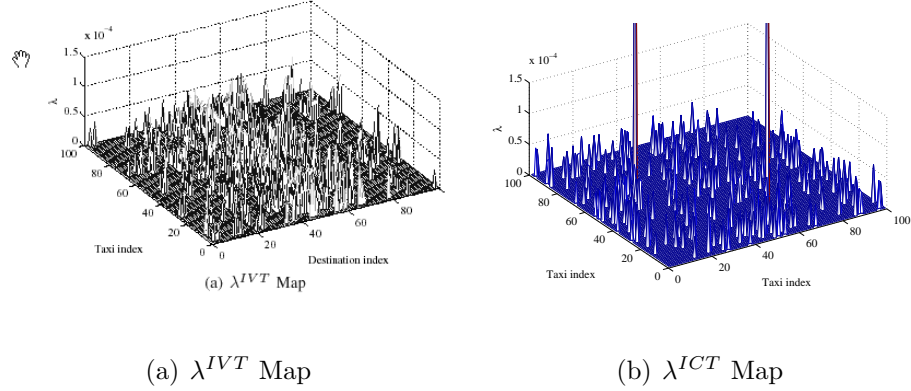


Figure 4.6: We plot the individual intensity values (λ^{IVT} and λ^{ICT}) of IVT and ICT exponential distributions from 100 taxies. IVT is plotted for 100 destination locations. A high intensity value of a particular location by a particular taxi in implies that a taxi has a high rate of visit to a particular location. Likewise, a high intensity value of a taxi with respect to another taxi meets they tend to meet very often. From the plots, we find that different taxies show different biases in the locations they visit and in the set of taxies they meet daily.

to the destination location, $d(m)$ of a packet m which implies the maximum potential

delivery probability. More precisely,

$$q_{m,i}(t) = \max\{q_{m,i}^1(t), q_{m,i}^2(t), q_{m,i}^3(t), \dots\}, \quad (4.13)$$

where $q_{m,i}^k$ denotes the delivery probability through k hops. For example, 1-hop probability, $q_{m,i}^1(t)$ is the probability that node i directly meets the destination location, $d(m)$ during the interval $[t, t_{\text{dl}}]$. For 2-hops or more, we find the path (sequence of nodes) with the maximum delivery probability by comparing all combinations of the intermediate nodes. Thus, the k -hop delivery probability is defined as follows (note that n_k denotes the k -th hop node and we replace $n_1 = i, n_{k+1} = d(m)$ for the ease of expression).

$$q_{m,i}^k(t) = \max_{\{n_2, \dots, n_k\} \in \mathcal{N}^{k-1}} \left\{ \int_{t_{n_{k-1}, n_k}}^{t_d - t} \dots \int_0^{t_d - t} \mathbb{P}[T_{n_1, n_2} = t_{n_1, n_2}] \prod_{j=2}^{k-1} \mathbb{P}[T_{n_j, n_{j+1}} = t_{n_j, n_{j+1}} - t_{n_{j-1}, n_j}] \mathbb{P}[T_{n_k, n_{k+1}} \leq (t_d - t) - t_{n_{k-1}, n_k}] dt_{n_1, n_2} \dots dt_{n_{k-1}, n_k} \right\} \quad (4.14)$$

where \mathcal{N}^k and $T_{n_j, n_{j+1}}$ denote the k -combinations of node sequences from the node set \mathcal{N} excluding the node i itself and a random variable indicating the inter-contact time or the inter-visit time between the j -th node and the $(j+1)$ -th node (or location).

4.4.2 Setup, Metric and Tested Algorithms

We implemented a resource allocation simulator for a DTN using MATLAB. Among 4000 taxies, we selected 1486 taxies that show valid GPS coordinates (included in the traces). By default, we use the communication range of WiFi (300 meters). Also, we selected 100 candidate locations (uniformly distributed) and 32 random pairs of S-D (source,destination) in the 100 candidate locations for our simulation. We also vary the number of packets per S-D pair to see the performance for different traffic loads. We set the deadline (i.e., t_{dl}) to be 24 hours. We make resource allocation decisions every 30 seconds. We also tested other intervals, and observed similar trends. We repeated ten simulations; each time, we varied S-D pairs randomly with different seeds.

We present the results for the max-delivery objective below. Two performance metrics are considered: (i) *delivery ratio* and (ii) *efficiency*. Delivery ratio is the ratio of the total delivered packets (counting only original packets) within a designated deadline to the total number of packets that sources initially have. Efficiency is the delivery ratio per unit cost where cost is simply the total number of *copy events* in the network. In the broadcast, we count the copy events at the receiver side.

We evaluate five algorithms summarized in Table 4.1. Some protocols do not have in their design the specifications for link/copy scheduling and value updates. Thus, for fair comparison, we additionally implemented the absent features. For

Table 4.1: Tested Algorithms (\star corresponds to the items that we added for fair comparison)

Algorithm	Link scheduling	Copy scheduling	Value update
Random	random	random	\times
DF [37]	\star greedy	\star difference	delegation
RAPID [19]	\star random	contribution	global
MC-Global	greedy	contribution	global
DMC	greedy	contribution	fusion

example, link scheduling has not been considered in DF and RAPID in [37] [19]. In Random algorithm, links and packets are randomly selected out of the connected links and packets that exist in either of two nodes that meet. In DF, link scheduling uses greedy, copy scheduling uses the differences of packet delivery probabilities. We used “delegation” originally proposed in DF for value updates, i.e., when a packet m is copied from v to w , the delivery probability of w for m is also copied to v . We intentionally use random (e.g., CSMA) for link scheduling at RAPID to quantify the impact of the joint copy and link scheduling.

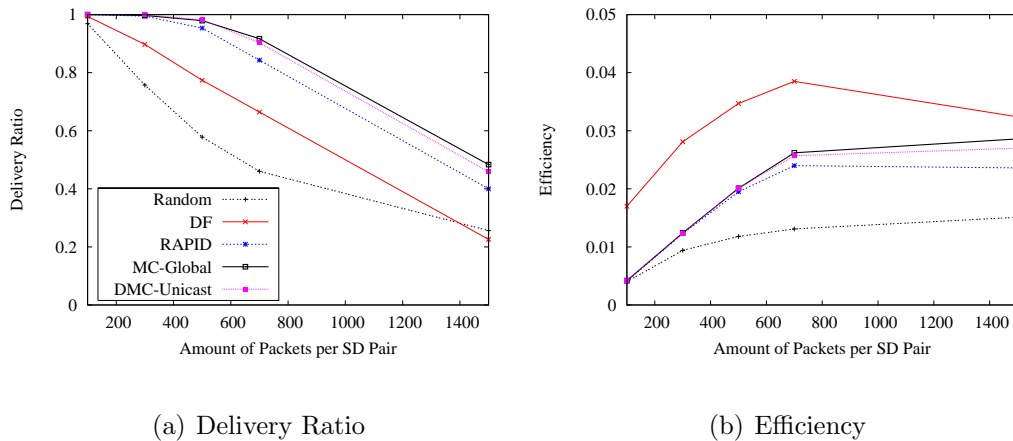


Figure 4.7: The Delivery ratio and Efficiency of algorithms listed in Table 4.1 against the offered load to 32 S-D pairs. Transmission range is 300m. Each value shows 95% confidence interval. We do not show cost as it is implied in the efficiency.

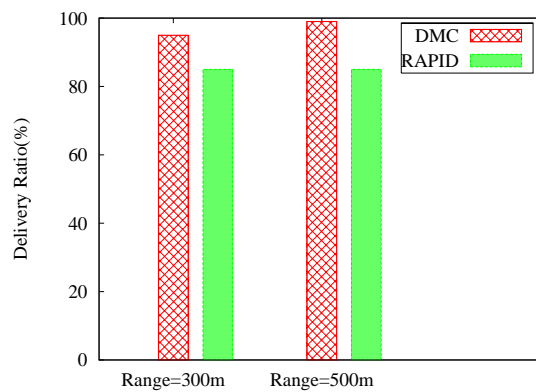


Figure 4.8: The delivery ratio of RAPID and DMC under different radio ranges. The input load is set to 700 packets.

4.4.3 Simulation Results

DMC-unicast

Fig. 4.7 shows the delivery ratio and efficiency of scheduling algorithms against the offered load (the number of packets each source generates). The delivery ratio

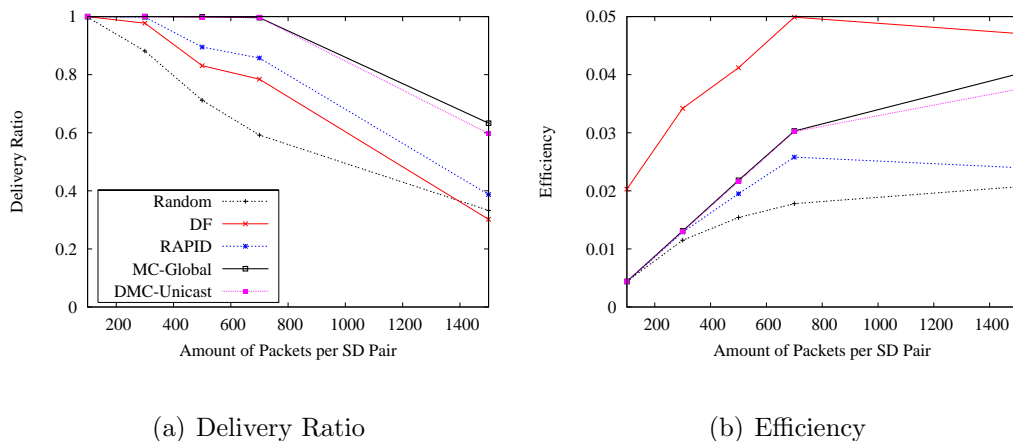


Figure 4.9: The Delivery ratio and Efficiency of algorithms listed in Table 4.1 against the offered load to 32 S-D pairs. Transmission range is 500m.

decreases as the offered load (the number of input packets) increases. MC-Global and DMC-unicast show better deliver ratios than any other protocols. DMC-unicast shows almost as good delivery ratio as MC-Global. This indicates that the localized information update, Fusion, can efficiently replace the expensive global knowledge update used in MC and also in RAPID. The main performance difference between DMC and RAPID is about 10% to 15% under high load and it arises from use of more intelligent link scheduling for DMC. We believe this effect will be more significant when the network density increases.

Among all tested algorithms, Random shows the worst performance in all cases. This was expected as it does not exploit the characteristics of IVT and ICT as shown in Figs 2 and 3. Figs 4 (a) and (b) are obtained when the radio range is 300 meter. We now modify the radio range from 300 meters to 500 meters to test the performance

of various protocols under high density environments. We are interested in studying the effect of more intelligent (but practical) link scheduling on the performance. Figure 4.8 compares the delivery ratios of DMC and RAPID as these two protocols are essentially different in two points: (1) DMC uses more lightweight metadata dissemination (called Fusion) than RAPID; the latter uses flooding and (2) DMC uses greedy link scheduling while RAPID uses random link scheduling. In general, the effect of the first point is minimal because the performance of DMC-unicast and MC-Global is not much different. Therefore, the performance difference between them most likely stems from the effect of the second point. As the density of the network increases, the interference becomes larger. Thus, we can observe from the figure that the performance gap between the two protocols almost doubles. The performance of DMC improves with the increased radio range because of a higher chance of meeting other nodes.

DMC-multicast

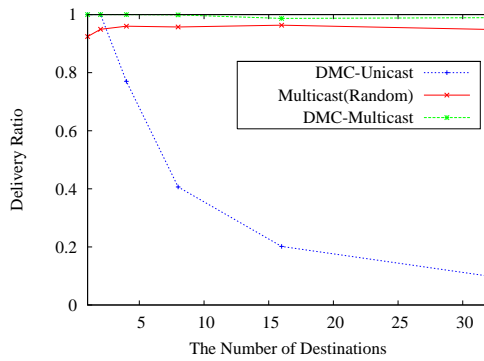
Using the same test environments as for the unicast case, we demonstrate the performance of DMC-multicast. The different thing is that we consider one source and different number of destinations which is varied from 1 to 32.

In the multicast experiments, we compare three schemes: DMC-unicast, Multicast (Random), and DMC-Multicast. To do the same thing as DMC-multicast, DMC-unicast scheme should generate/deliver packets whose number is that the number of

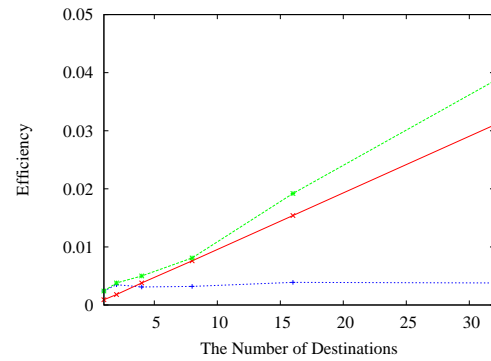
multicast packets in DMC-multicast times the number of destinations of a multicast packet. Multicast (Random) scheme selects links/packets randomly.

Figures 4.10 and 4.11 show the delivery ratio and efficiency of each scheme. As the number of destinations increases, so does the offered load on the networks in the unicast case. DMC-unicast shows that the delivery ratio decreases as the offered load increases. However, DMC-multicast has a smaller number of packets compared to that of DMC-unicast. In DMC-multicast, there is no change in the number of packets when the number of destinations changes. Only the size of a group that is a destination list of packets changes in the multicast. So the delivery ratio of DMC-multicast is affected much not by the size of a group but by the number of multicast packets than DMC-unicast. As shown in Figures 4.10 and 4.11, DMC-multicast scheme shows higher and fixed performance regardless of the number of destinations. As the number of destinations increases, the efficiency of DMC-multicast also increases. It means that when the same data should be delivered to multiple destinations, the efficiency of multicast schemes increases as much as the number of destinations. Figures 4.10 and 4.11 show the performance comparison results of each scheme when the offered load is 700 packet and 1000 packets for each destination respectively.

Through the experiments of DMC-multicast, we show that multicast is still an effective means in DTNs as well as cellular data networks.

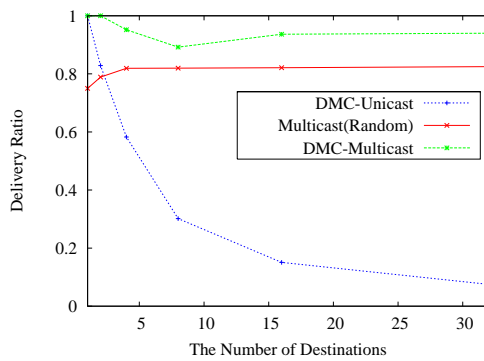


(a) Delivery Ratio

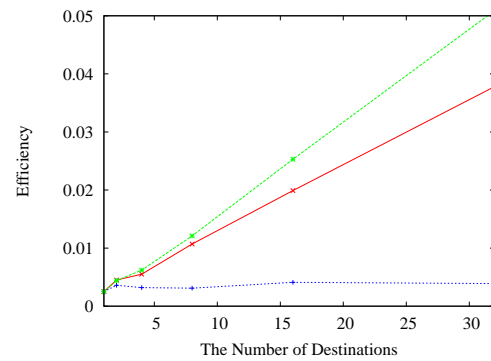


(b) Efficiency

Figure 4.10: The performance comparison of DMC-multicast and DMC-unicast. The number of destinations are varied from 1 to 32. Also the input load is set to 700 packets for each destination.



(a) Delivery Ratio



(b) Efficiency

Figure 4.11: The performance comparison of DMC-multicast and DMC-unicast. The number of destinations are varied from 1 to 32. Also the input load is set to 1000 packets for each destination.

4.5 Summary

In this Chapter, we consider resource allocation for jointly optimizing link scheduling, routing and replication for DTNs. The optimal resource allocation problem for

jointly optimizing link schedule and replication based routing is a *NP* hard problem in DTNs.

In this Chapter, we systematically approach the problem; we theoretically solve the optimal problem for "snapshot optimality" which restricts nodes to using only contemporarily available knowledge, and then approximate the optimal solution to reduce its complexity while minimizing the performance loss. This clearly shows how we derive our approximated solutions and provide some confidence over the expected performance. Based on the approximation, we propose an efficient and practical algorithm, called Distributed Max-Contribution (DMC). We then extend DMC to DMC-multicast for DTNs. We also demonstrate how our developed solutions can be applied to solving real world problems, such as information dissemination over a network of 1486 taxis. Each taxi is equipped with a WiFi radio. This is the biggest DTN network being simulated using real traces.

Chapter 5

Conclusion

In this thesis, we proved that multicast is an effective communication means in wireless networks. To show the feasibility of multicast in the networks, we chose cellular data networks and DTNs among the wireless networks. While cellular data networks are widely used now, DTNs are one of “challenged networks” that have different characteristics from conventional wireless networks. For both networks, unicast has been much studied, but multicast is not studied extensively. To make multicast feasible in the wireless networks, we studied multicast schedulers for cellular data networks and DTNs. Effective multicast schedulers play an important role in using less network resources. By efficiently using network resources through multicast scheduling, multicast can be an effective communication means in wireless networks.

First, in cellular data networks, we propose two sets of unified multicast propor-

tional fair scheduling algorithms, the (\vec{e}, α) Inter-Group Proportional Fairness (IPF) scheduler and the (\vec{e}, α) Multicast Proportional Fairness (MPF) scheduler. Each of these schedulers supports a different utility function. In particular, the IPF scheduler supports the utility function of log of aggregate group throughput that is computed by summing up all the user throughput in a group, and the MPF scheduler supports the utility of log of individual user throughput. These multicast scheduling algorithms can be applied to different scenarios depending on the application and business model of ISPs.

We proved the proportionally fairness of two multicast schedulers. By comparing the performance of these schedulers with several multicast scheduling algorithms in simulations, we showed that they achieve good balance between fairness and throughput of groups or users.

Second, we considered resource allocation for jointly optimizing link scheduling, routing and replication in DTNs. The optimal resource allocation problem is a *NP* hard problem. Many existing techniques in DTNs are based on intuition-driven heuristics in order to improve performance.

In this thesis, we systematically approached the problem; we theoretically solve the optimal problem for snapshot optimality which restricts nodes to using only temporarily available knowledge, and then approximate various components of the optimal solution to reduce its complexity while minimizing the performance loss.

This clearly shows how we derive our heuristic solutions and provide some confidence over the expected performance. Based on the approximation, we propose an efficient and practical algorithm, called Distributed Max-Contribution (DMC). We then extend DMC to DMC-multicast for DTNs. We also demonstrate how our developed solutions can be applied to solving real world problems, such as information dissemination over a network of 1486 taxis, each equipped with a WiFi radio. This is the biggest DTN network being simulated using real traces. From the traces, we extract statistical properties of taxi movements and apply them to formulate parameter values to the input of our algorithms. This work clearly demonstrates the ways in which our solutions would perform in real network settings.

By proposing efficient multicast schedulers to cellular data networks and DTNs respectively, we make multicast more feasible in the networks. Through this study, we demonstrate that multicast can be still an effective means of communication for wireless networks.

For future work, multicast Quality of Service (QoS) schedulers need to be considered for cellular data networks. In this thesis, we present proportional fair multicast schedulers. However, proportional fairness schedulers do not guarantee any throughput for each user or group in a cell. Applications such as multimedia streaming and downloading need the minimum throughput. To provide the minimum throughput or limiting the maximum throughput to each user or group in a system, Quality Of

Services (QoS) should be considered.

One way to provide Quality of Service (QoS) is to ensure that the average transmission rate that each group or user gets does not fall below specified rates. This type of QoS has been studied in unicast [84]. By extending proposed IPF and MPF multicast scheduling algorithm, the QoS multicast schedulers.

In this thesis, we assumed that there is no restriction on the storage (buffer) and the power of nodes in DTNs. In order to apply to more available and realistic environments, more restrictions need to be considered under the optimal framework in DTNs. Also, we considered only the objective function of maximum delivery rate in the thesis. To achieve goals of various applications in DTNs, other objective functions need to be considered such as minimum delay or maximum delivery ratio within a deadline. We leave them as future work.

Bibliography

- [1] 3g today. <http://www.3gtoday.com/>.
- [2] Delay tolerant network research group (DTNRG). www.dtnrg.org.
- [3] The network simulator- ns-2. <http://www.isi.edu/nsnam/ns/>.
- [4] Multicast extensions to OSPF, RFC1584, 1994.
- [5] Internet group management protocol, version 3, RFC3376, 2002.
- [6] Ad hoc on-demand distance vector (AODV) routing, RFC3561, 2003.
- [7] Optimized link state routing protocol (OLSR), RFC3626, 2003.
- [8] Protocol independent multicast - dense mode (PIM-DM), RFC3973, 2005.
- [9] Protocol independent multicast - sparse mode (PIM-SM), RFC4601, 2006.
- [10] 3GPP. High speed downlink packet access (HSDPA): overall description, Mar 2003. <http://www.3gpp.org>.

- [11] A. Doria A. Lindgren and O. Schelen. Probabilistic routing in intermittently connected networks. *SIGMOBILE Mobile Comput. Commun.*, 2003.
- [12] Muhammad Abdulla and Robert Simon. A simulation analysis of multicasting in delay tolerant networks. In *WSC '06: Proceedings of the 38th conference on Winter simulation*, pages 2234–2241. Winter Simulation Conference, 2006.
- [13] P. Agashe, R. Rezaiifar, and P. Bender. Cdma2000 high rate broadcast packet data air interface design. *IEEE Communications Magazine*, pages 83–89, Feb 2004.
- [14] R. Agrawal, A. Bedekar, R. La, and V. Subramanian. A class and channel-condition based weighted proportionally fair scheduler. In *ITC'01*, Salvador, Brazil, December 2001.
- [15] R. Agrawal, A. Bedekar, R. La, and V. Subramanian. A class and channel-condition based weighted proportionally fair scheduler for EDGE/GPRS. In *ITCOM'01*, Denver, CO, August 2001.
- [16] R. Agrawal and V. Subramania. Optimality of certain channel aware scheduling policies. In *40th Annu. Allerton Conf. Communications, Control, and Computing*, pages 1532–1541, Monticello, Illinois, USA, Oct 2002.

- [17] E. Altman, G. Neglia, F. De Pellegrini, and D. Miorandi. Decentralized stochastic control of delay tolerant networks. In *Proceedings of IEEE INFOCOM*, 2009.
- [18] M. Andrews, L. Quan, and A. Stolyer. Optimal utility based multi-user throughput allocation subject to throughput constraints. In *Infocom 2005*, Miami, FL, Mar 2005.
- [19] Aruna Balasubramanian, Brian Neil Levine, and Arun Venkataramani. DTN routing as a resource allocation problem. In *Proceedings of SIGCOMM 2007*, 2007.
- [20] A. Ballardie. Core based trees (CBT) multicast routing architecture, RFC2201, 1997.
- [21] S. Basagni, I. Chlamtac, V.R. Syrotiuk, and B.A. Woodward. A distance routing effect algorithm for mobility (DREAM). In *Proceedings of ACM MobiCom*, 1998.
- [22] P. Bender, P. Black, M. Grob, R. Padovani, N. Sindhushayana, and A. Viterbi. CDMA/HDR: A bandwidth-efficient high-speed wireless data service for nomadic users. *IEEE Communications Magazine*, 38(7):70–77, July 2000.
- [23] S. Borst. User-level performance of channel-aware scheduling algorithms in wireless data networks. *IEEE/ACM Trans. Networking*, 13(3):636–647, 2005.

- [24] S. Borst and P. Whiting. Dynamic rate control algorithms for HDR throughput optimization. In *IEEE Infocom'01*, pages 976–985, April 2001.
- [25] Jean-Yves Le Boudec. Rate adaptation. congestion control and fairness: A tutorial, September 2005.
- [26] J. Burgess, B. Gallagher, D. Jensen, and B.N. Levine. Maxprop: Routing for vehicle-based disruption-tolerant networking. In *Proceedings of IEEE INFOCOM*, 2006.
- [27] S. Burleigh, A. Hooke, L. Torgerson, K. Fall, V. Cerf, B. Durst, and K. Scott. Delay-tolerant networking: an approach to interplanetary internet. *IEEE Communications Magazine*, 41:128–136, 2003.
- [28] J. Byers, J. Considine, M. Mitzenmacher, and S. Rost. Informed content delivery across adaptive overlay networks. In *Proceedings of the 2002 conference on Applications, technologies, architectures, and protocols for computer communications (SIGCOMM)*, pages 47–60, Pittsburgh, PA, USA, August 2002.
- [29] J. W. Byers, M. G. Luby, M. Mitzenmacher, and A. Rege. Digital fountain approach to reliable distribution of bulk data. In *SIGCOMM*, pages 56–67, 1998.
- [30] R. Cohen and R. Rizzi. On the trade-off between benenergy and multicast effi-

ciency in 802.16e-like mobile networks. In *IEEE Infocom'06*, Barcelona, Spain, Apr 2006.

- [31] Elizabeth M. Daly and Mads Haahr. Social network analysis for routing in disconnected delay-tolerant manets. In *MobiHoc '07: Proceedings of the 8th ACM international symposium on Mobile ad hoc networking and computing*, pages 32–40, New York, NY, USA, 2007. ACM.
- [32] Stephen E. Deering and David R. Cheriton. Multicast routing in datagram internetworks and extended LANs. *ACM Trans. Comput. Syst.*, 8(2):85–110, 1990.
- [33] Stephen E. Deering, Deborah Estrin, Dino Farinacci, Van Jacobson, Ching-Gung Liu, and Liming Wei. The PIM architecture for wide-area multicast routing. *IEEE/ACM Trans. Netw.*, 4(2):153–162, 1996.
- [34] Q. Du and X. Zhang. Adaptive power and rate allocation for mobile multicast throughput optimization over fading channels in wireless networks. In *Proc. of ICCCN 2006*, pages 261–266, Oct 2006.
- [35] Q. Du and X. Zhang. Adaptive multicast with power and rate control over fading channels in mobile wireless networks. In *Proc. of IEEE WCNC 2007*, pages 3335–3340, Mar 2007.

- [36] Q. Du and X. Zhang. Cross-layer design based rate control for mobile multicast in cellular networks. In *Proc. of IEEE GLOBECOM '07*, pages 5180–5184, Nov 2007.
- [37] V. Erramilli, M. Crovella, A. Chaintreau, and C. Diot. Delegation forwarding. In *Proceedings of ACM MobiHoc*, 2008.
- [38] Vijay Erramilli, Mark Crovella, Augustin Chaintreau, and Christophe Diot. Delegation forwarding. In *MobiHoc '08: Proceedings of the 9th ACM international symposium on Mobile ad hoc networking and computing*, pages 251–260, New York, NY, USA, 2008. ACM.
- [39] Kevin Fall. A delay-tolerant network architecture for challenged internets. In *SIGCOMM '03: Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 27–34, New York, NY, USA, 2003. ACM.
- [40] Wei Gao, Qinghua Li, Bo Zhao, and Guohong Cao. Multicasting in delay tolerant networks: a social network perspective. In *MobiHoc '09: Proceedings of the tenth ACM international symposium on Mobile ad hoc networking and computing*, pages 299–308, New York, NY, USA, 2009. ACM.
- [41] C. Gkantsidis and P. Rodriguez. Network coding for large scale content distribution. In *INFOCOM'05*, 2005. Avalanche.

- [42] Song Guo and Oliver W. W. Yang. Energy-aware multicasting in wireless ad hoc networks: A survey and discussion. *Comput. Commun.*, 30(9):2129–2148, 2007.
- [43] Gagan Raj Gupta and Ness B. Shroff. Delay analysis for multi-hop wireless networks. April 2009.
- [44] H.Dubois-Ferriere, M.Grossglauser, and M.Vetterli. Age matters: Efficient route discovery in mobile ad hoc networks using encounter ages. In *Proceedings of MobiHoc 2003*, 2003.
- [45] Pi-Cheng Hsiu and Tei-Wei Kuo. A maximum-residual multicast protocol for large-scale mobile ad hoc networks. *Mobile Computing, IEEE Transactions on*, 8(11):1441–1453, Nov. 2009.
- [46] Pan Hui, A. Chaintreau, J. Scott, R. Gass, Jon Crowcroft, and C. Diot. Bubble rap: Social-based forwarding in delay tolerant networks. In *Proceedings ACM Mobihoc*, 2008.
- [47] Sushant Jain, Kevin Fall, and Rabin Patra. Routing in a delay tolerant network. In *SIGCOMM '04: Proceedings of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 145–158, New York, NY, USA, 2004. ACM.

- [48] A. Jalali, R. Padovani, and R. Pankaj. Data throughput of CDMA-HDR a high efficiency-high data rate personal communication wireless system. In *Vehicle Technology Conference (VTC2000-Spring)*, pages 1854–1858, Tokyo, Japan, May 2000.
- [49] Lusheng Ji and M.S. Corson. Differential destination multicast-a manet multicast routing protocol for small groups. In *INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 2, pages 1192–1201 vol.2, 2001.
- [50] David B. Johnson and David A. Maltz. Dynamic source routing in ad hoc wireless networks. In *Mobile Computing*, pages 153–181. Kluwer Academic Publishers, 1996.
- [51] Changhee Joo, Xiaojun Lin, and N.B. Shroff. Understanding the capacity region of the greedy maximal scheduling algorithm in multi-hop wireless networks. pages 1103–1111, April 2008.
- [52] Sung ju Lee, William Su, and Mario Gerla. On-demand multicast routing protocol in multihop wireless mobile networks. pages 1298–1302, 2001.
- [53] Gunnar Karlsson, Vincent Lenders, and Martin May. Delay-tolerant broadcasting. In *CHANTS '06: Proceedings of the 2006 SIGCOMM workshop on Challenged networks*, pages 197–204, New York, NY, USA, 2006. ACM.

- [54] F. Kelly. Charging and rate control for elastic traffic. *European Transactions on Telecommunications*, 8:33–37, Jan 1997.
- [55] F. Kelly. Charging and rate control for elastic traffic. *European Transactions on Telecommunications*, 8:33–37, 1997.
- [56] Minkyong Kim and David Kotz. Periodic properties of user mobility and access-point popularity. *Journal of Personal and Ubiquitous Computing*, 11(6):465–479, August 2007.
- [57] Amir Krifa, Chadi Barakat, and Thrasyvoulos Spyropoulos. An optimal joint scheduling and drop policy for delay tolerant networks. In *Proceedings of the WoWMoM Workshop on Autonomic and Opportunistic Communications*, 2008.
- [58] H. Kushner and P. Whiting. Asymptotic properties of proportional fair sharing algorithms. In *40th Annu. Allerton Conf. Communications, Control, and Computing*, Monticello, Illinois, USA, Oct 2002.
- [59] J. Lebrun, C.-N. Chuah, D. Ghosal, and M. Zhang. Knowledge-based opportunistic forwarding in vehicular wireless ad hoc networks. In *Proceedings of IEEE Vehicular Technology Conference*, 2004.
- [60] K. Lee, S. Hong, S. Kim, I. Rhee, and S. Chong. Slaw : A new human mobility model. In *Proceedings of IEEE INFOCOM*, 2009.

- [61] A. Lindgren, A. Doria, and O. Schelen. Probabilistic routing in intermittently connected networks. *ACM SIGMOBILE Mobile Computing and Communications Review*, 7(3):19–20, 2003.
- [62] Cong Liu and Jie Wu. An optimal probabilistic forwarding protocol in delay tolerant networks. In *Proceedings of ACM MobiHoc*, 2009.
- [63] X. Liu, E. K. P. Chong, and N. B. Shroff. Opportunistic transmission scheduling with resource-sharing constraints in wireless networks. *IEEE Journal on Selected Areas in Communications*, 19(10):2053–2064, 2001.
- [64] X. Liu, E. K. P. Chong, and N. B. Shroff. A framework for opportunistic scheduling in wireless networks. *Computer Networks*, 41(4):451–474, 2003.
- [65] Lucent. Dynamic rate scheduler design for broadcast and multicast in cdma-hdr, Sep. 2005.
- [66] PC Magazine. Verizon’s vcast: Video over evdo phones, Jan 2005. <http://www.pcmag.com/article2/0,1759,1749429,00.asp>.
- [67] J. Mo and Jean C. Walrand. Fair end-to-end window-based congestion control. *IEEE/ACM Trans. Netw*, 8(5):556–567, 2000.
- [68] MobiTV. Mobile television and radio service provider. <http://www.mobitv.com>.

- [69] M.J. Neely. Delay analysis for maximal scheduling in wireless networks with bursty traffic. pages 6–10, April 2008.
- [70] Y. Ohta, M. Tsuru, and Y. Oie. Framework for fair scheduling schemes in the next generation high-speed wireless links. In *the 8th International Conference on Cellular and Intelligent Communications (CIC 2003)*, Oct 2003.
- [71] Y. Ohta, M. Tsuru, and Y. Oie. Fairness property and TCP-level performance of unified scheduling algorithm in HSDPA networks. In *Universal Multiservice Networks: Third European Conference, ECUMN 2004, Porto, Portugal, October 25-27, 2004. Proceedings*, volume 3262, pages 185–195. Springer, 2004.
- [72] Jorg Ott and Dirk Kutscher. A disconnection-tolerant transport for drive-thru internet environments. In *in Proceedings of IEEE INFOCOM*, pages 1849–1862, 2005.
- [73] Charles Perkins and Pravin Bhagwat. Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers. pages 234–244, 1994.
- [74] R. Ramanathan, R. Hansen, P. Basu, R. R. Hain, and R. Krishnan. Prioritized epidemic routing for opportunistic networks. In *Proceedings of ACM MobiSys workshop on Mobile Opportunistic Networks (MobiOpp)*, 2007.
- [75] Ram Ramanathan, Prithwish Basu, and Rajesh Krishnan. Towards a formalism

- for routing in challenged networks. In *CHANTS '07: Proceedings of the second ACM workshop on Challenged networks*, pages 3–10, New York, NY, USA, 2007. ACM.
- [76] Sylvia Ratnasamy, Andrey Ermolinskiy, and Scott Shenker. Revisiting IP multicast. *SIGCOMM Comput. Commun. Rev.*, 36(4):15–26, 2006.
- [77] Elizabeth M. Royer and Charles E. Perkins. Multicast operation of the ad-hoc on-demand distance vector routing protocol. pages 207–218, 1999.
- [78] R.C. Shah, S. Roy, S. Jain, and W. Brunette. Data mules: Modeling a three-tier architecture for sparse sensor networks. In *Proceedings First IEEE Intl Workshop Sensor Network Protocols and Applications (SNPA 03)*, 2003.
- [79] S. Shakkottai and A. L. Stolyar. Scheduling for multiple flows sharing a time-varying channel: The exponential rule. *Analytic Methods in Applied Probability. In Memory of Fridrih Karpelevich. American Mathematical Society Translations*, 207:185–202, 2002.
- [80] U. Shevade, Han Hee Song, Lili Qiu, and Yin Zhang. Incentive-aware routing in dtns. In *Network Protocols, 2008. ICNP 2008. IEEE International Conference on*, pages 238–247, Oct. 2008.
- [81] T. Spyropoulos, K. Psounis, and C.S. Raghavendra. Spray and wait: An efficient

- routing scheme for intermittently connected mobile networks. In *Proceedings of ACM SIGCOMM*, 2005.
- [82] Thrasyvoulos Spyropoulos, Konstantinos Psounis, and Cauligi S. Raghavendra. Efficient routing in intermittently connected mobile networks: the multiple-copy case. *IEEE/ACM Trans. Netw.*, 16(1):77–90, 2008.
- [83] Thrasyvoulos Spyropoulos, Konstantinos Psounis, and Cauligi S. Raghavendra. Efficient routing in intermittently connected mobile networks: The single-copy case. *IEEE/ACM Transactions on Networking*, 16(1):63–76, 2008.
- [84] A. L. Stolyar. On the asymptotic optimality of the gradient scheduling algorithm for multi-user throughput allocation. *Operations Research*, 53(1):12–25, 2005.
- [85] Susan Symington, Robert C. Durst, and Keith Scott. Custodial multicast in delay tolerant networks. In *Consumer Communications and Networking Conference, 2007. CCNC 2007. 4th IEEE*, pages 207–211, Jan. 2007.
- [86] L. Tassiulas and A. Ephremides. Jointly optimal routing and scheduling in packet ratio networks. *Information Theory, IEEE Transactions on*, 38(1):165–168, Jan 1992.

- [87] Shanghai Jiaotong University Traffic Information Grid Team, Grid Computing Center. Shanghai taxi trace data. <http://wirelesslab.sjtu.edu.cn/>.
- [88] D. Tse and P. Viswanath. *Fundamentals of Wireless Communication*. Cambridge University Press, 2005.
- [89] A. Vahdat and D. Becker. Epidemic routing for partially-connected ad hoc networks. Technical Report, CS-200006, Duke University, April 2000.
- [90] P. Viswanath, D. Tse, and R. Laroia. Opportunistic beamforming using dumb antennas. *IEEE Transactions on Information Theory*, 48(6):1277–1294, 2002.
- [91] D. Waitzman, C. Partridge, and S. E. Deering. Distance vector multicast routing protocol, RFC1075, 1988.
- [92] Jeffrey E. Wieselthier, Gam D. Nguyen, and Anthony Ephremides. Energy-efficient broadcast and multicast trees in wireless networks. *Mob. Netw. Appl.*, 7(6):481–492, 2002.
- [93] Y., M. Tsuru, and Y. Oie. Fairness property and TCP-level performance of unified scheduling algorithm in HSDPA networks. In *ECUMN*, pages 185–195, 2004.
- [94] Peng Yang and Mooi Choo Chuah. Context-aware multicast routing scheme for disruption tolerant networks. In *PE-WASUN '06: Proceedings of the 3rd ACM*

international workshop on Performance evaluation of wireless ad hoc, sensor and ubiquitous networks, pages 66–73, New York, NY, USA, 2006. ACM.

- [95] Wen;Lin Yang and Lung Jen Wang. The investigation of delay constrained multicasting with minimum energy consumption in static ad hoc wireless networks. *Int. J. Ad Hoc Ubiquitous Comput.*, 4(3/4):237–250, 2009.
- [96] Y. Yi, A. Proutiere, and M. Chiang. Complexity in wireless scheduling: Impact and tradeoffs. In *Proceedings of ACM Mobihoc*, 2008.
- [97] Di Yuan, Joanna Bauer, and Dag Haugland. Minimum-energy broadcast and multicast in wireless networks: An integer programming approach and improved heuristic algorithms. *Ad Hoc Netw.*, 6(5):696–717, 2008.
- [98] X. Zhang and Q. Du. Adaptive low-complexity erasure-correcting code-based protocols for QoS-driven mobile multicast services over wireless networks. *IEEE Transactions on Vehicular Technology*, 55(5):1633–1647, Sep 2006.
- [99] W. Zhao, M. Ammar, and E. Zegura. A message ferrying approach for data delivery in sparse mobile ad hoc networks. In *Proceedings of ACM MobiHoc*, 2004.
- [100] Wenrui Zhao, Mostafa Ammar, and Ellen Zegura. Multicasting in delay tolerant networks: semantic models and routing algorithms. In *WDTN '05: Proceedings*

of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking, pages 268–275, New York, NY, USA, 2005. ACM.

Glossary

AODV Ad-hoc On-demand Distance Vector

AT Access Terminal

BS Base Station

CBT Core Based Trees

CCDF Complementary Cumulative Distribution Function

CDF Cumulative Distribution Function

DF Delegation Forwarding

DMC Distributed Max-Contribution

DRC Data Rate Control

DSDV Dynamic Destination-Sequenced Distance Vector

DSR Dynamic Source Routing

DTN Delay or Disruption Tolerant Networks

DVMRP Distance Vector Multicast Routing Protocol

EVDO CDMA Evolution Data Only

GMR Gradient algorithm with Maximum/Mimimum Rate constraints

GPS Global Positioning System

HSDPA High Speed Downlink Packet Access

ICT Inter-Contact Times

IGMP Internet Group Management Protocol

IPF Inter-group Proportional Fairness Scheduling Algorithm

IPF-MR Inter-group Proportional Fairness Scheduling Algorithm with the Minimum Rate

IVT Inter-Visit Times

MANET Mobile Ad-hoc Networks

MOSPF Multicast Open Shortest Path First

MPF Multicast Proportional Fairness Scheduling Algorithm

MPF-MR Multicast Proportional Fairness Scheduling Algorithm with the Minimum Rate

OLSR Optimized Link State Routing Protocol

OSPF Open Shortest Path First

PF Proportional Fairness

PFMR Proportional Fairness algorithm with the Minimum Rate constraint

PRoPHET Probabilistic Routing Protocol using History of Encounters and Transitivity

QoS Quality of Services

SNR Signal to Noise Ratio

SSM Source Specific Mode

TDM Time Division Multiplexing

Index

- AODV, 16
- broadcast, 1
- Bubble Rap, 23
- cellular data networks, 4
- challenged networks, 5
- DDM, 17
- Deering, 14
- Direct Transmission, 21
- DMC, 13
- DMC multicast, 13
- DRC, 6
- DSDV, 16
- DSR, 16
- DTNRG, 20
- DTNs, 4, 5, 20
- DVMRP, 15
- Epidemic Routing, 21
- EXP algorithm, 19
- GMR Algorithm, 19
- IGMP, 15
- IP Multicast, 14
- IPTV, 15
- MANET, 24
- MANETs, 3
- MOSPF, 15
- MRMP, 17
- ODMRP, 16
- OLSR, 16
- OSPF, 15
- PIM, 15
- ProPHET, 23

QoS, 20

RAPID, 26

SimBet, 23

snapshot optimality, 26

SNR, 5

spray and wait, 23

TDM, 5, 17

types of communications, 3

unicast, 1